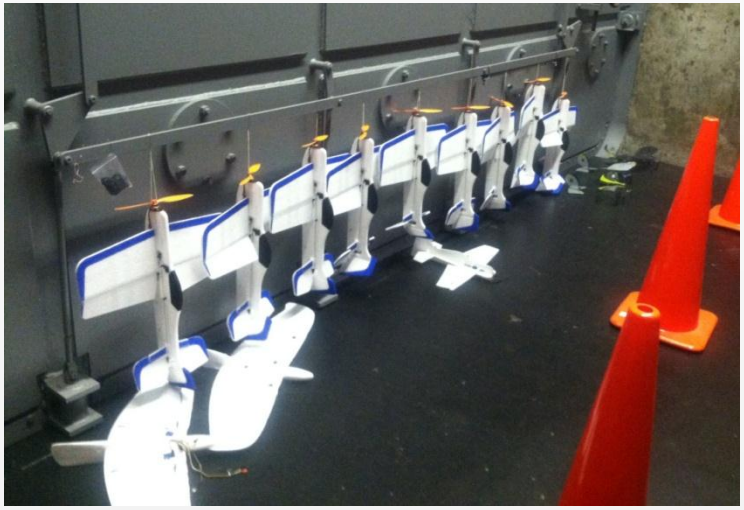
Using Commercial Off-The-Shelf (COTS) Radio Control Products and Microcontrollers to Develop & Test Small Unmanned Aerial Systems In an Indoor Flight Lab

Gabriel Staples, 1 Lt. United States Air Force
Aeronautical Research Engineer/SUAS Integration Engineer
Small Unmanned Aerial Systems Laboratory (SUASLab)
Air Force Research Laboratory, AFRL/RQVA
Wright-Patterson Air Force Base, OH
26 Feb. 2014



The views expressed in this presentation are those of the author and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the United States Government.

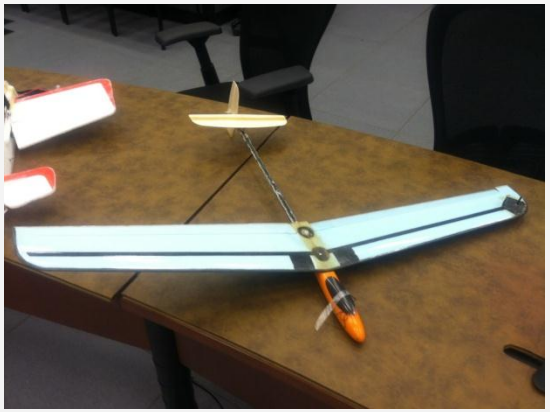
These slides are from a briefing which was given on 27 March 2014 to the Air Force Research Laboratory's Aerospace Systems Directorate (AFRL/RQ) Junior Force Council (JFC), as part of a "Lunch and Learn" "Present Your Project" initiative.

Purpose

1. To educate and inspire you, my fellow officers & engineers, to expand your skills and use technical solutions, such as Arduino microcontroller boards, to solve your technical challenges.
2. To remind you that education doesn't just happen in school, and that learning & research doesn't just happen at work.
3. To share what gets me excited to come to work everyday.

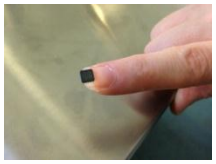
My Maxims

- *“Every engineer a programmer”*
- *"It should be the end goal of any professional to reach such a high level of skill at what you do that you are no longer limited by your abilities; but rather, you are limited only by your imagination.“*
- “Be a thinker, and be a doer.”



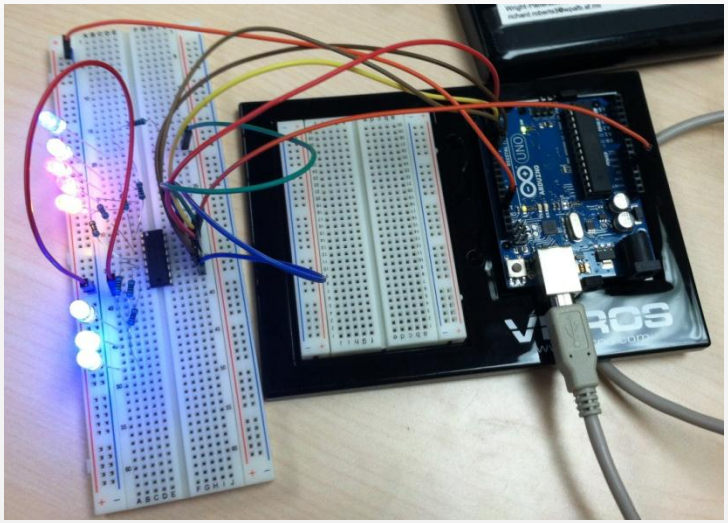
References: first two quotes are my own (see: http://electricrcaircraftguy.blogspot.com/2014/02/the-goal-of-a-lifetime.html), the third one is from Destin from his "Smarter Every Day" series on YouTube.

What is an Arduino microcontroller (mcu)?



Mcu's In General:

- A chip with integrated processor, peripherals, UART, RAM, EEPROM, Flash memory, etc.

