**Design Deliverable**

**EE 478 Final Project**

**Quadcopter Cameraman**

**Written By:**

Rebecca Chu

Phongsakorn Liewsrisuk

Jian Ma

**DESIGN SPECIFICATION FOR QUADCOPTER CAMERAMAN**

**Design Specification Overview**

The quadcopter cameraman system is a system that provides the ultimate automatic photographic and videotaping solutions for travelers, who share our resentment of devoting the majority of holiday time for taking pictures and recording videos. The quadcopter cameraman is a First-Person-View (FPV) quadcopter that is designed for easily capturing the environment around the proximity of the traveler. It is the first step towards the ultimate goal of fully automation of the work done by a professional cameraman, so that an average person can have his or her time of life well documented.

To achieve this goal, the quadcopter in this project incorporates a series of design that enable a user-friendly interface, robust camera functions, and most importantly, an agile quadcopter system that is able to perform flexible maneuvers and could be operated safely in indoor and outdoor environments. The quadcopter cameraman is composed of a FPV and a PC communication base the quadcopter connected to via WIFI, which sends commands to and receives data from the quadcopter. The quadcopter is able to be fully controlled from the platform. It can take off vertically and hover at a stable position before being commanded to land. The quadcopter is expected to fly between 2 to up to 20 minutes in the air depending on the complexity and intensity of maneuvers involved. Upon receiving command from the user, the quadcopter could perform picture-taking and other photographic functions. Additionally, the quadcopter supports a precise discrete 360 degrees turn maneuver, similar to a stepper motor in order to create panorama images. While in the air, the quadcopter is able to capture videos and store them in the on board PC. The quadcopter itself is designed to be as light as possible in order to maintain a long hovering time. Additionally, the quadcopter system itself is flexible to additional features as the whole system develops and updates.

**SPECIFICATION OF EXTERNAL ENVIRONMENT**

The quadcopter need to operate in an industrial environment in a commercial grade temperature.

Also, the quadcopter need to operate in an indoor environment which provides stable and high-bandwidth WIFI signal coverage to be able to communicate with the drone. The weather condition for this project is assumed to be relatively mild, which means, no strong wind, rain, snow. The temperature is assumed to be varying around room temperature and so does humidity.

Commercial grade: 0 °C to 70 °C

**SYSTEM INPUT AND OUTPUT SPECIFICATION**

**System Inputs**

The system the following environment inputs

Range to frontal objects ranging: up to 3 m **±** 0.1m.

Barometric pressure changing from: altitude 33.5m (Seattle ground) to 100m, **±**15cm.

Angular orientation of the quadcopter: for all axis up to 30**°** ± 0.01**°**

Angular velocity: 0 to ± 180 degrees/s

Acceleration: for all axis 10g ± 0.1g

Environmental Temperature: 0 to 100C**°, ±**1C**°.**

The system takes the following user inputs

Flight commands:

Zero quadcopter

Change altitude [expected altitude]

Start recording

Stop recording

Take single pictures

Take a panorama view (automatic rotation)

**System Outputs**

Physical roll, pitch and yaw change

Updating a website live streaming pictures and videos

10Hz

100kbytes

**User Interface**

The user will be able to type in different command through Secure Shell (SSH) from remote computer. The quadcopter will then be able to respond to each of the command.

**Mode**:

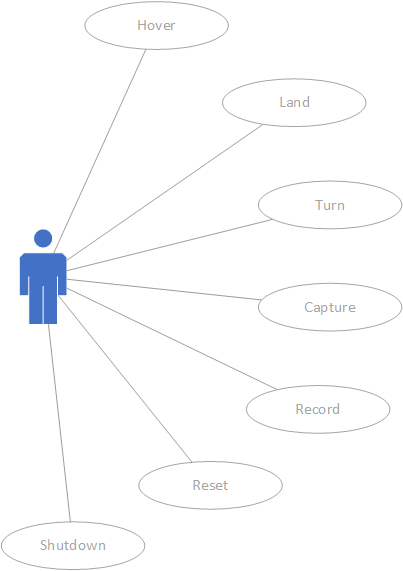
*Hover*, *Capture*, *Record, Land, Turn*

**Reset**:

The reset button will reset the electronic component on the board.

