Slide 1

These slides are publicly available on github from the hyperlink on the slide.

This presentation is about two boring activities: playing a flight simulator and riding a bike trainer. Combining them turns them into an enjoyable activity for me and I hope for you also.

I can now play flight simulator, “without time warping” in front of my family and they conclude incorrectly that the activity is just exercise.

Time warping allows the simulation time to be 4 times faster than clock time so a one hour flight takes on minute.

Combining bike trainer riding with operating flight simulator brought back the excitement of cycling. I now can plan the nature of my exertions [how far? How fast?]: in other words, “how fast can I climb that hill.”

This presentation is an attempt to record my quirky past time.

Please raise your hand to indicate you have a question when you have one. I will do my best to respond.

Slide 2

For me, it all began before my brain injury when my Uncle Steve was building a Piper Cub from plans. At that time, I was an avid cyclist living in northeast Ohio. I knew it was not unheard of to operate a home built aircraft without even a single pilot’s lesson. [ask me to cite two examples] So, I thought I should at least know something about operating a Piper Cub. [with flightgear] The winter was too cold for outdoor cycling and I had a bike trainer to stay in shape during the winter. Why not combine the two? And I did. At that time, I used a data acquisition device to translate the rotational speed of the bike trainer into a virtual throttle position. Here is a Logitech flight stick, commercially available. The throttle is here. Trainer resistance is not constant, it varies as the trainer heats up. At first I compared my speed to a record of a previous exertion on the trainer. Now I use a better method for this translation.

Slide 3

This configuration worked for me; but no one else knew about my quirk.

When I started to live with my parents, I sought to revive this activity

Virtual flights generated physical and mental challenges for me.

On this slide ther is a link so you can freely download flightgear.

Slide 4

When I move back home I wanted to revive my hobby. One of the first things I did after getting back on the bike trainer was purchase a logitech flightstick.

Slide 5

That still was not enough. Merely exerting myself (please note the bike is wrapped in towel to absorb sweat. I have replace two bottom brackets since moving back home.) was not satisfying enough.

Slide 6

I stated using a program called gatttool to record output from a Wahoo sensor I mounted on the rear hub of my bike with a Raspberry Pi. Raspberry Pi us a linux operating system, just like the operating system I had on my own personal computer. I move gatttool to my PC and modified gatttool to translate my cycling exertion into a throttle position. This required the use of administrative privileges. A way around this is to use an Arduino to do the same thing.

If you want, I can jump ahead to slide 9 to explain what an Arduino is/

Slide 7

The Wahoo sensor sends out 8 bytes of information every second when the rear wheel starts spinning.

The first and last byte were not relevant to me for achieving my goal: translating speed into a throttle position. But the middle six are. The first four bytes indicate the total number of rear tire rotations – how far I’ve gone. The last two combine with the first four give a high resolution speed indication

Slide 8

Merely capturing this information filled the programming space of Arduino Uno WIFI Rev. 2. A second Arduino is required to translate that information into a joystick throttle position.

Slide 9

Here is a succinct explanation of what an Arduino is.

Slide 10

Here is a picture show the Arduino Leonardo (joystick) and Arduino Uno Wifi Rev2 connected by wires that facilitate their communication.

Slide 11

I hope you ask questions, so I can illustrate the amount of thought I put into this feat.

Slide of HUD

Here I head up display I watch while playing flight gear. FlightGear includes a GPS to guide you to your virtual distanation. The GPS also shows ground speed. The HUD shows air speed, aileron position and its trim tab, elevator position and its trim tab, throttle position, aircraft heading, altitude above ground, altitude above sea level, and rudder position. I use auto-coordinated flight control so the rudder is not used.

Watch my hands closely, they never use the 9 and 3 keys to asjust the throttle. Keyboard presses are shown in the upper left part of the screen by ShowOff.

Now I will attempt a virtual flight Gaithersburg Airport to BWI

Check list

Turn on fanS!

Put headband and left foot brace

Put on cycling shoes

Turn on BLE Arduino

Rotate rear wheel by hand

Confirm blue light is on

From command prompt,

change to jump drive (E:)

change to flightgear\* directory

change to bin

type “fgfs –launcher”

While flightgear launches, launch app in showoff folder from jump drive

When launcher opens

choose Cessna

Confirm KGAI aiport as departure

When flight gear launches

activate parking break

from “Equipment” menu set GPS to KBWI

press “h” to turn HUD

twice if you want red

press “c” to turn off cockpit

press “x” to enlarge HUD

press “z” to change how far ahead you can see

start pedaling

if throttle is active take off

restart arduino until throttle is active