The University of Akron
IdeaExchange@UAkron

Honors Research Projects

The Dr. Gary B. and Pamela S. Williams Honors College

Spring 2017

Underwater Robot

Joseph E. Beck
The University of Akron, jeb135@zips.uakron.edu

Matthew Crislip
The University of Akron, mkc40@zips.uakron.edu

Cody Bobek
The University of Akron, cjb101@zips.uakron.edu

Peyton Lucas
The University of Akron, pal12@zips.uakron.edu

Underwater Robot

Final Design Report

Design Team #09

Joseph Beck

Cody Bobek

Matthew Crislip

Peyton Lucas

Faculty Advisor: Dr. Ryan Toonen

October 24, 2016

My role in this project team was the Team Leader. I was tasked with keeping everyone on track with the project timeline, facilitating necessary meetings with faculty, and keeping track of the project budget. As far as the actual design goes, I designed and programmed the GUI that was used to wirelessly control the ROV as well as view the video stream coming from its camera. This involved implementing several existing technologies into one application that interfaced with the ROV's wireless endpoints. Another major portion of the design I worked on was the physical design and layout of the entire ROV and all internal components. Lastly, I participated in the PCB design by overseeing other members laying them out and teaching them how to use the necessary software to do this.

Joseph Beck

Table of Contents

List of Figures

List of Tables

Abstract

Remotely Operated Vehicles (ROVs) are remote controlled drones operated by a non-local user. The ROV we plan to build is connected by a tethering wire to a floating buoy that contains an antenna which will send signals between the base station and the ROV. The ROV is equipped with a video camera, ballast system, propulsion system, lights, and a depth sensor. The ROV will transmit a live video feed to the user, while receiving input signals to control its movement from the base station.

[MKC, JEB]

I. Problem Statement

A. Need

Underwater inspection and reconnaissance is difficult but essential. Bridges have underwater components that regularly need to be checked for structural damage, deterioration, and growth of foliage. Usually divers are used for such a task, and also some underwater drones have been developed for this purpose. Many of these underwater drones have tether wires that reach back to the boats that are controlling them, which in turn limits their range. A need arrives for a wireless solution that allows for dispatch and communication from afar. This means that there is a need for an underwater drone with on board power systems, connected via wire to a buoy that carries a wireless antenna in order to provide for longer range usage and more maneuverability.

[PAL, JEB]

B. Objective

Design an ROV that is tethered to a buoy that sits on the water surface, carrying a long-range antenna. This antenna is then used as a connection to the nearby control station where a user interacts with a computer to manipulate the submerged vehicle. The ROV should have a camera that streams live video to the operator, along with high power lights in order to make visibility in the water higher. The control station should have full control of the ROV via a graphical user interface (GUI).

[JEB]

C. Background

Patent Search

The underwater drone will be able to monitor local environment and observe an object below much more closely than otherwise needing us to wear underwater suit which is bothersome. Working with anything that involves communication underwater is always going to be a difficult task in any project. This is mostly because radio waves do not travel through good conductors like water. Because of this a more indirect approach is needed in order to receive and transmit information.

The proposed solution is to have a buoy floating on the surface, above the underwater device attached via wire. The buoy has an antenna attached to it that will be able to easily receive and transmit the data through a free space, such as air.

Another proposed solution uses an underwater communication system that transmits electromagnetic signals to a remote receiver as described in [1]. An underwater communications system is provided that transmits electromagnetic and/or magnetic signals to a remote receiver. The transmitter includes a data input. A digital data compressor compresses data to be transmitted. A modulator modulates compressed data onto a carrier signal. An electrically insulated, magnetic coupled antenna transmits the compressed, modulated signals. The receiver that has an electrically insulated, magnetic coupled antenna for receiving a compressed, modulated signal. A demodulator is provided for demodulating the signal to reveal compressed data. A de-compressor de-compresses the data. An appropriate human interface is provided to present transmitted data into text/audio/visible form.

[JEB, MKC]

Article Search

When it comes to designing a relatively complicated underwater drone it is a good idea to have many design possibilities.

The preferred design, as mentioned above, is to communicate with the floating buoy via antenna which then sends information down to the underwater vehicle. The Remotely Operated Vehicle (ROV) has a camera attached which the human user can use to navigate accordingly. Other possible sensors for the ROV could be implemented as well, such as proximity sensors.

Another design suggests a fully autonomous underwater vehicle (AUV) [4]. This drone can be programmed to perform its task without the over watch of a human controller. A possible task that can be programmed into the drone could be to find an object with specific color using camera and image processing in Debian based operating system. Other than that as an AUV it is able to carry some load and transport it somewhere near as long as the power is sufficient. AUV Design compromised by planting artificial intelligence, planning the mechanical structure and electronics design. Component primarily discussed here are locomotion mechanism, particle density and buoyancy. Experiment data, measurement data, failures and lesson learned from the research are also included.

A possible backup design shows the implementation of a semi-submersible ROV which would eliminate the need for a buoy with an antenna attached thus making the design easier [5]. This can be remotely controlled from the ground, air, satellite and sea also during the semi-submersible operation. The vehicle with electric power is coupled with a jet propulsion, given the low draft, makes it possible to navigate in shallow waters or coastal shipping or sandbars. However, the drone would then be limited to water surface scanning.

[MKC, JEB, PAL]

D. Marketing Requirements

1. Stream live video back to some user interface.
2. Wirelessly controlled.
3. Waterproof hull.
4. Powerful lights for increased visibility.
5. Measure depth at any time.
6. Movement in 3 dimensions
7. Two batteries needed to power the whole system.
8. Easy to work with.
9. Have quick and reactive controls.
10. Durable
11. Has a failsafe system for 100% recoverability

[JEB, MKC]

E. Objective Tree

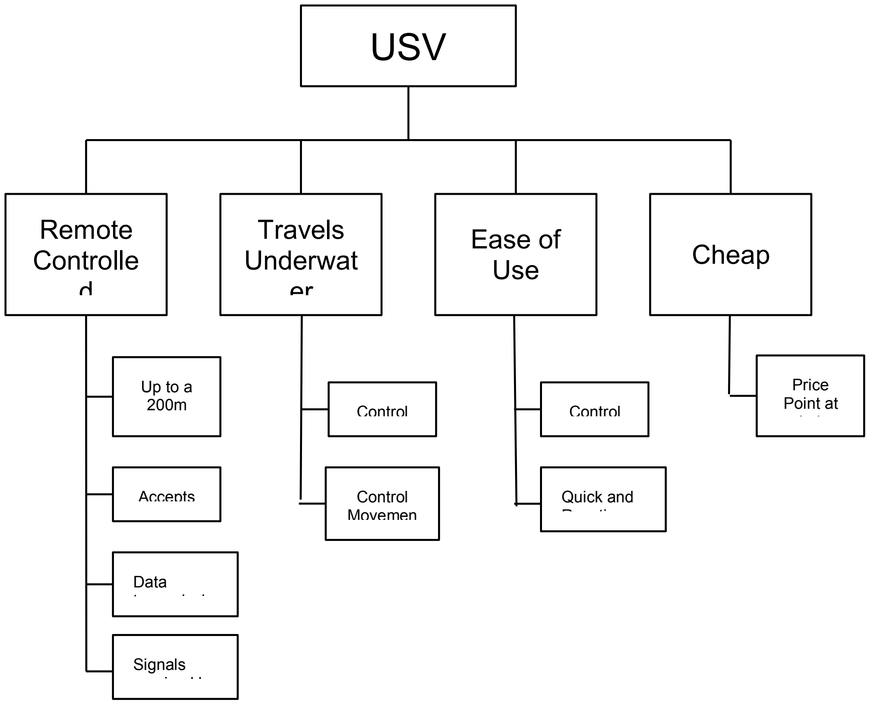


Figure 1: Objective Tree

[PAL, JEB, MKC]

