HARDWARE SYSTEM AND SOFTWARE DESIGN DESCRIPTION (SSDD): Incorporating Architectural Views and Detailed Design Criteria FOR

The Foliage Extracting Tele-Controlled Helicopter (FETCH)

Version 1.1
[4/23/2013]

Prepared for:
Dr. Katy Kavanagh, Professor Bruce Bolden, and Dr. Herbert Hess

Prepared by:
Kora Barnes, Elliot Dickison, Cable Johnson, Eric Johnston, Brian Lee, Theora Rice

University of Idaho
Moscow, ID 83844-1010

Template credited to:
Dr. Clinton Jeffery

CS383 SSDD

RECORD OF CHANGES (Change History)

Change Number	Date comple ted	Location of Change (e.g., page or figure #)	A M D	Brief description of change	Approved by (initials)	Date

A-Added **M-Modified** **D-Deleted**

FETCH

Table of Contents

1 Introduction

1.1 Identification

This is the design document of the project implemented by Team FETCH. The final product will be a fully functional remote-controlled quadcopter with the ability to collect sun foliage samples from treetops. It is composed of two general modules. The first is the quadcopter itself, built from off-the-shelf parts. The second is an extending arm attachment with cut-and-hold clippers which are able to extract the intended payload. This document provides a detailed outline of the design and intended implementation of the Foliage Extracting Tele-Controlled Helicopter project.

1.2 Document Purpose, Scope, and Intended Audience

1.2.1 Document Purpose

The Foliage Extracting Tele-controlled Helicopter is being developed according to a set of requirements outlined by the client, Dr. Katy Kavanagh of the College of Natural Resources department at the University of Idaho. This document provides detailed descriptions of how these requirements are being met.

1.2.2 Document Scope and/or Context

This document includes information regarding the overall design, as well as detailing the different components of the project. The context will include the overarching structure, as well as the interactions between the different modules.

1.2.3 Intended Audience for Document

The intended audience for this design document include the client, the project supervisors (Dr. Hebert Hess and Professor Bruce Bolden), and any other faculty or students who wish to use it as an educational resource.

1.3 System and Software Purpose, Scope, and Intended users

1.3.1 System and Software Purpose

The purpose of the Foliage Extracting Tele-Controlled Helicopter is to collect sun foliage from various trees to be studied by the College of Natural Resources. The foliage samples are used to study climate changes, as they get steady sunlight and retain consistent amounts of moisture. These samples are often gathered in the dark pre-dawn hours from trees up to 300 meters in height.

1.3.2 System and Software Scope/or Context

The system will be able to operate like a standard quadcopter, capable of flying with horizontal and vertical adjustments, with the addition of an extending clipping attachment. Once the quadcopter is hovering just above the tree of choice, the user will extend the arm downwards the desired distance, close the clippers on the target branch, and retract the arm. Once the foliage sample is collected, the user can either fly the quadcopter back and collect the sample by hand, or drop the sample from the air by releasing the clippers.

1.3.3 Intended Users for the System and Software

The intended users of the system include Dr. Katy Kavanagh herself, and anyone else the College of Natural Resources deems acceptable for its use in collecting foliage samples. It is strongly recommended that anyone using the system go through full training in its use, either by someone familiar with the system already, or by thoroughly studying the documentation and provided training video. It is also recommended that the user manual be available during all operation, not only as a reference for general use, but also in case something is wrong with the device and small modifications need to be made. If the quadcopter crashes, the documentation will cover the exact parts needed for repair, as well as detailed repair instructions.

1.4 Definitions, Acronyms, and Abbreviations

Acronym	Definition
FETCH	Foliage Extracting Tele-Controlled Helicopter (Also the name of the development team)
FCB	Flight Control Board
Sun Foliage	Foliage at the tops of trees which get the most sunlight and retain the most moisture
PIC32	Digilent MX3CK Cerebot Microcontroller
PWM Signal	Pulse Width Modulation Signal. Signal which is periodically received as a zero or one, but never anything in between. This is the typical signal transmitted by the quadcopter controller.
DC Motor	Electronic motor controlled by an incoming PWM signal.

1.5 Document Restrictions

This document is for LIMITED RELEASE ONLY to the University of Idaho College of Engineering and College of Natural Resources.