**USE CASES**

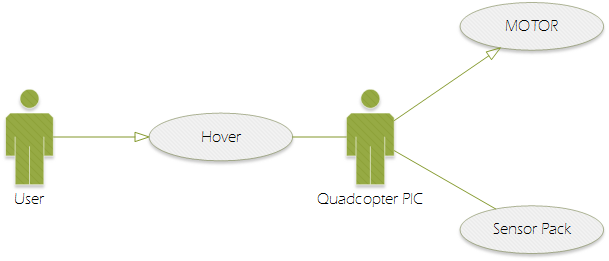
The use case below shows the features quadcopter can do as requested from the user from the PC. The user is represented by the human figure in the use case below. Each bubble connected to the user is the commands the quadcopter are capable of.



**Figure 3.1 Use Case Diagram for Quadcopter System**

***Hover***

The user sends hover command and specifies desired altitude from the PC and the quadcopter is going to fly off the starting platform and elevate to the requested height and remain a stable position. The quadcopter is going to suspend in air for a maximum of 20 minutes, the maximum amount of time the battery can support the hovering feature.

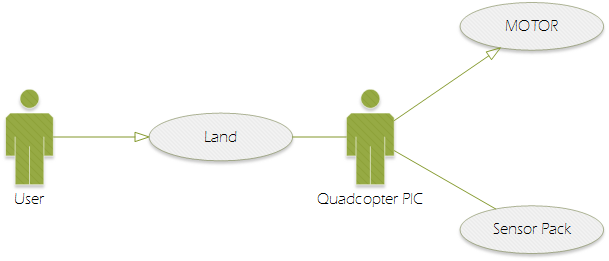


**Figure 3.2 Use Case Diagram for Hover**

If the flying time is approaching its max, the user should send the “Land” command to prevent the quadcopter from crashing.

***Land***

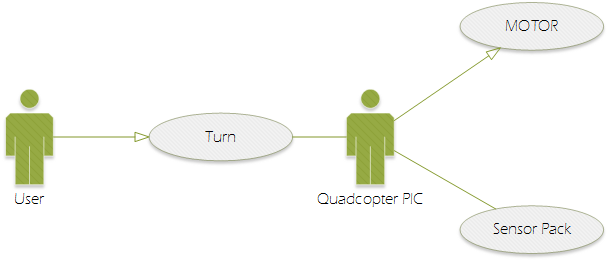
The “Land” command tells the quadcopter to descend from the original flight position back to the original departure platform. The landing feature will allow the quadcopter to safely drop in its’ altitude back to the ground where the four motor driven propellers will cease all its rotations. The quadcopter will terminate all its features exercised in the Hover mode.



**Figure 3.3 Use Case Diagram for Land**

***Turn***

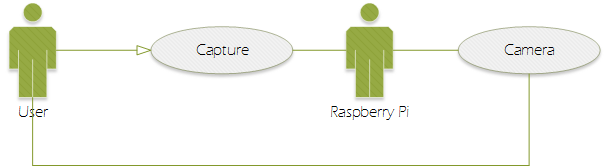
The “turn” command is only valid when the quadcopter is in the hovering state. This command tells the quadcopter to make a 360° turn mid-air. The turn can support either turn in counterclockwise rotation or clockwise rotation, determined upon the user’s request.



**Figure 3.4 Use Case Diagram for Turn**

***Capture***

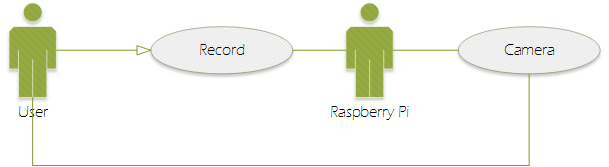
If the user sends a “Capture” command, the quadcopter will activate the camera on the quadcopter to taking a picture at the exact location the quadcopter is at. The camera used is the Raspberry Pi Camera and the captured camera is going to be sent back to the user to view on the PC.  The capture feature can be activated either when the quadcopter is in flight or is stationary on ground.



