Dr. Andrew Rawicz
School of Engineering Science

Simon Fraser University
8888 University Drive
Burnaby, BC
V5A 1S6

Re: ENSC 305/440 Design Specification for a Virtual Reality Bicycle Trainer

Dear Dr. Rawicz,

Enclosed is a document entitled *Design Specification for a Virtual Reality Bicycle Trainer*. This document outlines the technical design specification for the V-Cycle product we are prototyping. The V-Cycle Pro is a bicycle trainer that allows the user to use their own road bicycle and train indoors in a stimulating environment during the off-season.

The design specifications following refer to a previous document *Functional Specification for a Virtual Reality Bicycle Trainer* and outline how we will achieve each of those functional specifications. However they only will outline how we will create our proof-of-concept model. Some of our functional specifications were stated as not implementable in this design iteration and thus will not be discussed in this document in detail.

The V-Cycle team consists of Dan Edmond, Mike Henrey, Lukas-Karim Merhi and Jack Qiao; students from the School of Engineering Science at SFU. If you have any questions, please contact Mike by email at mah3@sfu.ca.

Regards,

Lukas-Karim Merhi
Chief Executive Officer
V-Cycle

Enclosure: *Design Specification for a Virtual Reality Bicycle Trainer*

Daniel Edmond
301027138

Michael Henrey
301026704

Jack Qiao
301025357

Lukas-Karim Merhi
301029870

date	March 11, 2010
contact person	Michael Henrey - mah3@sfu.ca
revision	1
submitted to	Dr. Andrew Rawicz
	Steve Whitmore
	School of Engineering Science
	Simon Fraser University

cycle

Design Specification

Executive summary

The design specification for the V-Cycle Pro provides a set of detailed descriptions for the design and development of our proof-of-concept model. The design specifications in this document are solely for the proof-of-concept model. Therefore, we will only discuss design considerations pertaining to the functional requirements marked A or B, as specified in the document *Functional Specification for the V-Cycle Pro* [1].

The mechanical design considerations for the V-Cycle Pro can be separated into those for the front lifting mechanism, the rear tilting linkage, and the variable resistance unit. The front lifting mechanism consists of a frame built around a linear drive system to raise and lower the front wheel with changes in terrain incline. In order to simplify the design of the motor control system, we have opted to use a DC stepper motor, operating in open loop, with a ball screw connected to the wheel platform on linear slide rails. The rear tilting linkage is built around an existing bike trainer which will securely mount the rear axle of any bike into our device. The bike trainer is mounted onto a passive tilting mechanism, a simple solution to incorporating core stability into the workout. The variable resistance unit will be built around the magnet eddy current braking system of the bike trainer. Repositioning the poles of the magnets and the distance between them with a simple motor system or a purely mechanical system coupled with elevation changes are solutions currently being investigated. These design choices have been implemented with readily available, easy to assemble, and inexpensive parts.

Virtual reality software has been implemented with open source 3D, physics, and sound engines to provide a total immersion experience for the user. A microcontroller will provide the bridge between the software and the user's movement, updating the elevation and resistance from the software by actuating motors and interfacing with sensors to provide feedback to the software for determining current position. Sensors will only be placed on the structure of our system to make set-up simple for all users without the need to attach sensors to their bike. Sensors inputs and data processing by the microcontroller will obtain information on speed with a reed switch tachometer, tilting with a 2D-accelerometer, and turning with a potentiometer on the wheel platform. It will also drive motors for the front lifting mechanism and variable resistance unit and communicate with the virtual reality software. A microcontroller has been selected to meet these above requirements.

Good progress has been made in all aspects of the subsystem design and implementation with the exception of the variable resistance unit, which may see some delays in completion. Otherwise, the V-Cycle team has detailed individual and system test plans and has planned for an adequate integration period to ensure that our system functions to the best of its abilities. We will meet the targeted delivery date of Thursday, April 15 for the proof-of-concept model of the V-Cycle Pro.

Table of Contents

Executive summary ii
List of Figures v
Glossary vi
1. Introduction 1
1.1. Scope 1
1.2. Intended Audience 1
2. System Specifications 2
3. Overall System Design 3
3.1. General Design 3
3.2. High level system design 4
3.3. Mechanical Design 4
3.4. Actuators 5
3.5. Sensors 6
3.6. Electronics System 7
3.7. Power System 8
3.8. Safety Systems 8
4. Front Lifting Mechanism 10
4.1. Mechanical Design 10
4.1.1. Linear Motion and Ball Screw System 10
4.1.2. Anti Gravity Device 11
4.1.3. Wheel Platform 11
4.1.4. Other Designs Considered 12
4.2. Safety Considerations 13
5. Side Tilting Linkage 14
5.1. Mechanical Design 14
5.1.1. Passive Tilting Bushings 14
5.1.2. Other Designs Considered 15
5.2. Safety Considerations 15
6. Variable Resistance Device 17
6.1. Mechanical Design 17
6.2. Heat Dissipation Considerations 18
7. Electrical Hardware and Control System 19
7.1. Overall System Design 19
7.1.1. List of Processes 19
7.2. Motor Control System Design 19
7.2.1. Motor Control Hardware 19
7.2.2. Motor Control Software 20
7.3. Input Signals System Design 21
7.3.1. Table of Input Signals 21
7.3.2. Input Signal Conditioning 22

7.3.3. Sampling Frequencies and Interrupts 22
7.4. Communication System Design 23
7.4.1. Communication Overview 23
7.4.2. Communication Protocol 23
7.5. Safety System Design 24
8. User Interface and Visual Display 24
8.1. Software Design 24
8.1.1. Visual Interface 24
8.1.2. List of Processes / threads 25
8.1.3. Status of Development 25
9. System Test Plan 28
9.1. Unit Tests 28
9.1.1. Front Lifting Mechanism 28
9.1.2. Side Tilting Mechanism 28
9.1.3. Sensors 28
9.1.4. Motors 28
9.1.5. Variable resistance device 29
9.2. System Test 29
9.2.1. Normal Operation 29
10. Conclusion 30
11. References 31

List of Figures

Figure 1: V-Cycle Pro System Overview 3
Figure 2: High Level V-cycle System 4
Figure 3: V-Cycle pro rear tilting system 5
Figure 4: Front Lifting Mechanism 10
Figure 5: Front Lifting Platform 12
Figure 6: Side Tilting Linkage 14
Figure 7: Flowchart for motor control software 20
Figure 8: State Machine 21
Figure 9: Serial Communication State Machine 24
Figure 10: Dynamic Lighting and Atmospheric System 26
Figure 11: Terrain heightmap generation, colour and normal textures, forest generation 26
Figure 12: Terrain physics collision (tested with collision cylinders) 27

List of Tables

Table 1: Analog and Digital Inputs 21

Glossary

CM – Center of Mass

Pinch Point – A place where the hand (or even the entire body) can be crushed between two moving objects, or between a moving object and a stationary one

Standard road bike:

610mm - 720mm radius wheel rims
Conventional road tires with a smooth tread pattern, inflated to 80-100PSI
Mass of less than 15Kg
Max seat height 110 cm
Wheelbase less than 110cm
Quick release rear axle
Bike is in acceptable standard of upkeep (wheels trued, operational brakes, drive train oiled calibrated and greased, no stress fractures on frame or crankset, etc.)

Average user :

Body mass is between the 5th percentile of females and the 95th percentile of males all aged 20-39 in Canada [1]
Height is between the 5th percentile of females and the 95th percentile of males all aged 20-39 in Canada [1]
Familiar with simple bike repair and bike tools
Owns their own standard road bike and preforms regular basic maintenance
Owns and fully utilizes a helmet

Inexperienced user:

Has not used the V-Cycle bike trainer for more than 10 hours

Experienced user:

Has used the V-Cycle bike trainer for at least 10 hours

Standard operating environment:

15 – 25 degrees Celsius
Indoors
Stable, non-slip floor for using device

Critical error condition:

A condition where continued use of the device will result in physical harm to the user or machine

Non-critical error condition:

A condition where the device detects a configuration error or invalid information, but continued use will not cause any error conditions

For bike terminology, please refer to Sheldon Brown's Online Resource:

http://www.sheldonbrown.com/

1. Introduction

The Virtual Reality Bicycle Trainer (V-Cycle Pro) is a bicycle trainer for use indoors in the off-season. Incorporating physical and visual feedback elements into the system provides the user with an exciting alternative to conventional exercise bicycles. With our device, both amateur and professional riders are able to enjoy training during the cold winter months, and obtain a realistic exercise routine while doing so. The requirements for the Bicycle Trainer proof-of-concept are described in the following specification.

