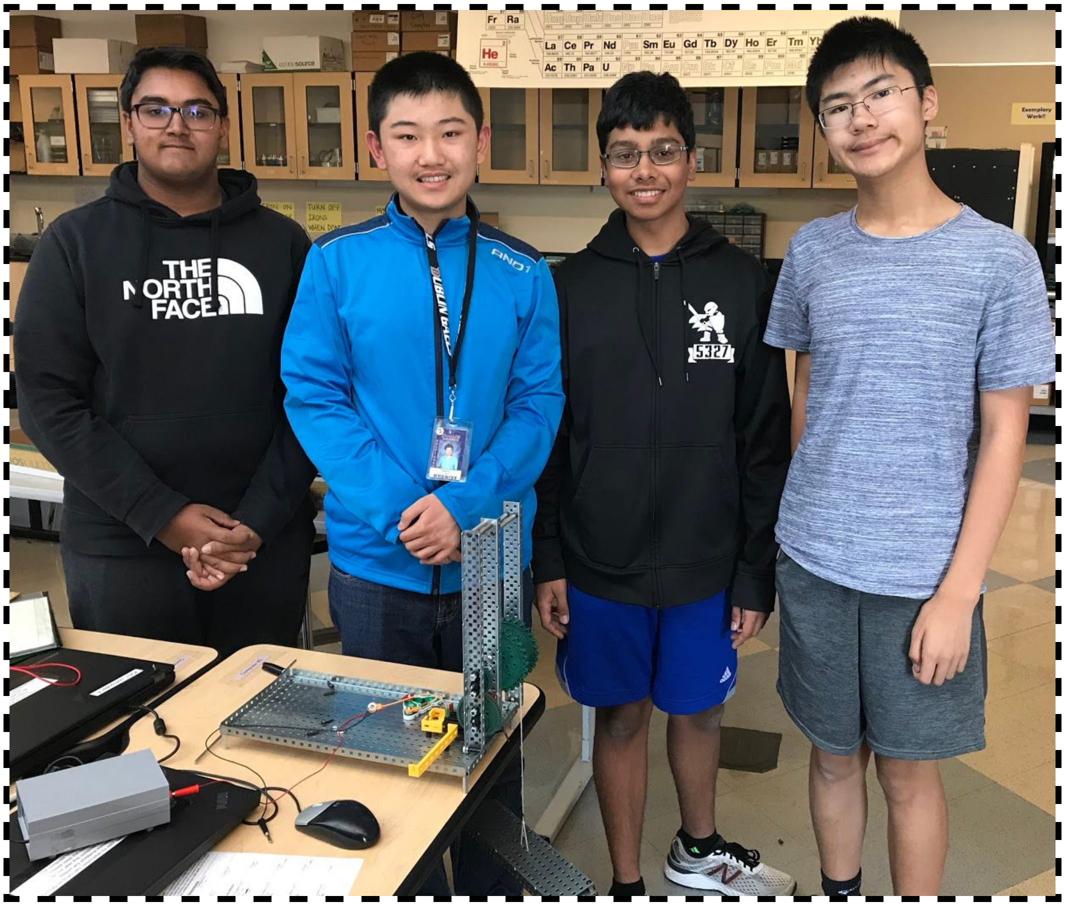
Project 1.2.5: Electric Winch



Honors Principles of Engineering Period 7

10/14/2019 - 10/22/2019

Vedant Agrawal, Sai Balakumar, Alexander Lim, Derek Xu

Table of Contents

Design Brief

Client: Local Construction Company

Designers: Vedant Agrawal, Sai Balakumar, Alexander Lim, Derek Xu

Problem Statement: It is impossible to move large, heavy objects up and down multiple floors of unfinished structures that are under construction.

Design Statement: Create a small-scale prototype winch to move a small weight upwards.