**Figure 3.5 Use Case Diagram for Capture**

***Record***

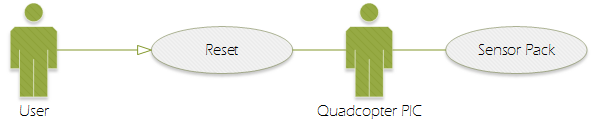
If the user sends a “Record” command, the quadcopter will activate the camera on the quadcopter to start recording the video at the exact location the quadcopter is at. The camera used is the Raspberry Pi Camera and the captured camera is going to be sent back to the user to view on the PC.  To terminate the video recording, the user needs to send another record stop command. The capture feature can be activated either when the quadcopter is in flight or is stationary on ground.



**Figure 3.6 Use Case Diagram for Record**

***Reset***

The Reset feature is to calibrate all the sensors on the quadcopter such as barometer and gyroscope. These sensors determines the altitude and positioning of the quadcopter. The sensors will be leveled in respect to the quadcopter ground position. For example, the quadcopter determines its’ height through finding the difference in distance between the start and the current position. Therefore, reset feature is to zero the starting distance. As a result, the sensor on quadcopter will have a more accurate measurement of relative location from the starting off platform.

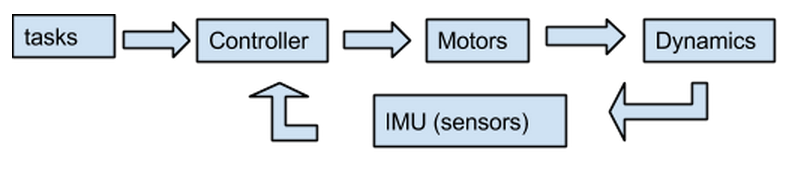
  
  
**Figure 3.7 Use Case Diagram for Reset**

***Exceptions***

When the user specifies a command that the quadcopter does not specify, the quadcopter will not respond but instead continues performing the previous valid command.

**SYSTEM FUNCTIONAL SPECIFICATION**

**Flight Control System Architecture**



**Figure 4.1 Block Diagram**

The task block represents the input command from the users. The tasks can be changed by the command of the user such as hover and land through secure shell from the remote computer to Raspberry Pi. The Raspberry Pi then decode the command then send the signal to the Master pic through UART.

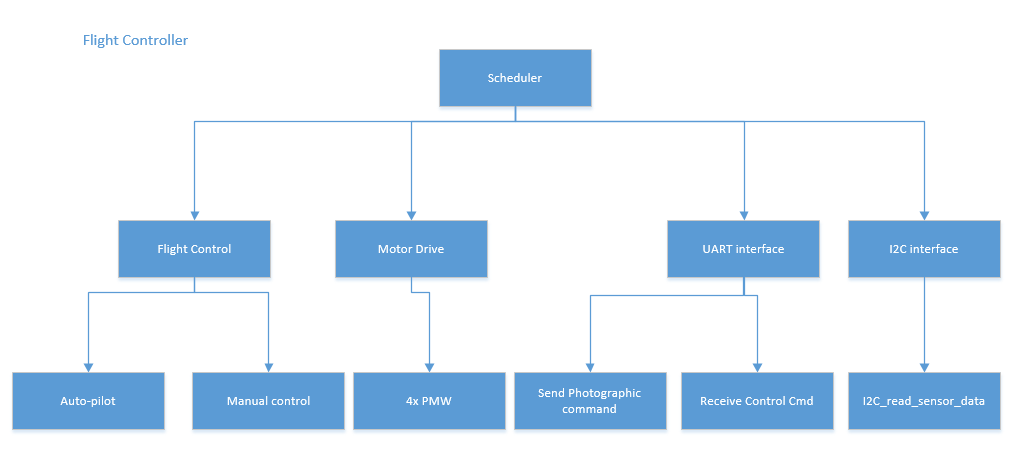
The control block represents the control task to stabilize the quadcopter.  From the sensors, it is possible to have a feedback on the position and provide the automatic stabilization.

The motors block compose of the power board and ESCs which provides the voltage to various components in the system. The Electronic Speed Controller controls the speed of the motor according to the controller.

The dynamic block refers to the propeller's velocity in reference to the initial position of the quadcopter.

