

Hackflight: A Simple Software Ecosystem for Miniature Aerial Vehicles



Simon D. Levy
Julio Hidalgo Lopez
Matt Lubas
Alfredo Rwagaju

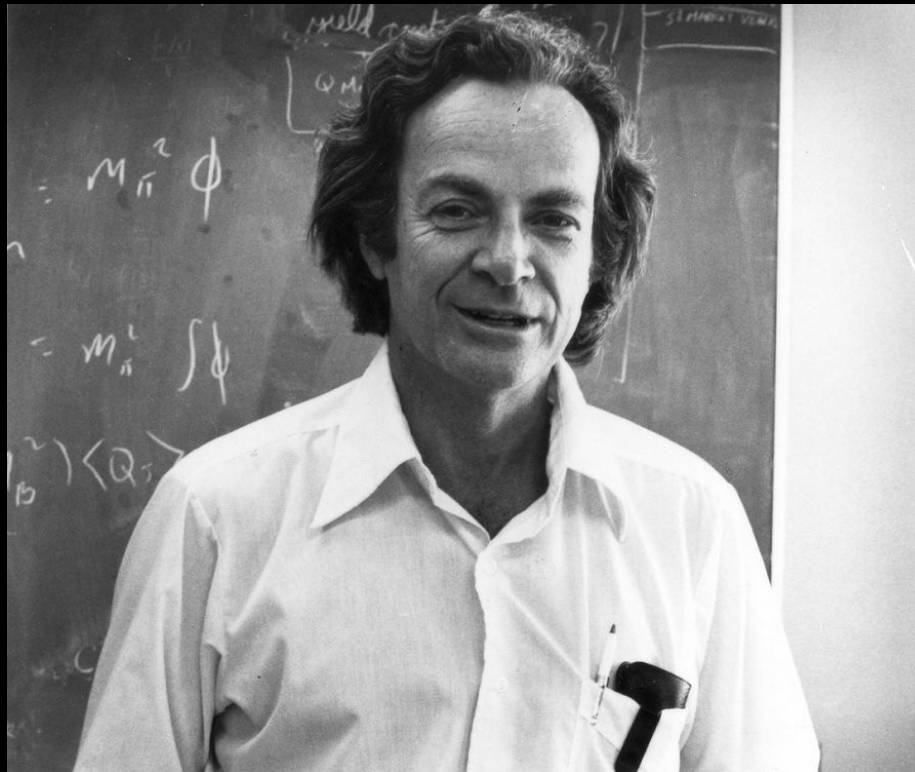


Washington & Lee University

Intercollegiate Dronefest
18 August 2016

Motivation #1:

What I Cannot Build, I Do Not Understand



R. Feynman (1918-1988)

K_{keep} **I**_t **S**_{imple,} **S**_{tupid!}

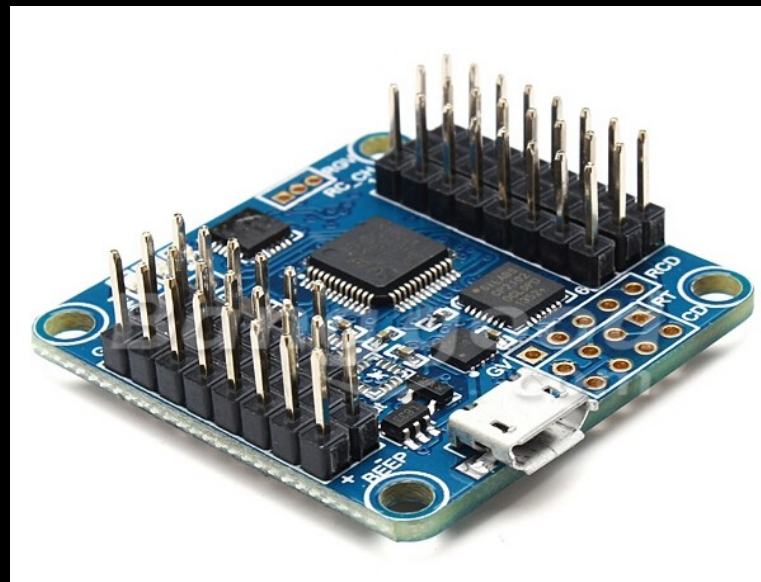
- PX4 Firmware: > 100K lines of code
- Cleanflight: > 50K

Motivation #2:

*I'm not gonna pay a lot for this
muffler flight controller!*



\$200



\$15

Hackflight Genealogy

MultiWii

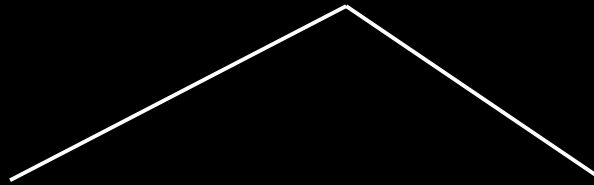


Baseflight (~14K l.o.c.)