2 Constraints

This section identifies and describes in detail the architectural and usability constraints that are imposed by the physical environment and system requirements and the user characteristics.

2.1 Environmental constraints

This system can be used in any environment in which line of sight can be ensured between the operator and quadcopter. It can operate in non-precipitate weather, with winds up to 10 mph, and can fly up to 300 meters (higher, with legal licenses). The system was designed so that it can collect a sample sun foliage branch up to 20cm in length and 2.5cm in width.

2.2 System requirement constraints

The system requires fully charged batteries in order to operate its embedded electronics. It also requires a transportation device in order to be used in remote locations. Examples of this device could be a car or truck.

2.3 User characteristic constraints

The operator must have received the FETCH flight training (to be provided in the form of a video), as well as have knowledge about how laptops are run. They must also understand the collection of sun foliage, as that is the system's intended purpose.

3 System and Software Architecture

3.1 Developer's Architectural View

From the developer's viewpoint the architecture of this system is modular, and involves three main modules: the quadcopter flight control, cutting attachment control, and communication structure. The cutting attachment control system can be subdivided in two parts: the arm extension controls and the cutting controls. The communication system, which is closely integrated into the overall product, can be subdivided into three parts: communication with the flight controller, the cutting attachment, and video relay.

3.1.2 Flight Control Module

The computer in control of the flight control module is called the flight control board (FCB). This is a special, integrated circuit board (an Ardu-Pilot Mega 2.5) that contains a pre-programmed microchip that accepts input from the user through an on-board transceiver (see section 3.1.4), as well as several on-board sensors. These sensors include a gyroscope, accelerometer, magnetometer, barometer, and GPS unit. The output of the flight controller microchip is sent to four electronic speed controls. These translate and relay the signals to the four motors, along with the appropriate amount of power for the desired speed.

3.1.3 Cutting Attachment System

The cutter arm control system utilizes a PIC32, which has been programmed using the C language to automate the extension and contraction of the arm, as well as the triggering the cutting device.

3.1.3.1 Arm Extension

When input signals are sensed by the on-board transceiver (see section 3.1.4), it immediately relays instructions to its connected units in the form of PWM signals. If the operator has specified that the arm is to extend or retract via one of the radio controller toggles, the transceiver sends a PWM signal to the PIC32. These signals are then read into the PIC32 via an input pin, and decoded to determine an arm movement direction. To do this, the microcontroller has been programmed to measure the frequency of the PWM signal, depicted in Figure 1.

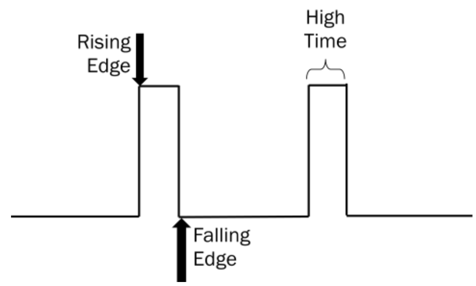


Figure 1: PWM Reading Diagram

Using this measurement, the program writes a new, constant PWM signal to an H-bridge, which is connected to DC motor, in order to extend and contract the arm. The circuit diagram of this motor output is provided in Appendix A. The motor turns a worm gear that powers a rack and pinion setup. The rack is fixed to one side of the arm lattice, and the other side of the arm lattice is fixed. Moving the rack via the DC motor and worm gear then causes the base of the lattice to slide together or apart, causing lattice extension and retraction.

3.1.3.2 Cutting Controls

The PIC32 is triggers the cutting mechanism based on input from another PWM signal generated by the on-board transceiver. The exact method used to accomplish the cutting has not been determined, as of spring 2013.

3.1.4 Communication System

The communication system utilizes two main signals between the user and the quadcopter. The first is a 2.4GHz 8 channel signal sent to the quadcopter from a standard two-stick flight radio operated by the user. The second is a 900MHz video signal sent back from the quadcopter to a receiver/laptop ground station so that the user may receive live video feed from the quadcopter cameras.

3.1.4.1 Communication with the Flight Controller

There are four servo connections between the 2.4GHz receiver (circuit diagram provided in Appendix B) and the FCB. These connections control the roll, pitch, throttle, and yaw of the quadcopter. Because so much of the input used to fly the quadcopter comes from the on-board sensors, these are the only signals needed from the user.

