**FPV Drone Project**

**Project Goal:** The goal of this project is to learn more about embedded real-time systems, flight mathematics/computation, computer vision, radio signal communication, electronics and machine learning. To do so I will research and develop a DIY FPV drone focusing on the firmware and applications to further advance my skills as a software engineer and have fun!

**Definitions:**

LVL 1: Basic Drone – Capable of manual flight using a flight controller

LVL 2: Video Capture Drone – Capable of recording and saving video during a flight

LVL 3: FPV Drone – Capable of live streaming, and saving video during a flight

LVL 4: Autonomous FPV Drone – Capable of performing instructions using camera(maybe LIDAR) and onboard(maybe offload to laptop) AI systems to discern what actions need to be taken.

**Requirements:**

**Functional:**

* All systems must use a real-time operating system
* The drone must be capable of traveling at least 3km away from RF Controller while still under control(long range frequency)
* The drone must be capable of recording video
* The drone must be capable of live-streaming video feed via a short range frequency at a minimum distance of 1km(might need an amateur radio license to do this)
* Drone must be able to continuously fly under perfect conditions for 45 minutes
* Drone must be able to function and complete tasks autonomously
  + Does this require an app to do so?
  + What tasks?
    - Follow a person/object(5m following distance(adjustable?)
    - Dodge incoming projectiles(Object avoidance system)
    - Map out a living space(rooms, floors, etc)
* The drones systems must output and save flight logs that can be used to debug, verify and fine-tune flight functionalities

**Non-Functional:**

* The drone’s camera must be capable of recording 4k 30fps footage
* The drones camera must be capable of recording 1080p >200fps footage
* The video feed from drone must be smooth
* The drone should be cost-effective and competitively priced compared to pre-built market
* The drone should charge in < 1hour
* The drone should be capable of withstanding crashes
* The drone should hover/slowly descend when connection is lost via RC controller
* The drone should not be too loud
* The drones autonomous systems should interface with Unix based Application(maybe windows) that can display the drones live feed, communicate flight instructions and receive mappings of rooms/floors in a home
* Status should be transmitted back to the controller/ application including battery status, distance from controller, altitude, pitch, yaw, roll

**Research & Development Plan:**

**Stage 1: LVL 1**

1. Research and aggregate the best value/reasonably priced hardware needed to make a *Basic Drone and a Video Capture Drone*
2. Perform Cost benefit analysis on each item necessary for building a *Basic Drone*
3. Ensure final hardware specifications align with the functional/non-functional requirements of this project
4. Order parts needed for a *Basic Drone*

**Stage 2: LVL 1**

1. Research needed software development tools to create firmware for flight controller, and auxiliary systems monitoring
2. Learn mathematic principles behind quadcopter flight:
   1. <https://www.quora.com/How-difficult-is-it-to-build-a-flight-controller-from-scratch-and-programming-it-our-own>
      1. Control Theory(Closed Loop-Control)
      2. Observability and Controllability
      3. Nyquist Stability Criterion
      4. Rotation angles(Quaternions)
      5. Direction Cosine Matrix
      6. Integral Wind-up -> big issue for others
      7. Tune PID loops(Ziegler-Nichols Method)
      8. Digital Filtering using Extended Kalman Filter
      9. Quake fast inverse square root function
3. Develop **state machines, use case and sequence** **diagrams** for the expected actions and performance of drone manual flight and communication
4. Ensure flexibility in flight controller I/O so that another system can communicate with it in the future
5. Develop Flight Controller, Electronic Speed Controller, and IMU firmware

**Stage 3: LVL 1**

1. Assemble drone
2. Integrate software into drone
3. Test drone for all use cases and reiterate/fix and improve on software until all requirements have passed

**Stage 4: LVL 2**

1. Research compact versatile, durable video capturing devices
2. Research microcontrollers, and short range video transmitter/ receivers
3. After performing a cost-benefit analysis purchase a short range video Tx/Rx set, camera, and microcontroller
   1. Microprocessor must in theory be capable of performing **LVL 4** tasks
4. Integrate video-radio Tx/Rx devices and camera into drone and controller
5. Have camera output video to a SD card(on drone) while also live-streaming over short radio wave to RC Controllers screen(try to use a smartphone or better Laptop screen for this)
   1. This needs to all be possible using just the controller no manually pressing record etc
   2. So a dedicated **video capturing system**(with a lot of extra processing/memory capabilities) needs to be implemented
   3. Again design state machines, use case diagrams, and state machines to help guide development process
6. Test and reiterate on solution as needed