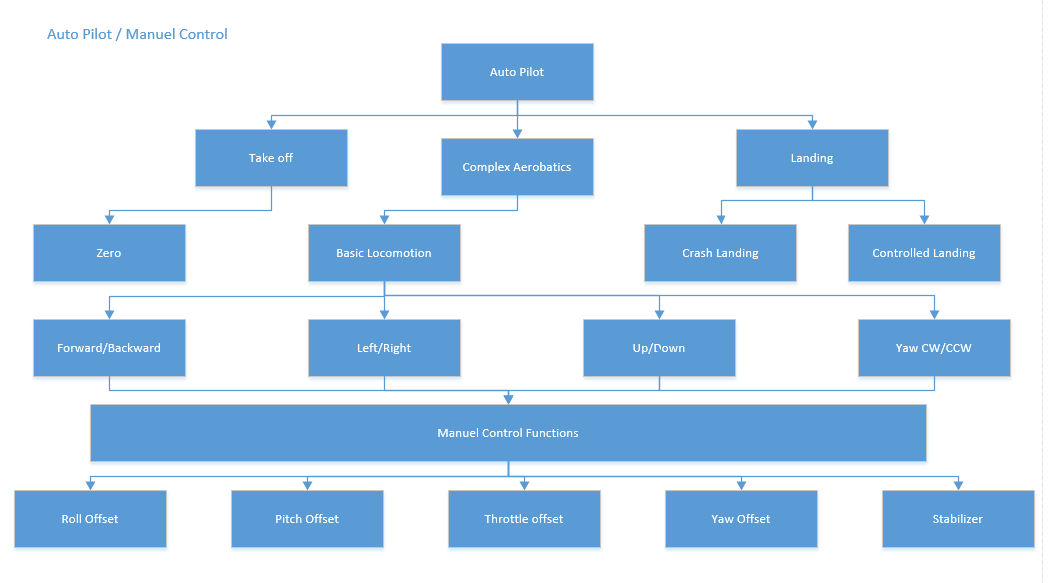
The IMU block provides the information about the quadcopter’ attitude and heading. It has three-axis accelerometer, three-axis gyroscope, three-axis compass and a barometer. The IMU would provide enough information to calculate roll-pitch-yaw angles and send them through I2C from the slave Pic to master Pic.

We then created the following function decomposition diagram to specify the tasks for the system modules, including the flight controller, sensor unit and the remote end (raspberry PI).



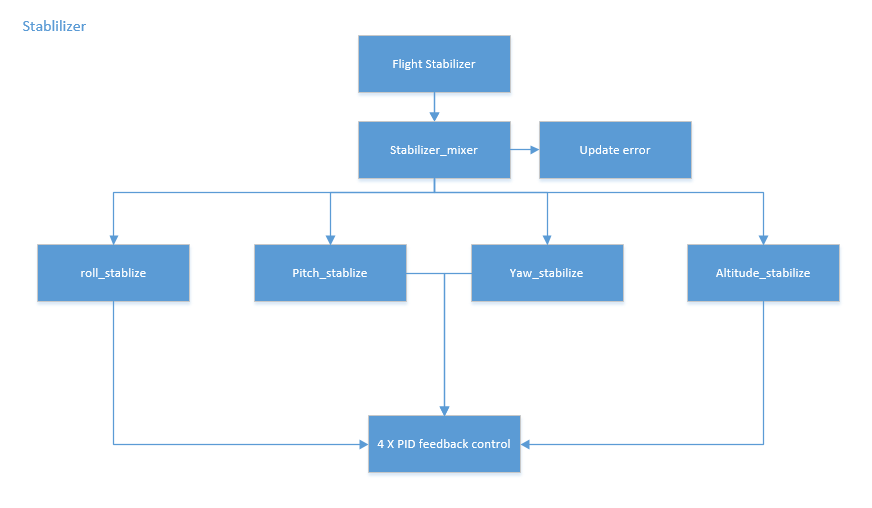
**Figure 4.2 Function decomposition diagram for the flight controller**

Due to the complexity of the flight control function, which includes both auto-pilot and manual control. We listed function decomposition for the flight control separately.