3.1.4.2 Communication with the Cutting Attachment

The act of extending the cutting attachment arm is controlled from an extra servo (Boolean) channel from the 2.4GHz receiver, sent to the radio control receiver. The PIC32 then decodes this signal and converts it for use. The cutting mechanism is triggered from another identical channel.

3.1.4.3 Video Communication

There are two small, wide-angle, CCD cameras mounted on the quadcopter: one facing forward for general flying and one mounted on the cutting attachment, facing straight down. A single 900MHz signal is used to send the feed from one of these cameras back to a receiver at the ground station. A hard-wired switch is used to select between the forward facing camera during normal flight, and the camera mounted on the arm when it is extended. The video feed is converted to an RCA signal by a receiver at the ground station and sent to a laptop for viewing by the user.

In order to provide adequate light in the pre-dawn hours, two high-intensity white LEDs are mounted next to the cameras. This not only provides the cameras with light, but also provides a visual stimulus so that the operator may see the quadcopter in the dark sky. These LEDs are controlled via a hardwired button on the quadcopter frame itself, so that they can only be affected when the quadcopter is on the ground. The circuit diagram modeling this setup is represented in Appendix C. This is to ensure that during operation the user will never lose visual sight of the quadcopter or where it's flying.

3.2 User's Architectural View

From the point of view of the user, the system is composed of the following units: the quadcopter frame, cutting attachment, and ground station. The interactions between the operator and FETCH are represented in the Figure 2 use case diagram.

Figure 2: UML Use Case Diagram for operation

3.2.1 The Quadcopter Frame

The frame has a center pod with four carbon fiber arms roughly 90° from each other. Motors connect to the end of each arm pointing upwards, and have propellers attached. The extension arm connects to the center pod of the frame on the downward-facing side. A flight controller, electric speed controllers (ESCs), batteries, microcontroller and radio signal receiver are also attached to the center of the frame. One small camera lens and white high-powered LED are also mounted at the center of the frame, pointing in a “forward” direction. When operating, the flight controller will receive input from the receiver, and then cause the ESCs to operate and balance the motor thrusts to compensate for maneuvering, wind, and mass distribution.

3.2.2 The Arm Attachment

The arm frame has been constructed in a lattice shape that connects to the central pod of the quadcopter. It extends and contracts based on how its attached DC motor, worm gear and pinion set-up moves. These, in turn, are controlled by a signal sent by the user from the radio controller to a receiver attached to the microcontroller. The motor and microcontroller are powered by a dedicated battery set.

Cut-and-hold clippers are mounted at the end of the arm frame, and may be opened or closed based on signals sent by the radio controller. When they cut a sample, they have a special plastic attachment that grips the branch after separation. The clippers will continue to hold the sample until the operator triggers them to open. How these clippers will be driven, and how they are connected to the bottom of the lattice work are undecided as of spring 2013. A small camera lens and white high-powered LED are also attached at the bottom of the arm, to provide detailed downward-facing video feed of the cutting process.

3.2.3 The Ground Station

The radio controller is the user interface that sends the wireless signals to the receiver. It is powered through an encased battery. The receiver will then receive the radio controller signals, and output directions to the connected flight controller and microcontroller. This receiver is powered through its connection to the flight controller.

Another aspect of the ground station is a laptop, with video feed receiver attachment. This attachment receives signals from either the forward-facing or downward-facing camera lens. It then relays the input video feed to a laptop screen. This laptop will be placed next to the operator, so that they may both operate the quadcopter and view the screen simultaneously.

3.3 Developer's View Identification

From the developer’s view, this system was designed to be built and programmed separately, and then integrated in stages. Besides the receiver, the construction of the quadcopter, cutting attachment, and ground station are autonomous. Therefore, the designs can be worked on in parallel in order to save time.

The quadcopter parts have been ordered off the shelf in order to provide for easy repair if damage occurs. It will be built and tested separately from the other attachments, so that its stability may be optimized before further advancement. These tests will take place both indoors and out-of-doors.

The cutting attachment will be constructed in stages. The extension of the lattice-work arm will be verified before incorporation of the clippers. It will then be tested separately from the quadcopter to ensure proper functionality.