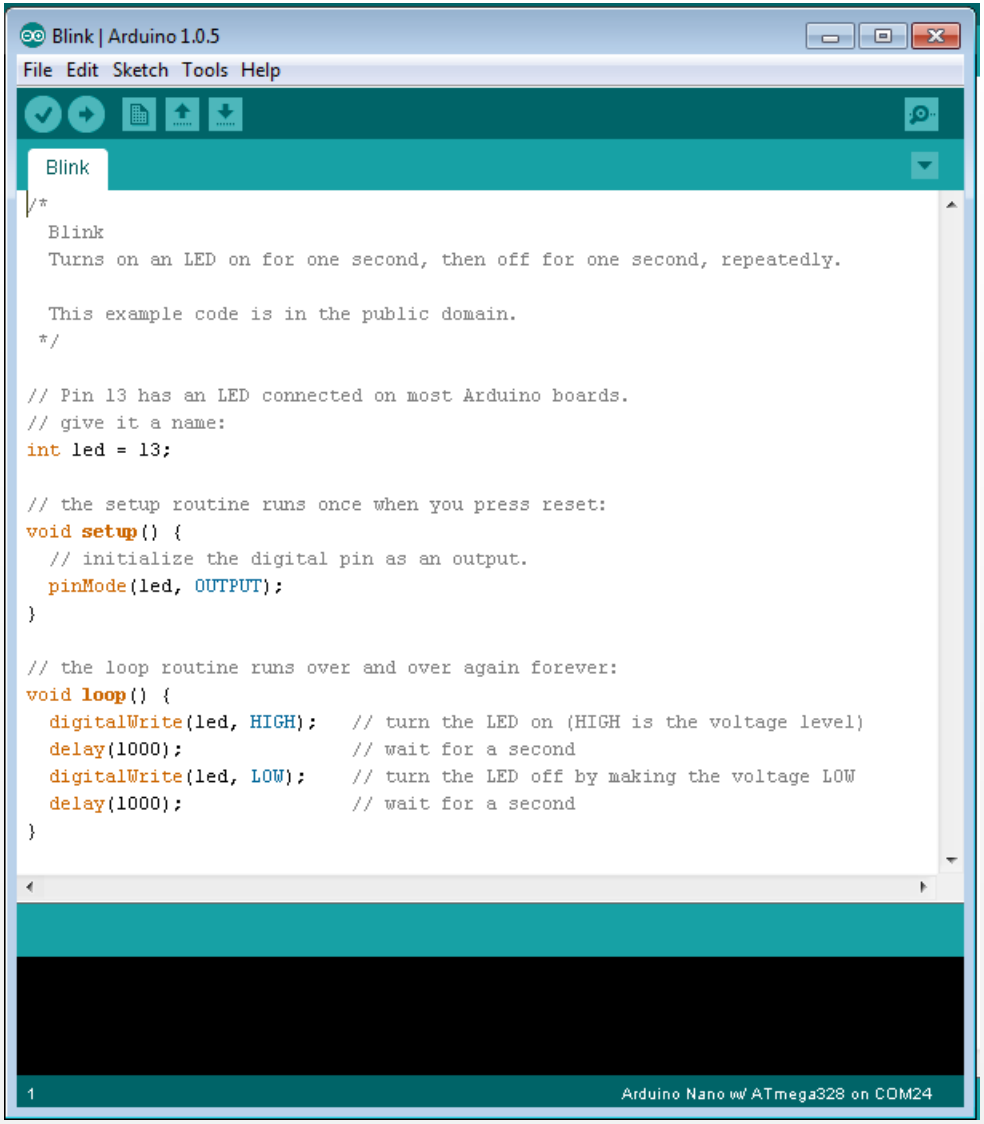


Arduino specifically:

- An open source development board that uses (usu.) an Atmel ATmega microcontroller
- Open source hardware
- Open source software
- Hundreds of open-source libraries
- Hundreds of thousands of users, including professionals and hobbyists
- Fantastic documentation, support, & tutorials
- Designed for ease of programming, with the non-engineer in mind
- Programs are called "sketches," meant to be a way to introduce artists to programming.
- Cost of only $3.50~$150 per board (~$30 avg.)
- This is a HUGE cost savings, as other systems I've seen are not open source, and are $75~$100 to get started.

See http://en.wikipedia.org/wiki/Arduino for more info.

Sample Code



-Based on C/C++, with a basic Arduino core library meant to be easy to understand and use.

-Yet you still have the ability to use all of the AVR-libc functions and access the basic, low-level Atmel microcontroller registers, variables, timers, functions, input capture, output compares, etc.

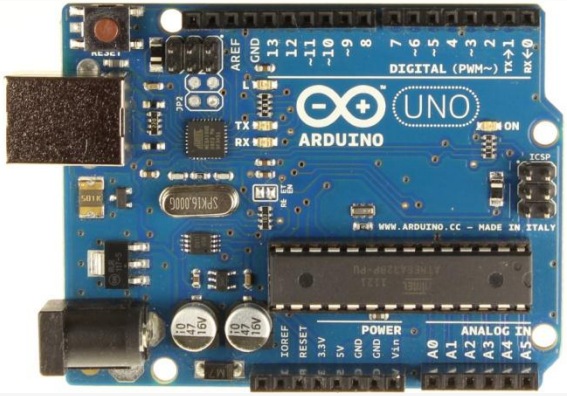
-Simple, one-click compile and upload button—no additional programmer required – Arduino plugs straight into your computer via a USB cable, and communicates via serial.

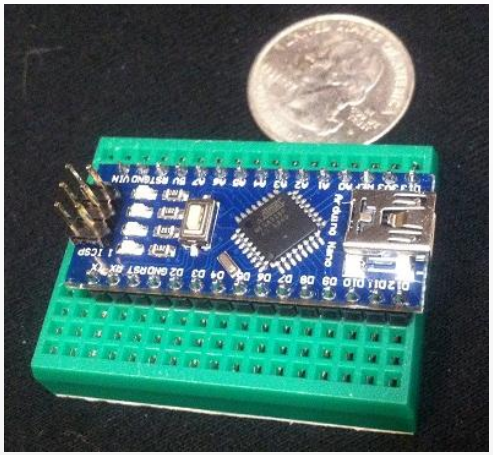
The Arduino Models

(see arduino.cc → Products)

- MANY Arduino, or Arduino-compatible boards are available, but here are just a few of the primary ones I use:

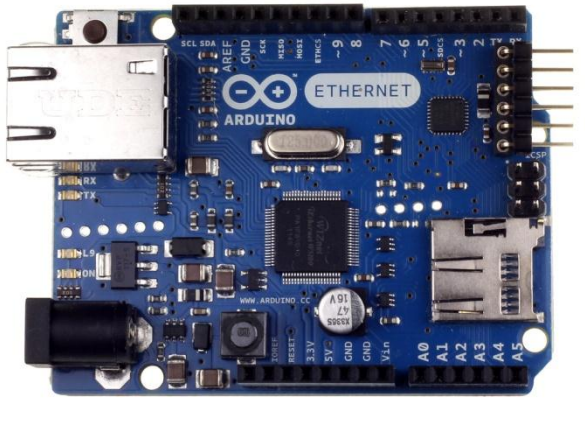
Uno

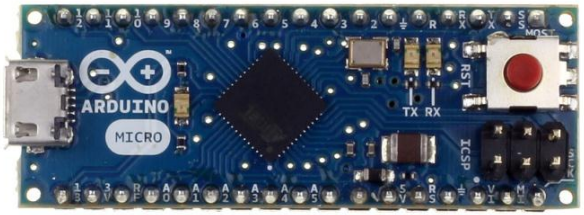




Nano (my personal favorite)

Ethernet





Micro

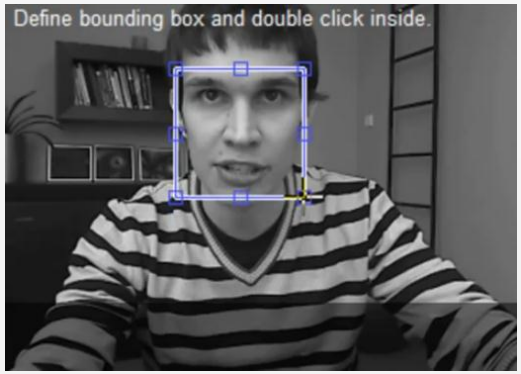
The Ardupilot

- Open source, Arduino-compatible
- Basically is an Arduino Mega 2560 w/integrated sensors & IMU (Inertial Measurement Unit), barometric pressure sensor/altimeter, etc.
- VERY Low cost, ~$250, as compared to similar commercial UAS autopilots which cost $5k~$15k.