Constraints

- Winch must be able to lift a weight of 100 grams dangling off of a table up by 30 cm
- Use FT and VEX parts to build machine, must be powered by a switch-controlled FT motor
- Complete within 5 workdays

Deliverables

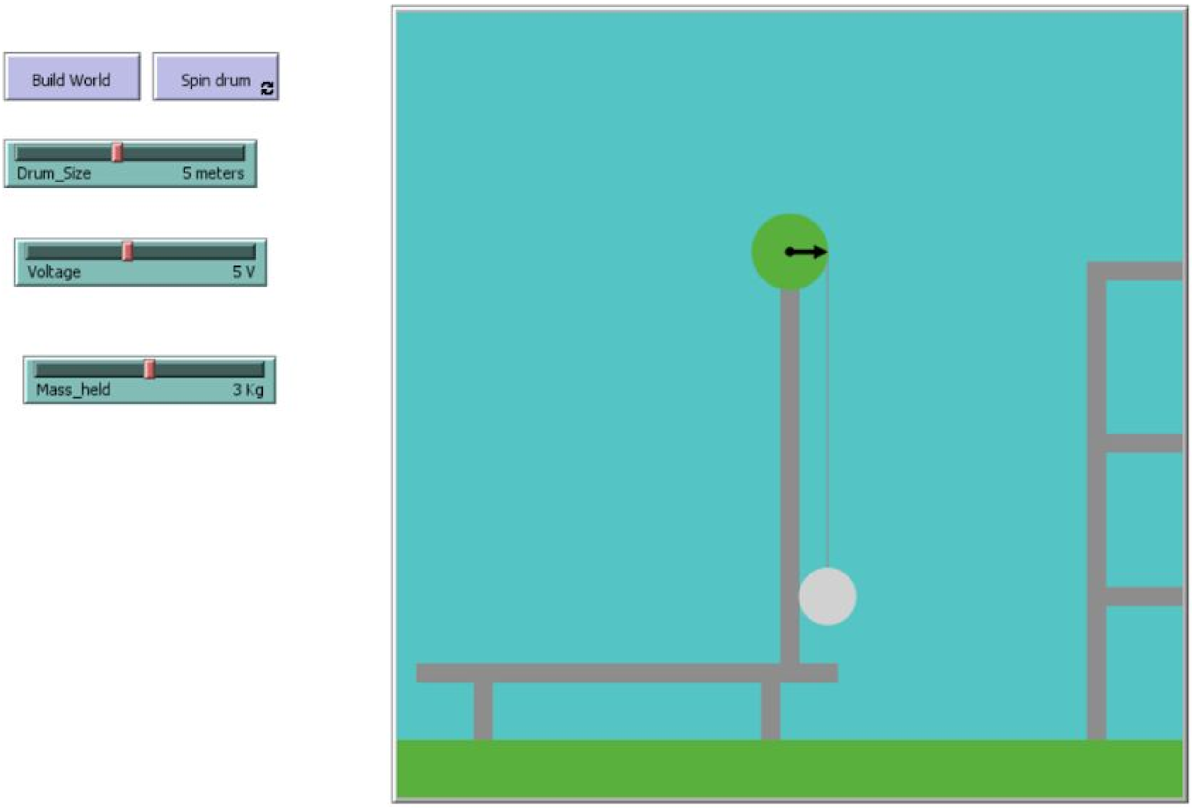
- Prototype
- Netlogo Simulation
- Project Report
- Documentation

Brainstorm

- High torque ratio to lift more weight
- High speed ration to lift less weight but faster
- Mix to achieve an optimal combination of speed and torque