The quadcopter and arm can then be integrated and tested together, to provide a cohesive, yet modular, final product. It will be controlled via the laptop-and-radio-controller ground station. This amount of system independence allows the quadcopter to be built and tested efficiently before the final product is released.

3.4 User's View Identification

The user's view of the FETCH system is mostly centered on ground station/quadcopter interaction. The radio-controller is how the operator interfaces and controls the system to achieve its intended function. They will be able to assess distance and clipper position by monitoring the laptop with video feed receiver. Live-streaming technology will thus create a more detailed view of the sample-collection process.

4 Software Detailed Design

4.1 Developer's Viewpoint Detailed Software Design

The desire to have the interface be as simple as possible is the driving force behind the software design. For user simplicity one control interface system was picked. A need for video feed from the quadcopter sparked the need for a laptop receiver. From there the desire to promote a modular system separated the quadcopter and arm controllers. This allowed for maximum system modularity and efficiency. The overall software interface design is shown in Figure 3.

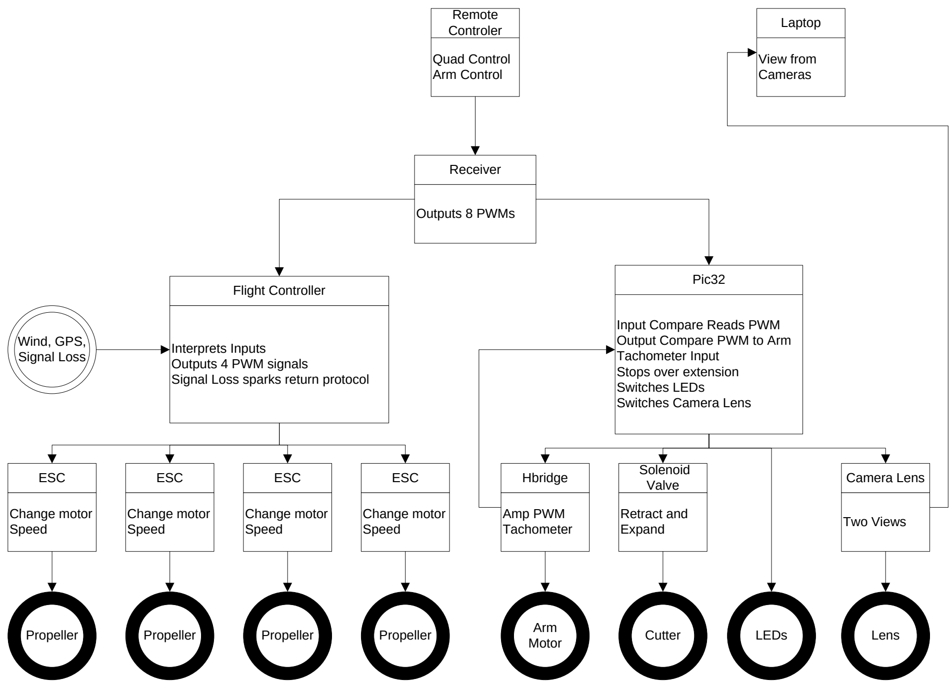


Figure 3: Software Interface Diagram

As the flight controller comes pre-programmed, and only requires configuration, software development on Project FETCH focused mostly on the PIC32. The different states the PIC32 can be in are represented by the UML state chart in Figure 4.