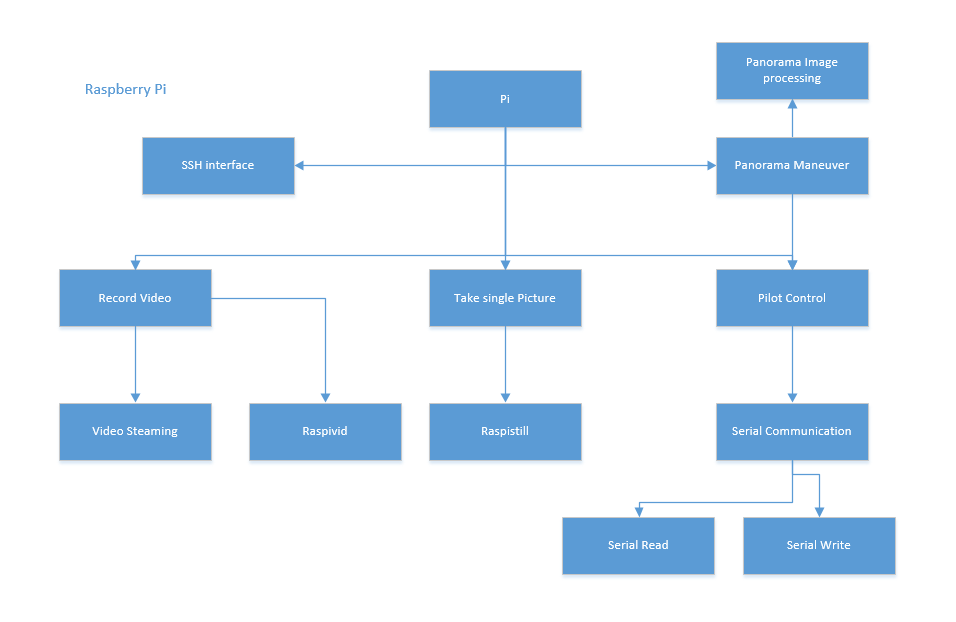


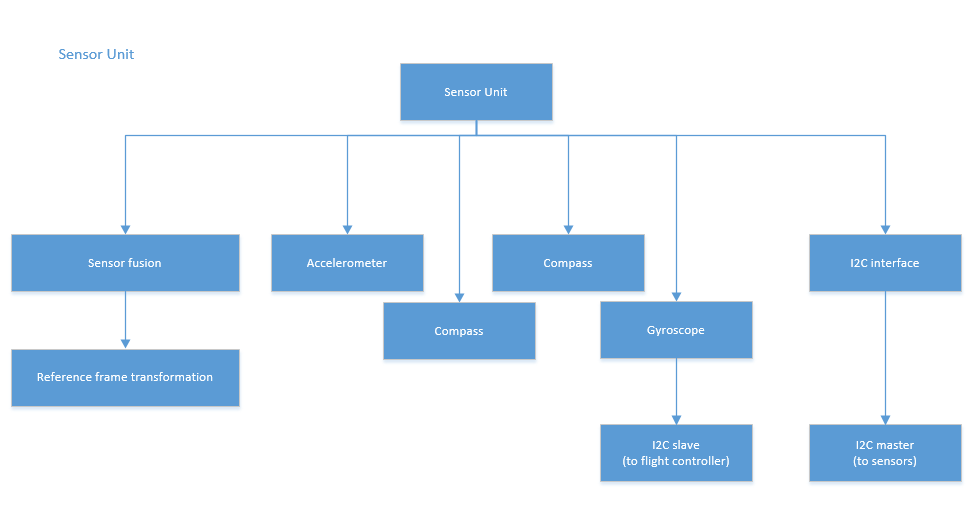
**Figure 4.3 Function decomposition diagram for the Autopilot and Manuel control function in the flight controller**

An important component in the Autopilot/Manuel control function is the stabilizer to maintain the quadcopter in the hovering state. Due to the complexity of the stabilizer. It also decomposed separately below.

 **Figure 4.4 Function decomposition diagram for the stabilizer in the flight controller**

The function decompositions for the remote controller and the sensor unit are also listed below.