1.1. Scope

This document outlines how we aim to achieve our functional specifications set out in a previous document (*Functional Specification for a Virtual Reality Bicycle Trainer)*. However, our proof-of-concept model will differ slightly from a production model, and thus this specification will only focus on requirements for the proof-of-concept. Specifically, we will outline how we intend to meet or exceed specifications marked A or B, but not provide implementations details for Production Only specifications marked C. At the end of this document are appendices providing further technical drawings to clarify any implementation details.

1.2. Intended Audience

The audience for this document is the entire V-Cycle team. The document will be used as a benchmark to ensure that the designs are capable of satisfying all of the requirements. During integration this specification will ensure that sub-systems are compatible with each other. During testing, the test plan outlined in this document will be used.

2. System Specifications

The V-Cycle Pro device will allow the user to experience realistic indoor cycling. To achieve this, the user shall place their own road bike in our trainer. They can see where they are virtually cycling on an LCD display. As they traverse hills, the front of the bike tilts up and down. In addition, the bike will allow tilting side to side, with a restorative spring force which balances the bike in the vertical position. The resistance of the pedaling will vary as a function of velocity (simulating wind resistance) and angle of attack (simulating climbing and descending hills).

3. Overall System Design

This section provides an overall look at our system. Design specifications that are related to every aspect of the V-Cycle product are discussed here. More specific details, where appropriate, are provided in future sections.

3.1. General Design

The V-Cycle pro system is shown in Figure 1. The user's bike is mounted securely onto the rear platform into an existing bike trainer. The front wheel will rest on the front lifting section, which will raise and lower with elevation changes. The front and back systems are mechanically unconnected so that different sized bikes can be accommodated. The microcontroller and power supply will be housed safely in the front lifting unit, and cables will connect electronics on the rear section.

Figure 1: V-Cycle Pro System Overview

3.2. High level system design

A high level diagram of our system is shown below. The data flow is described from left to right. Note that the microcontroller and computer communicate with each other. This allows the computer to use the tilting measurement generated by the 2D accelerometer to steer the virtual display, and the microcontroller knows what lift the front of the bike should have from the virtual world.

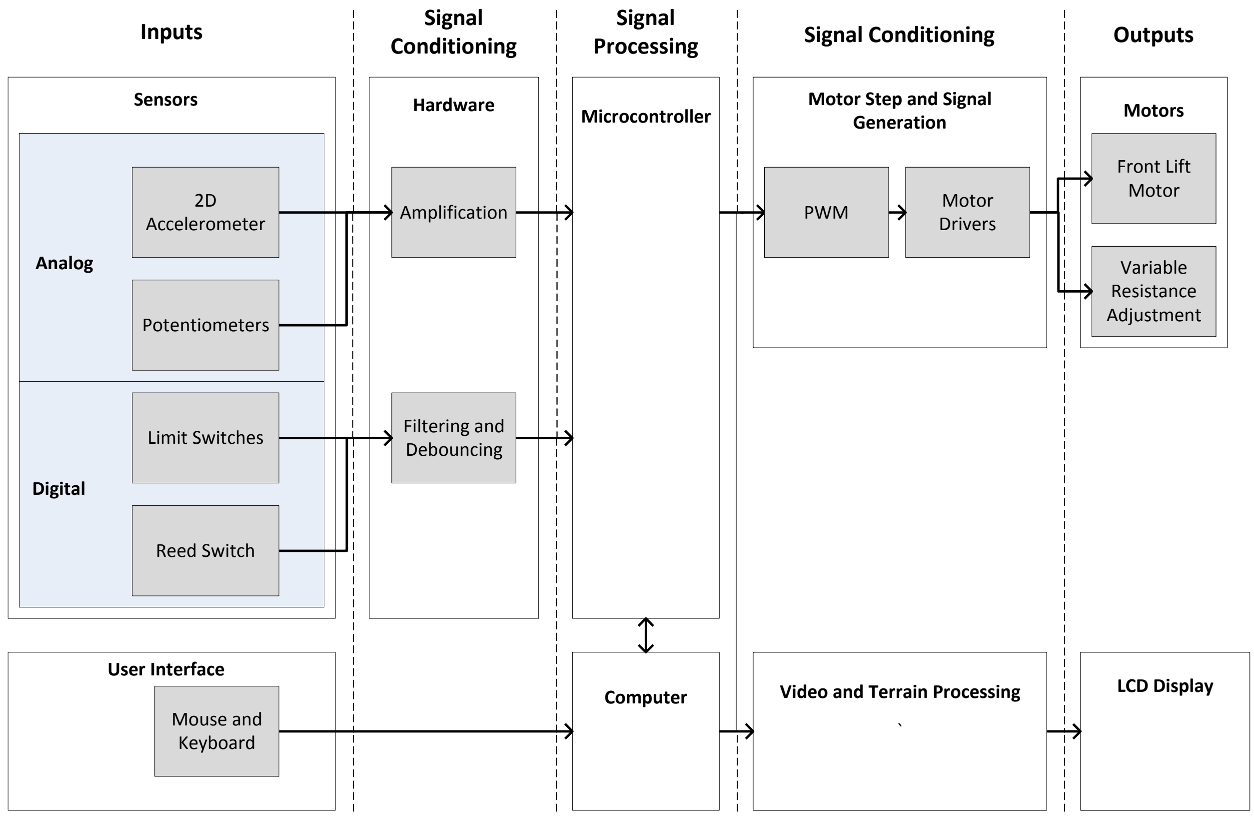

Figure 2: High Level V-cycle System

3.3. Mechanical Design

The mechanical design is divided into 2 major sections. The front section consists of a platform under the front wheel which raises and lowers on a set of linear glides. A ball screw mechanism is responsible for generating the appropriate forces. The rear section sits under the existing bicycle trainer and allows tilting side to side. The tilting action is made possible by a set of skateboard bushings acting in a similar fashion to skate bushings in skateboard trucks. The bushings compress as the user leans side to side and act as restorative springs. The rear section is shown in detail in Figure 3.

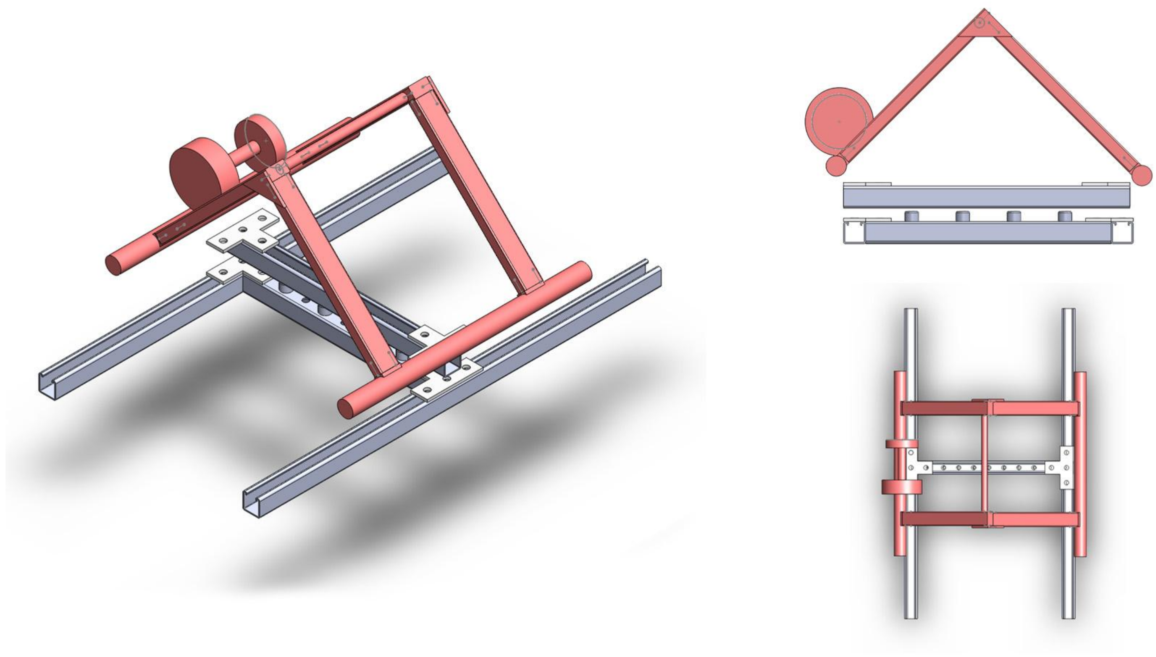

Figure 3: V-Cycle pro rear tilting system

Both structures have been designed to withstand forces generated by the average user, with a safety factor considered. The frame is formed from 1-5/8” steel struts, with a movable spring-nut system that allows us some flexibility in final design parameters.