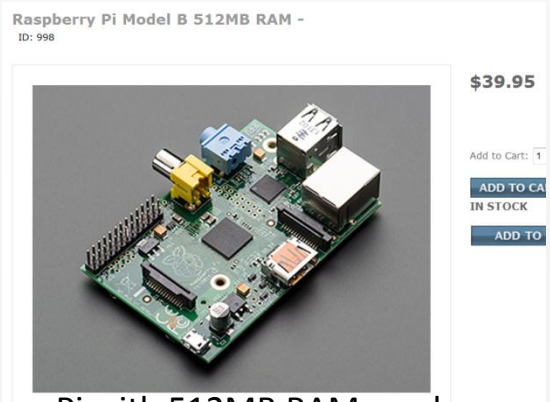
- More info at ardupilot.com
- Purchase at 3drobotics.com or hobbyking.com

~~What Can Arduino Do?~~
What Can Arduino NOT Do?

- In short: IT CAN DO ANYTHING (almost).
- It is limited only by (in this order):
  - #1 your imagination
  - #2 your ingenuity and skill
  - #3 its processing speed, memory, and hardware capabilities
    - This is nearly always the *least* important of them all, but if you want to do something more processor-intensive, such as **visual image processing**, look into using something more powerful like the Raspberry Pi with the OpenCV (Open Source Computer Vision) Library, Open TLD, or Predator, for instance.



Dr. Zdenek Kalal's (from the Czech Republic) Open TLD algorithm demonstrating facial Tracking, Learning, & Detection.



Raspberry Pi with 512MB RAM, as shown on the sales page at Adafruit.com – runs a light version of the Linux OS – can do real-time vision processing, if required, on a UAV.

Where are microcontrollers used now? -based on my understanding

- Digital microwaves, ovens, toasters
- Digital blenders
- Automobile fuel injection (this is your car's "computer")
- Radio Control Devices
  - Tx/Rx, ESC
- Printers, phones
- Some watches
- "smart" clothing (GPS-guided backpacks, blinking ties, shoes, shirts, etc.)
- TV remotes
- Satellites, robots, calculators, etc.
- Pretty much anything with a digital display…

Get Involved with Arduino--Use it In Your Research: Lab Use Examples

- Read in analog and digital voltages
  - Ex: all sorts of sensors, buttons, potentiometers (scroll wheels/knobs), etc.
- Write out analog and digital voltages (analog is done via PWM)
- **Controls, interfaces, & displays, incl. touch screens**
- Display information on LCD displays, Nokia cell phone TFT displays
- Commutate/control DC and AC motors, servos
- **Transmit wireless information (such as for datalogging, telemetry, wireless coms, etc)**
- **Communicate with MATLAB or FreeMat via a USB cable and serial communication**
- **Talk over Ethernet cables or internet WiFi**
- I^2C (2-wire), Serial, & SPI communication protocols; IR-communication; Dallas 1-Wire
- Act as a web server
  - Display info. on your laptop in a browser
  - Receive commands from your laptop via a web browser

Get Involved with Arduino--Use it In Your Research: Lab Use Examples (continued)

- **Detect/sense pressure, altitude, temperature, humidity, GPS coordinates, capacitance, liquid levels, soil moisture levels, human touch, light, color, whatever…just** find the right sensor or use your own ingenuity to make a sensor!
- **Datalogging and instrumentation (replace a $4500 LabVIEW License and DAQ with a $35 Arduino, for instance)**
  - Log directly to a file on your PC, via a USB serial connection from Arduino to MATLAB/FreeMat, Processing, MegunoLink, MakerPlot, Java, Python, etc.
    - Commerical licenses for MegunoLink/MakerPlot are on the order of $40~$90 per computer
  - Live data plotting using any of the above
- Controls/Run control & regulation algorithms
- **Fly an airplane or SUAS (ex: Ardupilot)**
  - Read and Write Radio Control PWM and PPM signals
  - Read brushless motor pulses to calculate motor RPMs
- Robotics: virtually all robots use microcontrollers somewhere, though some rely on more powerful microprocessors
- **Power control – turn AC or DC circuits on and off via transistors/relays**
- **Open & close programs on your computer, by directly emulating a mouse/keyboard** --very useful lab applications: remotely control computer processes & applications

More on datalogging w/Arduino

- Arduino has max analogRead of 10kHz. Max digitalRead is much faster.
- If using serial to send the data to a PC, you are limited to a baud-rate of 115200. **In my Arduino to MATLAB code, I can send 9x 4-byte floating point values at a rate of ~310Hz in either direction.**
- If using an Arduino Ethernet and **UDP** packets, rather than serial, this rate could be **MUCH higher.**

What is a Small Unmanned Aerial System (SUAS)?

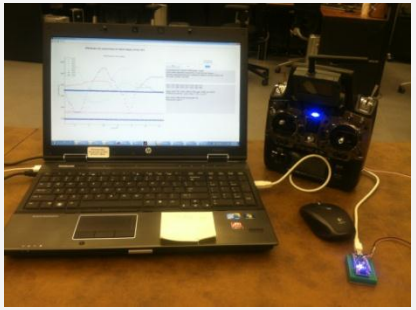
Vicon Flight Lab

- Includes:
  - UAV (Unmanned Aerial Vehicle)
  - GCS (Ground Control Station)
  - Sensors, communication/telemetry, GNC (Guidance, Navigation, & Control) algorithms, etc.

Fearless Foamie, “FF5000”

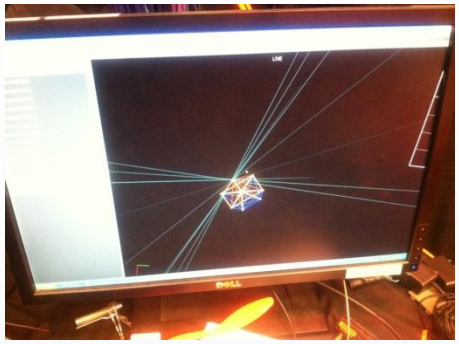
Arducopter

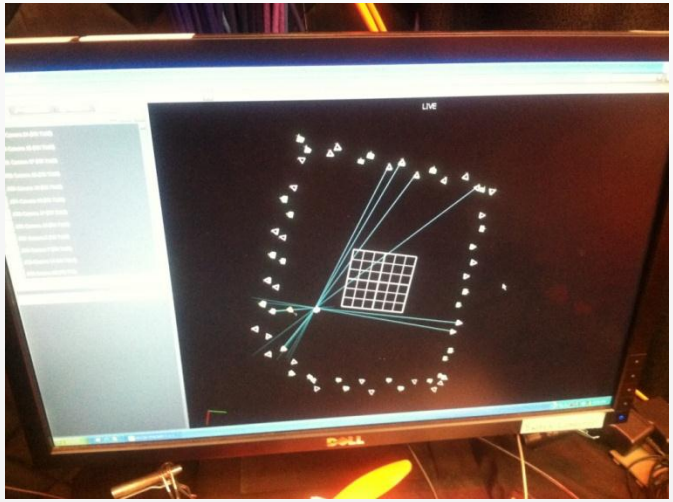
Fearless Foamie FF5000 (left) & ARDrone (right)



Laptop, Arduino Nano, & Turnigy 9XR Radio

Vicon Motion Capture System: like an indoor, sub-millimeter*-precision GPS system & 10-DOF** IMU (Inertial Measurement Unit) all in one!