 **Figure 4.5 Function decomposition diagram for the remote unit (Pi)**



**Figure 4.4 Function decomposition diagram for the sensor unit in the flight controller**

**Power**

The power supply will provide the voltages at the specified current levels to various component. On the quadcopter, the power board will be mainly supplying power to:  1 Sensor pack, 4  motors, 1 Raspberry Pi, 2 PICs, and 4 ESCs.

|  |  |  |
| --- | --- | --- |
| **Component** | **Voltage** | **Approx. Current** |
| Sensor Pack | 5.0V/3.3V | 1A |
| Motor | 5.0V | 10A |
| Raspberry Pi | 5.0V | 1A |
| PIC | 5.0V | 0.2A |
| ESC | 11.1V | 0.2A |

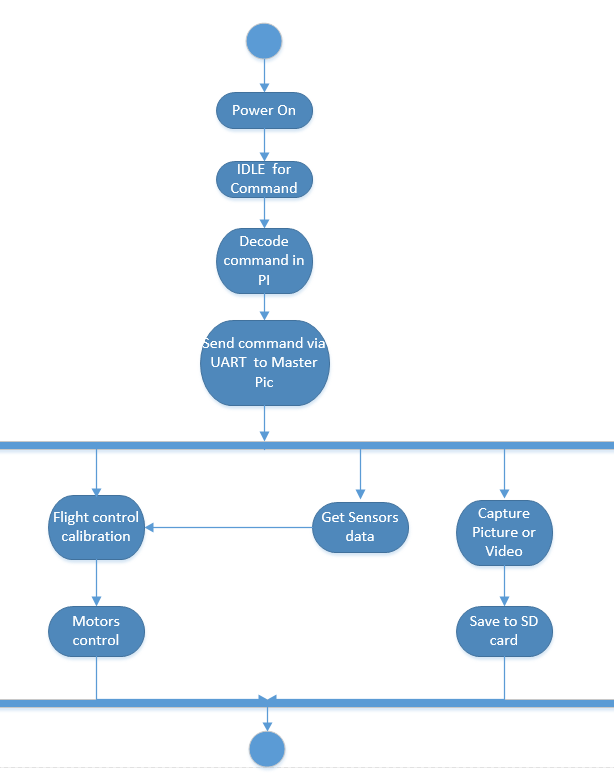
**Table 4.1 Power Requirements for each component of Quadcopter**

The entire quadcopter and its’ components are powered by the 7A Lithium Polymer Battery which outputs 11.1V. The Electronic Speed Controllers (ESC) that controls the four motors will be powered with 11V, straight from the battery pack. The remaining electronics on the quadcopter are all supplied with 5V. The 5V supply is achieved with a 5V voltage regulator that inputs 11.1V from the battery pack and outputs 5V. The 5V voltage regulator is a step down buck switching regulator.

**High Level Diagram**

Upon power on the quadcopter, the system will stay in IDLE mode for command. The command will be received in Raspberry Pi through Secure Shell (SSH) protocol from a remote computer.

The system will have a closed loop feedback control for the stability and flight control of the quadcopter from the input of the collected data from sensors.If the capture command is entered, the Pi camera will respond to the command accordingly.



**Figure 4.2 Functional decomposition for Quadcopter System**

**OPERATING SPECIFICATION**

Temperature Range: 20-50 C° typical, 90 max.