3.4. Actuators

We opted to use a stepper motor with a ball screw and linear glide to lift the front platform. This is due to the low cost and high torque of such a motor. The stepper motor has the added advantage that its position can be known quite precisely, and is often used in open-loop configurations.

The primary designs considered were:

- 4 bar linkage and gear train
- 4 bar linkage and ball screw
- Ball screw and linear glide

The ball screw with linear glides was chosen for a number of reasons. One of our group members has experience building CNC machines, and is quite familiar with the calculations required for such a setup. The use of linear glides also allowed for the simplest structure to lift the required load.

The main disadvantage of a linear track is that the bike is fixed at the rear so the path of the front wheel is not linear. To overcome this, we are allowing the front wheel to move forward and backward along the platform, but constraining it within a track. While this problem may have still occurred with a 4-bar linkage, we could have designed the linkage output path in a way that mimics the natural motion of the front wheel.

After choosing the ball screw and linear glide components, we preformed calculations to decide which motor to buy. We wish to have a quick acceleration so that our device can respond well to changes in slope in the virtual terrain. The system must also be capable of lifting at least 65Kg over the front wheel. However we wished to keep costs down, and if possible operate the system in an open loop.

The systems considered were:

- DC stepper motor
- DC servo motor
- DC brush motor and feedback system
- Hydraulic or pneumatic system

In the end we choose the DC stepper motor. Again, one of our members had experience with this motor type from CNC machine design. Additionally, the stepper motor requires less maintenance than a brushed motor, and has a large holding torque. The servo motor was also a strong candidate; however the cost was significantly higher than that of a stepper motor. While the hydraulic or pneumatic system was considered seriously at the start, none of our members were confident in their ability to implement such a system, and the safety risks of a high pressure/temperature system in a consumer product were inhibitive.

The main limitation of the stepper motor is the need for more complicated programming and a motor driver. Our solutions to the stepper motor control are discussed further in the appropriate sections; however we found a sufficiently powered motor driver and are implementing a stack-based step system for controlling the stepper motor.

3.5. Sensors

We require sensors for a number of conditions. A 2D accelerometer is used to measure tilt angle, so that we know how much the user is leaning to each side. Limit switches help ensure the front platform does not exceed its travel limits. A magnet and reed switch pair is mounted on the flywheel to generate a pulse for a specified distance of travel. A rotary potentiometer will be used to provide feedback as to the turning angle of the front wheel.

One of the requirements of our device is that the user shall be able to remove their bike from the trainer and use it on the road directly. To achieve this, we have decided that there shall not be any sensors mounted on the bicycle itself. As a result, the sensor placement is a critical part of our design.

We have chosen to mount the accelerometer on the rear support structure. This should be quite sensitive to tilts in both possible directions. In addition, the signals do not have to travel far from the accelerometer to the microcontroller which should help preserve signal integrity.

Limit switches are mounted at each end of the linear track. These prevent the system from overrunning its tracks and causing damage to either the system or user.

The reed switch is mounted on the frame, and the magnet is mounted directly on the trainer's flywheel. While this generates many pulses, especially as the user is pedaling quite fast, the user is not forced to make any modifications to their bike to use our system. In addition, compared to the traditional location of such a sensor pair which is on the bike wheel itself, our system does not need to be calibrated for individual wheel and tire sizes. The main disadvantage is that we must provide extra filtering and read the pulses more often because the flywheel diameter is significantly smaller than the wheel diameter.

The rotary potentiometer is mounted under the front lifting platform. The platform is designed in such a way that the bike wheel is able to rotate, and as it does so the angle of rotation is measured with the potentiometer. Again this system does not require the user to make any modifications or connect anything to their own bike – the system is quite self-contained.

We also have considered the possibility of adding feedback directly to our stepper motor. The options considered for this were:

- Optical displacement sensor
- Incremental encoder
- Flexible stretch sensor

In the end we opted to choose a suitable sized stepper motor and operate it in open loop with a set of limit switches at each end for safety. The optical displacement sensors were either expensive (laser) or not appropriate for the range we were looking at (IR). Incremental encoders are also quite expensive and require additional processing. The flexible stretch sensor does not offer sufficient resolution compared to the step size that our stepper motor offers. Therefore we would not be able to detect missing steps, only gross errors that the limit switches will already detect. Stepper motors are often used in open loop applications, provided the motor torque is sufficient, so we have decided to operate in this mode.

3.6. Electronics System

A microcontroller is at the center of our control system. This microcontroller collects data from the sensors, and outputs to the motors. In addition it is responsible for communicating with a computer, to synchronize the output lift with the display on the user's screen and tell the computer how fast the user is moving.

When selecting a microcontroller a number of options were considered. The primary decision was between:

- Arduino
- PIC 18F1320

While both had sufficient on board features and I/O pins, the Arduino had a number of advantages. Being an open source device, and often employed in hobby projects, much support is available on-line from code samples to hardware suggestions. In addition, hardware shields and software libraries available for the Arduino allows for easy communication with a PC which was a big concern. With the PIC, this would have been much more difficult. Finally the cost of the PIC was projected at being higher, when the costs of a programmer and designing a PCB were included. The Arduino also boasts integrated PWM outputs with maximal configurability, which will greatly reduce the complexity of the motor driver code.

3.7. Power System

According to our functional specification, two separate power supplies are acceptable for the proof-of-concept model, though in a production model having a single power supply and wall plug in is desirable. Currently we have the Arduino and sensors powered from a single 9V DC supply, while the motor has a separate 48V DC power supply. Currently our system requires less than 200W, which is suitable for a single household circuit (household circuits can service about 1500W).

A second requirement of our power system is that it must be safe. All of the power supplies and power components are kept in a metal box, which is attached to the V-Cycle trainer. Suitable warning signs will be posted. The power cables are of suitable type for the application (14 AWG, flexible) and bushings and connectors are used to prevent any abrasion near box edges.

Fuses and switches are accessible from the outside of the box. This means the user is able to turn the device mains on and off and replace the fuses without any risk of injury.

3.8. Safety Systems

Safety systems are an important requirement for such a system. We have identified the main potential hazards to the user/system as:

- Tipping/instability
- Electrical shock
- Pinching
- Linear parts exceeding their limits

To address the tipping issue, the appropriate base width has been calculated and a safety factor has been added to ensure the user's CM is always inside the base footprint, even as they are leaning into a turn.

To prevent any chance of electrical shock, the power components are located in a suitable metal enclosure and proper wiring practices are followed. Warning notes are posted, notifying the user of potentially dangerous areas of the device.

Pinching may occur when two or more parts are moved relative to each other and there is the potential for the human body to be crushed between them. Because our device has moving parts, we must design it to minimize any risk of pinching. The main danger of pinching occurs near the ball screw mechanism, so the ball screw shall be covered with a telescoping guard. Pinching can also occur where the flywheel of the trainer meets the bike tire. Adequate warnings about these areas shall be placed on the device and in the user manual.

When linear parts exceed their limits, there is the potential for damage to be caused to the system or user. To prevent this from happening, limit switches are placed near the top and bottom of the linear tracks to indicate when the limits are reached.