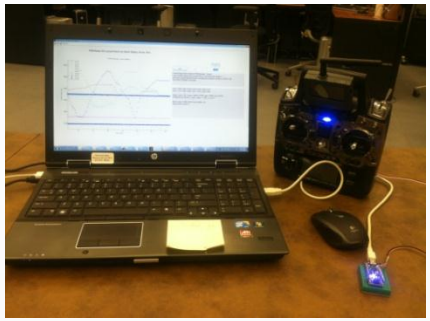


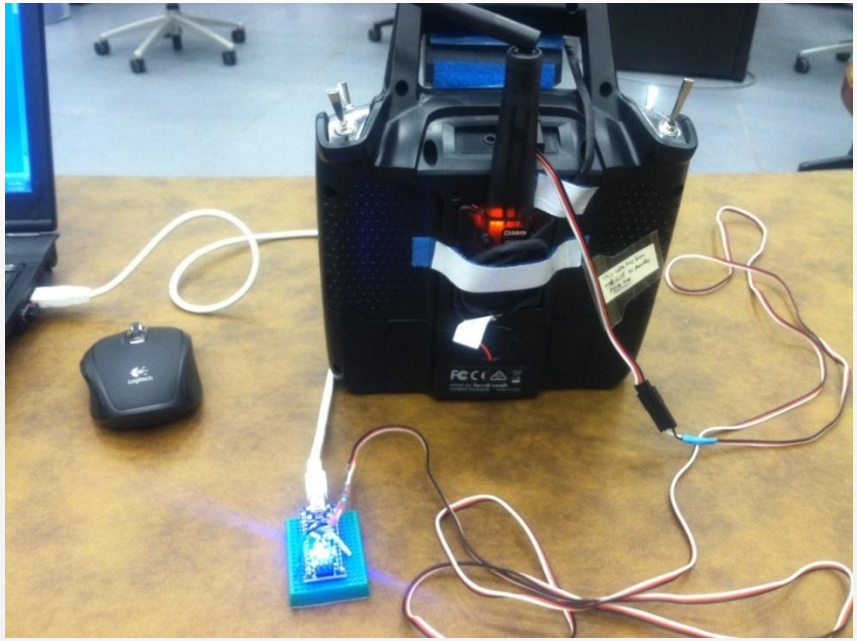


**Vicon precision in our lab is <0.1mm*

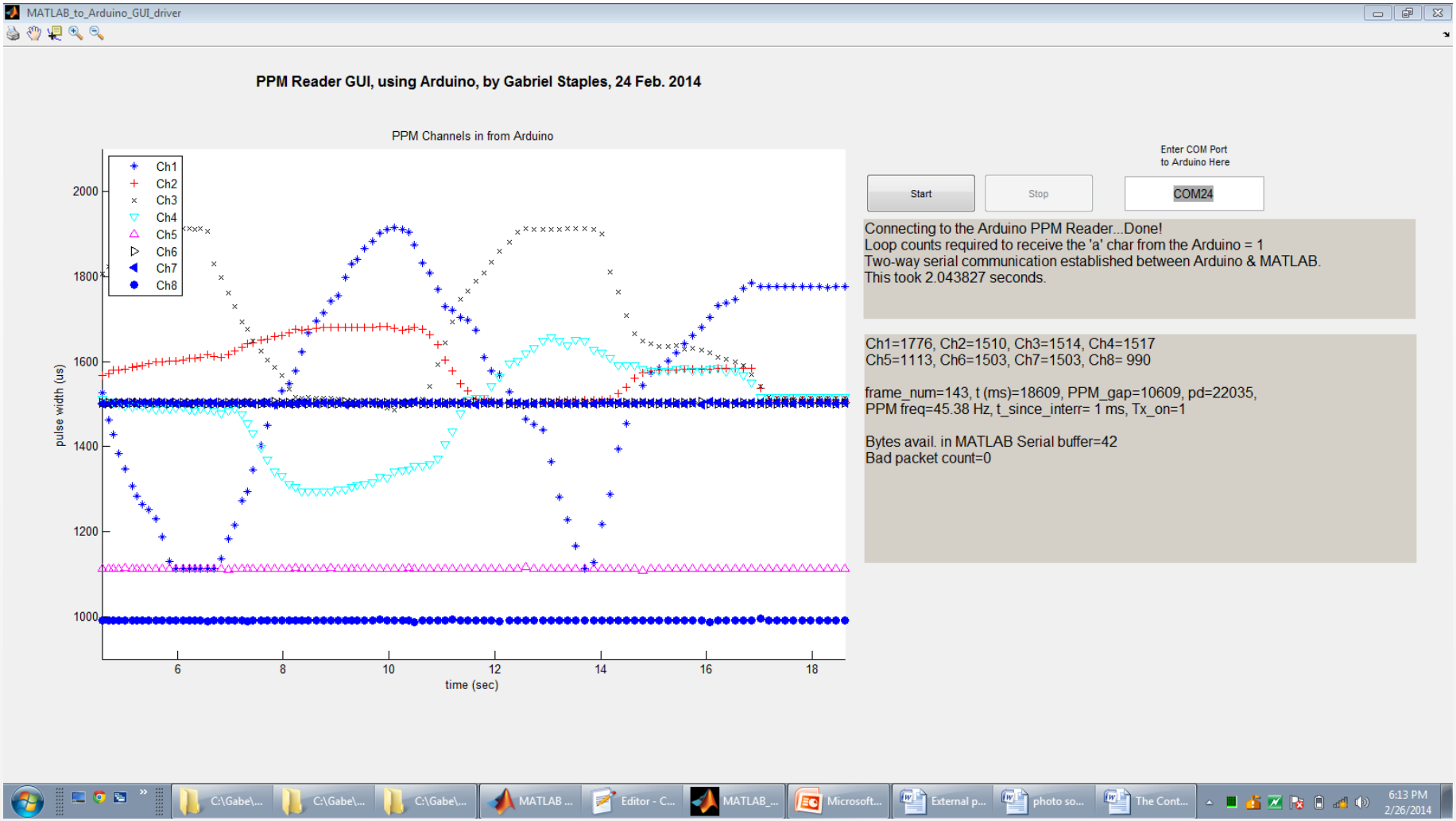
***Note: to get on-board a vehicle, what Vicon can provide off-board, you'd need a GPS and 10-Degree-of-Freedom IMU, which would entail a 3-axis gyro, 3-axis accelerometer, 3-axis magnetometer, barometric pressure sensor, & temp. sensor.*

