

FUFO

System Architecture

**Project Code: FUFO**

**Document Code: FUFO\_SA >– v1.0**

**Hanoi, August 6th, 2012**

Record of change

\*A - Added M - Modified D - Deleted

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Effective Date | Changed Items | A\* M, D | Change Description | New Version |
| Aug 6th | The whole document | A | Newly created | 1.0 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

SIGNATURE PAGE

**ORIGINATOR:** Phongvm August 6th, 2012

Project Manager

**REVIEWERS:** YenTT

Quality assurance

**APPROVAL:** HungPD

Supervisor / Professor of FPT University

TABLE OF CONTENTS

1 Introduction 3

1.1 Purpose 3

1.2 Scope 3

1.3 Definitions, Acronyms and Abbreviations 3

1.4 References 3

1.5 Overview 3

2 Architectural Representation 3

3 Architectural Goals and Constraints 3

4 Use-Case View 3

4.1 Use-Case Realizations 3

5 Logical View 3

5.1 Overview 3

5.2 Architecturally Significant Design Packages 3

6 Process View 3

7 Deployment View 3

8 Implementation View 3

8.1 Overview 3

8.2 Layers 3

9 Data View (optional) 3

10 Size and Performance 3

11 Quality 3

12 Other Considerations 3

# Introduction

## Purpose

This document provides a comprehensive architectural overview of the system of FUFO project, using a number of different architectural views to depict different aspects of the system. It is intended to capture and convey the significant architectural decisions that have been made on the system, including both hardware and software.

## Scope

This document describes the highest level architecture of the software and hardware regards to the System Requirement Specification document. However, the mechanical design will be specified as clearly as possible. The following aspects of the system will be discussed:

- Hardware:

+ Quadrocopter overall design and guides.

+ A clearly described Mechanical design.

+ A list of hardware components with little descriptions.

+ A context diagram of the influenced hardware components.

- Software:

+ A component diagram of the whole software system that shown the connection of each independent software.

+ Component diagrams of every independent software.

+ Descriptive specification of each independent software.

## Definitions, Acronyms and Abbreviations

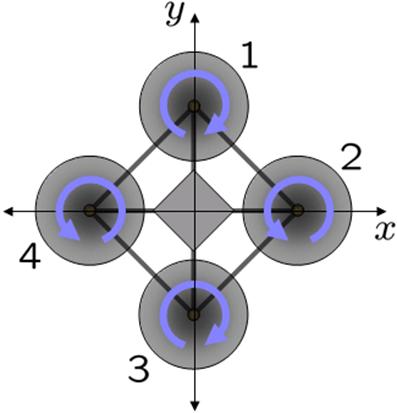
[This subsection should provide the definitions of all terms, acronyms, and abbreviations required to properly interpret the **Software Architecture Document**.  This information may be provided by reference to the project Glossary.]

## References

[This subsection should provide a complete list of all documents referenced elsewhere in the **Software Architecture Document**. Each document should be identified by title, report number (if applicable), date, and publishing organization. Specify the sources from which the references can be obtained. This information may be provided by reference to an appendix or to another document.]