II. Design Requirement Specifications

Marketing Requirements	Engineering Requirements	Justification
1, 2	ROV should be able to operate up to 200m away from the control station.	In order to warrant a wireless solution, the range of the ROV should be large enough that the reduction in cable required is worth it.
3, 6, 10	ROV can obtain a depth of 50ft safely.	50ft is a good target depth to allow for a wide range of use on the ROV.
6	ROV can move in 3 dimensions	The ROV must be able to move in 3 dimensions to be of any use to the user.
1, 4	Video stream of up to 720p quality is shown to the user via some user interface.	The user will need to be able to see where the ROV is looking and going in order to properly control it.
11	Onboard storage keeps a 1080p version of the video taken on a SD card for later review.	Since the wireless stream is broadcast in more compressed format, the ROV should have onboard storage for complete HD video so that it can be later viewed by the user in higher resolution.
5, 6	A depth sensor is used to monitor depth of the ROV.	The user will need to know what depth the ROV is operating at.
6, 8	The ROV should be capable of remaining at a stable depth with no input from the user.	If the user stops inputting controls, the depth of the ROV should remain the same so that it does not drift up or down from natural buoyancy forces.
11	The ROV should have a failsafe that returns it to the surface.	In the event of a signal loss or dead battery, the drone should be recoverable in such a way that it will always return to the surface.
3, 10	The hull of the ROV should be waterproof to prevent damage to internal components	Water easily damages computer systems and should be kept out of critical component bays.
9	The controls should be reactive and have latency of no more than 200ms from the time the control input is taken, till the user sees the effect on the video stream.	The video reaction to control input should feel fast and normal.
4	The ROV should have a light source strong enough to depict images on the video stream	The camera is going to need some kind of light source in order to better see objects in the water when the water clarity isn't very high.
6, 9	The ROV can travel at a speed of 3.5 mph	Needs to be able to control itself in water that has currents.
7	The ROVs battery system should rechargeable, and be able to last roughly 30 minutes under regular use.	30 minutes of total time is a fair amount to exercise the full use of the ROV.

Table 1: Engineering Requirements and Justifications

[JEB]

III. Accepted Technical Design

A. Hardware

Theory of Operation (Level 0): The ROV will be able to collect a video stream from a camera and relay that video data with a radio signal back to the user. The user will be able to control the ROV by sending radio control signals to the ROV. The ROV will need to change its density by taking in and expelling water to make the ROV's density more than, less than, or equal to the density of the surrounding water in order to change depth. The ROV will be able to decode the user sent radio signal to actuate motion to control movement below and at a given body of water's surface.

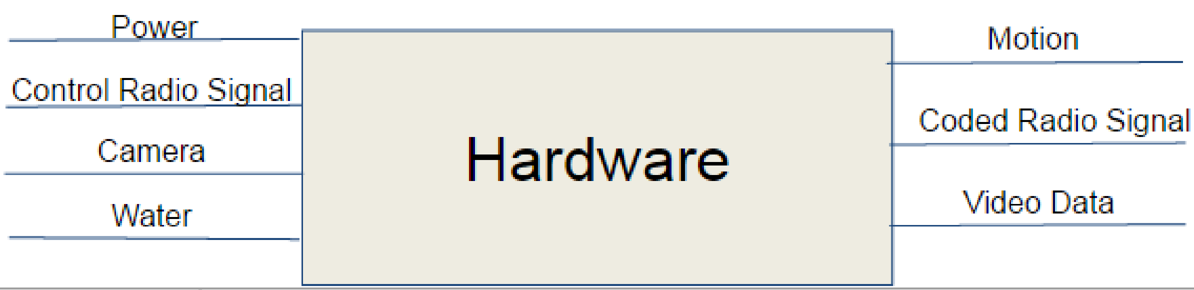


Figure 2: Hardware Level Zero Block Diagram

Module	**Hardware**
Inputs	● Power ● Control Radio Signal ● Camera ● Water
Outputs	● Motion ● Coded Radio Signal ● Video Data
Functionality	● Uses Power and Water in order to move to users specifications ● Receives and sends Radio signal and codes it for software use. ● Receives Video data via camera for software to code.

Table 2: Hardware Level Zero Functional Requirements

Theory of Operation (Level 1): The hardware will allow for user input to be transmitted to the ROV. The user input will dictate the way in which the ROV will move in the XYZ plane, with X/Y being forward/backward and turning and Z being the depth of the ROV. The ROV will take data from Peripheral devices and transmit them back to the user via the communication system.

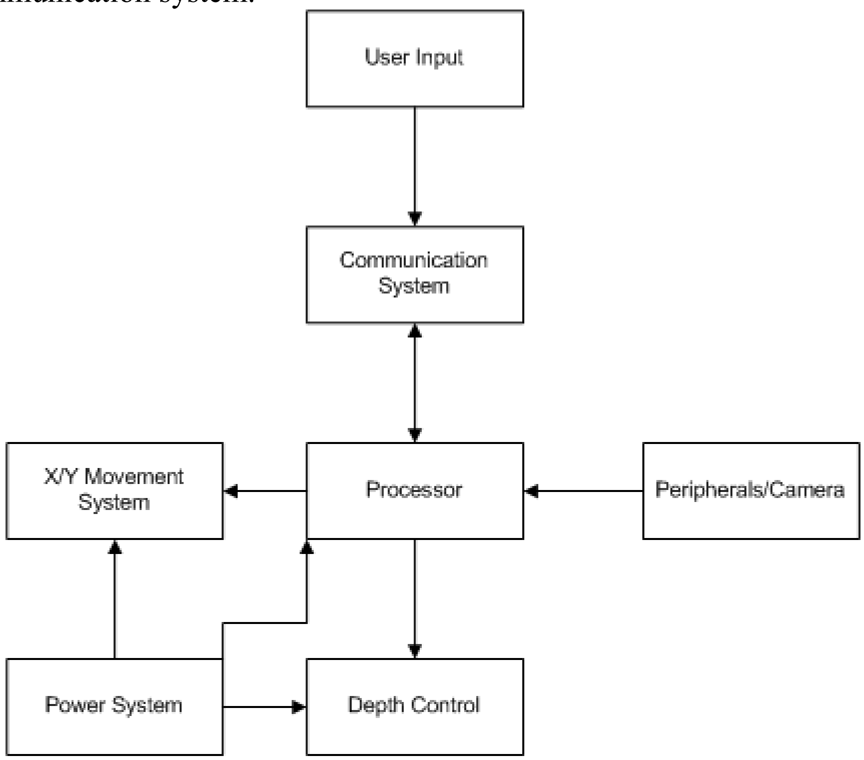


Figure 3: Hardware Level One Block Diagram

Module	**Processor**
Inputs	• Data from Communication System • Data from Peripherals/Camera • Power from Power System
Outputs	• Data to Communication System • Controls to X/Y Movement System • Controls to Depth Control
Functionality	• Processor will need to transmit and receive data to and from wireless communication