Key Interfaces – Arduino/MATLAB PPM Reader (reading PPM signal *from* RC Radio)



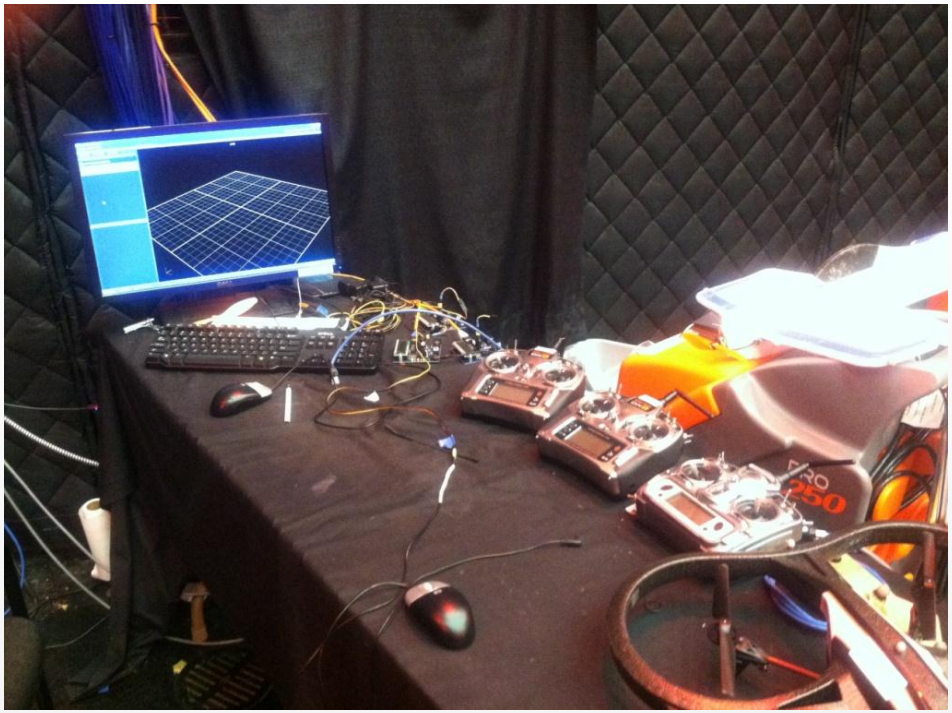


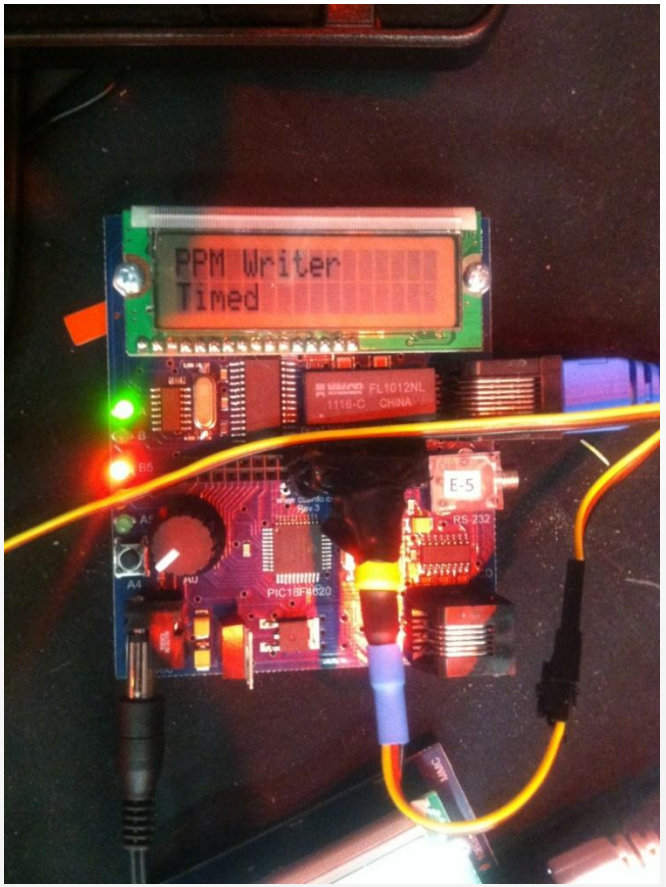
Key Interfaces – ex: Arduino/MATLAB PPM Reader from RC Radio



Arduino/MATLAB PPM Reader Demo

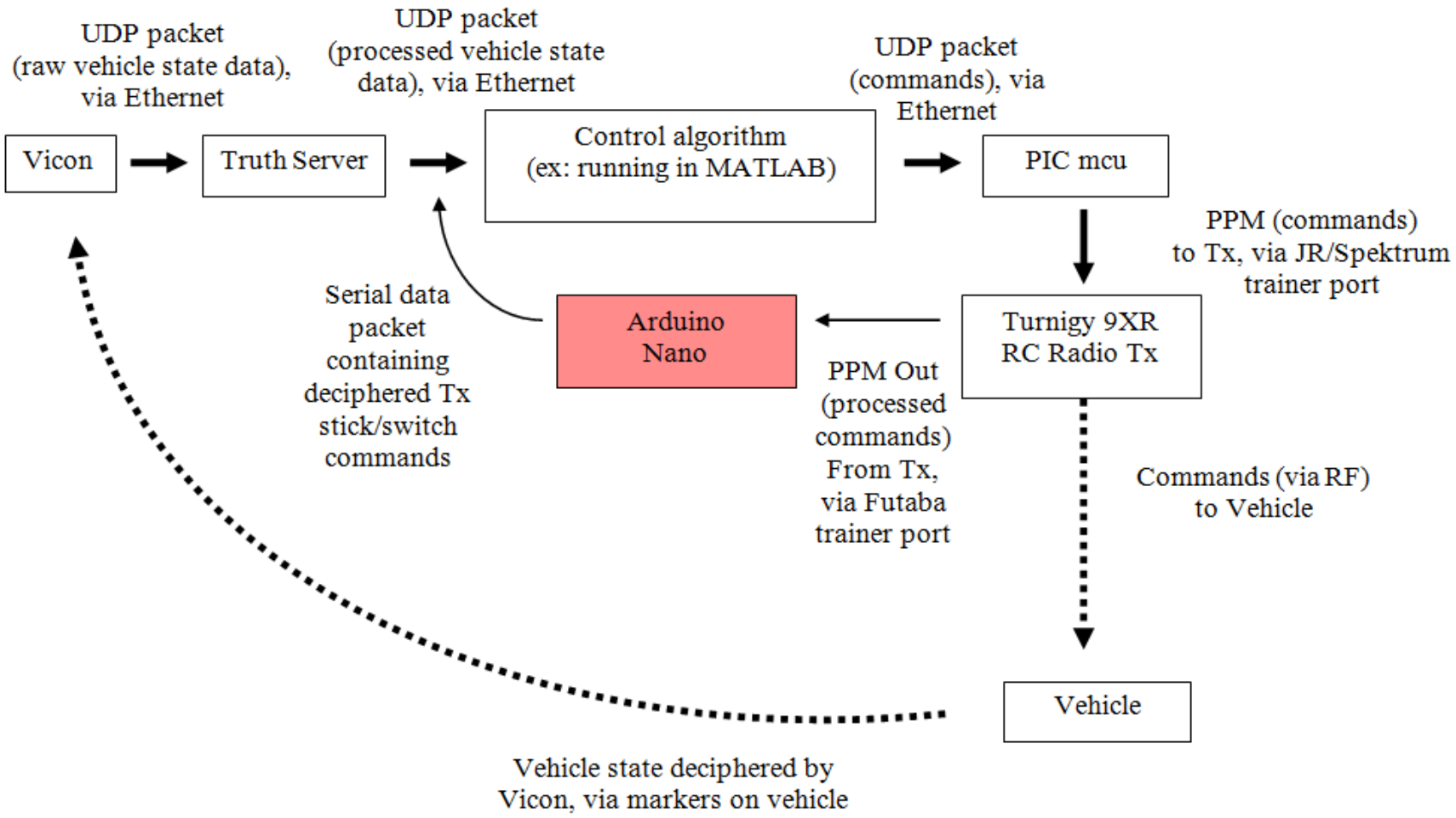
Key Interfaces – PIC-brand microcontroller, creating PPM signal to go *to* the trainer port on an RC Radio – allows autonomous control of a standard RC vehicle, by means of a computer flight controller algorithm which flies the airplane!