Netlogo Program

Program Interface

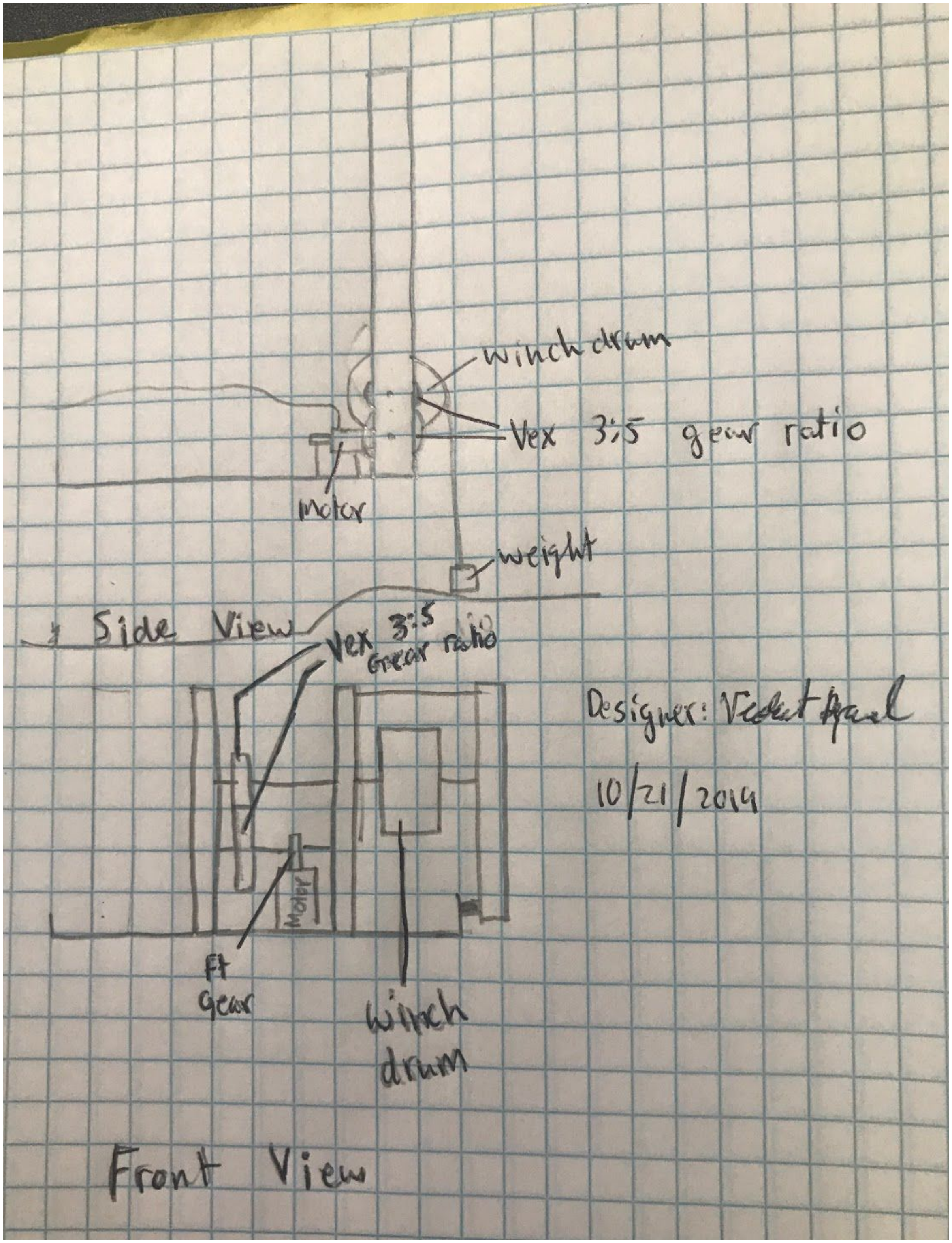


Program Code

```
;Vedant Agrawal, Sairam Balakumar, Alexander Lim, Derek Xu
;POE 7
;P125 Mechanical Winch

globals [
  ;starts list of global variables to be used in the program
  Drum
  ;Global variable representing the trutle representing the drum
  Mass
  ;Global variable representing the trutle representing the weight/mass
  Connection
  ;Global variable representing the trutle representing the dot which is used for connection purposes to connect the drum and weight together
  amperage
  ;Global variable representing the value of the current (A)
  angular_speed
  ;Global variable representing the angular speed (rad/s)
  time_unit
  ;Global variable that represents each unit of time between each time the go function repeats itself
  power
  ;Global variable that represents the value of the power (W)
]

to setup
  CA
  ;Clears the world of everything previous
  ask patches [set pcolor cyan]
  ;Makes all patches in the world turn cyan
  ask patches with [pycor < -17] [set pcolor green]
  ;Makes all patches in the world with a y-coordinate below -17 turn green
  ask patches with [pxcor < 17 and pxcor > 15 and pycor > -18 and pycor < 8] [set pcolor grey]
  ;Makes all patches in the world with an x-coordinate between 15 and 17 and a y-coordinate between -18 and 8 turn grey
  ask patches with [pxcor > 16 and pycor > 6 and pycor < 8] [set pcolor grey]
  ;Makes all patches in the world with an x-coordinate greater than 16 and a y-coordinate between 6 and 8 turn grey
  ask patches with [pxcor > 16 and pycor > -3 and pycor < -1] [set pcolor grey]
  ;Makes all patches in the world with an x-coordinate greater than 16 and a y-coordinate between -3 and -1 turn grey
  ask patches with [pxcor > 16 and pycor > -11 and pycor < -9] [set pcolor grey]
  ;Makes all patches in the world with an x-coordinate greater than 16 and a y-coordinate between -11 and -9 turn grey
  ask patches with [pxcor < 1 and pxcor > -1 and pycor > -15 and pycor < 9] [set pcolor grey]
  ;Makes all patches in the world with an x-coordinate between -1 and 1 and a y-coordinate between -15 and 9 turn grey
  ask patches with [pxcor > -20 and pxcor < 3 and pycor > -15 and pycor < -13] [set pcolor grey]
  ;Makes all patches in the world with an x-coordinate between -20 and 3 and a y-coordinate between -15 and -13 turn grey
  ask patches with [pxcor < 0 and pxcor > -2 and pycor > -18 and pycor < -14] [set pcolor grey]
  ;Makes all patches in the world with an x-coordinate between -2 and 0 and a y-coordinate between -18 and -14 turn grey
  ask patches with [pxcor < -15 and pxcor > -17 and pycor > -18 and pycor < -14] [set pcolor grey]
  ;Makes all patches in the world with an x-coordinate between -17 and -15 and a y-coordinate between -18 and -14 turn grey
  crt 1 [
  ;Creates 1 turtle, which represents the drum
    setxy 0 8
    ;Puts the drum in the upper center of the world
    set shape "clock"
    ;Makes the drum a circle that has an arrow to show its direction
    set color green