4. Front Lifting Mechanism

4.1. Mechanical Design

4.1.1. Linear Motion and Ball Screw System

The linear motion system and actuator comprises the main components of the lifting mechanism. The linear motion system is designed to move in a single linear direction – ie. it transfers all non-linear forces to the support structure, and allows the actuator system to exert force only in one direction. Because the load is several inches away from the actuator system (the front wheel must have at least 8" of room for natural motion), the linear motion system must also handle dynamic moments of up to 320 Nm, which translates to a force of 5kN in the radial direction. The linear motion system as described above is pictured in Figure 4.

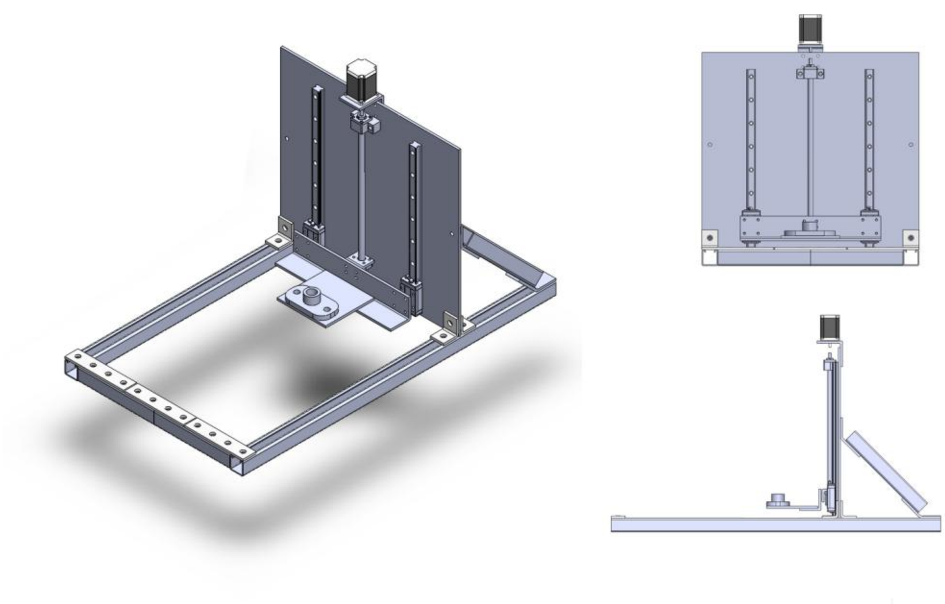

Figure 4: Front Lifting Mechanism

For the sake of simplicity, we have decided to purchase an off-the-shelf industrial linear motion system, the THK SR25W linear motion guides. This system allows for over 10kN of dynamic load in the radial direction and is more than adequate for our application, with a generous margin of safety.

The linear actuator is also off-the-shelf: A Nema 23 Stepper motor from Keling Motors, combined with a THK KX Ground Ballscrew to convert motor torque to linear force.

The motor must overcome the inertia of the bike and rider at an acceleration of at least 1g, which corresponds to 0.72Nm of motor torque. Due to the nature of the stepper motor, its maximum holding torque decreases as rotational speed increases. At a torque of 0.72Nm, the motor can drive the lifter at 7.8 cm/s, adequate for most terrain slope changes.

4.1.2. Anti Gravity Device

Without a passive system for countering gravity, the linear actuator would have to fight gravity when going up, as well as when coming to a stop when going down. This necessitates a motor with a much higher torque, or alternatively a passive system for countering the force of gravity.

The proposed design uses a shock cord coiled around 16 pulleys for 16 feet of pre-extension (that is, the cord will be extended to 16 feet at the top of linear travel, at 17 at the bottom). This ensures that the elastic force experienced at the top of travel is very close to the force experienced at the bottom.

Using a 9 foot unextended cord with 100% elasticity (typical of shock cords) that has a max load of approximately 65kg, the weight of the rider should be largely nullified.

4.1.3. Wheel Platform

For smooth and natural movement, the rider must be able to freely turn the handlebars. This motion is integral to keeping balance when riding a street bike, and must be simulated in order to create an immersive virtual environment.

The front wheel will rest on a rotating platform with a passive bearing system as shown in Figure 5. This allows the bike to turn freely, similar to a normal street bike.

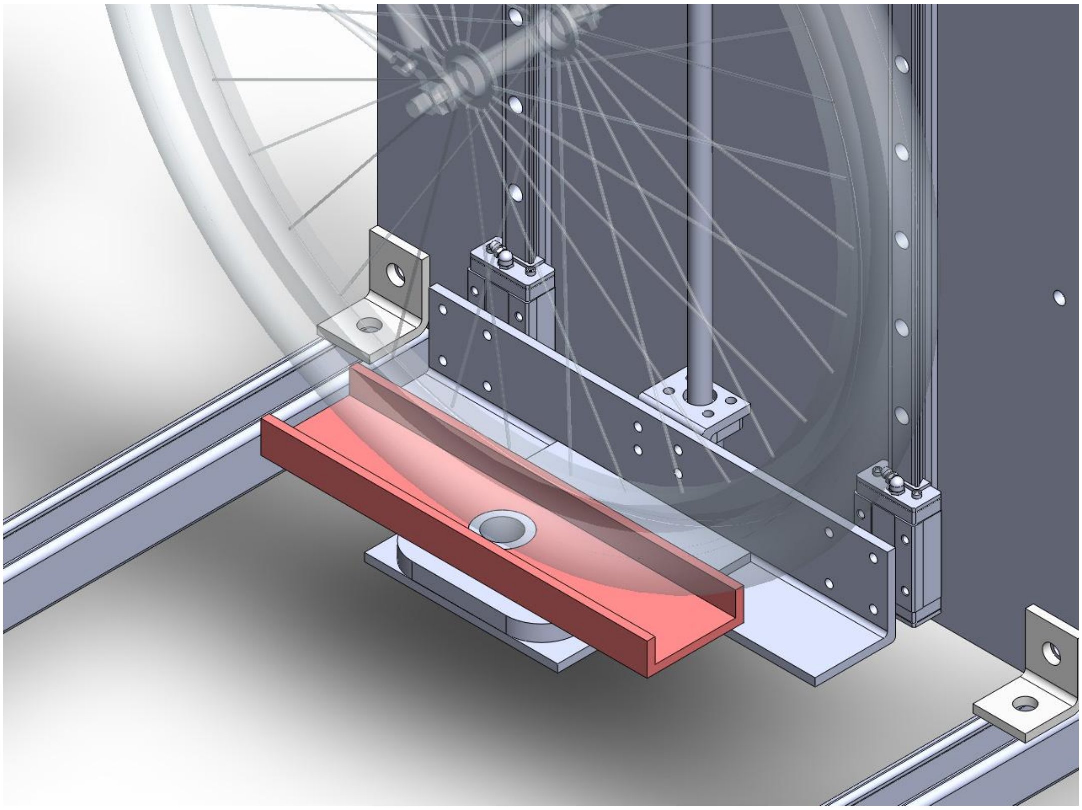

Figure 5: Front Lifting Platform

4.1.4. Other Designs Considered

Alternative solutions were considered for several components of the system. These are discussed by category in the following section.

4.1.4.1. Motor

As linear displacement is extremely important for both performance and safety, a servo motor is much better suited for our application than a stepper. A servo motor has built-in feedback and a simpler control scheme, as well as an overload mode suited for high-acceleration-high-frequency road texture. However, a servo motor with comparable torque to a stepper is much more expensive, both in terms of the motor and drive electronics.

4.1.4.2. Motion System

Instead of a ballscrew, several possible drive systems were considered, including belt-drive, chain-drive, and rack-and-pinion. All these systems are mechanically complex, requiring multiple pulleys to attain the required gear ratio, and in the case of rack-and-pinion requires spring tensioners. The most elegant and direct system is using a screw, coupled directly to the shaft of the motor. Acme screws were also considered instead of ballscrews, however the lower efficiency of an acme screw (50% vs 90% for a ballscrew) presents problems for long-term durability and motor torque requirements.

4.1.4.3. Anti-Gravity Device

Springs and counterweights were considered for this purpose. The main problem with springs is that they only work linearly – a spring with similar properties to an shock cord design will need to extend over 5 feet above the device. If a counter-weight is used, the effective inertial load of the motor will be doubled, which will limit the maximum acceleration of the lifting system.