# Hardware architecture

## Quadrocopter



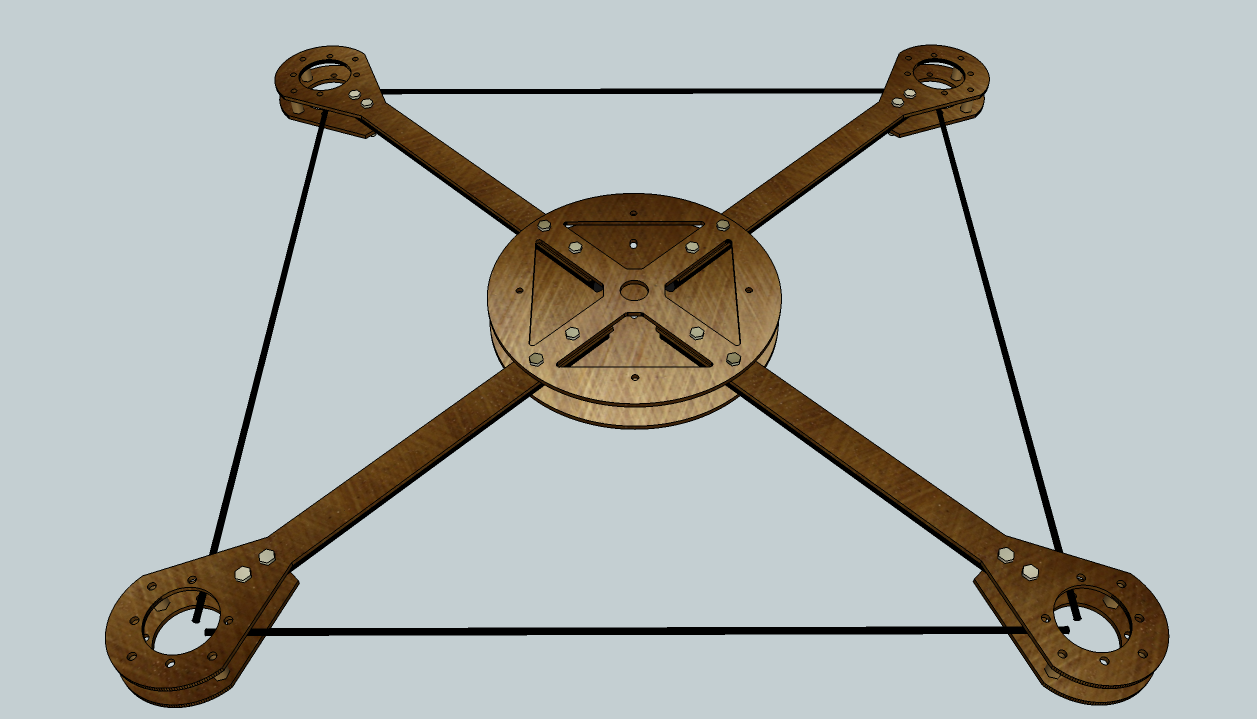
**Figure 2.1.1**

This Quadrocopter is a multicopter lifted and propelled by four motors. The motors are mounted on the same plane, forming a square.

As shown in **Figure 2.1.1,** the rotational direction of the four motors is not the same. Two opposite motors are having the same rotational direction while the other two are inversed. This design is necessary to compensate the angular acceleration generated by the motors. This way the total angular acceleration should be zero given the same speed of the propellers, ensure the "stable" characteristic of this flying structure.

The flight dynamic of a Quadrocopter is the rotation in three dimension of the plane about its center of mass, all of which depends solely on the momentum generated by the motors. More information on this matter can be found in the Quadrocopter Hovering Capability report.