	• Processor will need to turn send peripheral device data back to the user • Processor will need to send controls for movement

Module	**Communication System**
Inputs	• Data From User Input • Data from Processor
Outputs	• User Input Data to Processor • Processor Data to User
Functionality	• Gives user ability to communicate with the processor on the ROV

Module	**User Input**
Inputs	• Controls from user • Data from Communication System
Outputs	• Controls to Communication System
Functionality	• Allows for user to control ROV using the Communication System

Module	**X/Y Movement System**
Inputs	• Power from Power System • Controls from Processor
Outputs	• Horizontal underwater movement for the ROV

Functionality	• Will allow the ROV to move through water

Module	**Depth Control**
Inputs	• Power from Power System • Control Signal from Processor
Outputs	• Vertical/Depth motion
Functionality	• Allows ROV to submerse under the water and maintain a depth

Module	**Peripherals Camera**
Inputs	• Sensory data from world
Outputs	• Digital signal data to Processor
Functionality	• Allows ROV to interface with the world and collect sensory data

Table 3: Hardware Level One Functional Requirements

Theory of Operation (Level 2): The hardware will consist of 2 parts: A ground system where the user will: control the ROV, and the user will be able to view data from the ROV including a live video feed. The controls will be originated from a computer which will be sent via an Antenna to the Remote ROV. The Antenna will also receive data from the ROV and transfer it to the computer where the live video stream will be decoded and viewed.

The ROV will receive control data sent from the ground system's antenna and send it to an onboard computer (processor) to be decoded. This antenna will also be responsible for relaying data from the onboard processor to the ground system. On board processor will decode received commands and actualize signals to mechanical components in order to control movement of the ROV. The movement of the ROV will be actualized by the motors and ballast system of the ROV.

The processor will continually decode information from a camera and store it in data as well as compress the video feed to be sent back to the ground system via the antenna.

The Battery will be a power system with the capability of powering motors, the ballast system, lights, and the processor.

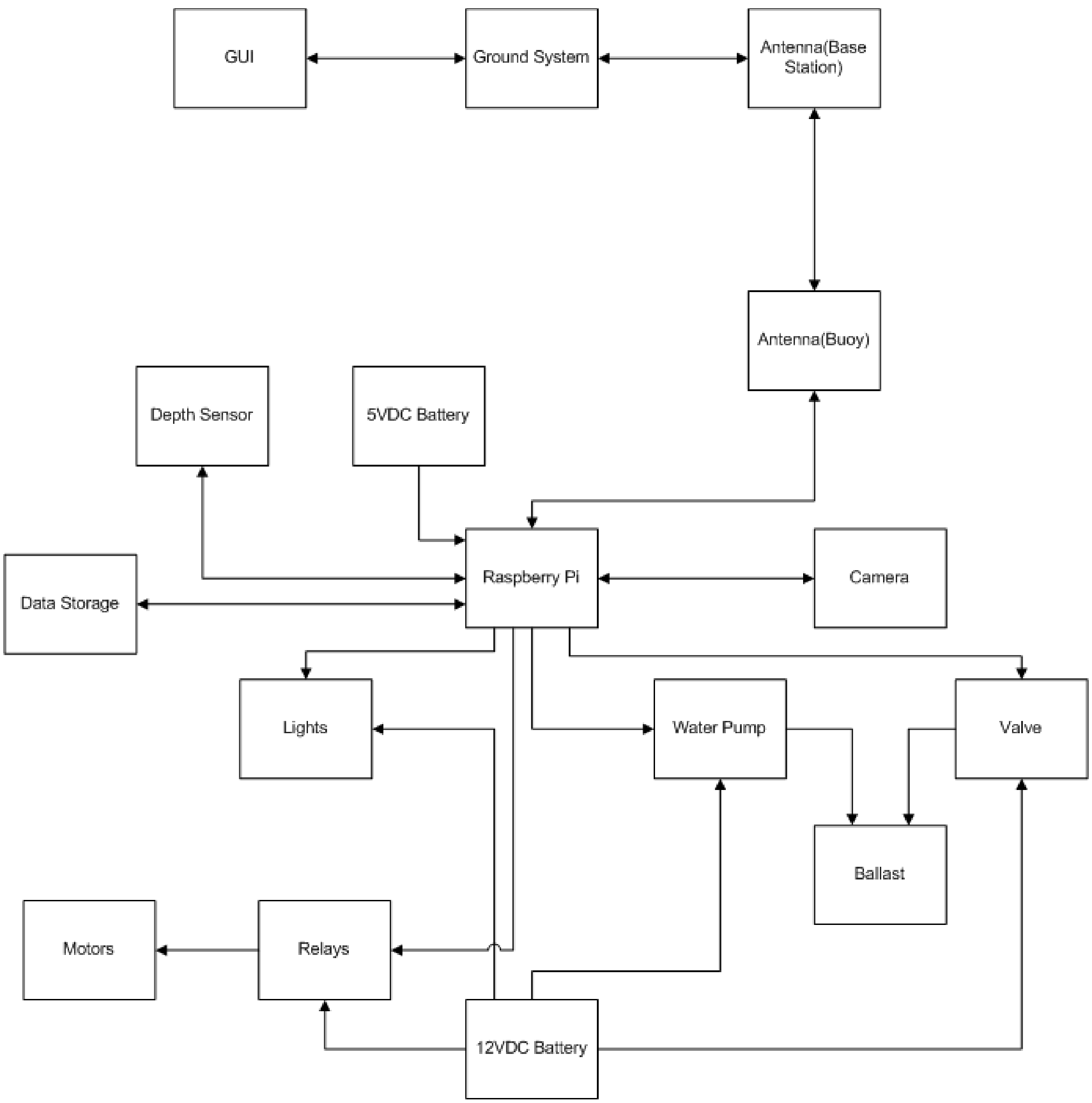


Figure 4: Hardware Level Two Block Diagram

Module	On-Board Computer
Inputs	• User Control • Bit stream from ground system/ Antenna
Outputs	• Bit stream to Ground System/ Antenna
Functionality	• Computer with keyboard to receive user controls and process controls into signals to be sent across the wireless link • Computer will be able to decode wireless signals received from the ground system and produce Video on a monitor.

Module	Ground System
Inputs	• Bit stream from Computer • Signal from Antenna
Outputs	• Bit stream to Computer • Signal to Antenna
Functionality	• Translates signal from/to Antenna from/to Computer

Module	Antenna(s)
Inputs	• Wireless signal • Signal from Ground System • Signal from On-Board Computer
Outputs	• Signal to Ground System • Signal to On-Board Computer
Functionality	• Allows for wireless link between ROV and Ground System

Module	**On-Board Computer**
Inputs	• Data from Depth Sensor • Data from Camera • Power from Battery
Outputs	• Signals to Motors • Signals to Ballast System • Signals to Antenna
Functionality	• Central hub of the ROV all signals will end or originate from this onboard computer • Receives and sends Radio signal and codes it for software use. • Receives Video data via camera for software to code.