4.2. Safety Considerations

The main safety hazards of the lifting mechanism include:

- Risk of electrical shock
- Physical pinch points
- Danger from a "runaway" motor

A grounded metallic outer covering for all electronics and mechanical assemblies will be implemented. This will act as a faraday cage, preventing electrical shock as well as reducing any RF interference. A physical cover can also hide pinch points and protect the user mechanically.

Switches will be used to control the behavior of the system. Limit switches at the ends of travel will prevents the motor from overrunning the track and causing damage to the user or machinery. An emergency stop button will also be implemented. This immediately cuts power to the system in case of an emergency.

5. Side Tilting Linkage

5.1. Mechanical Design

This aspect of our system is designed to allow the user to lean side to side as though they were navigating a corner. In addition, this freedom of motion requires the user to engage their core muscles in the activity and have a more realistic indoor cycling experience. The system must also measure the amount of tilt for use by the computer in determining what path the user is taking in the virtual world. Finally the tilting linkage would be mounted under the rear trainer section, and must raise the rear wheel to the neutral position, so that the front lifting platform can tilt the bike up and down. The side tilting linkage is shown in Figure 6.

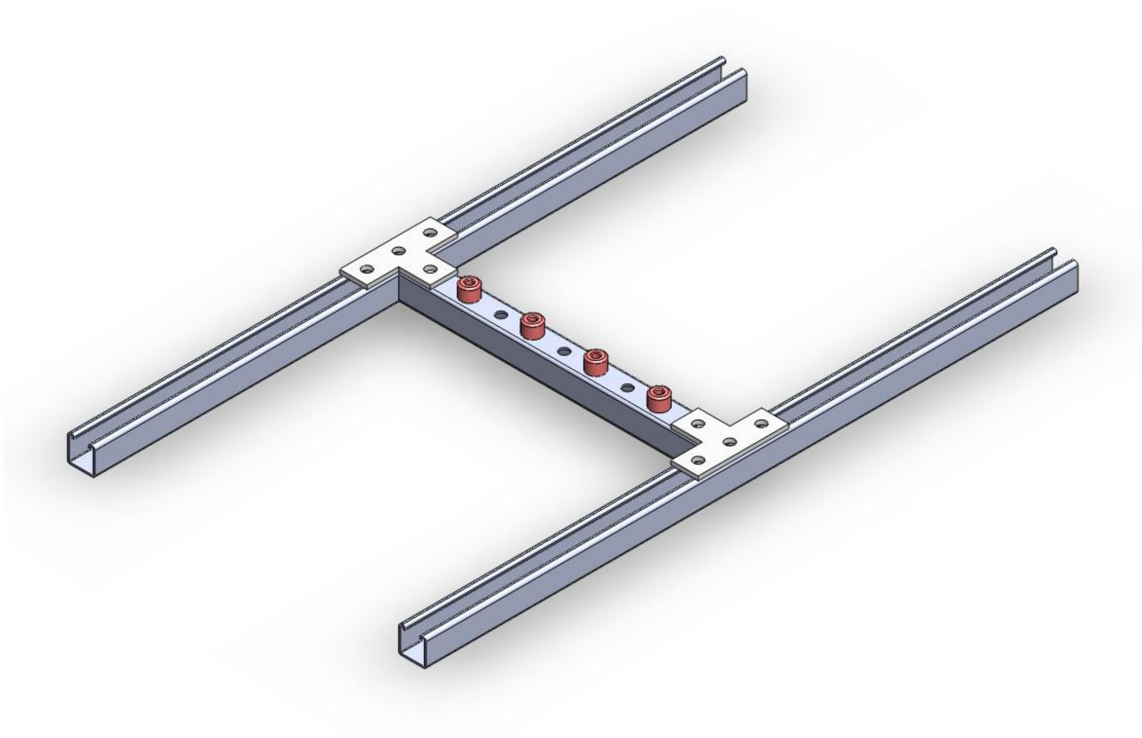

Figure 6: Side Tilting Linkage

5.1.1. Passive Tilting Bushings

The design we settled similar to the design of a skateboard truck. It consists of 4 kingpin and skate bushing assemblies, mounted between a stable base frame and the rear section of the bike trainer. The design limits the tilt angle to 12.6 degrees, which ensures that the average user will be in no danger of tipping. This system was quite simple and low cost to implement; the main costs were in the structure which was required anyway to lift the trainer to the proper height.

To increase the spring force, we are considering the use of surgical tubing or shock cords and a turnbuckle. After testing, we will determine if this is a necessary addition to our device, in order to provide realistic forces on the bike while turning.

A 2D accelerometer mounted on the frame measures the actual tilt and uses this information to help the computer decide where the user is trying to go.

The skate bushings provide a realistic effect for the user as they act as restorative springs. As the user tilts further to the side, the bushings become more compressed and the spring force becomes greater. This is similar to the forces that a user feels outdoors as they are leaning into a corner or riding intensely causing the bike to tip slightly from side-to-side.

5.1.2. Other Designs Considered

In addition to this design, other designs were considered. The most notable of these were:

- Springs on back axle
- Shocks/struts under the trainer
- Leaf springs under the trainer

Springs on the back axle would have been connected between the axle and the trainer. This would allow the wheel to tilt side to side, but provide a restorative force upon the bike frame. One main disadvantage is that the contact point of the wheel and the trainer flywheel would change. In addition, there is not much room for springs, so the spring force would have to be quite strong.

Shocks or struts under the trainer was a potential solution, however it was an expensive option. One shock or strut would be required under each corner of the base, so that they could move independently depending on how the user was shifting his or her weight. Many shocks are designed for bicycles so the spring constants required were in the correct region, and they have the advantage of also providing some damping effect too. The cost of this system was the main drawback.

Leaf springs under the trainer was also a strong candidate, however the system would be costly to implement. Leaf springs require a significant amount of hardware (hangers etc.) and are most commonly available for cars, making procurement of leaf springs with the proper spring constant difficult.

5.2. Safety Considerations

Failure modes were considered for the system, and the following hazards were identified:

- Tipping if the user leans too far
- Pinching when the bike tilts

Because the design naturally limits the tipping to 12.6 degrees (with a hard stop) and the largest of the expected users will not tip until at least 15.3 degrees, we have a small safety factor. The hard stop is adjustable, so we can change the values during testing if it proves to be unsafe.

Pinching is a dangerous condition because of the hard stops. It is possible for a bystander to inadvertently become pinched between the moving bike and the stationary structure as the bike is tilting. Adequate labeling will be provided, and a guard is being considered for this location.

6. Variable Resistance Device

6.1. Mechanical Design

This aspect of our device shall allow the pedaling resistance to vary based on what the user is doing in the virtual environment. For example, if the user is pedaling up a virtual hill, the resistance would increase, and while they are going downhill the resistance should decrease.

Currently our trainer has the capability to adjust the resistance level through a mechanical lever. The resistance unit consists of an eddy current brake, with a metal disk between two sets of permanent magnets. When the magnets are aligned, the flux through the disk is a maximum, and the resistance is greatest. When the lever is moved so the magnets are no longer aligned (offset position), the flux is reduced and the resistance is lower.

6.1.1. Lever Coupled to Front Platform

Our preferred method of varying the resistance automatically is to couple the adjustment lever directly to the front platform. To do this we will employ Bowden Cable (i.e. the same cable used in bicycle brakes) to link the lever to the platform. As the platform lowers, the lever will be pulled by the cable and the magnets will be rotated into the offset position and low resistance is felt. As the platform raises, the magnets will naturally return to the aligned position (via attractive magnetic forces), taking up any slack in the cable.

This design was selected because of its reliability, simplicity and low cost. The use of cables as control lines is well known in industry, and is even employed in mission critical applications like small airplane control. This simple linkage is lightweight, and requires very low cost parts (cable, pulley and connectors).

6.1.2. Other Designs Considered

Aside from our preferred design, a number of other designs were considered for controlling the resistance level. The most notable of these are discussed below:

- Generator and electronic load
- Ball screw, motor and separating magnets linearly

Using a generator and electronic load provides a lot of flexibility to our design. We would couple the shaft to a generator shaft and then use the back EMF to power an electronic resistance. By varying the electronic resistance (i.e. through a power transformer), we could change the mechanical resistance that the rider feels. With an electronically controlled load, we can vary the forces the user feels very accurately. Also, with a battery charging system, we could use the energy generated to power components such as our microcontroller. However this system is employed on a number of exercise equipment machines and according to our research it often fails. In addition, the use of a generator and power circuitry increases costs significantly.