Hackflight

Cleanflight



Raceflight

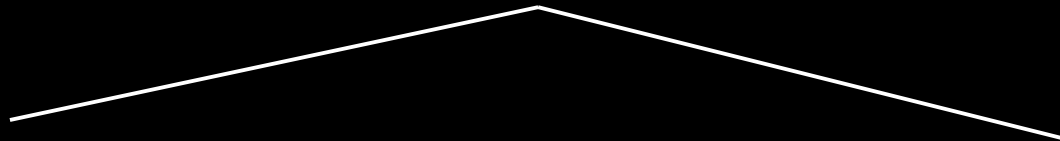
Betaflight

Hackflight Genealogy

Baseflight



Hackflight



Platform-
Independent

Firmware

(~1500 lines of C++)

BreezySTM32

(~4200 lines of C)

Branch: master

hackflight / firmware /

Create new file

Upload files

Find file

History



simondlevy Fixed typo in comment

Latest commit ad16c1f 13 hours ago

..

baro.cpp	Made all Board methods static	7 days ago
baro.hpp	Made all Board methods static	7 days ago
board.hpp	Add support for displaying aux-switch status	3 days ago
crossplatform.h	Debugging console working with Windows	a month ago
filters.cpp	Add Hover class, module for filters	a month ago
filters.hpp	Add Hover class, module for filters	a month ago
hover.cpp	Fixed typo in comment	13 hours ago
hover.hpp	Built Windows release	13 hours ago
imu.cpp	Computing vario, alt-hold PIDs	15 hours ago
imu.hpp	Computing vario, alt-hold PIDs	15 hours ago
mixer.cpp	Made all Board methods static	7 days ago
mixer.hpp	Made all Board methods static	7 days ago
msp.cpp	Renamed Navigate to Hover; removed alt-hold implementation	19 hours ago
msp.hpp	Renamed Navigate to Hover; removed alt-hold implementation	19 hours ago
mw.cpp	Renamed Navigate to Hover; removed alt-hold implementation	19 hours ago
mw.hpp	Renamed Navigate to Hover; removed alt-hold implementation	19 hours ago
rc.cpp	Made all Board methods static	7 days ago
rc.hpp	Made all Board methods static	7 days ago
sonars.cpp	Better variable names	6 days ago
sonars.hpp	Begin supporting four horizontal sonars	6 days ago
stabilize.cpp	More cleanup	22 days ago
stabilize.hpp	More cleanup	22 days ago

BreezySTM32: An Arduino-like API for STM32 flight controllers

```
#include <breezystm32.h>

static bool sonar_present;
mb1242_t mb1242;

void setup(void)
{
    i2cInit(I2CDEV_2);
    delay(500);
    sonar_present = mb1242_init(&mb1242, 0); // Use default address
}

void loop(void)
{
    if(sonar_present)
        printf("%d\n", mb1242_poll(&mb1242));
    else
        printf("no sonar\n");
    delay(100);
}
```



```
/*
ms4525.c : Airseed Measurement Values

Copyright (C) 2016 James Jackson

This file is part of BreezySTM32.

BreezySTM32 is free software: you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.

BreezySTM32 is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with BreezySTM32. If not, see <http://www.gnu.org/licenses/>.
*/

#include <breezystm32.h>

void setup(void)
{
    delay(500);
    i2cInit(I2CDEV_2);
}

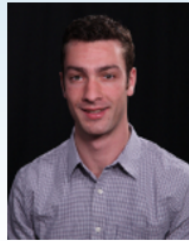
int16_t velocity;
int16_t temp;

void loop(void)
{
    if( ms4525_detect() )
    {
        ms4525_read(&velocity, &temp);
        printf("velocity = %d, temp = %d\n", velocity, temp);
    }
    else
    {
        printf("no airspeed\n");
    }
    delay(10);
}
```