    ;Makes the drum green
    set heading 90
    ;Makes the arrow on the drum point directly towards the right
    set size Drum_Size
    ;Sets the size of the drum to be that specified by the slider
  ]
  crt 1 [
  ;Makes a turtle that is used for connection purposes
    setxy (pi / 8 * Drum_Size) 8
    ;Puts the point the of connection on the right of the drum
    set size 0.1
    ;Initializes the point of connection to be of a very small size
    set shape "dot"
    ;Makes the turtle have a shape representing that of a very small dot
  ]
  crt 1 [
  ;Makes a turtle that represents the weight
    setxy (pi / 8 * Drum_Size) -10
    ;puts the weight at a place lower than the drum, while also placing it at the x-coordinate with the rightmost part of the drum
    set size Mass_held
    ;Makes the weight's diameter length directly porportional to its mass
    set shape "circle"
    ;Makes the weight take a circular shape
    set color 8
    ;Makes the weight have a grey color
    set heading 0
    ;Makes the weight always move up, to show that it is being pulled up by the winch
  ]
```

```
  set Drum one-of turtles with [shape = "clock"]
  ;Makes the variable name "Drum" correspond with the turtle representing the drum
  set Mass one-of turtles with [shape = "circle"]
  ;Makes the variable name "Mass" correspond with the turtle representing the weight
  set Connection one-of turtles with [shape = "dot"]
  ;Makes the variable name "Connection" correspond with the turtle representing the point connecting the weight to the drum
  ask Mass [create-link-with Connection]
  ;Establishes a visible connection between the far right side of the drum and the weight
  set amperage 0.37
  ;Sets the winch's current to be 0.37 A
  set time_unit 0.01
  ;Sets the go function to repeat itself every 0.01 seconds
  set angular_speed amperage * Voltage / Mass_held
  ;Sets the value for the angular speed of the winch
  set power Voltage * amperage
  ;sets the value for the power of the winch
end

to go
  ;;;Monitors the motion and status of the winch system;;;
  ask Mass [fd (angular_speed * time_unit * 180 / pi) * Drum_Size / 128
  ;Makes the mass move a little bit each time the go function is called
    show ycor]
    ;Shows the y-coordinate of the mass every time it moves a little bit
  ask Drum [set heading heading - angular_speed]
  ;Makes the drum rotate as it lets the string holding the weight coil
  if [ycor] of Mass > 8 [stop]
  ;Makes the drum stop rotating and the string stop coiling once the weight has been raised to the highest possible height
  if power < 0.5 [ask Drum
  ;Checks if the power of the winch is lower than 0.5 W, calls on the Drum's turtle to do the following actions
    [set color red
    ;Tells the drum to trun red
      ask patch 10 10 [set plabel "Stall!"]]
      ;Tells the patch 10, 10 to show that the drum has stalled and can't move due to lack of power
      stop]
      ;Stops the program's operation
  wait time_unit
  ;Sets a 0.01 second break between each time the go function updates the turtles in the world
end
```