JR/Spektrum trainer in (PPM in) on Turnigy 9XR Radio

Control Loop (as it is) – Arduino parts in Red

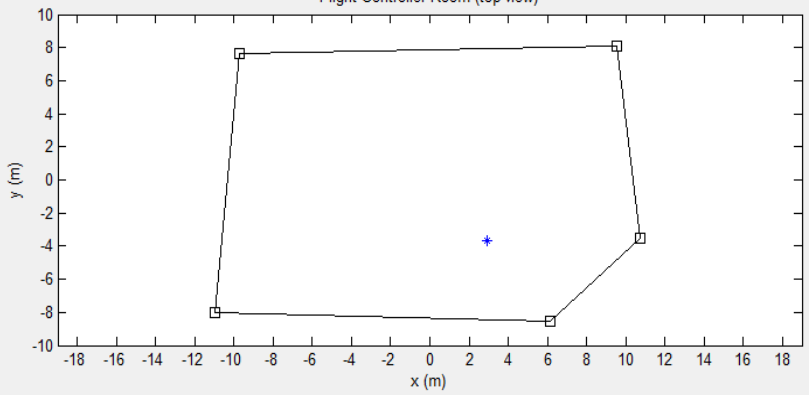


My MATLAB Flight Controller GUI
-for Arducopter; incomplete at this time;
at the moment, this is an altitude controller only

Flight_Controller_GUI2

Arducopter Flight Controller GUI, by Gabriel Staples, 25 Feb. 2014

Flight Controller Room (top view)



Alt (m)
0.077

Enter COM Port
to Arduino Here

Start | Stop | COM24

Connecting to the Arduino PPM Reader...Done!
Loop counts required to receive the 'a' char from the Arduino = 1
Two-way serial communication established between Arduino & MATLAB.
This took 3.076112 seconds.

Ch1=1113, Ch2=1512, Ch3=1511, Ch4=1513
Ch5=1114, Ch6=1503, Ch7=1502, Ch8= 990

packet_num=11835, t (ms)=553908, PPM_gap=11278, pd=22038,
PPM freq=45.38 Hz, t_since_interr= 7 ms, Tx_on=1

Bytes avail. in MATLAB Serial buffer=42
Bad packet count=0

elapsed time = 552.580 sec, n = 11835, Loop pd = 47.104, commanded freq = 21.429 Hz, actual freq = 21.230

Aircraft (quad)
position (x,y,z) = (2.915, -3.678, 0.077) m
velocity (u,v,w) = (-0.000, -0.000, 0.000) m/s
euler (phi,theta,psi; roll,pitch,yaw/heading) = (0.428, -0.392, -62.416) deg

Altitude Controller:
  alt_des = 2.000 m, alt = 0.077 m
Proportional:
  Kp = 40.000, e_alt = 1.923 m, PWM_thro_Kp = 76.915
Integral:
  Ki = 3.000, Z_dot_int_lim = 0.500 m/s, integral_err = 5.428 m*s, PWM_thro_Ki = 16.285
Derivative:
  Kd = 40.000, Z_dot_manual = 0.000 m/s, Z_dot_TS = 0.000 m/s, PWM_thro_Kd = -0.003