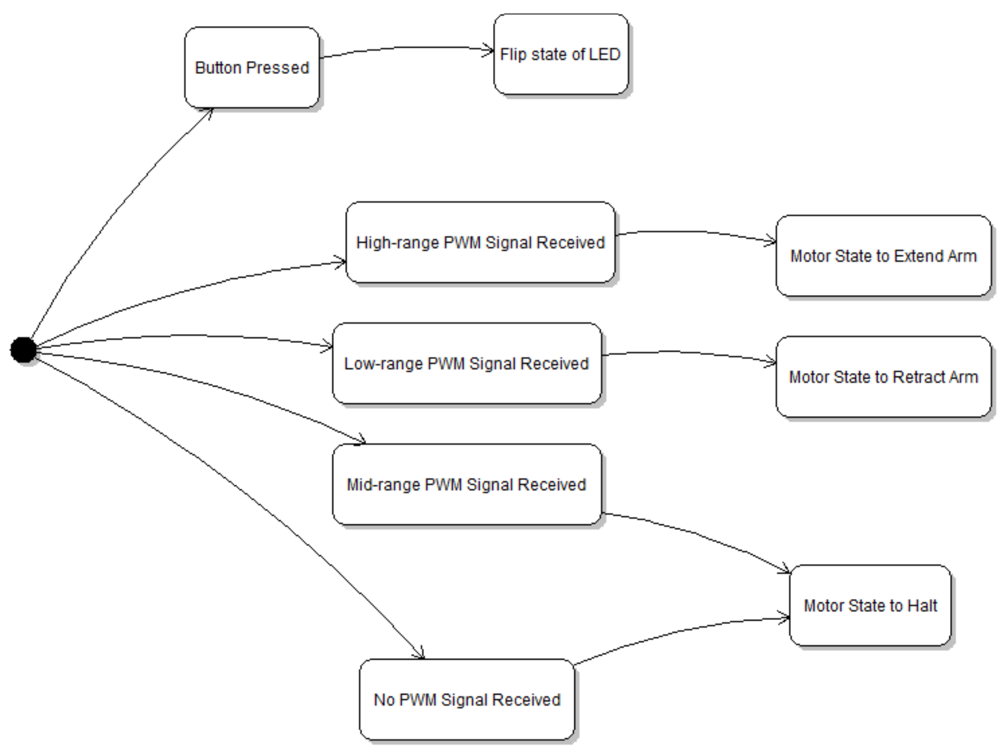


Figure 4: State Chart Diagram

4.2 Component/Entity Dictionary

Component/Entity Dictionary				
Name	**Type Interface**	**Purpose/Function**	**Dependencies**	**Subordinates**
Remote Controller	Physical interaction, radio signal	User control of quadcopter	Battery power	Receiver
Receiver	Radio signal, hard wired	Translates wireless signals	Remote Controller signal	PIC32, Flight controller
Flight Controller	Hard wired	Stabilizes flight, controls movement	Receiver	ESC
Electronic Speed Control (ESC)	Hard wired	Control Flight Motor Speed	Flight Controller	Flight Motors
Flight Motor	Hard wired	Provides Lift, turns propellers	ESC	Arm Motor
PIC32 MX3CK (PIC32)	Hard wired	Arm/LED Control	Receiver/motor position	Arm Motors, LED, Camera, Solenoid
H-Bridge	Hard wired	Converts PIC32 signals for Arm Motor	PIC 32	Arm Motor
Arm Motor	Hard wired	Move Arm	H-Bridge	None
Solenoid Valve/DC Motor/?	Hard wired	Provide Cutting Force	PIC 32	Cut-and-hold clippers
Cut-and-hold Clippers	Unknown	Provides cutting blades for separating tree branches, provides attachment for maintaining grip on sample	Solenoid Valve/DC Motor/?	None
LEDs	Hard wired	Provides light for capturing video feed and quadcopter operation	PIC32, button	None
Cameras	Hard wired	Capture video feed	PIC 32 (for power)	Video feed receiver
Video feed receiver	Wired	Relay video feed	Cameras	Laptop
Laptop	Wireless	Receive video feed	Video feed receiver	None

4.3 Component/Entity Detailed Design

4.3.1 Detailed Design for Component/Entity: Remote Controller

4.3.1.1 Introduction/Purpose of this Component/Entity: Receive user input and transmit signal to quadcopter for flight commands. Will also control cutting attachment extension and clipper use.

4.3.1.2 Input for this Component/Entity: Direct user input for movement controls and arm commands.

4.3.1.3 Output for this Component/Entity: Remote signal to the receiver on the quadcopter.

4.4.1.4 Component/Entity Process to Convert Input to Output: Internal circuitry.

4.3.1.5 Design Constraints and performance requirements of this Component/Entity: Comfortable, easy and intuitive to use.

4.3.2 Detailed Design for Component/Entity: Receiver

4.3.2.1 Introduction/Purpose of this Component/Entity: Receives signal from remote controller and converts it to PWM signals for use by the microcontroller and flight controller.