Final Design Sketch



Final Prototype Summary

Our final prototype consists of an FischerTech (FT) motor controlled by a switch, which is meshed to a torque gearbox. In order to keep this gearbox meshed to the first gear, FT structure pieces were shifted slightly to line up with the VEX structure pieces so that they could be attached via screws, while an allen wrench was used as a wedge between the base plate and motor to create pressure between the gearbox and its connecting gear, ensuring that the gears didn't slip. The allen wrench was used because the FT motor did not align with the VEX parts perfectly enough the gearbox and connecting gear were meshed reasonably well. This gearbox is meshed to an FT gear, generating torque, and that FT gear shares an axle with a VEX gear in order to convert to the VEX parts, creating a compound gear train from the gearbox to the drum. In order to keep the speed high enough to lift objects quickly, we used a speed-oriented gear train of two VEX gears. The second VEX gear shares an axle with the drum, which uses a moderate to small circumference in order to be able to wind up a higher amount of weight since our gear system was already speed oriented. Our initial measurements and tests indicate that the winch could lift a weight of 4.3316 N (442 grams) over a distance of .3 m (30 centimeters) within 15.18 seconds, even this weight caused some miniscule slips in the end. The circuit powering the motor for this winch used 7.45 V and .37 A. By multiplying the force and distance together, we found that the total work done was

$$4.3316\ N \cdot .3\ m = 1.29948\ J$$

or about 1.29948 Joules. We divided work by time to find the mechanical output power, which was

$$\frac{1.29948\ J}{15.18\ s} = .0856\ W$$

or about 0.856 watts. The electrical input power was calculated by multiplying current and voltage:

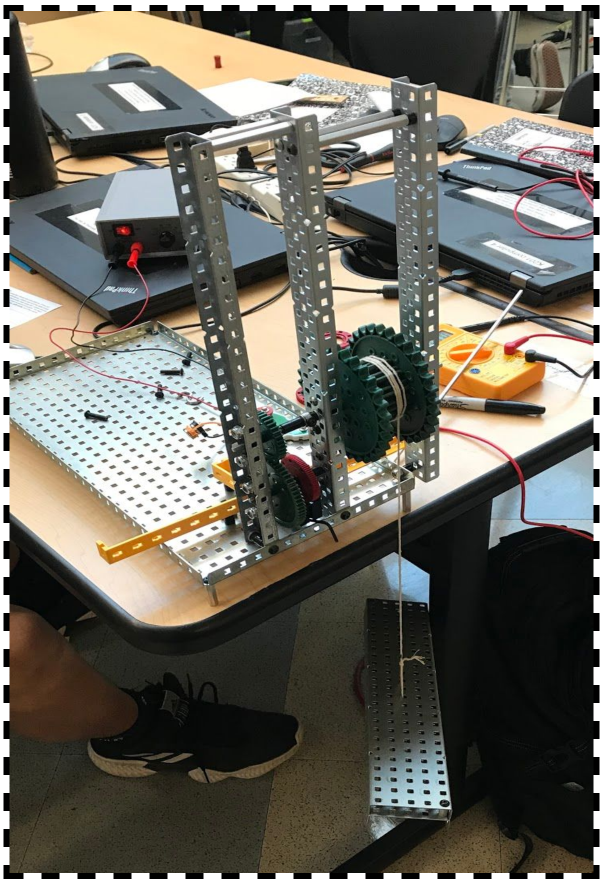
$$7.45\ V \cdot .37\ A = 2.7565\ W$$

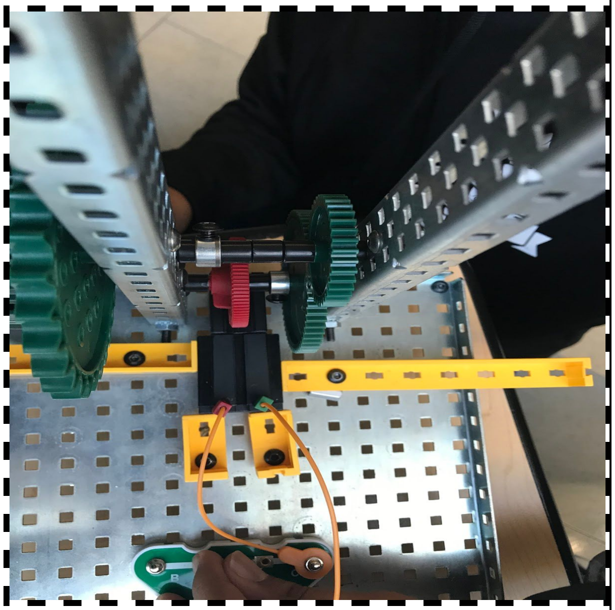
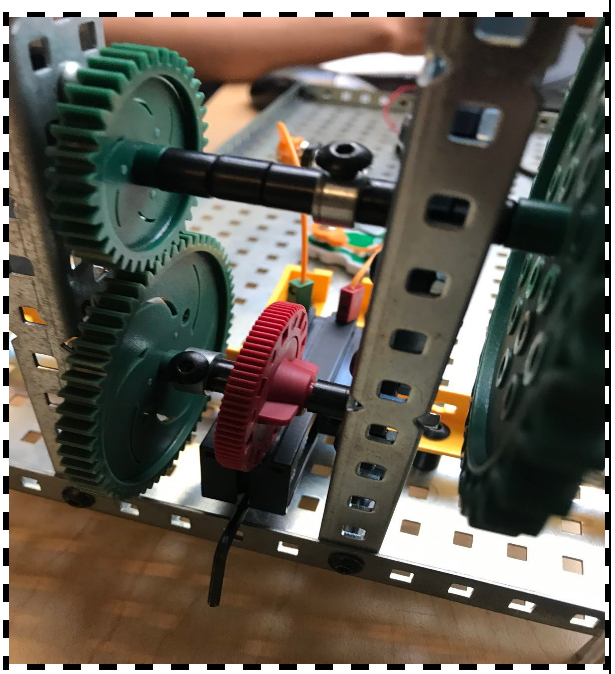
or about 2.7565 watts. Finally, we calculated efficiency by dividing the output power by the input power and converting it into a percentage, as follows:

$$\frac{.0856\ W}{2.7565\ W} \cdot 100\% = 3.2\%$$

Thus, the total efficiency for this prototype was about 3.2%.