## Mechanical design

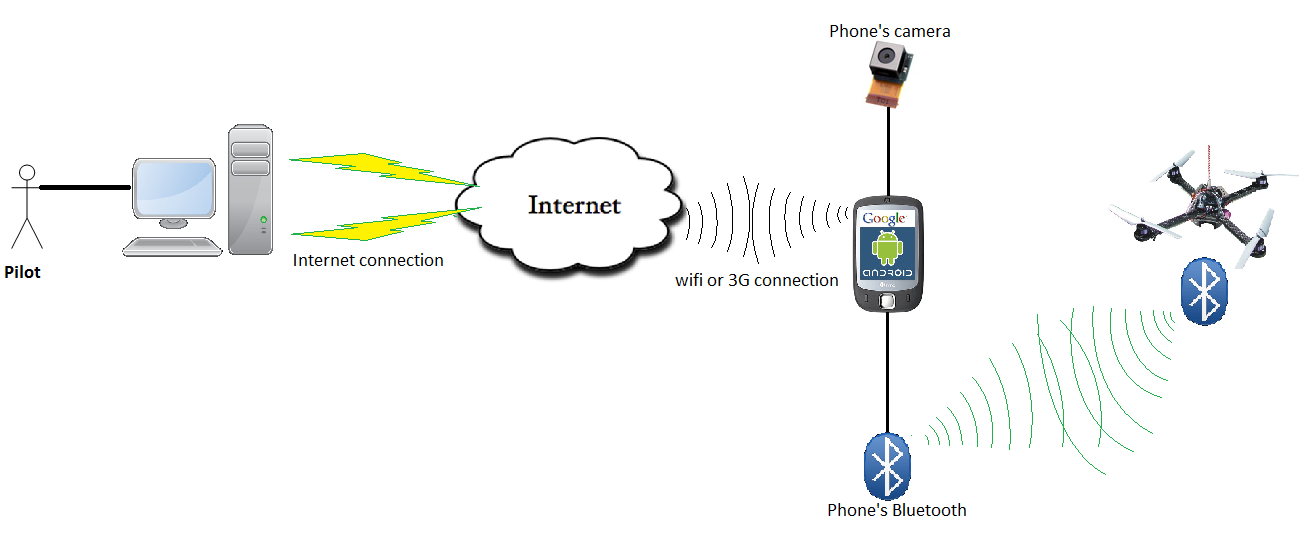


**Figure 2.2.1**

A 3D mechanical design has been sketched as shown in **Figure 2.2.1**. The main materials are being used in this design are 2mm Glass Fiber plates and 10mm Carbon Fiber tubes. The latter can be also replaced with Glass Fiber tubes in case of scarcity of Carbon Fiber tubes in Vietnam. The exact dimensions of each component in this design can be found in the Mechanical Design document in AutoCad format.

A realized version of this design may not have to be exactly the same in every aspect of the sketched design because of the poor capability in mechanic of software student. Due to the dynamic nature of the Quadrocopter, the hovering capability will not be affected if the moment arms are remained the same.