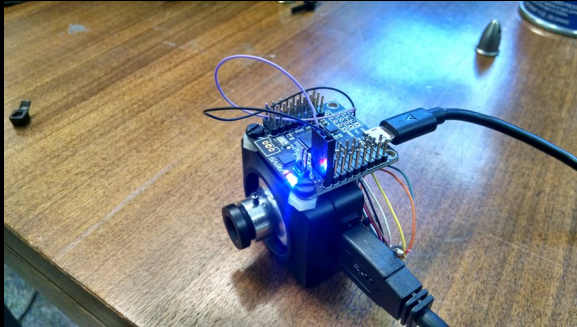
James Jackson

Degree: MS Mechanical Engineering
Email: james.s.jackson@byu.edu

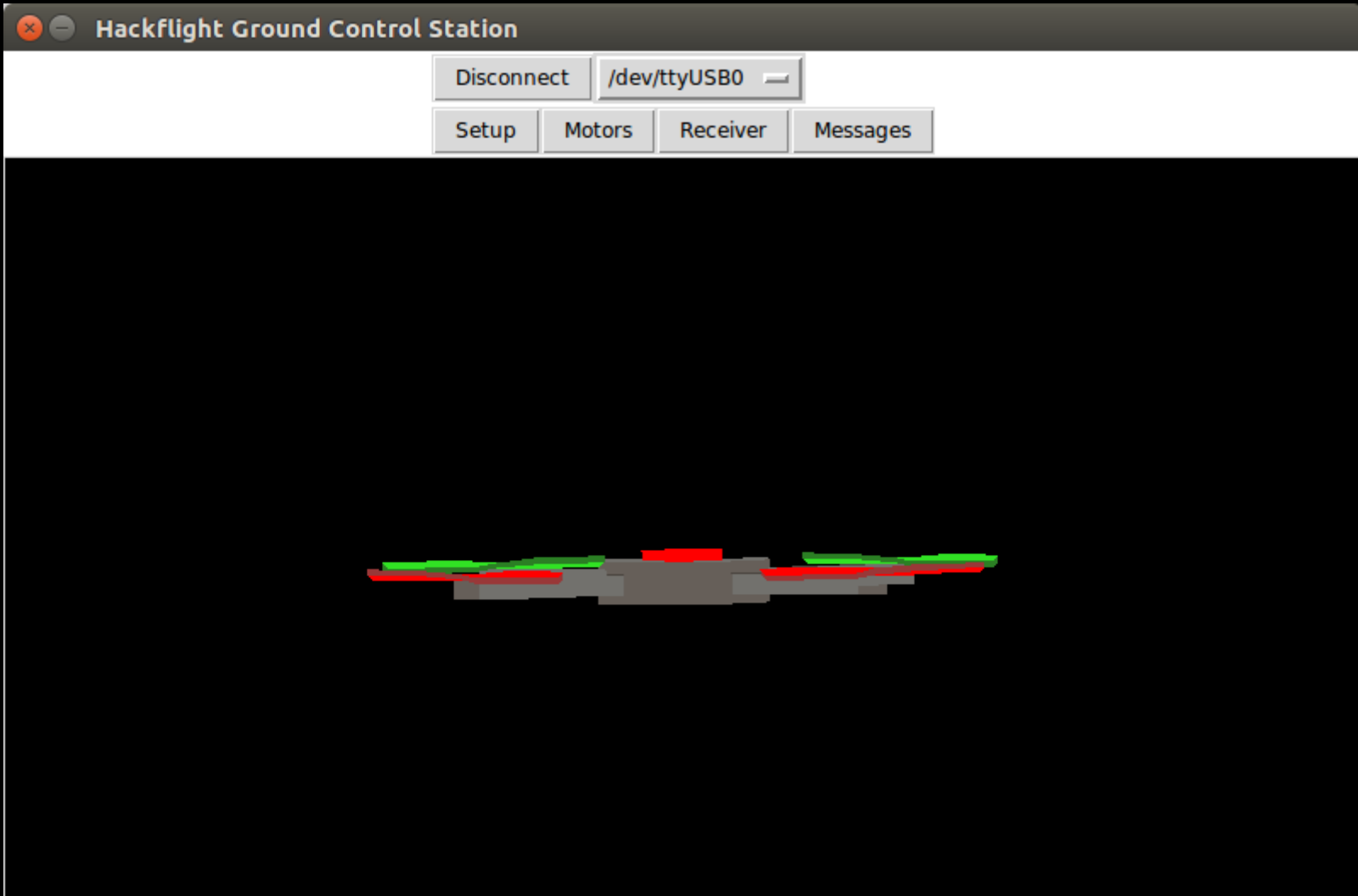


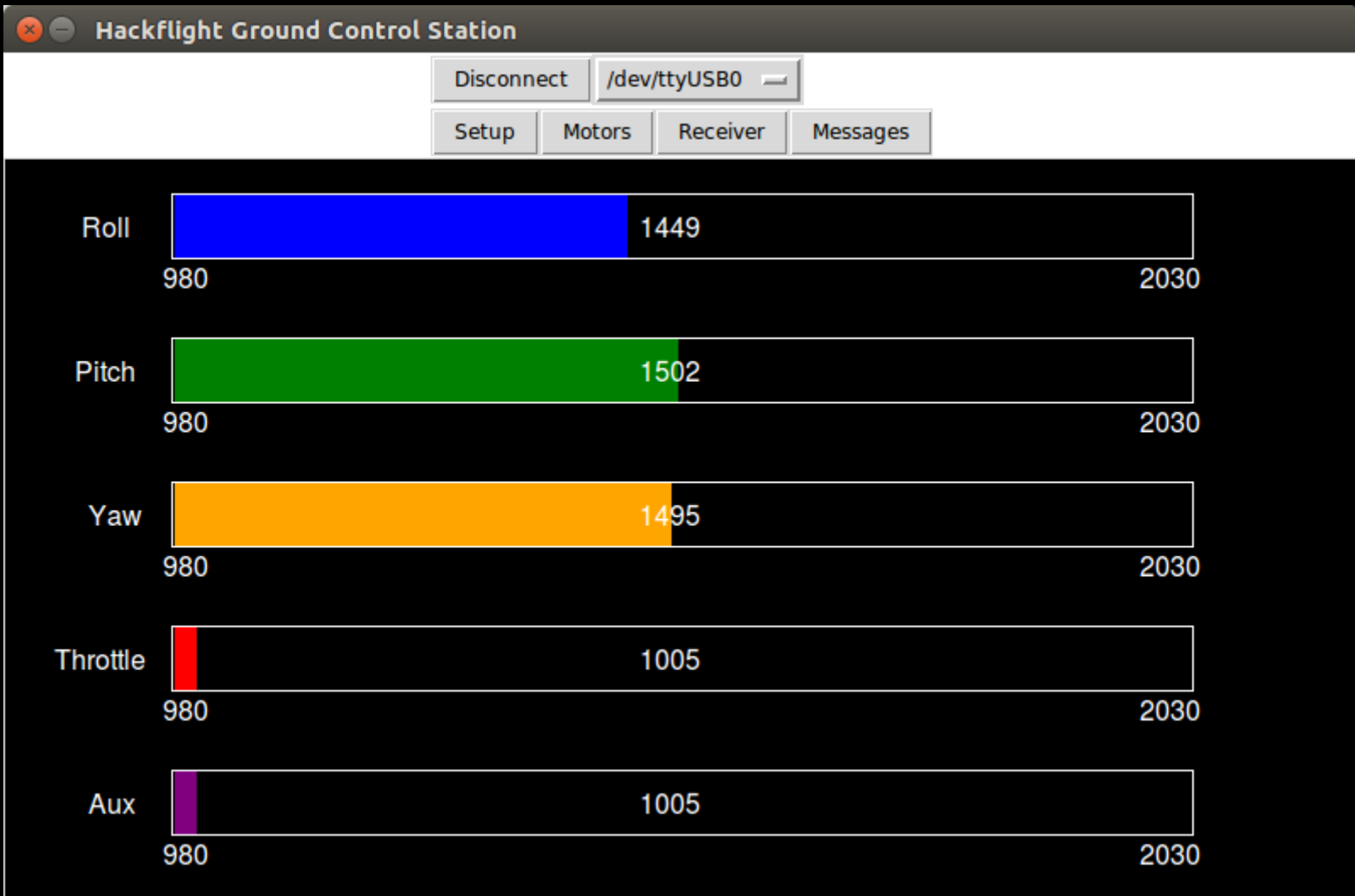
Daniel Koch

Degree: MS Mechanical Engineering
Email: daniel.koch@byu.edu



Python Ground Control Station





MSPPG: Parser (Code) Generator for MultiWii Serial Protocol

```
{
  "RC": [{"ID": 105},
        {"comment": "16 channels in http://www.multiwii.com/wiki/index.php?title=Multiwii\_Serial\_Protocol"},
        {"c1": "short"},
        {"c2": "short"},
        {"c3": "short"},
        {"c4": "short"},
        {"c5": "short"},
        {"c6": "short"},
        {"c7": "short"},
        {"c8": "short"}],

  "ATTITUDE": [{"ID": 108},
               {"comment": "angles should be divided by 10"},
               {"roll": "short"},
               {"pitch": "short"},
               {"yaw": "short"}],

  "ALTITUDE": [{"ID": 109},
               {"altitude": "int"},
               {"vario": "short"}],

  "SONARS": [{"ID": 127},
             {"comment": "four horizontal-facing sonars"},