PWM_thro_trim = 1470.0, PWM_thro = 1563.2

Alt. Controller

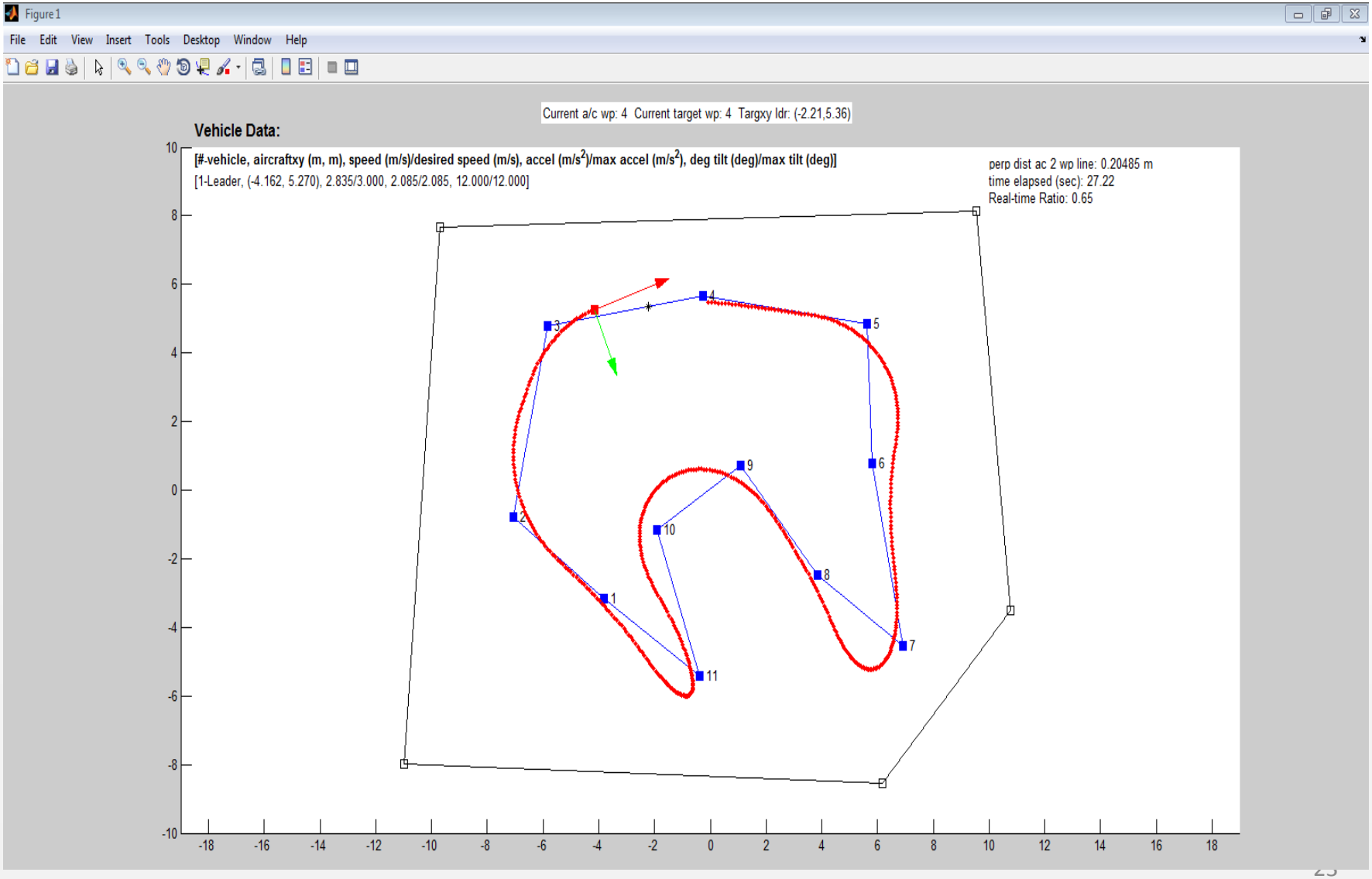
Kp_alt:	40
Ki_alt:	3
Kd_alt:	40
alt_des (m):	2
PWM_thro_trim	1470
Z_dot_int_lim (m/s)	0.5

My MATLAB Flight Controller for Arducopter

1. Already done:
    1. Altitude Controller (PID Feedback only)
2. Still Need to be implemented for full flight controller:
    1. Heading Controller (PID Feedback only)
    2. Pitch/Roll Controller (Physics-based Feed-Forward [I command angles directly, via Ardupilot autopilot], w/passive position feedback only)
3. Additional improvements thereafter:
    1. Alt. Feed-Forward (veh. tilt angle → throttle)
    2. Pitch/Roll PID Feedback on angles, to remove angle bias directly, and vehicle position error indirectly.

Note: All inner loops & vehicle stabilization is automatically handled by the Arducopter autopilot on-board, thereby making point 2.2. above possible, as roll and pitch *angles* can be commanded directly with no external feedback loops, since that is taken care of on-board.

Physics-based Modeling & Simulation, to check path-following ability using my navigation algorithm, given vehicle thrust constraints
– Optional: live simulation demo



Video Demonstrations

1. Dr. Mark Mears's FF5000 Perching

2. Dr. Rich Roberts's FF5000 flight controller

3. Dr. Rich Roberts's RC truck controller

Problem: only one vehicle per RC Radio Tx

- Note: the ARDrone, off-the-shelf, already has on-board WiFi capability, and is controlled via a standard smart phone or tablet---we are currently hacking their control system.
    - But what about standard off-the-shelf RC vehicles?

Possible Solutions for multi-vehicle control, and in-house telemetry:

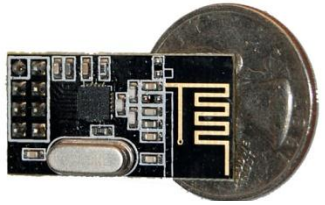
- Wireless WiFi router on ground, w/Arduino WiFi boards on vehicles; 1 router → many vehicles
- Flutter Wireless
- Spark Core (Arduino-compatible WiFi)
- Xbee/Zigbee radios
- RF24 transceivers (my preferred choice) w/Arduino

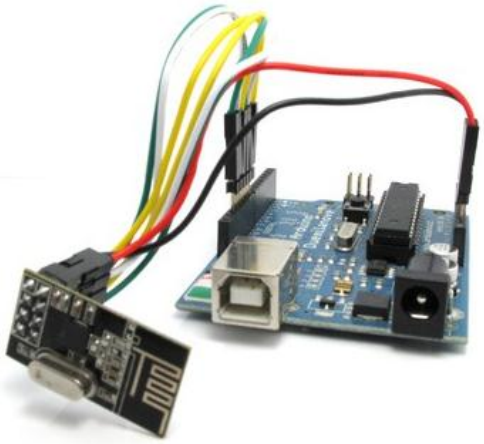
What Next? Telemetry, and multivehicle control— we need coms: Inexpensive 2.4GHz Wireless Transceiver - Nordic Semiconductor nRF24L01+

- Up to 2Mbps data transfer, broadcast out; Rx from up to 6 other devices at once; 128 2.4GHz channels—can be made to be spread-spectrum/frequency hopping if you write your own custom algorithm

- Short range: 100m-range version is as little as $1.50 OR LESS, w/FREE shipping, on Ebay, brand-new

- Long range: 1km range, ~$18; (ex: http://yourduino.com/sunshop2/index.php?l=product_detail&p=190)





More info:
http://arduino-info.wikispaces.com/Nrf24L01-2.4GHz-HowTo
http://maniacbug.github.io/RF24/

Telemetry - Sensors

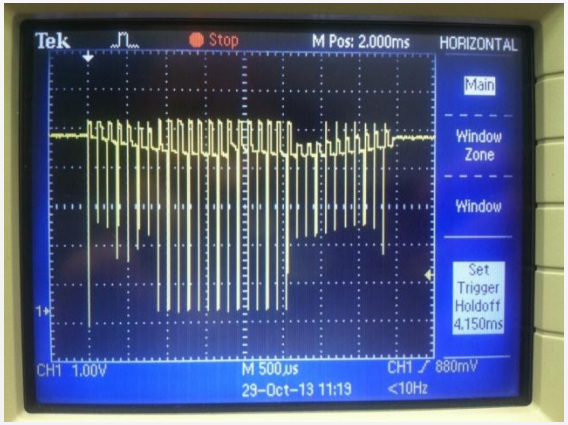
1) Collect the data using Arduino
2) Send the data to the ground using an nRF24L01+ connected to the Arduino