## System overview



**Figure 2.3.1**

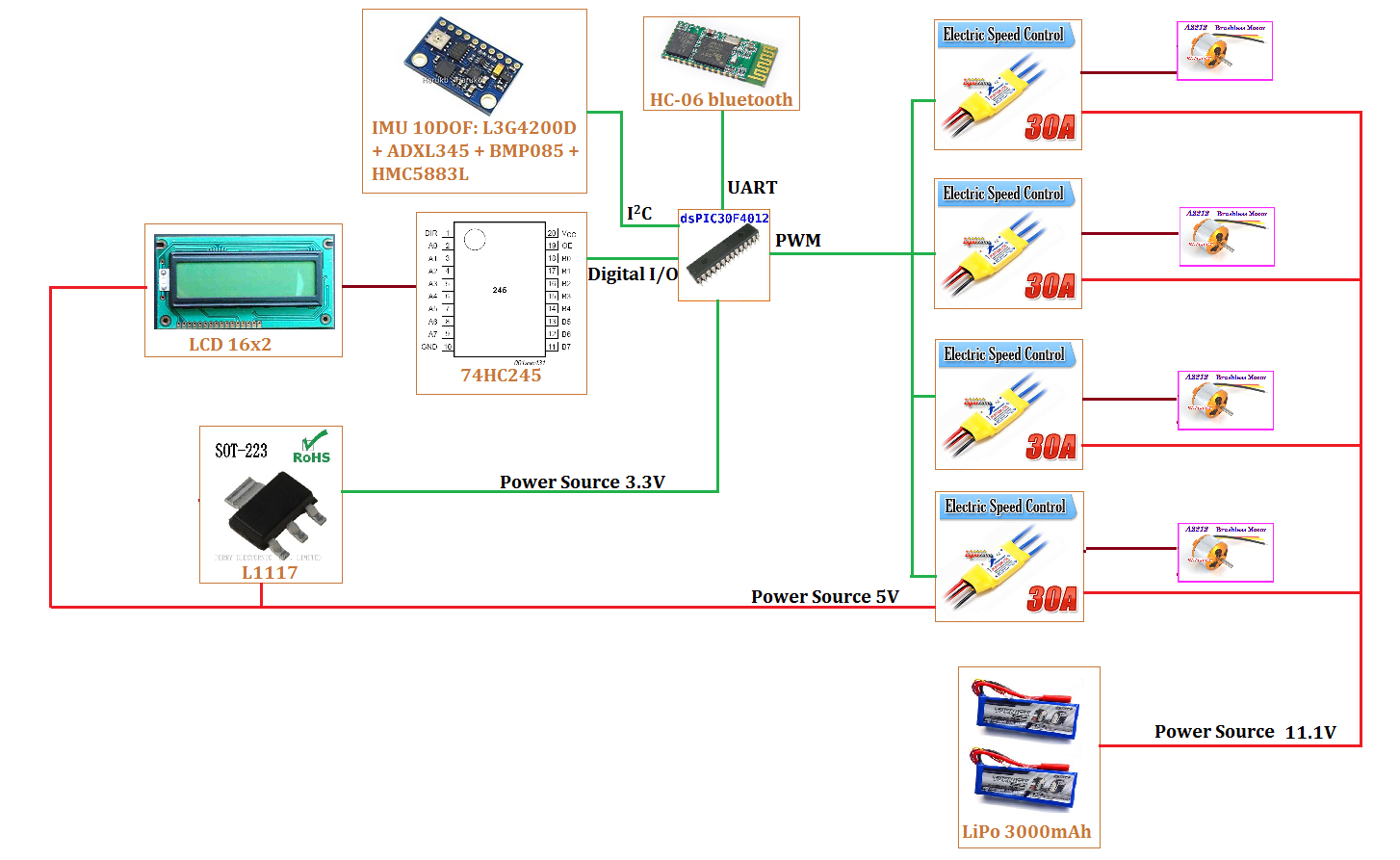
Figure above shows the context of the whole system.

## Hardware components

|  |  |  |  |
| --- | --- | --- | --- |
| Component Name | Quantity | Interface | Descriptions |
| dsPIC30f4012 | 1 | 3 PWM Generators  5 Timers  2 UART Interfaces  1 SPI/I2C Interface  40 Pins in total | - 16-bit wide data path.  - 48 Kbytes on-chip Flash program space.  - Up to 30 MIPs operation.  ­- 16 x 16-bit working register array. |
| Gyroscope L3G4200D | 1  (IMU Unit) | I2C | - 16 bit-rate value data output  - Integrated low- and high-pass filters with user selectable bandwidth  - supply voltage: 2.4 V to 3.6 V  - Working temperature range: -40 to +85 Celsius degree.  - 3 Dimensions Euler angle Velocity. |
| Acceleration ADXL345 | I2C | - 3-axis accelerometer.  - Measurement at up to ±16 *g*.  - supply voltage: 2.4 V to 3.6 V |
| Barometer BMP085 | I2C | - Pressure range: 300 - 1100 hPa.  - supply voltage: 2.4 V to 3.6 V |
| L1117 | 1 | VCC/VDD | - 3.3v Regulator.  - Input voltage up to 15V. |
| 74HC245 | 1 | I/O pins | - For communication between 5V LCD and 3.3V Microcontroller. |
| LCD 16x2 5V | 1 | I/O pins | - need 74HC245 to communicate with Microcontroller. |
| HC-06 Bluetooth module | 1 | UART | - supply voltage: 2.4 V to 3.6 V |
| HW30A ESC | 4 | PWM | - Motor speed controller module.  - Can provide a 5V 1A power source. |
| 1400Kv Motor | 4 | Indirect PWM via ESC |  |
| 8x4 Propeller | 4 |  | - 2 normal and 2 inverse |
| 3S1P 3000mAh LiPo Battery | 2 | Power source | - Need to be parallel connected to form a 3S2P battery system. |
| Android 2.2 phone | 1 | Bluetooth | - Has a 320x240 image supported camera.  - Has a 2.0 Bluetooth module.  - Has Wifi connection. |

**Table 2.4.1**

## Hardware context



**Figure 2.5.1**

Context of the FUFO Quadrocopter has been shown in **figure 2.5.1**. The electronic schematic must be designed from this context.