Another way of varying the magnetic flux through the metal disk would be to control the separation of the sets of magnets. To do so, we would use a ball screw and stepper motor to accurately position one set of magnets with respect to the other. Again, this system has a high cost associated with it, and increased complexity because of the extra parts.

6.2. Heat Dissipation Considerations

One of the biggest problems with such a system is that the entire user's energy generated is transformed into thermal energy. This means that we need to dissipate this energy into the environment. A cyclist is able to produce in the range of 100 to 500 Watts, which is similar to the energy consumption of a computer.[2] The resistance unit has a built in fan, which is able to keep the metal parts cool. Using the current cooling functionality, we do not anticipate high temperatures being a problem.

7. Electrical Hardware and Control System

7.1. Overall System Design

The electronic hardware, microcontroller and specific electric circuitry is designed to continuously monitor wheel speed, tilting coordinates, and turning, and send the data to the virtual reality software. It will also receive elevation and resistance information from our virtual reality software and actuate the motor for the elevation mechanism and variable resistance unit accordingly. Additionally, the microcontroller shall be able to run without computer software in standalone mode.

7.1.1. List of Processes

The following processes will be performed by the microcontroller:

- Serial Communication
  - Send sensor data
  - Receive game data
- Stepper Motor Control
  - Update stepping stacks with new data
  - Modify PWM output controlling stepper motor steps based on acceleration profile
- Poll Analog Sensors
  - Read potentiometer
  - Read accelerometer and process data
- Reed Switch tachometer interrupt service routine
- Safety limit switch interrupt service routine

7.2. Motor Control System Design

A compact and robust algorithm was developed to drive the stepper motor. Built in counter/timers and low level registers in the Arduino microcontroller was utilized to output pulses (waveforms). The code was designed for maximum configurability, so it could be integrated easily and efficiently with other code and work for a range of different loads (bike and an average user). The stepper code as it is called required intricate understanding of stepper motors and their accelerations curves. These were emulated with the use of multiple stacks. One of the most complex dimensions was creating a state machine that would keep track of the direction of the motor and execute steps accordingly.

7.2.1. Motor Control Hardware

The motor control hardware consists of the micro controller output steps and direction to the motor Gecko motor driver G251.

7.2.2. Motor Control Software

The flowchart in Figure 1 demonstrates how the motor control software works. New Data is incoming stepping data from the 3D software when there is an elevation change in the terrain.

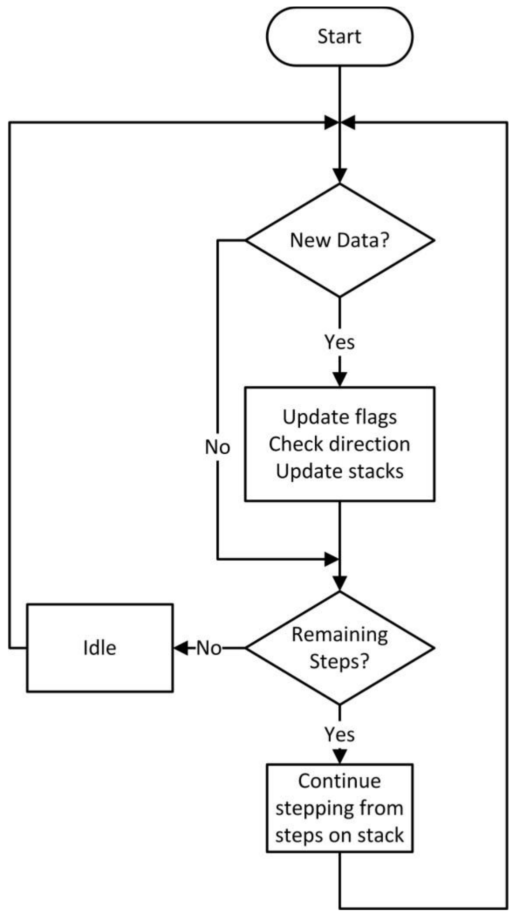

Figure 7: Flowchart for motor control software

The state machine in Figure 2 demonstrates the different states with which the front lifting motor can be in when V-Cycle is being used.

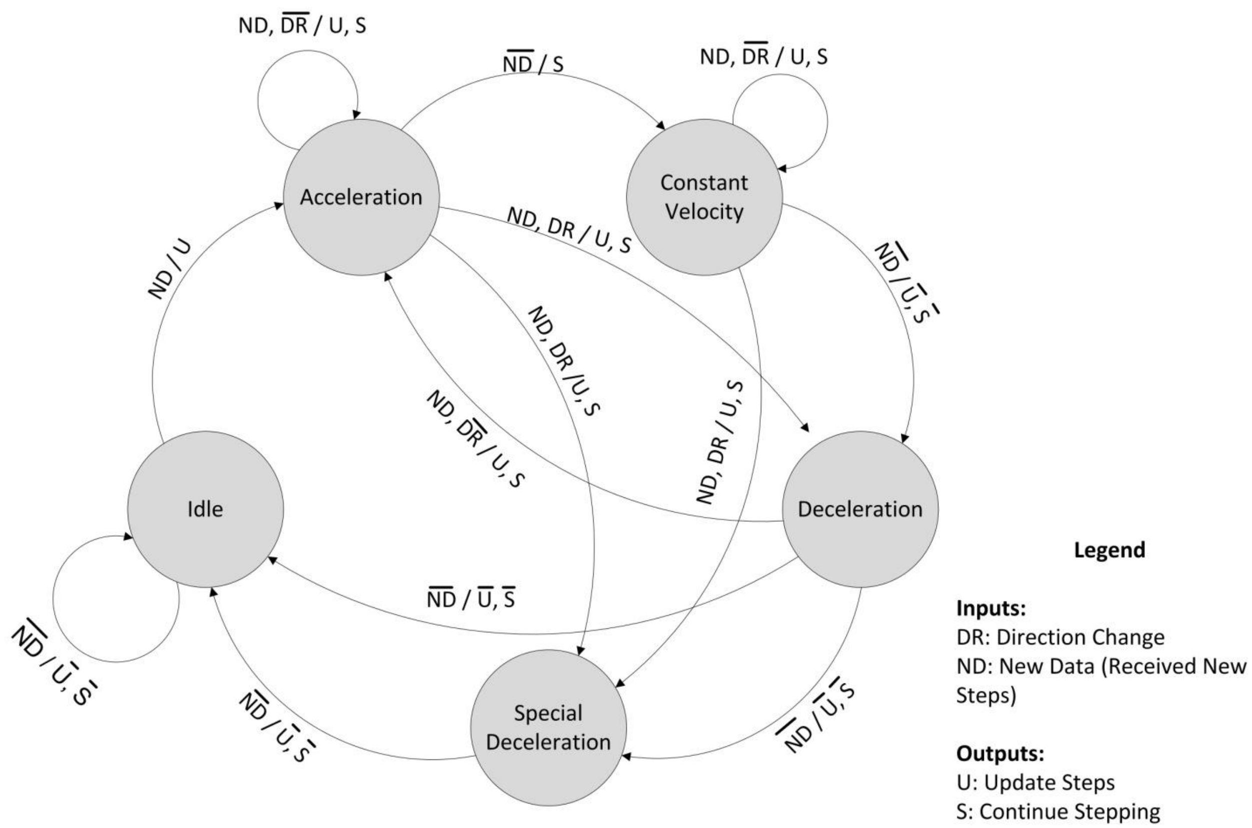

Figure 8: State Machine

7.3. Input Signals System Design

The signals from the various sensors, the user interface, and the motor drivers (for current sensing) need to be modified before they can be processed by the signal processing unit.