4.3.2.2 Input for this Component/Entity: Signal from the remote control.

4.3.2.3 Output for this Component/Entity: PWM signals on multiple channels.

4.4.2.4 Component/Entity Process to Convert Input to Output: Internal circuitry.

4.3.2.5 Design Constraints and performance requirements of this Component/Entity: None

4.3.3 Detailed Design for Component/Entity: Flight Controller

4.3.3.1 Introduction/Purpose of this Component/Entity: Manages flying the quadcopter, maintains stability, provides control signals to the Electronic Speed Controls

4.3.3.2 Input for this Component/Entity: PWM signal from the Receiver

4.3.3.3 Output for this Component/Entity: PWM and control signals for the ESCs

4.4.3.4 Component/Entity Process to Convert Input to Output: Internal Circuitry and pre-programmed software

4.3.3.5 Design Constraints and performance requirements of this Component/Entity: Return to home feature, hover ability, high quality, light weight

4.3.4 Detailed Design for Component/Entity: Electronic Speed Control (ESC)

4.3.4.1 Introduction/Purpose of this Component/Entity: converts the signal from the flight controller into usable signals for the flight motors.

4.3.4.2 Input for this Component/Entity: Signal from flight controller

4.3.4.3 Output for this Component/Entity: Signal to flight motors

4.4.4.4 Component/Entity Process to Convert Input to Output: Internal Circuitry

4.3.4.5 Design Constraints and performance requirements of this Component/Entity: light weight

4.3.5 Detailed Design for Component/Entity: Flight Motor

4.3.5.1 Introduction/Purpose of this Component/Entity: Turn propellers and provide lift and flight to quadcopter.

4.3.5.2 Input for this Component/Entity: Signal from Electronic Speed controls.

4.3.5.3 Output for this Component/Entity: Rotational motion/lift

4.4.5.4 Component/Entity Process to Convert Input to Output: Internal Circuitry

4.3.5.5 Design Constraints and performance requirements of this Component/Entity: Need to provide enough lift to fly with added cutting attachment weight.

4.3.6 Detailed Design for Component/Entity: PIC32MX3CK (PIC32)

4.3.6.1 Introduction/Purpose of this Component/Entity: Controls all arm motion, as well powers the video cameras.

4.3.6.2 Input for this Component/Entity: PWM signals from the receiver.

4.3.6.3 Output for this Component/Entity: Sends PWM/logical signals to all dependant components, which include H-bridges, LEDs, and camera feed.

4.4.6.4 Component/Entity Process to Convert Input to Output: Designer-generated C code interprets PWM signals from the receiver and manual button into various outputs for the components.

4.3.6.5 Design Constraints and performance requirements of this Component/Entity: Small size, complex computing ability

4.3.7 Detailed Design for Component/Entity: H-Bridge

4.3.7.1 Introduction/Purpose of this Component/Entity: Converts PWM and logic signals from PIC 32 into a directional DC signal for the arm motor.

4.3.7.2 Input for this Component/Entity: PWM and logic signals from PIC32

4.3.7.3 Output for this Component/Entity: Directional DC signal for the DC motor.

4.4.7.4 Component/Entity Process to Convert Input to Output: Internal circuitry.

4.3.7.5 Design Constraints and performance requirements of this Component/Entity: small size, directional output

4.3.8 Detailed Design for Component/Entity: Arm Motor

4.3.8.1 Introduction/Purpose of this Component/Entity: Provides directional force to move the cutting arm.

4.3.8.2 Input for this Component/Entity: Directional DC signal from the H-Bridge.

4.3.8.3 Output for this Component/Entity: Rotational Force to be used in arm motion.

4.4.8.4 Component/Entity Process to Convert Input to Output: Internal circuitry.

4.3.8.5 Design Constraints and performance requirements of this Component/Entity: None

4.3.9 Detailed Design for Component/Entity: Solenoid Valve/Cutting motor/?

4.3.9.1 Introduction/Purpose of this Component/Entity: Provides force to cut branches.

4.3.9.2 Input for this Component/Entity: Signal from the PIC32

4.3.9.3 Output for this Component/Entity: Cutting force

4.4.9.4 Component/Entity Process to Convert Input to Output: Electronic signal opens valve, which in turns allows pressurized CO_2 to move a piston and provide cutting force. (Rotational force to be used in cutting motion?)