             {"back": "short"},
             {"front": "short"},
             {"left": "short"},
             {"right": "short"}],

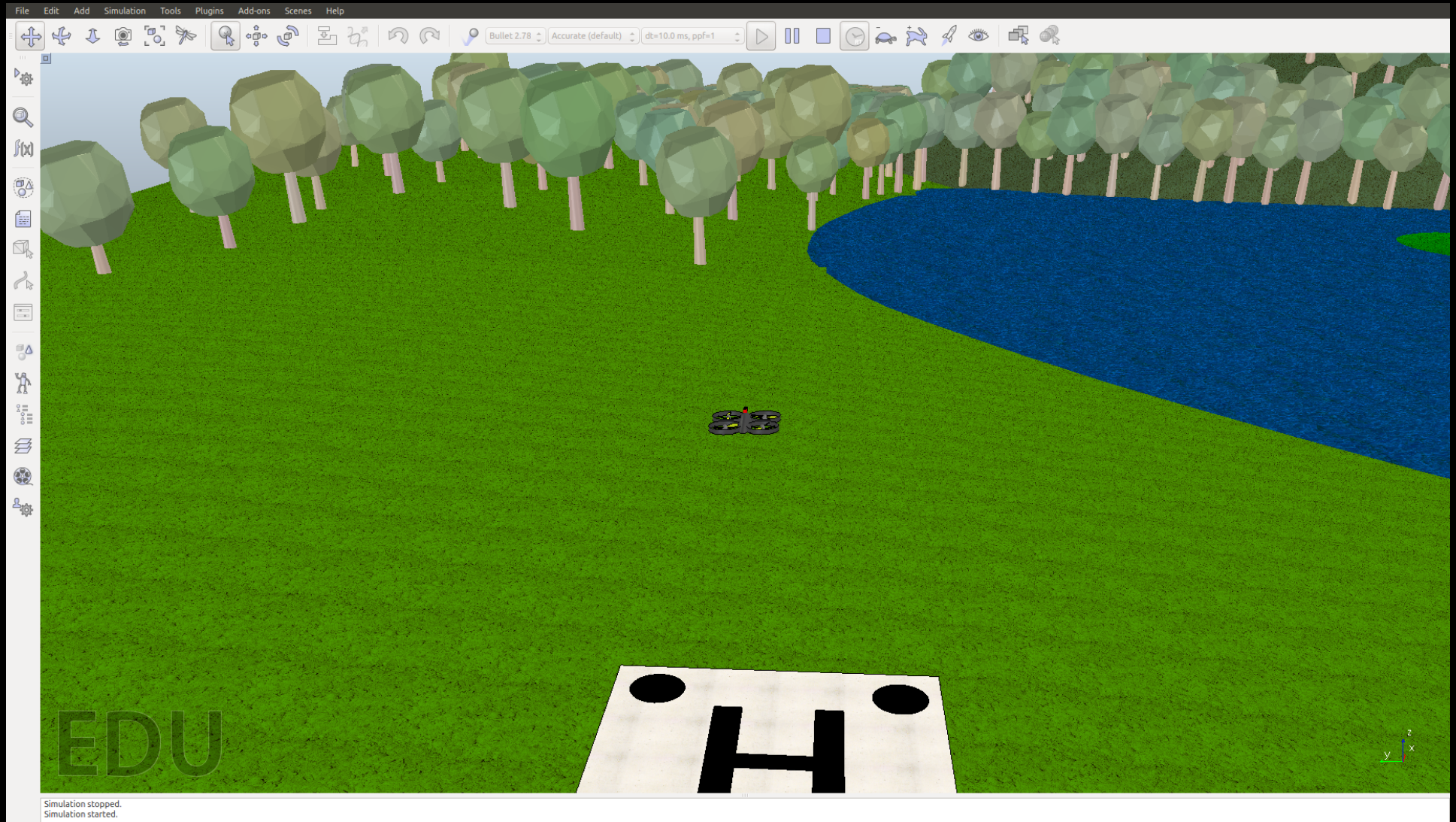
  "SET_RAW_RC": [{"ID": 200},
                 {"comment": "16 channels in http://www.multiwii.com/wiki/index.php?title=Multiwii\_Serial\_Protocol"},
                 {"c1": "short"},
                 {"c2": "short"},
                 {"c3": "short"},
                 {"c4": "short"},
                 {"c5": "short"},
                 {"c6": "short"},
                 {"c7": "short"},
                 {"c8": "short"}],

  "SET_HEAD": [{"ID": 205},
               {"head": "short"}],

  "SET_MOTOR": [{"ID": 214},
                {"m1": "short"},
                {"m2": "short"},
                {"m3": "short"},
                {"m4": "short"}]
}
```

```
levy@kern: ~/Desktop/hackflight/parser
levy@kern:~/Desktop/hackflight/parser$ ./msppg.py
Creating file output/python/Makefile
Creating file output/python/getimu.py
Creating file output/python/getrc.py
Creating file output/python/imudisplay.py
Creating file output/python/blueimudisplay.py
Creating file output/python/setrc.py
Creating file output/python/setup.py
Creating file output/python/msppg/__init__.py
Creating file output/cpp/Makefile
Creating file output/cpp/example.cpp
Creating file output/arduino/MSPPG/examples/imuexample/imuexample.ino
Creating file output/cpp/msppg/msppg.cpp
Creating file output/cpp/msppg/msppg.h
Creating file output/arduino/MSPPG/msppg.cpp
Creating file output/arduino/MSPPG/msppg.h
Creating file output/c/Makefile
Creating file output/c/example.c
Creating file output/c/msppg/msppg.c
Creating file output/c/msppg/msppg.h
Creating file output/java/Makefile
Creating file output/java/example.java
Creating file output/java/edu/wlu/cs/msppg/Parser.java
Creating file output/java/edu/wlu/cs/msppg/ATTITUDE_Handler.java
Creating file output/java/edu/wlu/cs/msppg/ALTITUDE_Handler.java
Creating file output/java/edu/wlu/cs/msppg/SONARS_Handler.java
Creating file output/java/edu/wlu/cs/msppg/RC_Handler.java
levy@kern:~/Desktop/hackflight/parser$ █
```