Humidity up to 90% RH non-condensing

Battery charging 120 – 240 VAC 50 Hz, 60 Hz, 400 Hz, 15 VDC

The system shall operate for a minimum of 15 mins on a fully charged battery

The system time base shall meet the following specifications

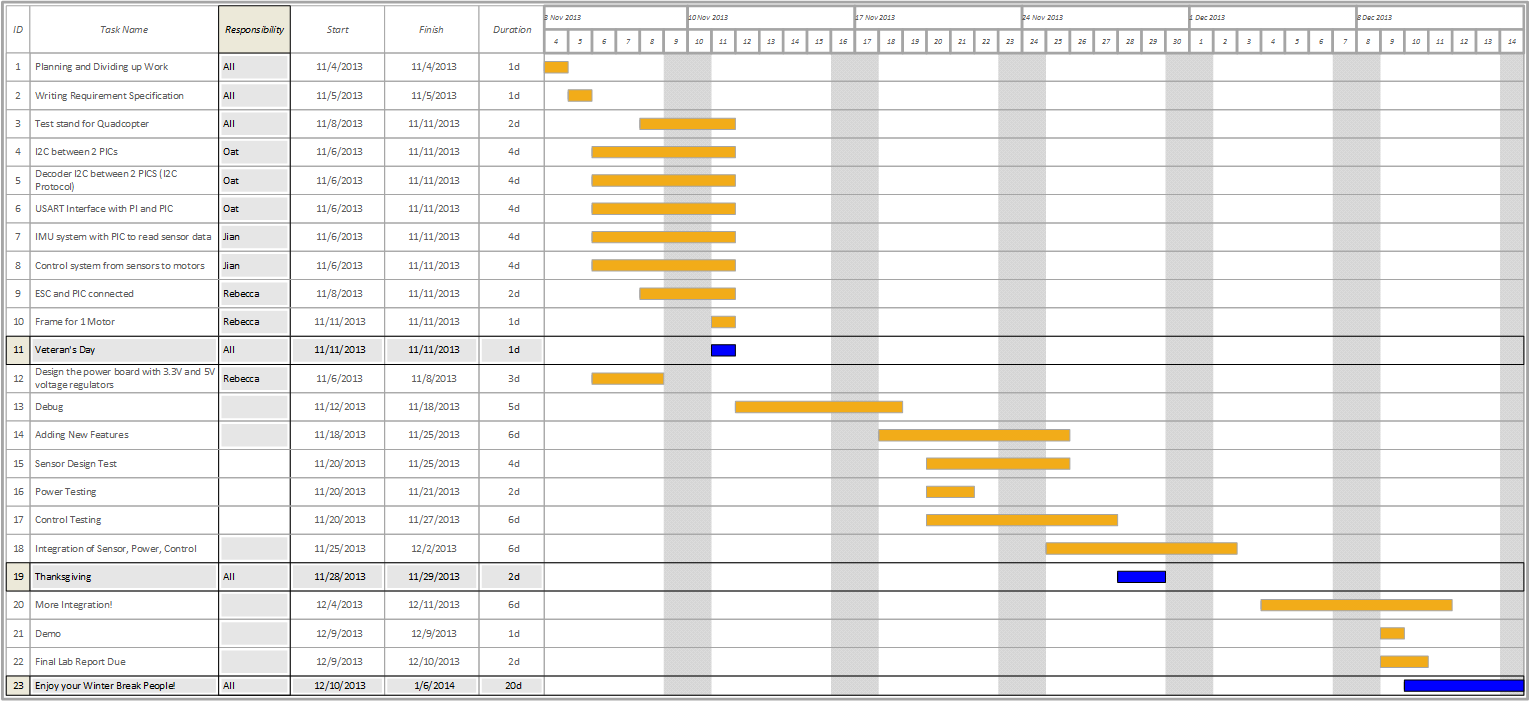
**RELIABILITY AND SAFETY SPECIFICATION**

The quadcopter should have full blade protection for propellers and an emergency landing plan.

The quadcopter should be compliant with the AMA National Model Aircraft Safety Code.

The use of quadcopter should be compliant to state with federal privacy laws regulations.

**UPDATED SCHEDULE**



**Figure 5.1 Schedule for Project**

**BILL OF MATERIALS**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TOTAL COST** | **$377.81** |  |  |  |
| Part | Unit Cost | Number | Total Cost | Vendor |
| Microcontroller |  |  | 0 |  |
| ESCs | $11.99 | 5 | $59.95 | Hobby Wing |
| On-board PC (Pi) | $30 | 1 | $30 | Ebay |
| Motors |  | 4 | $58.25 | SunnySky |
| Battery | $28.8 | 2 | $57.6 | Turnigy |
| Propeller | $1 | 5 | $5 | HobbyKing |
| Camera | $34.49 | 1 | $34.49 | Amazon |
| Ultrasonic Ranger Finder | $3.47 | 1 | $3.47 | Vetco |
| Speaker | $1 | 1 | $1 | Vetco |
| Sensor pack | $36.3 | 1 | $36.3 | Vecto |
| T7A80W - Turnigy MAX80W 7A Lithium Polymer Batteries |  | 1 | $27.99 | HobbyKing |
| shipping for propeller, battery and charger |  | 1 | $5.72 | HobbyKing |
| Wifi adapter |  | 1 | $10.95 | ebay |
| Aluminium rod |  | 2 | $24 | Truevalue hardware store |
| Perf Board | $6.98 | 1 | $6.98 | Ebay |
| EC3 | $4.88 | 1 | $4.88 | Ebay |
| Propeller | $4.1 | 1 | $4.1 | HobbyKing |
| Power Connectors | $7.11 | 1 | $7.11 | HobbyKing |

**Figure 6.1 Bill of Material**