4.3.9.5 Design Constraints and performance requirements of this Component/Entity: Small size

4.3.10 Detailed Design for Component/Entity: Cut-and-Hold clippers

4.3.10.11 Introduction/Purpose of this Component/Entity: To separate foliage sample from tree, and maintain hold of it until otherwise instructed by operator.

4.3.10.2 Input for this Component/Entity: Solenoid/dc motor/?.

4.3.10.3 Output for this Component/Entity: Cutting force for sample separation, holding force for sample collection

4.4.10.4 Component/Entity Process to Convert Input to Output: Mechanical design

4.3.10.5 Design Constraints and performance requirements of this Component/Entity: Need to be sharp and able to maintain grip of sample

4.3.12 Detailed Design for Component/Entity: LEDs

4.3.12.1 Introduction/Purpose of this Component/Entity: To provide adequate light for video capture and quadcopter operation

4.3.12.2 Input for this Component/Entity: Logic signal from microcontroller

4.3.12.3 Output for this Component/Entity: White light

4.4.12.4 Component/Entity Process to Convert Input to Output: Internal circuitry

4.3.12.5 Design Constraints and performance requirements of this Component/Entity: Need to be intensely bright

4.3.13 Detailed Design for Component/Entity: Cameras

4.3.13.1 Introduction/Purpose of this Component/Entity: Provide video feed to user. One camera provides forward feed, the other provides downward feed.

4.3.13.2 Input for this Component/Entity: Power from external batteries

4.3.13.3 Output for this Component/Entity: Video signal to video feed receiver

4.4.13.4 Component/Entity Process to Convert Input to Output: Video signal generated by internal circuitry, hardwired switch chooses between camera feeds.

4.3.13.5 Design Constraints and performance requirements of this Component/Entity: Automatically switch between feeds.

4.3.14 Detailed Design for Component/Entity: Video feed receiver

4.3.14.1 Introduction/Purpose of this Component/Entity: Receive video input from quadcopter cameras and relay it to connected laptop

4.3.14.2 Input for this Component/Entity: signal from quadcopter cameras.

4.3.14.3 Output for this Component/Entity: laptop-translatable streaming video

4.4.14.4 Component/Entity Process to Convert Input to Output: Internal Circuitry

4.3.14.5 Design Constraints and performance requirements of this Component/Entity: Need to provide clear, streaming video