7.3.1. Table of Input Signals

Table 1: Analog and Digital Inputs

Analog Inputs		
Name	Number (Quantity)	Description
Potentiometer	1	Used for detecting turning

Accelerometer	1	Used for detecting tilting
Serial Input	1	Receive data from computer software such as steps data, and variable resistance data
Digital Inputsernal		
Name	Number (Quantity)	Description
Limit Switches	Multiple 3	Used as a safety switch to disable the system and calibration purposes due to error accumulation from incorrect motor steps. They are external interrupt signals.
Reed Switch*	1	Debounced and an external interrupt signal.

7.3.2. Input Signal Conditioning

Amplification

The signals from the analog sensors need to be amplified in order for the signal to be processed properly. We will use a custom amplifier circuit to amplify the sensor signals where required. This will likely be required for the accelerometer, which will exhibit only small changes in voltage for the very small accelerations of the tilting platform.

Filtering

Filtering out noise will be of particular importance when taking analog voltage reading from the accelerometer, which will accumulate error in the tilting position processed by the Arduino.

Digital Debouncing

The signal from the reed switch tachometer must be debounced by an analog circuit or by software since it is connected to an external interrupt pin on the microcontroller. Otherwise, more revolutions will be counted by the microcontroller than should be.

7.3.3. Sampling Frequencies and Interrupts

The reed switch signal and safety limit switch signal are external interrupt signals that will call interrupt service routines in the microcontroller.

Other analog/digital signals from sensor will be polled by the microcontroller software. The polling rate will be equal to the rate of sending user packet data from the microcontroller to the virtual reality software, which will be roughly equal to the frame rate of the virtual reality software. If it is seen that polling at a higher rate and filtering the data yields smoother performance, this option will be considered.

7.4. Communication System Design

Reliable and fast communication from the computer running the virtual reality software and the microcontroller, which will feedback data used to determine the user`s position, is essential for our system to function in real time. All design considerations of the serial communication protocol emphasize speed and data integrity.

7.4.1. Communication Overview

The Arduino microcontroller serial communication libraries allows for the sending serial data over the USB interface at set baud rate. This allows us to focus our efforts on the development of a higher level packet based communication protocol so that data can be sent in packets at the refresh rate that we set, and so that we can implement acknowledgement of receipt.

Packets from the computer software contain the following data:

- Start of packet byte (character `G`)
- Change in elevation (in number of steps, or scaled number of steps)
- Step direction
- Resistance quantity

Packets from the computer software contain the following data:

- Start of packet byte (character `D`)
- Number of reed switch interrupts since last packet sent
- Tilting position
- Turning potentiometer reading

Error and Ack will be sent with characters ‘E’ and ‘A’ respectively.

7.4.2. Communication Protocol

The microcontroller and the computer software follow the serial communication protocol outlined in the state machine in Figure 9.

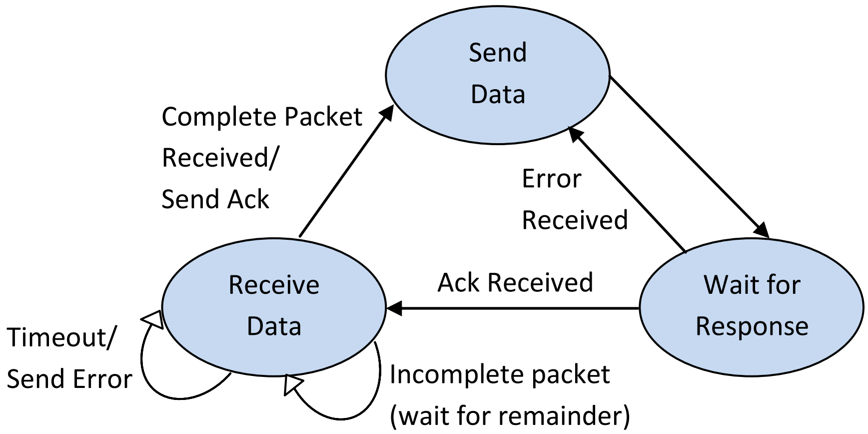

Figure 9: Serial Communication State Machine

To start the state machine properly, the microcontroller starts in the send state, and the computer starts in the receive state. The microcontroller enters the serial state machine to check for data in the Serial buffer or to send data every time a hardware timer expires. The computer will enter the state machine to check for data in the serial buffer and to send data at a twice the rate of the microcontroller to ensure that the microcontroller never waits too long for data.

7.5. Safety System Design

To protect electronic components from damage and prevent harm to developers and users, electrical safety in the V-Cycle will be enforced by taking the following precautions:

- Use of fuses in high current areas
- Proper insulation of wires
- Use of heat sinks to dissipate heat from motor drivers
- Use of interrupt controlled safety limit switches to stop motors immediately

8. User Interface and Visual Display

8.1. Software Design

The software on the V-Cycle computer will consist of 3 main components:

- The 3d rendering engine
- The audio processing engine
- The physics processing engine

8.1.1. Visual Interface

The 3d engine and physics engine currently employed are Ogre 3d and the Newton physics engine. The planned audio engine is the FMOD Soundmanager plugin. The main application will run on the Ogre 3d platform, and the physics and audio engines will hook into the application via an event-based api.

A simple description of the 3d terrain generation process follows:

- A terrain heightmap is painted in photoshop. A heightmap represents an area of terrain by using grayscale values, where white represents the tallest area and black represents the lowest.
- Ogre imports the heightmap and converts it to a triangle mesh, then applies the appropriate colormaps and blendmaps.
- A forest map is used to import billboard-based trees. The forest map consists of a black and white image, where each fully white pixel represents the location of a single tree.
- A live sky-cloud-weather system is loaded via an Ogre plugin called Caelum.
- The terrain triangle mesh is exported to the physics engine for collision detection.
- Different terrains can be loaded by simply changing the heightmap, forest map, and the colourmaps and blendmaps used to define terrain texture.

8.1.2. List of Processes / threads

The graphical rendering, audio processing, as well as physics will all run on separate threads with independent frame rates. The physics processing in particular needs to run at a much higher framerate to avoid the effects of tunneling - that is, a collision mesh may "ghost" or "tunnel" through another mesh if they happen to collide in between processing cycles. The graphical rendering thread is extremely processor-intensive. In order to achieve a high framerate a newer computer with a fast graphics card may be required. Certain effects may be disabled to accommodate older computers. The minimum recommended requirements are as follows:

- OS: Windows XP, Vista, or Windows 7 with DirectX 9 support
- Hardware: Minimum 512Mb RAM, P4 2.66 GHz or equivalent processor, and a Graphics Card with minimum of 256Mb video memory

8.1.3. Status of Development

The following features have already been implemented in our software:

- Dynamic lighting and atmospheric system, shown in Figure 10
- Terrain heightmap generation, colour and normal textures, forest generation, shown in Figure 11
- Terrain physics collision (tested with collision cylinders), shown in Figure 12