HackflightSim

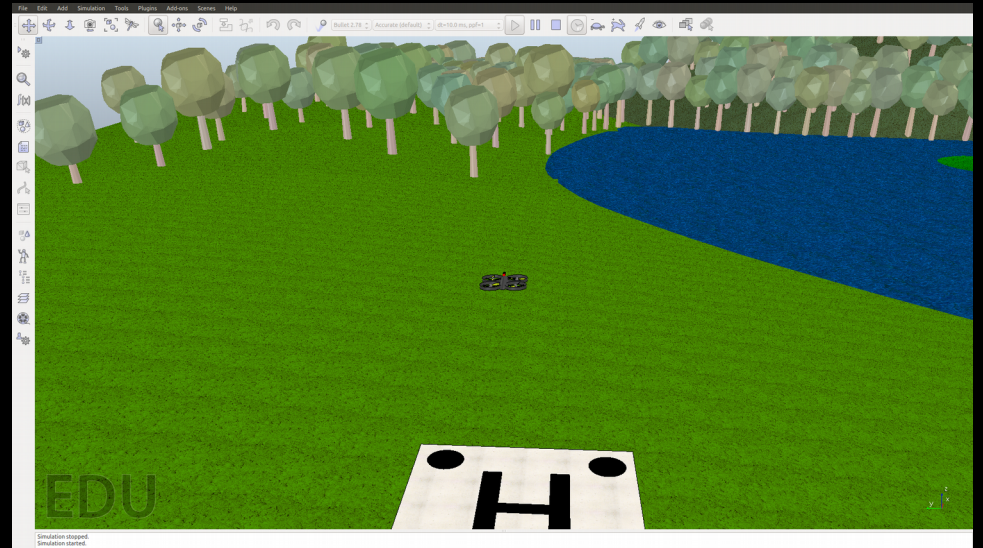
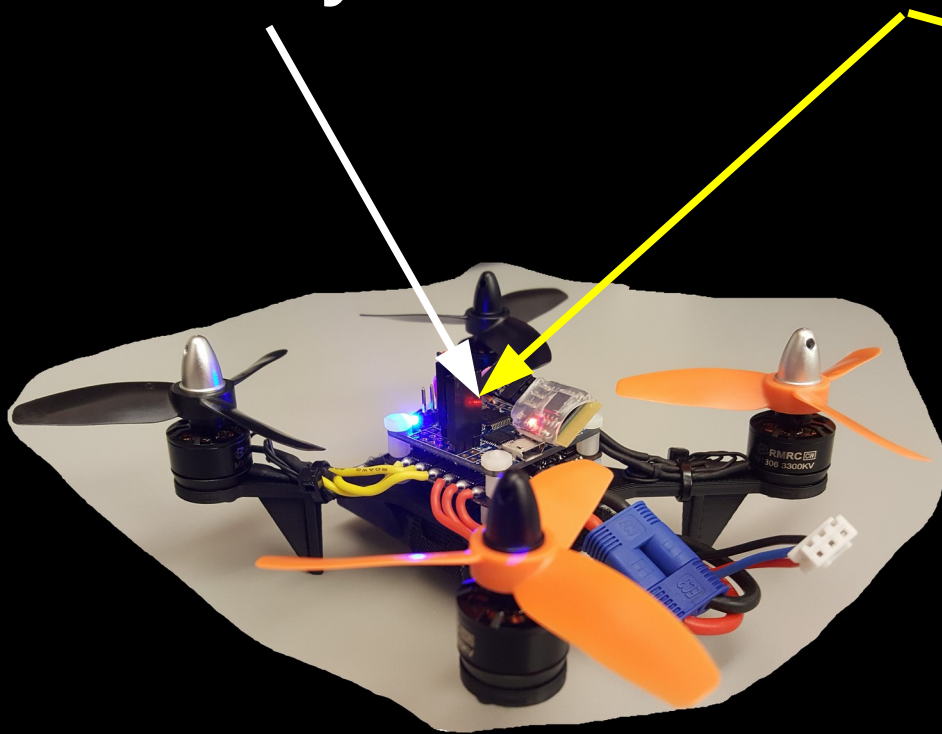