4.3.15 Detailed Design for Component/Entity: Laptop

4.3.15.1 Introduction/Purpose of this Component/Entity: Provides visual feedback to the user for control and navigation.

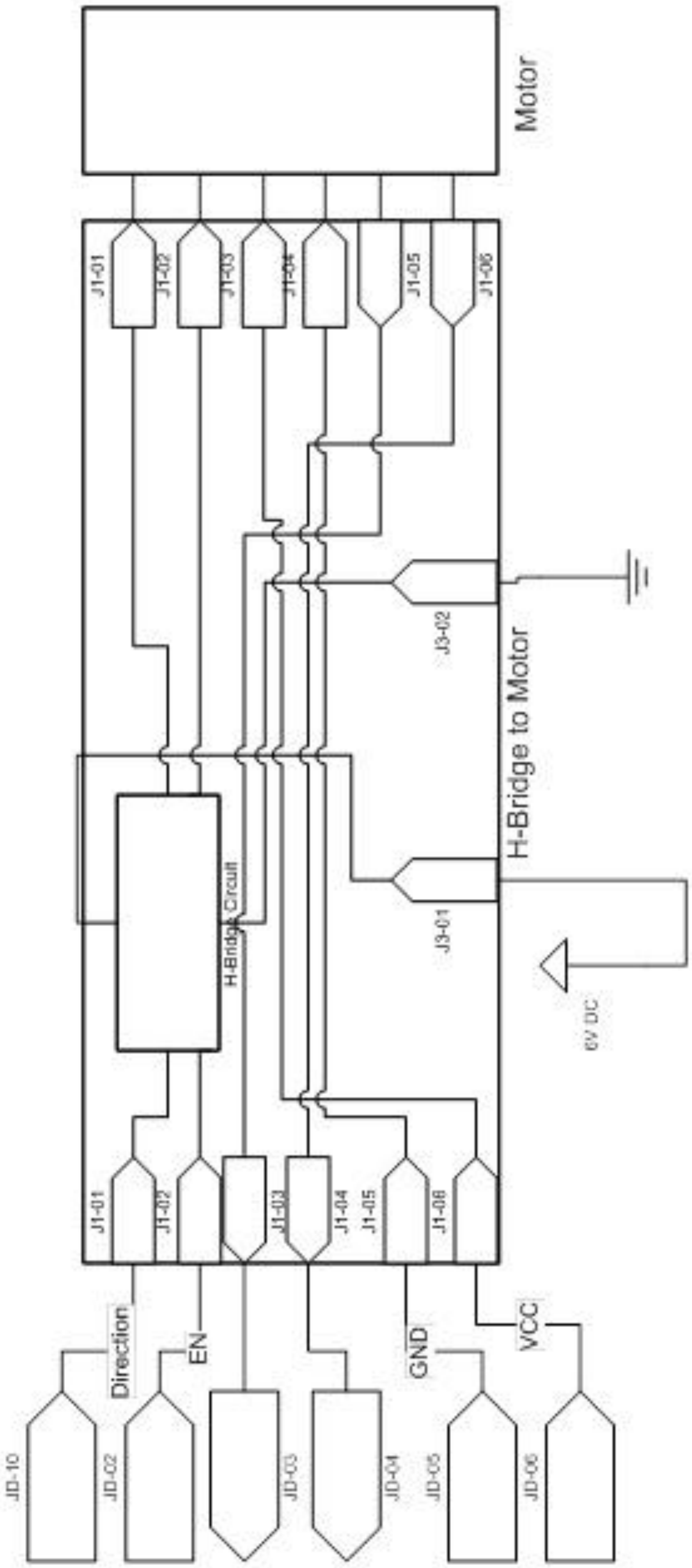
4.3.15.2 Input for this Component/Entity: data from video feed receiver

4.3.15.3 Output for this Component/Entity: Real time video streaming

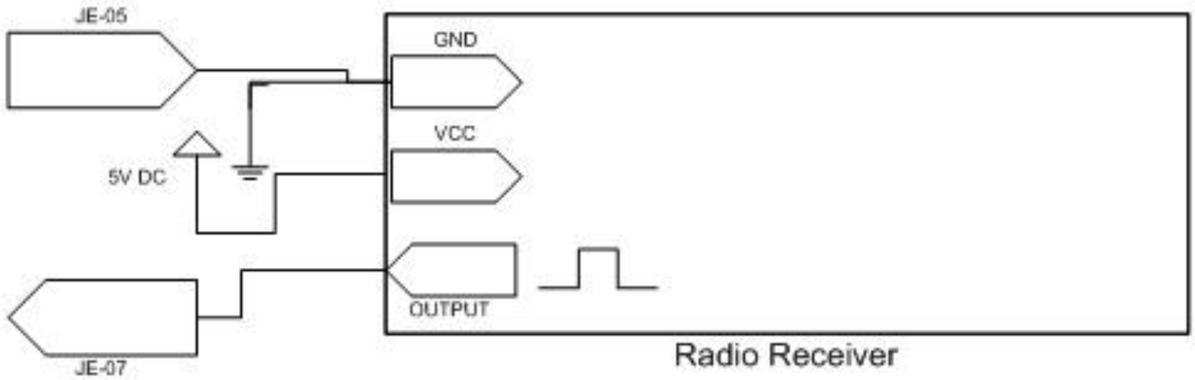
4.4.15.4 Component/Entity Process to Convert Input to Output: Installed software and internal circuitry.

4.3.15.5 Design Constraints and performance requirements of this Component/Entity: Clear image for operator.

Appendix A


Motor
J1-01
J1-02
J1-03
J1-04
J1-05
J1-06
J3-02
H-Bridge to Motor
H-Bridge Circuit
J3-01
6V DC
J1-01
J1-02
J1-03
J1-04
J1-05
J1-06
Direction
EN
GND
VCC
JD-10
JD-02
JD-03
JD-04
JD-05
JD-06

Appendix B



Appendix C