Module	**Depth Sensor**
Inputs	• Pressure from environment
Outputs	• Bits to onboard computer
Functionality	• Will sense pressure of water to allow the On-Board Computer to determine depth

Module	**Motors**
Inputs	• Power from Battery • Control Signal from On-Board Computer
Outputs	• Motion

Functionality	• Motors will actuate based on controls from the On-Board Computer to create thrust by moving water

Module	**Ballast System**
Inputs	• Power from Battery • Water • Control signal from On-Board Computer
Outputs	• Water
Functionality	• Ballast will control water taken/expelled to control the ROV's density relative to surrounding water

Module	**Lights**
Inputs	• Control signal from On-Board Computer • Power from Battery
Outputs	• Light
Functionality	• Produces light so the Camera can detect images

Module	**Data Storage**
Inputs	• Control from On-Board Computer
Outputs	• Information to On-Board Computer

Functionality	• Stores data from the On-Board Computer to be recovered when the ROV is retrieved

Module	**Camera**
Inputs	• Power from On-Board Computer • Light reflected from objects in environment
Outputs	• Data to On-Board Computer
Functionality	• Will allow the ROV to receive video data which can be transferred to the user using other modules.

Module	**Battery**
Inputs	• Power charged while ROV is not in use
Outputs	• Power to other modules
Functionality	• Will power modules of the ROV

Module	**Relay(s)**
Inputs	• Control Signal from On-Board Computer • Power from Battery
Outputs	• Power to higher voltage devices (Valve, Motors, Water Pump)

Functionality	• Will convert a low voltage control signal into a high voltage signal

Module	**Valve**
Inputs	• Power from Relay • Water
Outputs	• Water • Valve open/close
Functionality	• The Valve will control whether water can be pumped into the Ballast or released from the Ballast

Module	**Ballast**
Inputs	• Water
Outputs	• Water
Functionality	• Holds air and allows for water to be pumped inside, pressurizing the air, the water will allow for the ROV to control density

Module	**Water Pump**
Inputs	● Power from Battery through Relay ● Water
Outputs	● Water
Functionality	● Pumps water into the Ballast pressurizing the contained air

Table 4: Hardware Level Two Functional Requirements

[CJB PAL,MKC]

B. Software

Theory of Operation (Level 0): The software block will take user data and convert it into motion on the ROV. It will decode video and display it to the user. The software block will decode/encode signals to be sent across a wireless link from/to user and ROV.

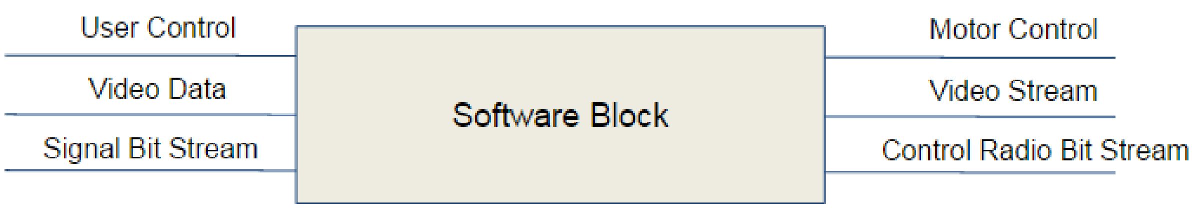


Figure 5: Software Level Zero Block Diagram

Module	**Software Block**
Inputs	● User Control ● Video Data ● Signal Bit Stream

Outputs	• Motor Control • Video Stream • Control Radio Bit Stream
Functionality	• Decodes user control signal to give drive to motors. • Decodes video data and creates a video stream to be sent back to user

Table 5: Software Level Zero Functional Requirements

Theory of Operation (Level 1): The software will consist of a GUI that will allow for the user to view video and a depth reading from the ROV. The GUI will transfer user controls to the transmitter to be sent to the ROV transmitter via a wireless signal. The ROV transmitter will receive the data and produce it to the decoder. The decoder will decode the received messages and send control signals to the pins that actuate movement controls. Data will be gathered from the depth sensor and camera in a data gathering loop. The data gathering loop will store the camera data and send it to a compressor to be compressed before being sent to the transmitter and sent over the wireless link. Memory storage will store the raw video data in storage.

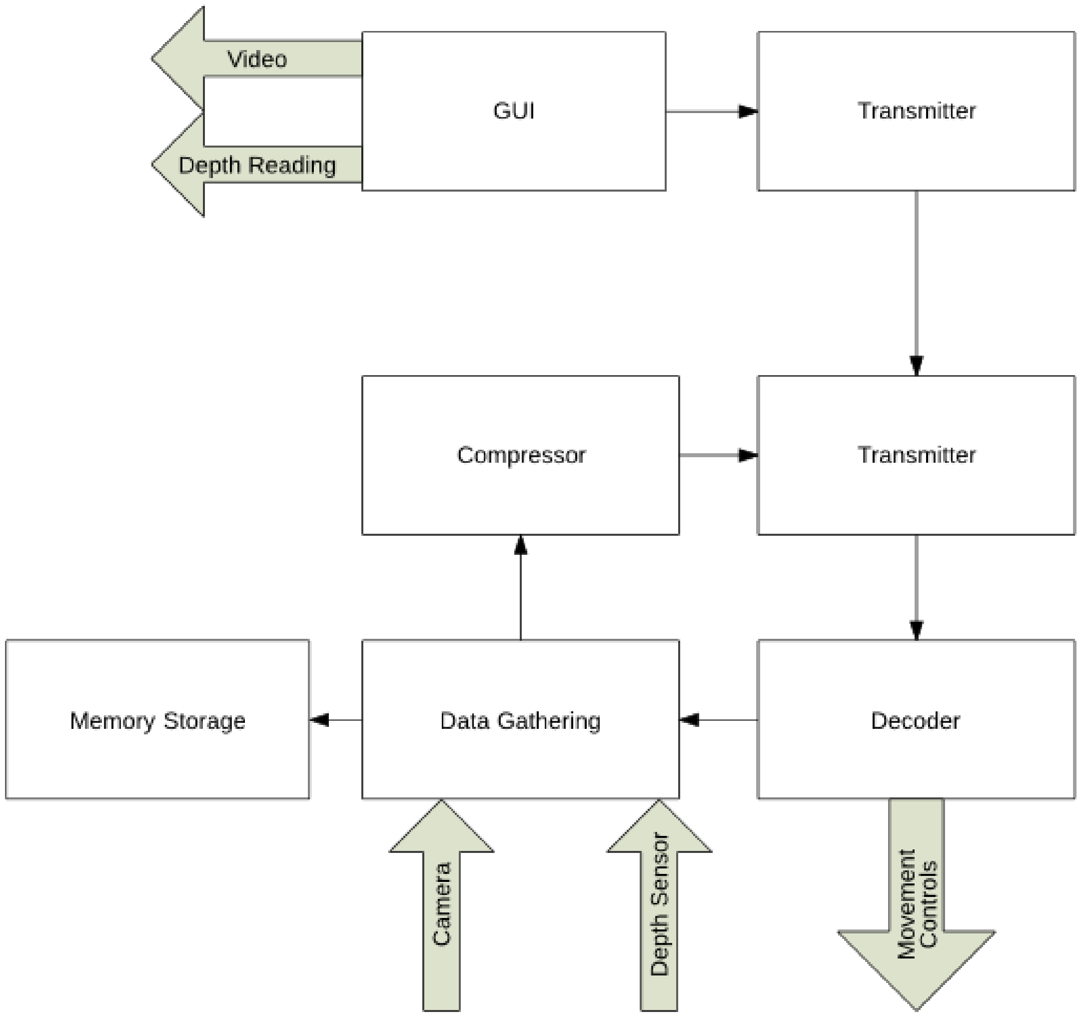


Figure 6: Software Level One Block Diagram

Module	**GUI**
Inputs	• User Controls • Data from Transmitter
Outputs	• Video • Depth Reading • Data to Transmitter