Photos of Design and Descriptions

Mechanism:	**Design Description:**
Final Prototype  Signatures: Derek Xu 10/23/19	This is a close up image of our winch system with all parts assembled and functioning. The system was constructed using a compound gear train from the gearbox to the drum, emphasizing speed over the torque.
Motor and Circuits	Attaching the motor was an issue that we encountered because FT parts and VEX parts were not designed to be able to perfectly attach to each other. We had to get creative in our use of the FT parts ability to slide on and off the

<table>
<tr>
<td>
Signatures: Derek Xu 10/23/19</td>
<td>motor. Our circuitry was quite simple, with a gate that allowed us to change the flow of current by flipping the switch and changing the connection point of the ground.</td>
</tr>
<tr>
<td>Compound Gear Train:

Signatures: Derek Xu 10/23/19</td>
<td>In order to transfer energy from the FT system to the VEX system, we utilised a compound gear train to enable a VEX gear and FT gear to lie on the same axle. This allowed us to take the input from the FT gearbox onto an FT gear and transfer it to the VEX simple gear train. The gear ratios on this system are more focused on a better speed value than maximum torque output.</td>
</tr>
</table>

Functionality Analysis

Our prototype was functional, but was not efficient at an optimal level. Besides the very low efficiency (3.2%), there are some technical issues. The motor is a bit loose, and while this was mostly fixed by holding it in place by wedging an allen wrench under it, it does tend to rapidly unwind and strip the drive gear when disturbed too much. The circuitry is also functional, since we used a basic switch circuit as shown in the directions, but the wires need to be rearranged to change the winch's direction, which is a practicality issue. This could be easily fixed if we had another dual gate switch but the lack of this piece caused this restriction of having to change wires in order to reverse the direction of the motor. If scaled to a larger, practical size, this winch would likely be slightly impractical to power, operate, and maintain compared to other designs. However, it does carry a significant amount of weight at a decent speed, and with more time and effort, this design could be easily improved to remove the issues with the circuitry and motor, as well as increase efficiency. The low efficiency of our prototype had many sources. First of all, the gears of the machine likely lost much energy as friction and heat. Not only were there many gears and multiple axles, but the FT drive gear that was used was slightly stripped due to the issue with the loose motor, and the conversion from FT to VEX parts also caused energy to be lost during the transfer. Also, the circuitry factored into the low efficiency. The slipping of the FT gear that was attached to the gearbox and the imperfect alignment of the VEX and FT parts could be seen as some of the main causes of the low efficiency. Adding on to this, the system was set to the highest power setting allowed (7.5 V), which in a larger system could be a lot more power, and the circuit did have some resistance at about 20 ohms, which indicates that energy was lost. Another detail we noticed was that our multimeter measured that our system had about .37A, where we were supposed to get a value closer to .1A. This potential misreading could be due to either a misreading or a broken multimeter. If the actual amperage were to be about .1A, our efficiency would far higher. The fact that many parts were broken and had to be replaced constantly also contributed to this. Finally, the mechanical advantage

(MA) of the system wasn't optimal, and the efficiency would have been higher if the MA was different. Our system was geared a nice middle ground between speed and torque, slightly favoring speed. Ways we could have combated this is by utilising a simple gear train to reduce friction and points of contact, use better gear parts so that we don't have any slips when running the system, and finally try to make the transition between VEX and FT more smooth or just use one type of part in order the avoid the issues caused by the disportionality between the parts that affect system in a negative manner.