HackflightSim

BreezySTM32

Firmware

V-REP Plugin



**slow*

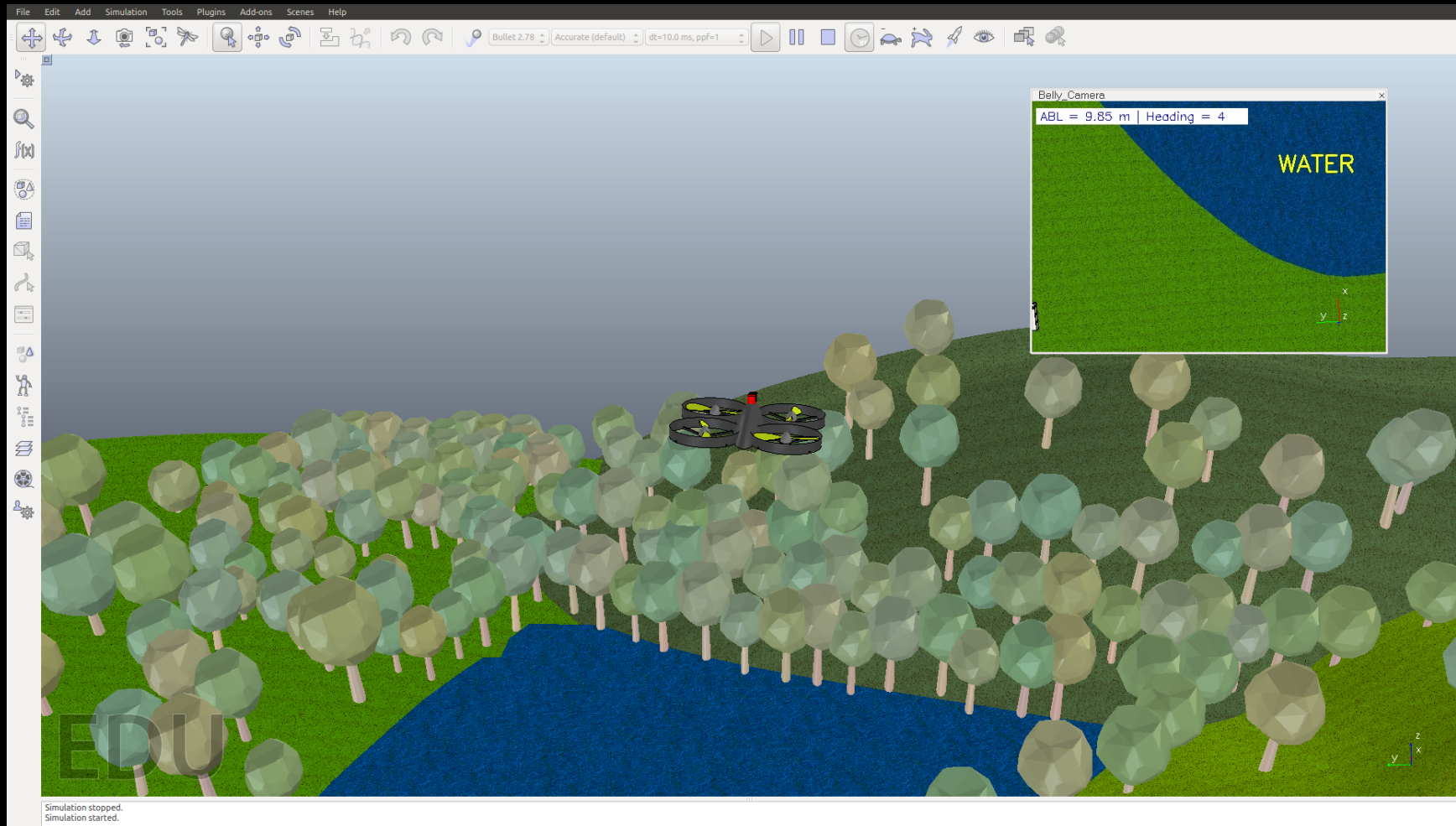
HackflightSim



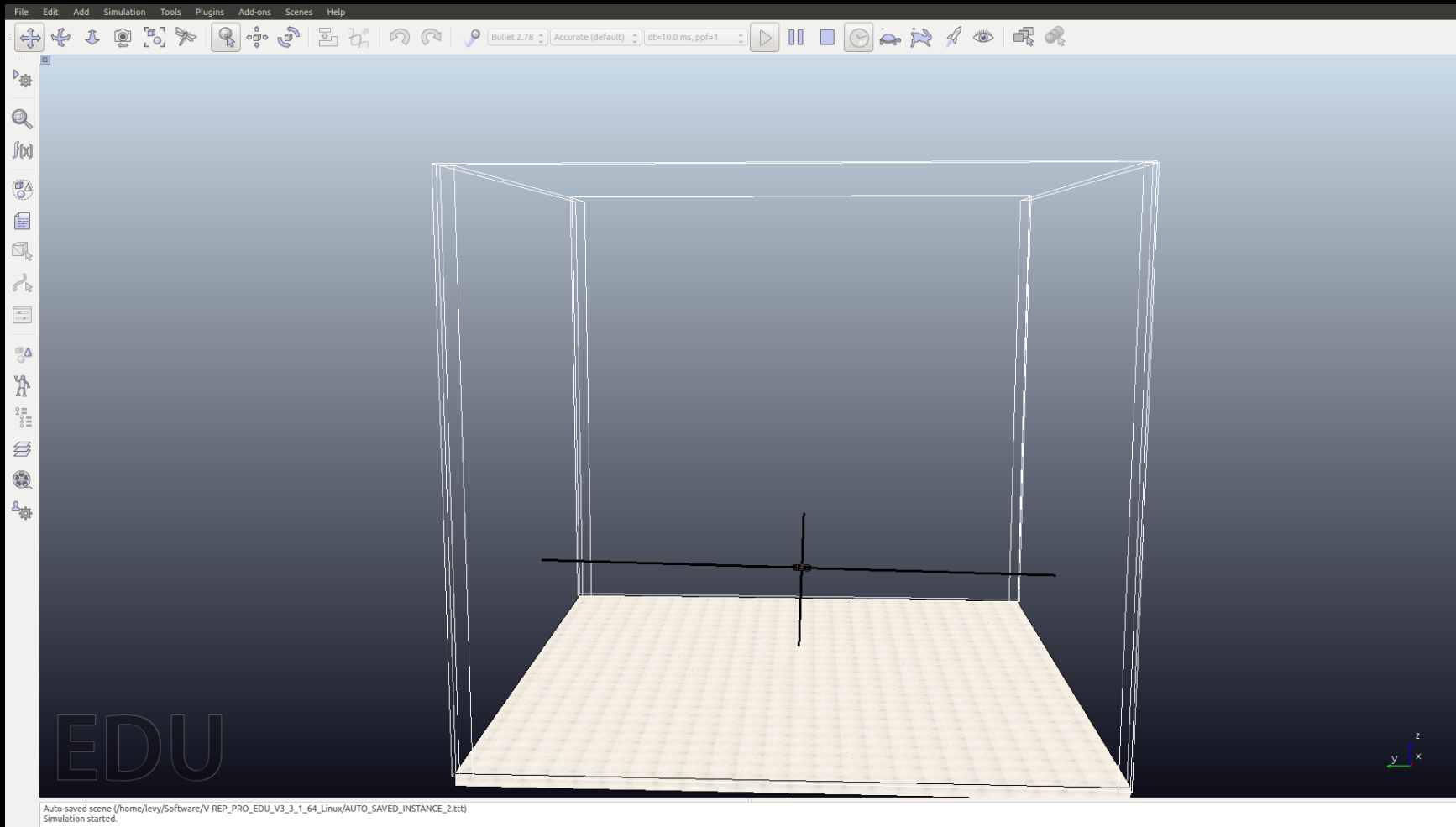
Board Abstraction

```
class Board {  
  
    public:  
  
        // your implementation should support these methods  
  
        static void    init(uint32_t & imuLoopTimeUsec, uint32_t & calibratingGyroMsec);  
  
        static bool    baroInit(void);  
        static void    baroUpdate(void);  
        static int32_t  baroGetPressure(void);  
        static void    checkReboot(bool pendReboot);  
        static void    delayMilliseconds(uint32_t msec);  
        static uint32_t getMicros();  
        static void    imuInit(uint16_t & acc1G, float & gyroScale);  
        static void    imuRead(int16_t accADC[3], int16_t gyroADC[3]);  
        static void    ledGreenOff(void);  
        static void    ledGreenOn(void);  
        static void    ledGreenToggle(void);  
        static void    ledRedOff(void);  
        static void    ledRedOn(void);  
        static void    ledRedToggle(void);  
        static uint16_t readPWM(uint8_t chan);  
        static void    reboot(void);  
        static uint8_t  serialAvailableBytes(void);  
        static uint8_t  serialReadByte(void);  
        static void    serialWriteByte(uint8_t c);  
        static void    showArmedStatus(bool armed);  
        static void    showAuxStatus(uint8_t status);  
        static bool    sonarInit(uint8_t index);  
        static void    sonarUpdate(uint8_t index);  
        static uint16_t sonarGetDistance(uint8_t index);  
        static void    writeMotor(uint8_t index, uint16_t value);  
  
}; // class Board
```