# Architectural Representation

[This section describes what software architecture is for the current system, and how it is represented. Of the **Use-Case**, **Logical**, **Process**, **Deployment**, and **Implementation Views**, it enumerates the views that are necessary, and for each view, explains what types of model elements it contains.]

# Architectural Goals and Constraints

[This section describes the software requirements and objectives that have some significant impact on the architecture, for example, safety, security, privacy, use of an off-the-shelf product, portability, distribution, and reuse. It also captures the special constraints that may apply: design and implementation strategy, development tools, team structure, schedule, legacy code, and so on.]

# Use-Case View

[This section lists use cases or scenarios from the use-case model if they represent some significant, central functionality of the final system, or if they have a large architectural coverage - they exercise many architectural elements, or if they stress or illustrate a specific, delicate point of the architecture.]

## Use-Case Realizations

[This section illustrates how the software actually works by giving a few selected use-case (or scenario) realizations, and explains how the various design model elements contribute to their functionality.]

# Logical View

[This section describes the architecturally significant parts of the design model, such as its decomposition into subsystems and packages. And for each significant package, its decomposition into classes and class utilities. You should introduce architecturally significant classes and describe their responsibilities, as well as a few very important relationships, operations, and attributes.]

## Overview

[This subsection describes the overall decomposition of the design model in terms of its package hierarchy and layers.]

## Architecturally Significant Design Packages

[For each significant package, include a subsection with its name, its brief description, and a diagram with all significant classes and packages contained within the package.

For each significant class in the package, include its name, brief description, and, optionally a description of some of its major responsibilities, operations and attributes.]

# Process View

[This section describes the system's decomposition into lightweight processes (single threads of control) and heavyweight processes (groupings of lightweight processes). Organize the section by groups of processes that communicate or interact. Describe the main modes of communication between processes, such as message passing, interrupts, and rendezvous.]

# Deployment View

[This section describes one or more physical network (hardware) configurations on which the software is deployed and run. At a minimum for each configuration it should indicate the physical nodes (computers, CPUs) that execute the software, and their interconnections (bus, LAN, point-to-point, and so on.) Also include a mapping of the processes of the **Process View** onto the physical nodes.]

# Implementation View

[This section describes the overall structure of the implementation model, the decomposition of the software into layers and subsystems in the implementation model, and any architecturally significant components.]

## Overview

[This subsection names and defines the various layers and their contents, the rules that govern the inclusion to a given layer, and the boundaries between layers. Include a component diagram that shows the relations between layers. ]

## Layers

[For each layer, include a subsection with its name, an enumeration of the subsystems located in the layer, and a component diagram.]

# Data View (optional)

[A description of the persistent data storage perspective of the system. This section is optional if there is little or no persistent data, or the translation between the Design Model and the Data Model is trivial.]

# Size and Performance

[A description of the major dimensioning characteristics of the software that impact the architecture, as well as the target performance constraints.]

# Quality

[A description of how the software architecture contributes to all capabilities (other than functionality) of the system: extensibility, reliability, portability, and so on. If these characteristics have special significance, for example safety, security or privacy implications, they should be clearly delineated.]

# Other Considerations

[This section provides a description of other approach/ solutions that were considered in selection process for the above architecture, i.e. a brief explanation of advantages and disadvantages of the selected architecture in comparison with others. It should be a clear answer to the question why the above architecture is selected for this system, not the others.]