Sensor Data to Collect:

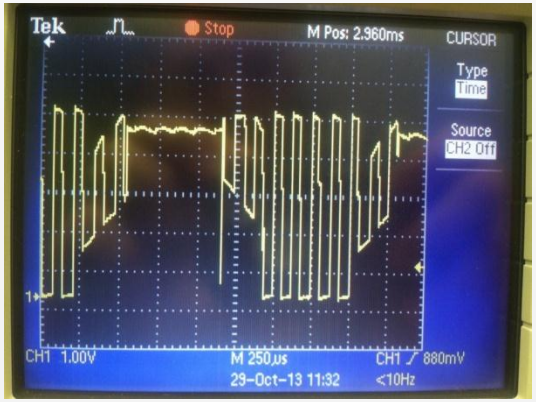
- Aircraft battery voltage – easy, use voltage divider and an analog input on the Arduino
- Current – a little harder, use low-ohm resistor (possibly with an op-amp), and an analog input on the Arduino
- Motor RPMs – much harder, but I've already solved it – use voltage divider and digital inputs on Arduino

Motor pulses (voltage across 1 motor lead & battery ground)

High-freq signal = throttle PWMing @ 8kHz, low-freq signal = commutation pulses

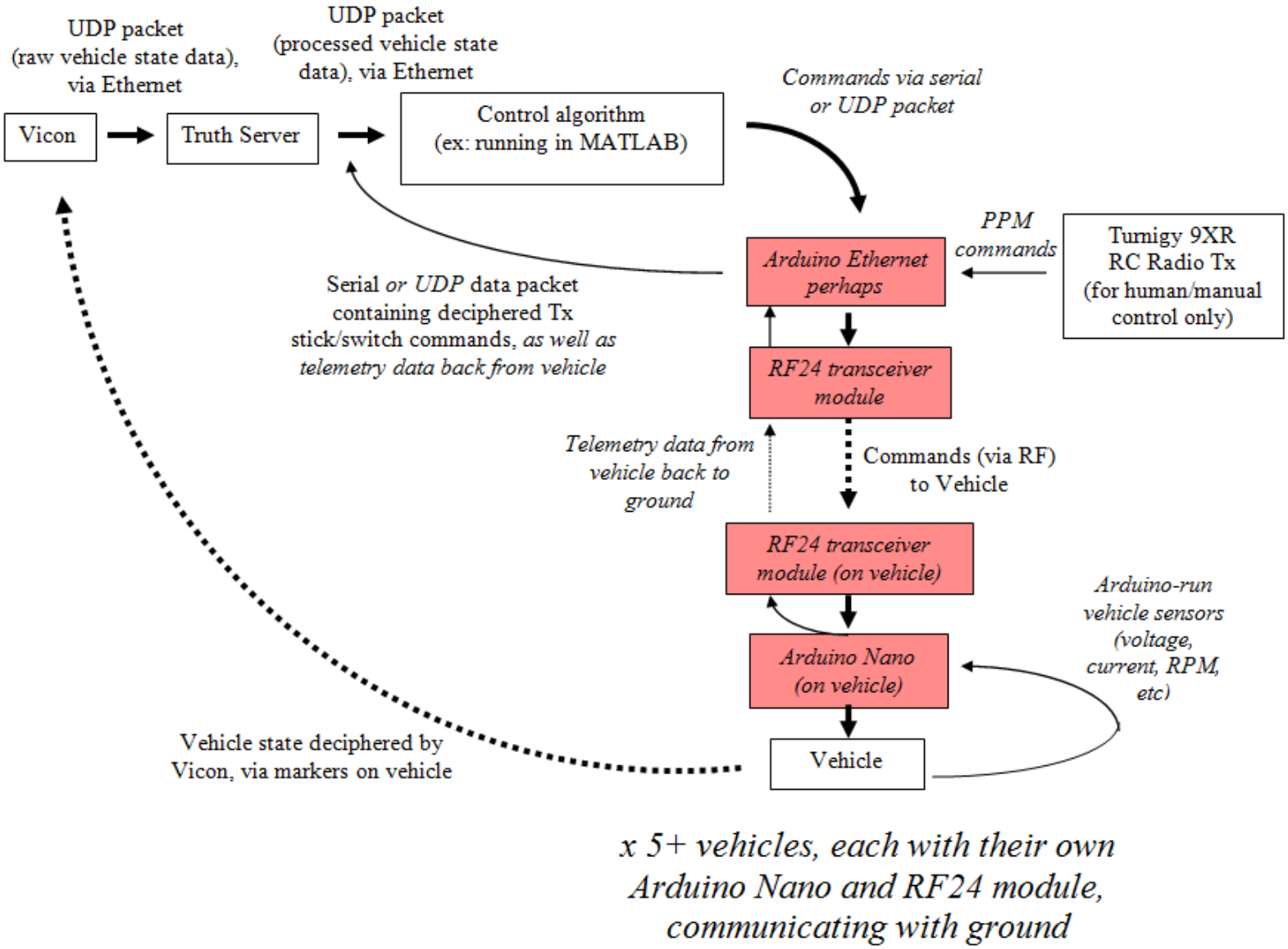


Low throttle setting



Mid-throttle setting

Control Loop (as it will be) – Arduino parts in Red, *other changes in italics*



You could even…

- Get rid of the off-board controller (MATLAB in our case) altogether
  - Have the Arduino nano process the flight path on-board
  - Just send the Truth Sever Data (vehicle state, as determined by Vicon) to the vehicle, and the Arduino does the rest
- Add on-board sensors, optical flow, computer vision
  - may require upgrade from 16 MHz Arduino Nano to 84 MHz Arduino Due, 400 MHz Arduino Galileo, or 700 MHz Raspberry Pi, etc.

Contacts & Resources

- Gabriel Staples, AFRL/RQVA – gabriel.staples.1@us.af.mil
- Learn More: http://electricrcaircraftguy.blogspot.com/2014/01/the-power-of-arduino.html --lots of general info. & many great, additional links here
- Tutorials & Help:
  - http://arduino.cc/
    - → Reference
    - → Products
    - → Learning – Examples
    - → Learning – Playground
    - → Support – Forum
  - http://learn.adafruit.com/category/learn-arduino - ADAFRUIT IS A TREMENDOUSLY VALUABLE RESOURCE FOR THE BEGINNER TO ARDUINO….ESPECIALLY TO THOSE NOT WELL-VERSED IN ELECTRONICS, PROGRAMMING, AND ELECTRICAL ENGINEERING.
  - https://www.sparkfun.com/
- Where to Purchase:
  - Personal:
    - Amazon (fastest), Ebay (cheapest), Adafruit and SparkFun (best tutorials, code, & support), or the below sites
  - Work/Other:
    - http://www.adafruit.com/
    - https://www.sparkfun.com/
    - Amazon.com
    - Digikey
    - Mouser