**Stage 5: LVL 3**

1. Research FPV drone goggle options(maybe goggles are not necessary save costs and develop a UI on laptop instead)
2. Perform cost-benefit analysis and make a purchase ensuring non-functional requirements are met
3. Add new feature in Video Capturing System enabling livestreaming of feed to both a screen and goggles
4. Ensure functional and non-functional requirements are met

**Stage 6: LVL 4**

1. Research computer vison and AI topics:
   1. Object tracking
   2. Room mapping
   3. LIDAR vs camera or both
2. If any additional purchases need to be made to enable LVL 4 Drone capabilities perform a cost-benefit analysis and purchase any necessities
3. Create a C++ application that can run on the video capturing processor to enable LVL 4 FPV Drone specifications
   1. Reroute local drone 4k SD card output to the processor
   2. Research and develop object tracking and room/house mapping applications using C++
   3. Integrate Computer Vision system with current Video Capturing System
4. Integrate Computer Vision System into FPV UI
5. Capture AI Logs which describe what the drone is “thinking” after being given commands
6. Test, verify and refine application on hardware

**Stage 7: Field Testing**

\*Although the drone will be tested throughout find the limits of what the drone is capable of and compare to initial requirements

1. Max Speed

2. Max Altitude

3. Max Video Live-Stream Distance

4. Max Travel Distance

5. Flight Duration

**Stage 8: Reflection**

Reflect on what was learned, how the drone can be approved and what should have been done differently.

**Timeline:**

**LV1 –** 1.5 months

**LVL2 –** 1 week

**LVL3 –** 3 weeks

**LVL4 –** 1 month++++++this may be ongoing

**Jots**

* Links
  + Reddit Comments with processor specifications needed: <https://www.reddit.com/r/diydrones/comments/141u83p/beginner_looking_to_build_a_drone_from_scratch/>
  + Flight controller processors: <https://oscarliang.com/f1-f3-f4-flight-controller/#:~:text=in%20Flight%20Controllers-,F1%2C%20F3%2C%20F4%2C%20G4%2C%20F7%2C%20and%20H7,of%20FPV%20drone%20flight%20controllers>.
  + Quora processor specifications and flight coordination principles: <https://www.quora.com/How-difficult-is-it-to-build-a-flight-controller-from-scratch-and-programming-it-our-own>
  + Someone’s notes on building a drone: <https://www.instructables.com/Quadcopter-and-DIY-Flight-Controller-Basics/>
    - Motors
    - Battery
    - ESC (Electronic Speed Controller) uses pulse width modulation
    - IMU (Internal Measuring Unit)
* Camera:
  + GoPro Bones (54g)
  + GoPro Hero 11 Mini (133g)
  + Dji osmo action 4
* HDZero Goggles for flexibility
  + Is open source ☺
  + Discount link: <https://atxairborne.com/hdzero/hdzero-goggle/hdzero-goggle-review/#:~:text=The%20mechanism%20works%20well%2C%20and,them%20when%20handling%20the%20goggle>.
* Radio Controllers
  + TX 12 MK 2
    - Open-source
    - $99
    - OS preinstalled for tx/rx
* Radio Controller(Rx/Tx Drone side)?
* Audio/Video Transmitter and receiver
* Frame (Gotta be carbon fiber ☺)
  + ???

Research(Books, vids etc)

* Designing flight controller from scratch : <https://www.youtube.com/watch?v=mMRLaRf0kvc>
* “Feedback Control of Dynamic Suystems” by Gene F. Franklin
* “Control Systems Engineering” by Norman S. Nise
* <https://github.com/rizacelik/STM32F411CEU6_INAV_Firmware>
  + Manz made a whole guide but for the STM32F411CEU6
* Guy made a whole ass tutorial: <https://www.reddit.com/r/diydrones/comments/17dw3o8/coding_custom_flight_controller/>
* https://iopscience.iop.org/article/10.1088/1742-6596/2483/1/012034/pdf