Functionality	• Produces to the user and allows for the user to communicate with the ROV system

Module	**Transmitter (1)**
Inputs	• Data from hardware • Data from GUI
Outputs	• Data to hardware • Data to GUI
Functionality	• Allows for data to be transmitted through wireless transceivers

Module	**Transmitter (2)**
Inputs	• Data from hardware • Data from compressor
Outputs	• Data to hardware • Data to Decoder
Functionality	• Allows for data to be transmitted through wireless transceivers

Module	**Decoder**
Inputs	• Data from Transmitter
Outputs	• Signals to movement devices

Functionality	• Decodes signals from the transmitter to create words and send signal bits to the movement motors/devices

Module	**Data Gathering**
Inputs	• Data from Depth Sensor • Data from Camera
Outputs	• Camera data to Compressor • Camera data to Memory Storage
Functionality	• Sends data to memory storage to be saved once ROV is retrieved and the storage is removed • Sends Camera data to the Compressor to be compressed before

Module	**Compressor**
Inputs	• Retrieves data from Data Gathering
Outputs	• Compressed data to the Transmitter
Functionality	• Takes raw data and compresses it to be sent to the Transmitter

Table 6: Software Level One Functional Requirements

Theory of Operation (Level 2): The theory behind this level is to separate the the different blocks from the Level 1 diagram into their particular parts. Each part will be a method or class that will handle an operation.

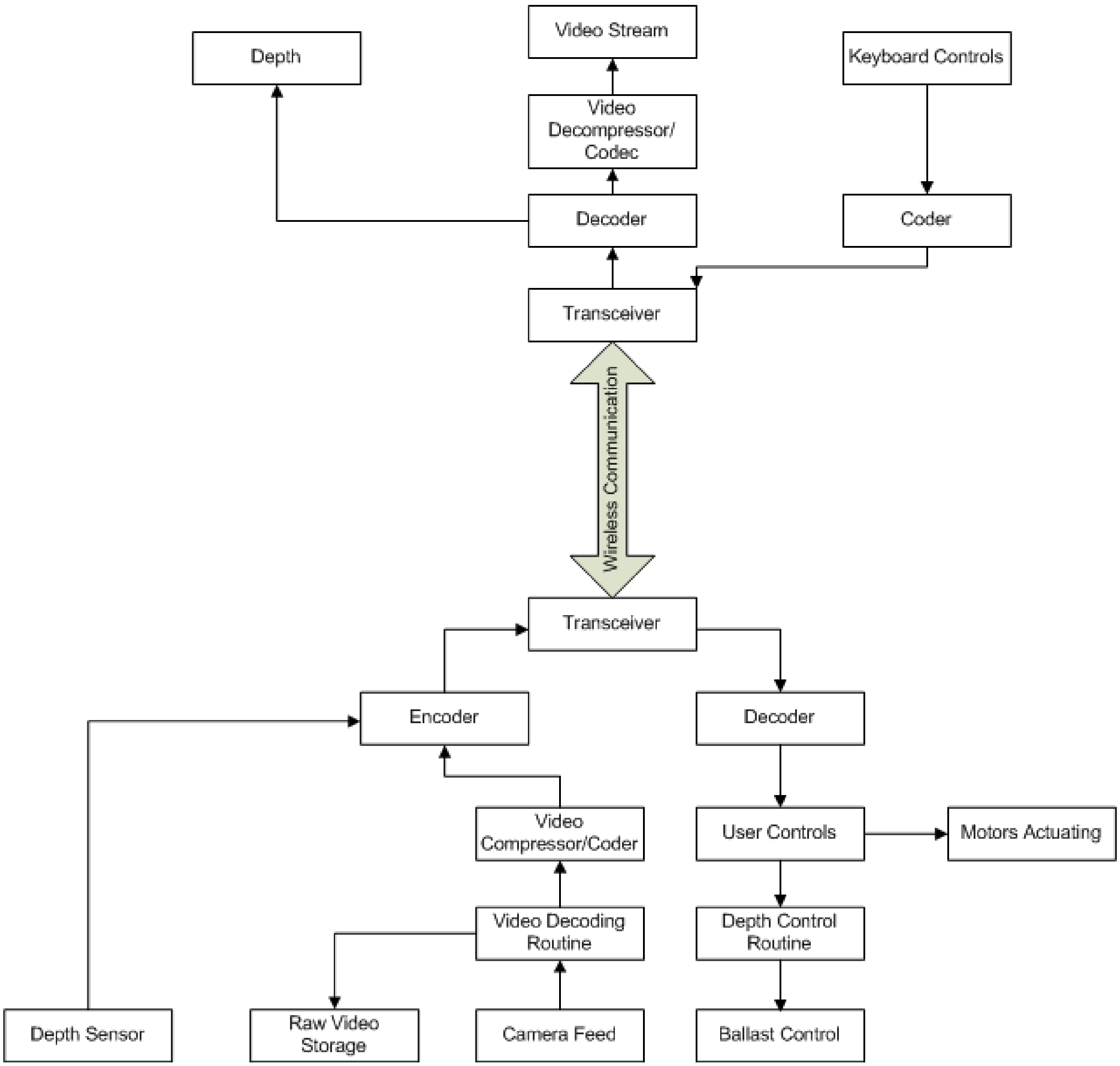


Figure 7: Software Level Two Block Diagram

Module	**Depth**
Inputs	• Variable value from Decoder
Outputs	• String
Functionality	• Shows depth reading

Module	**Video Stream**
Inputs	• Video from Decompressor
Outputs	• Video to User
Functionality	• Displays the decompressed video

Module	**Keyboard Controls**
Inputs	• User controls from keyboard strokes
Outputs	• ASCII characters to Coder
Functionality	• Accepts user keystrokes

Module	Coder
Inputs	● ASCII characters from Keyboard Controls
Outputs	● Coded Messages to Transceiver buffer
Functionality	● Codes ASCII characters into controls for the ROV

Module	Decompressor
Inputs	● Words from Decoder
Outputs	● Decompressed video to Video Stream
Functionality	● Decompresses compressed signal

Module	Decoder
Inputs	● Data from Transceiver buffer
Outputs	● Data to Depth variable ● Data to Decompressor
Functionality	● Decodes produced data in Transceiver buffer

Module	**Transceiver (1)**
Inputs	● Data from hardware (wireless signal) ● Data from Coder
Outputs	● Words to Decoder ● Data to hardware (wireless signal)
Functionality	● Has output/input buffers to receive and send data over hardware

Module	**Transceiver (2)**
Inputs	● Data from hardware (wireless signal) ● Data from encoder
Outputs	● Data to hardware (wireless signal) ● Words to decoder
Functionality	● Has output/input buffers to receive and send data over hardware