Missions (“Scenes”)

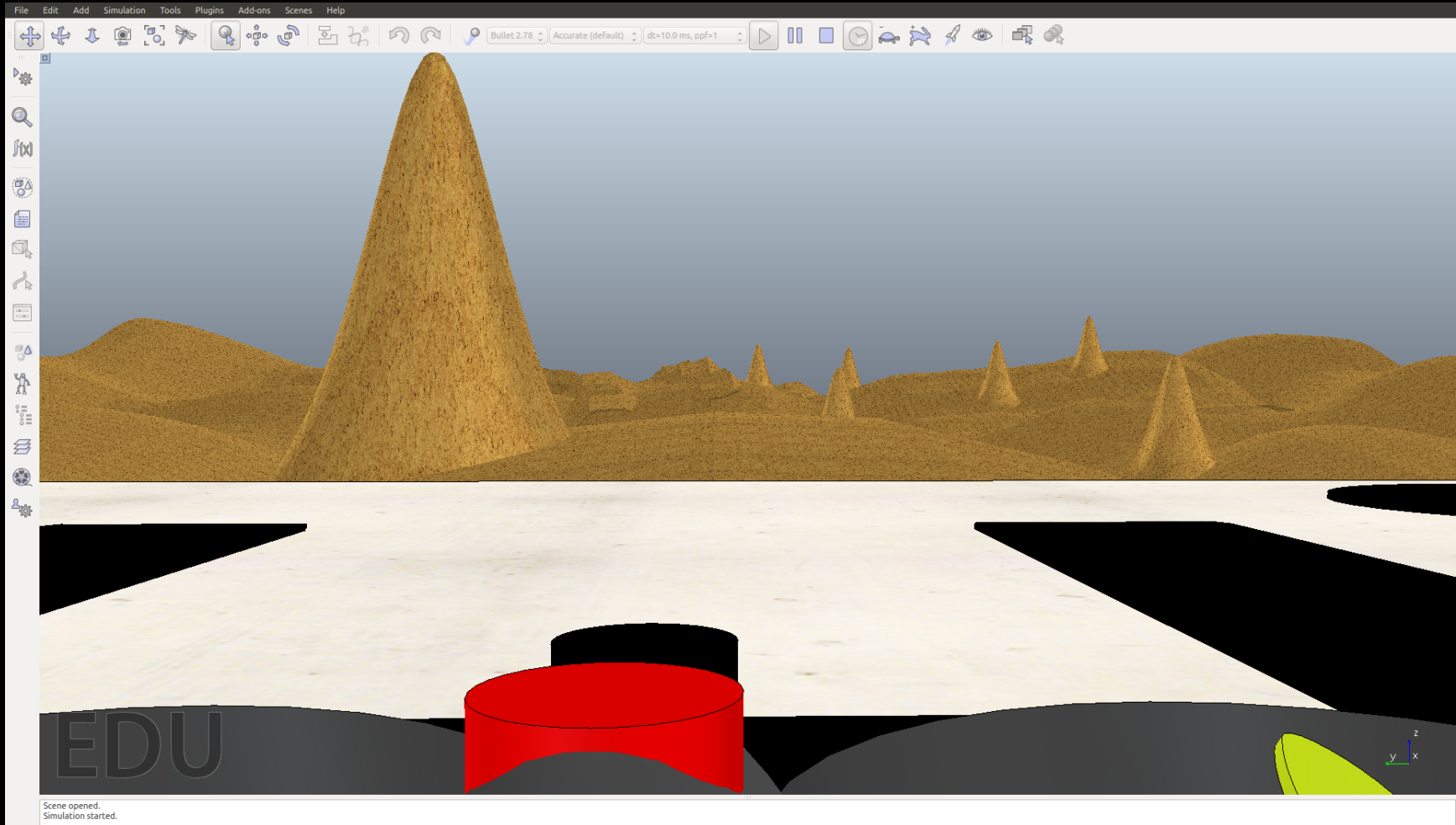


Python Computer Vision via
“companion board”



Sonar-Based SLAM

(Simultaneous Localization And Mapping)



FPV

Personnel & Acknowledgements



Prof. Simon D. Levy



Commonwealth Research Commercialization Fund (2014)



Bipeen Acharya '15



Shannon Nollett '15



Fred Gisa '16



Summer Research Scholars Program
Gerry Lenfest Summer Research Grant
Advanced Research Cohort Program



Bert Wagner
Advanced Aerials, Inc.



Rob Jones '17



Alec Singer '18



Matt Lubas '18



Alfredo Rwagaju '18



Julio Hidalgo Lopez '20