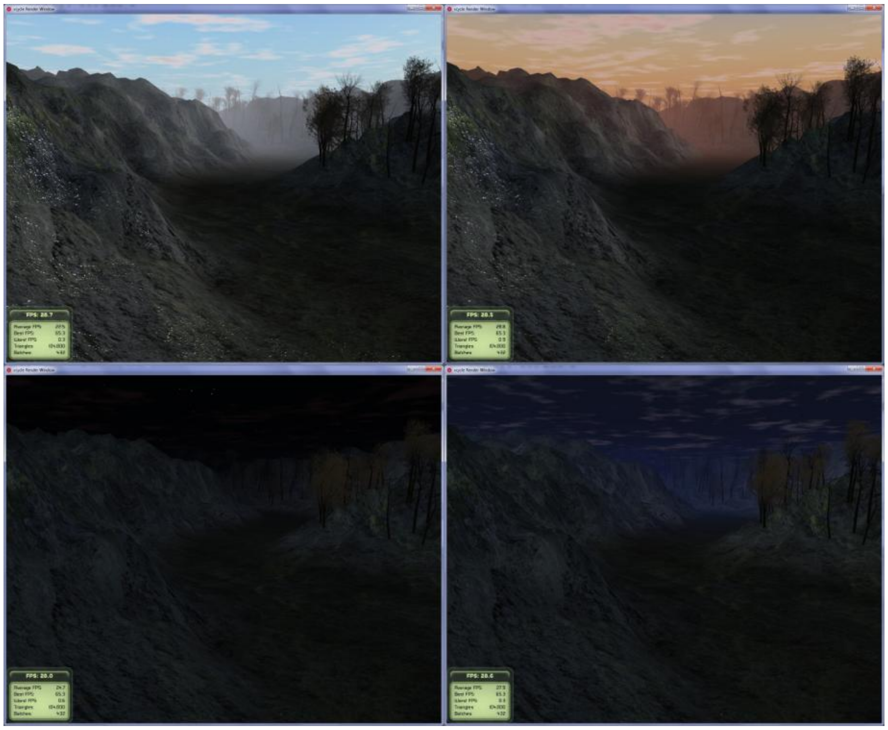

Figure 10: Dynamic Lighting and Atmospheric System

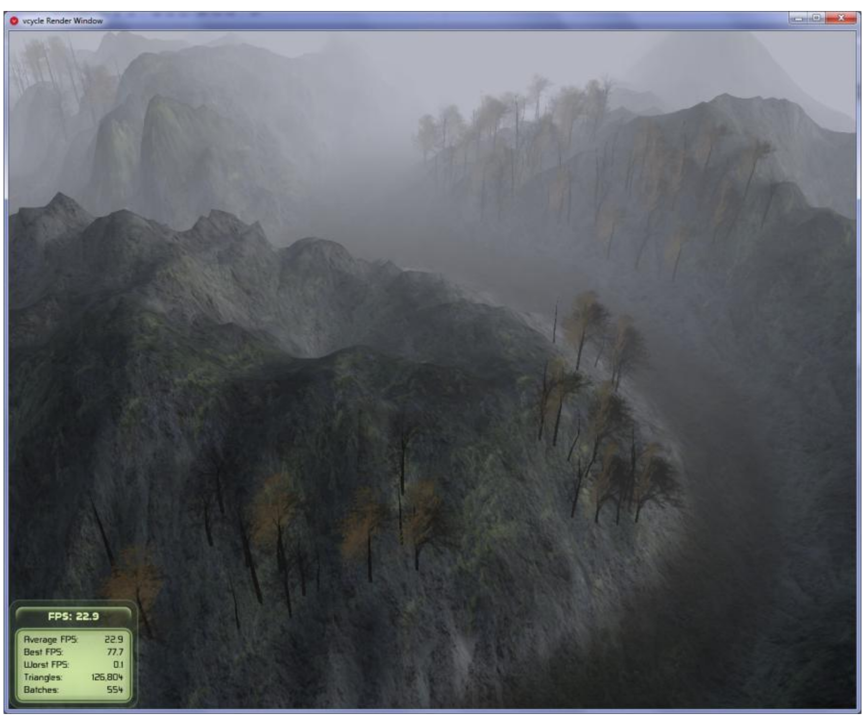

Figure 11: Terrain heightmap generation, colour and normal textures, forest generation

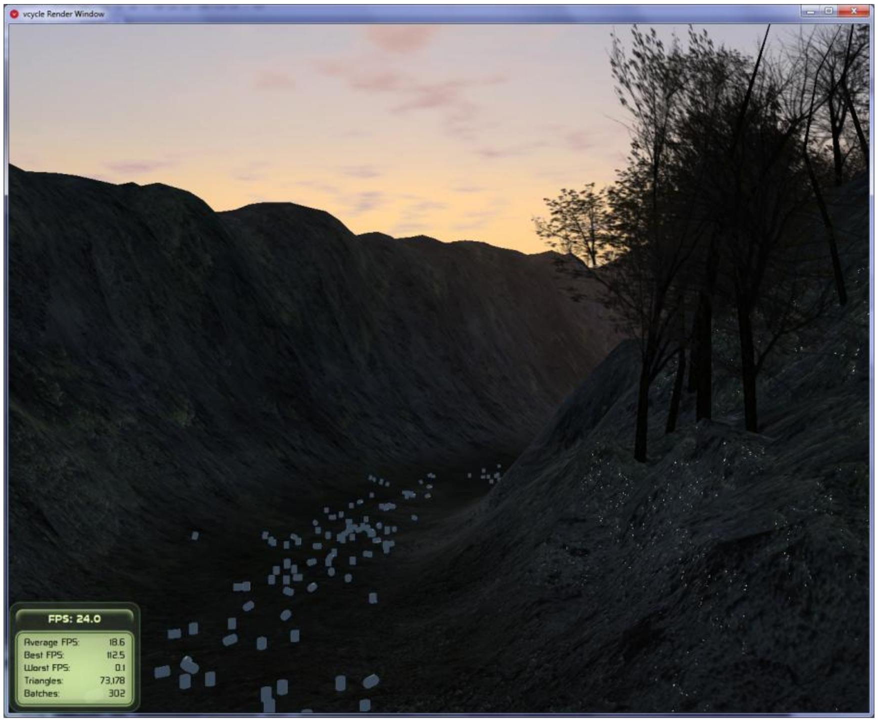

Figure 12: Terrain physics collision (tested with collision cylinders)

9. System Test Plan

Each part of the V-Cycle will be tested individually as specific milestones are reached. These are the unit tests outlined below. Then once the sub-systems have been integrated, system tests will be performed to ensure compatibility between different parts of the V-Cycle product.

9.1. Unit Tests

9.1.1. Front Lifting Mechanism

Internal acceptance testing for the side tilting mechanism will consist of the following tests:

- When maximum torque is applies to motor, linear slides continue to roll smoothly despite applied non-linear force.
- Gravity cancelling of the linear drive system reduces holding torques required such that the motor driver draws no more than 1 A when holding linear position.
- Position is held even if user applies downward or upward force on the bike handlebars.

9.1.2. Side Tilting Mechanism

Internal acceptance testing for the side tilting mechanism will consist of the following tests:

- Dead weight (up to maximum acceptable load) does not allow unit to tip over
- Dead weight (up to maximum acceptable load) does not cause structure to fail
- Unbalanced force shall not cause device to tip beyond 13 degrees on either side

9.1.3. Sensors

Internal acceptance testing for the sensors will consist of the following tests:

- The reed switch gives accurate counts of revolutions up to cycling speeds of 60 km/hr.
- The accelerometer and the post processing by the microcontroller, accurately tracks the tilting angle within 0.5 degrees without accumulating error after tilting in either direction many times.
- The potentiometer tracks handlebar angle very accurately from 15 to -15 degrees from going straight.

9.1.4. Motors

Internal acceptance testing for the variable resistance device will consist of the following tests:

- The motor is able to switch directions as dictated by the virtual reality software without the motor stalling.
- The motor does not stall when at the maximum speed with the maximum allowable load on the bike.
- The motor code does not accumulate error in absolute position, even when receiving consecutive movement commands from the computer software before the motor has completed the steps from the previous command.

9.1.5. Variable resistance device

Internal acceptance testing for the variable resistance device will consist of the following tests:

- When the front platform is fully raised, the magnet pairs will be aligned (determined by observation)
- When the front platform is fully lowered, the magnet pairs will be fully offset (determined by observation)
- Endurance testing: after one hour of cycling, the heat energy created shall not cause the device to fail

9.2. System Test

9.2.1. Normal Operation

- The device never becomes unstable for the user and never fails as a result of overloaded, even when performing high intensity activities such as hill climbing.
- The device always stops safety and quickly when the user employs the safety limit switch.
- Changes in resistance are closely coupled with changes in elevation.
- Turns caused by the user turning handlebars and/or tilting are realized real-time in the virtual reality software.
- Elevation changes in terrain are realized in adequate time by the linear drive system. Where abrupt changes in elevation occur, the linear drive system algorithm adequately smooths the movement to operate within the limits of the motor.

10. Conclusion

This document proposes the designs solutions to the functional specification of the V-Cycle Pro. The components in this design were selected based on their ability to meet the project's functional specifications while staying within the budget of the project.

All functional requirements are expected to be met by the end of the design cycle, and through the test plans included in the design specifications, the required functionality and safety of our product will be ensured. We believe that our V-Cycle Pro will provide users with a fun and interactive home exercising option.

11. References

[1] Statistics Canada: Canadian Health Measures Survey 2007-2009

[2] Pedal Power Generator [Online document]. [Accessed March 5, 2010].
http://www.pedalpowergenerator.com/how-to-build-a-bicyle-generator-free-power.html#much