Module	**Decoder**
Inputs	● Words from Transmitter
Outputs	● Words to User Controls block
Functionality	● Decodes received buffer from transmitter

Module	User Controls
Inputs	● Data from Decoder
Outputs	● Sends on/off bits to actuate motors ● Sends signal to Depth Control Routine to submerge/emerge into/from water
Functionality	● Translates user control into signals to hardware

Module	Depth Control Routine
Inputs	● Signal from User Controls
Outputs	● Signals to Ballast Control in a particular order
Functionality	● Translates a dive/surface command to proper order of events to perform the desired task

Module	Ballast Control
Inputs	● Data in a particular order from the Depth Control Routine
Outputs	● Signals to hardware
Functionality	● Sends the signals to the ballast system hardware to dive or surface

Module	Encoder
Inputs	• Data from Depth Sensor • Data from Video Compressor
Outputs	• Words to Transmitter
Functionality	• Combines data and places it in the Transmitter to be sent via the wireless link

Module	Depth Sensor
Inputs	• Data from hardware
Outputs	• Data to Encoder
Functionality	• Translates data from hardware to be encoded and sent

Module	Video Compressor
Inputs	• Video data (RAW)
Outputs	• Compressed video to be sent over wireless link
Functionality	• Compresses video data so it can be sent over link

Module	**Video Decoding Routine**
Inputs	• Data from Camera Feed
Outputs	• RAW data to be stored • RAW data to be compressed
Functionality	• Receives data from the Camera Feed

Module	**Camera Feed**
Inputs	• Data from hardware
Outputs	• RAW video data to the Video Decoding Routine
Functionality	• Translates data from hardware into workable words

Module	**Raw Video Storage**
Inputs	• RAW video data from Video Decoding Routine
Outputs	• Video storage (data and location) to hardware storage
Functionality	• Stores RAW video data in order on storage device

Table 7: Software Level Two Functional Requirements

[PAL]

C. Lighting Theory of Operation

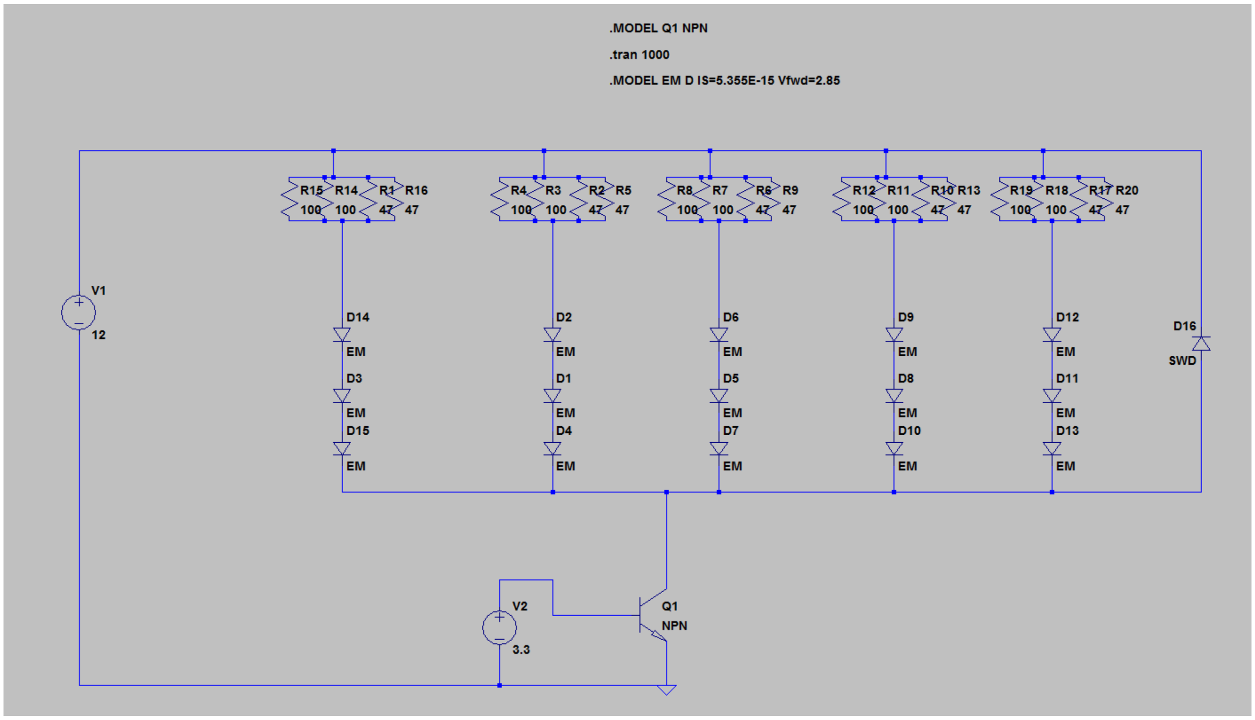


Figure 8: LED Circuit

The Lighting from the LED lights will need to supply a sufficient amount of light to allow the ROV Camera to clearly detect images. The lights will originate at points close to the camera and will need to illuminate an area 2 meters away. The lights will be orientated in such a way that a cone will be created with an angle θ such that at 2 meters a circle with radius r will be illuminated with 320 to 500 lux. The angle θ can be found with the equation $sin(r/2) = \theta$. The area illuminated will be $A=\pi r^2$. Without any particles reflecting light the amount of lumens the LEDs would need to generate is $Lux=LED/A$ where LED is the Lumens generated from the LEDs, Lux is the amount of light at the surface being illuminated, and A is the Area of the surface displayed. Because the ROV is underwater there may be particles that reflect the light generated by the LEDs. Here the Absorption A is equal to $A=log10(I_0/I)=a*b*c$ [5] where I_0 is the Incident Light, I is the Light transmitted from the LEDs, a is the ability of a particle in the water to absorb light, b is the path length of the light, and c is the concentration of the water. Solving for I we have $I=I_0/10^{abc}$ this I will need to be taken over the surface area A if any absorption occurs. The desired viewing radius will be .5 meters. Solving for the angle θ yields a value of 14-15 degrees. The desired lumens in air is 500=LED/(3.14*.5^2) or 392.5 lm. With a high absorption of 1.0, the lms made from the LEDs will be 10=I_0/392.5 Meaning the desired output in lumens will 3925 lms a minimum for a webcam style camera to view objects will be 2512 lms.

[PAL]

D. Data Transmission/Storage needs

The data that will be stored will be based on quality of video and the camera used and can be viewed in the chart below:

Definition	Resolution	Pixels Per Image	Bits per Frame	MB Per Frame	MB Per Second at 30 fps	GB Per 30 Minutes
480p	640 X 480	307200	7372800	.88	26.4	46.41
720p	1280 X 720	921600	22118400	2.64	79.2	139.22
1080p	1920 X 1080	2073600	49766400	5.93	177.9	312.7

Table 8: Data Transmission/Storage Needs

These values are based on using a 16-bit RGB Camera in its RAW format. To transmit a video feed from the ROV a bandwidth of $79.2 * 8 = 633.3$ Megabits per second (Mbs). Using a standard codec such as H.264 the pixels can be compressed to only require .1 [7] bit per pixel (instead of the 16-bit per pixel monochrome X 3 for RGB, i.e. 16x3 bits per pixel in RAW). Using a codec such as H.264 the needed data transfer rate for 720p could become $30 * (921600 * .1/1024) = 2700\ Kbps$ or roughly 2.7 Mbps.

[PAL]

E. Ballast Tank Requirements

In order to make the ROV float at the surface, and also allow the ROV to sink, a ballast tank must be capable of changing the density of the ROV to that of less than water (ballast tank full of air), and greater than water (ballast tank full of water). The density of fresh water is assumed to be a maximum 1000 kg/m^3. The ballast system will be able to pump water into the ballast in order to manipulate the density of the ROV forcing the ROV to submerge under the surface of the water. The density of the ROV (D_s) will have 3 values relative to the density of water (D_w):

$D_s < D_w$ where the ROV will float

$D_s = D_w$ where the ROV will be at neutral buoyancy

$D_s > D_w$ where the ROV will sink

In order to achieve buoyancy, the ROV must have a density less than the water, which can be any arbitrary amount less than the assumed water density. The less dense the ROV, the faster it will rise to the surface due to the net force acting upon the ROV defined below. The same principal applies to sinking, when the density of the ROV is greater than that of water, the ROV will begin to sink.

To determine the size of the ballast tank required for our application, four things must be known, the total volume displacement (V_w) of the ROV (including the ballast tank), the desired minimum density (ρ_{min}), the desired maximum density (ρ_{max}), and maximum water to compressed air ratio inside of the tank (R). To find a relationship for the total net force acting on the ROV in the z axis using the assumed densities, a relationship must be defined between the buoyancy forces (F_b) and gravitational force (F_g) equations.

$$F_b = \rho_w V_{ROV} g$$

$$F_g = \rho_{ROV} V_{ROV} g$$

Where the density of water is ρ_w, the density of the ROV is ρ_{ROV}, the volume of water displaced by the ROV is V_{ROV},and the acceleration of gravity is g. Using these two force equations, the net force acting on the ROV is simply F_n, where forces in the positive direction are downwards.

$$F_n = F_g - F_b$$

When the ROV is neutrally buoyant, the value for F_n is 0, and the ROV will remain at the current depth, unless if other forces such as water currents or motor thrust acts upon it. When F_n is negative, the ROV will float upwards, and when F_n is positive, the ROV will sink.

Now, to determine the ballast tank size using the arbitrary densities picked earlier, a necessary amount of water to change the ROV density from the minimum desired density to the maximum desired density must be calculated. To do this, we must first assume a volume displacement (V_{ROV}) for the ROV, and then use the below equation to find the mass of water (M_w) needed.

$$M_w = ((\rho_{max} V_{ROV}) - (\rho_{min} V_{ROV})$$

Then, the volume of water in cm^3 can be directly determined by the relationship of water kg to cm^3 shown below.

$$V_w = 1000 * M_w$$

Using this volume, the dimensions of the ballast tank can be found. The ballast is assumed to be a cylindrical body, so the volume can be found from a simple geometric principle.

$$V = \pi r^2 h$$

Using this relationship and the ratio of water to air, a necessary cylinder height can be determined, when using standard inner diameter values for PVC piping. The total volume of the cylinder necessary would be the volume of necessary water multiplied by the inverse of the water to air ratio.

$$V_t = V_w R^{-1}$$

$$h = \frac{V_t}{\pi r^2}$$

$$h = \frac{V_w R^{-1}}{\pi r^2}$$

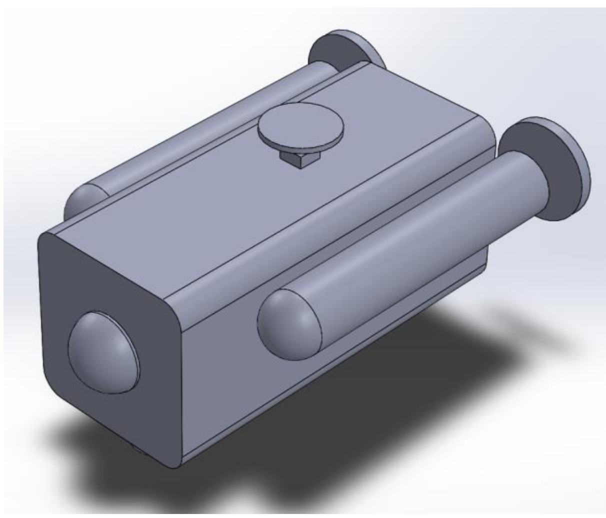
With the above relationship, a necessary minimum ballast length (cm) with regards to a standard PVC pipe inner diameter.

An important thing to note about the calculated ballast volume must be included in the already assumed volume displacement of the ROV.

[JEB, PAL]

F. Structural Design

The structure of the ROV is not a core design component besides being aerodynamic enough to allow for the minimum top speed of 3.5mph, and capable of submerging to 50 ft without water damage. Because of this, PVC piping will be used for the majority of the hull, while the core of the sub body will involve a plastic box sealed with an O ring. The general design concept can be seen in Figure 9.



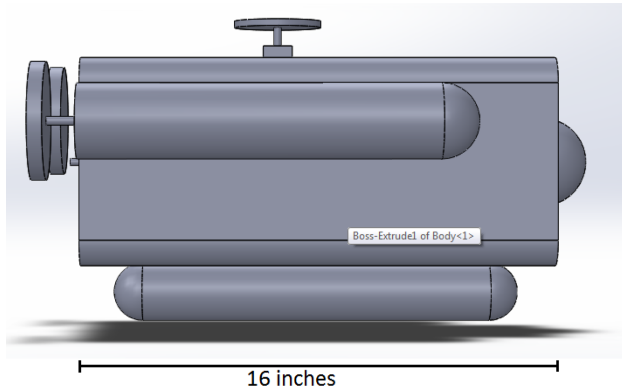


Figure 9: ROV Structure Design Concept

The structure attached to the bottom will act as the ballast, while the two structures on the sides will house the motors and have the main propellers on the back. The propellers are shown as the circular disks, two on the back for horizontal movement, and one on top for vertical movement. The Dome in the front of the ROV will serve as the port where the camera resides, along with the lighting system. Inside of the main body, will be both batteries, the Raspberry Pi, and necessary circuitry to accompany the motor and ballast controls. Extending from the back of the body will be a 1 ft pole where the antenna buoy line leaves the ROV. This is used to keep the ROVs propellers away from the wire as much as possible to prevent damage, since the 50ft of cable will just be floating in the water alongside of the ROV.

[JEB]

G. Wireless Communication Method

The ground system and ROV will communicate via wireless communication using transceivers that communicate using the IEEE 802.11n standard. The IEEE 802.11n can allow for a stream data rate of 600 Mbps (theoretical) and a range of up to 250m. The coded H.264 bit stream will be sent to the ground system using the Real-time Transport Protocol (RTP) over User Datagram Protocol (UDP). RTP is a common protocol used in streaming video between endpoints often implemented with UDP. UDP allows for unicast or broadcast/multicast behavior. As our ROV will only be networked with our ground system, either will sufficiently work to reliably allow for data transmission. RTP allows for the correction of out of order words to be correctly ordered before it is viewed/heard by an endpoint.

The wireless link will be accomplished with a 300Mbps High Power USB based wireless adapter with a 2T2R multiple input multiple output (MIMO) package. This wireless device is capable of 900ft of line of sight connection, offers USB 2.0 data speeds of 480 mbps on the USB connection, and transmits at 300Mbps. Using two of these devices, connected in an ad-hoc wireless connection, is how the link will be established.

The last problem to address with the wireless link is the 50ft USB connection from the buoy antenna, to the ROV. Since USB signals deteriorate quickly, the signal needs to be regenerated every so often, or sent across a different wire type, such as CAT5. An active repeater cable will be used here, because sending the USB signal across a CAT5 can cause USB speeds to slow down to USB 1.1 speeds of 12 Mbps. Using an active repeater cable causes no data transfer speed loss, at the cost of being expensive part.

[PAL, JEB]

H. ROV Computer and Camera

The ROV will implement a Raspberry Pi 3B for the main computer running the core software. The Raspberry Pi will control the motors, the ballast, camera, and wireless for the ROV, as shown in earlier block diagrams.

The camera will be a Raspberry Pi NoIR Camera Module V2. This is a 1080p30 camera, meaning it is 30 frames per second, with HD quality and progressive images. This allows for the stored video to have good quality and complete images.

[JEB]

I. Power Analysis

One of the main issues surrounding the project is the power consumption of the ROV. From the lights to the motors to the valve and a whole bunch of other components, this ROV consumes a lot of power. Because of this, two batteries are to be put into place in order to bypass voltage regulators and distribute power consumption.

The first battery to be used is a 5VDC Lithium Ion Battery, specifically the USB Battery Pack for the Raspberry Pi. Coincidentally, this battery pack is going to power the Raspberry Pi in the ROV. The Raspberry Pi will then be responsible for supplying the power to the camera, depth sensor, data storage, and all of the electrical switches (BJTs, relays, etc.). The referenced battery pack will be able to supply the Raspberry Pi with power at 1A for 4 hours (4000mA Battery Pack).

The Raspberry Pi is capable of supplying 2.5A to all of its subsequent devices. This could be possibly problematic due to the fact that the Raspberry Pi NoIR Camera that is to be used is recommended to be ran at 2A at all times for maximum quality and efficiency. This doesn't leave too much room for the other devices that are going to be implemented into this ROV.

The other battery to be used is going to be a 12VDC Gruber Power Systems 18Ah Battery. This battery will be used to power all of the main ROV subsystems. It's going to power the lights, the water pump and the valve for the ballast, and the motors for the propulsion system. With a total power consumption of about 30 amps of current draw which will meet the projected run time of 30 minutes easily with the 18Ah battery. With the choice of this battery the power requirements are meant but also out weight is not affected too much with an overall weight of 12.57 lbs it will not add too much weight to the ROV.

[MKC,CJB]

J. Motor Control

The motor control design is actually very simple. It only consists of four separate parts: the Raspberry Pi, 12VDC battery, magnetic relays, and the motors. The motors we selected are high torque 12VDC geared motors. A simple electrical diagram using Circuit lab displays the motor design in Figure 10. Along with these motors we will be using larger propellers to create a greater thrust because the ROV is larger and heavier compared to most commercial ROVs.

To control these motors we will use the 12VDC battery that is ran through a relay to power them. The magnetized relay will be controlled by the Raspberry Pi via 5VDC signals to the relays to magnetize them. The 5VDC signals from the Raspberry Pi will be sent out with the GUI and the use of a keyboard using WASD for X and Y axis control, and Shift and Control for Z axis control.

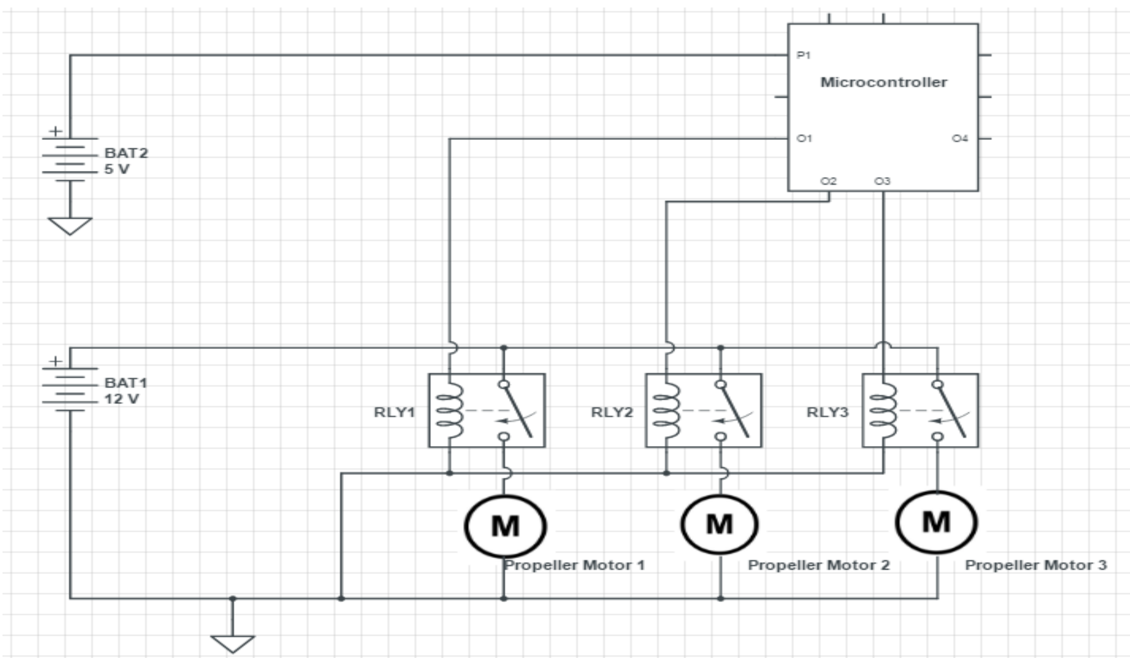


Figure 10: Simple Design of the Motor Control System

[CJB]

K. Ballast Control

The ballast is a fairly complicated part of the design. First, the ballast tank size was determined using the equations in Section D and it was determined that standard SCH40 PVC pipe would be used. After determining the size, the ballast system need to be implemented. There are a few possible ways to create a proper ballast. The method chosen was to have a 12VDC water pump connected to an external hose that's capable of pumping water into the tank. The projected flow rate is set at a max of 780 L/h. The hose would then be connected to a 12VDC ½ in. solenoid valve that is normally open. Once the valve receives an electrical signal it will open up and will allow the water into the ballast tank.

The valve needs to be normally open as a safety measure. If the ROV were to ever lose power or signal, the valve will open up and the compressed air inside the ballast tank will dispel the water inside and will make the ROV rise to the surface. A simple MS Paint diagram of the ballast design can be seen in figure 11.

Figure 11: Simple Design of the Ballast System

. [MKC]

L. Graphical User Interface (GUI)

The GUI will be a simple, WPF based application. As far as what the user sees, the window will comprise entirely of a video feed with a depth measurement (in feet) in one of the corners. Any warnings, such as a connection failure, will appear as dialog boxes telling the user the error that is being encountered.

Under the hood, the GUI is a loop that can be seen in the flowchart figure 12. It simply sends regular TCP control signals, and receives a constant stream of UDP video feed packets, and displays them. Another core feature is the video is saved locally (along with the video feed stored on the ROV). This is just another redundancy factor in case of ROV loss.

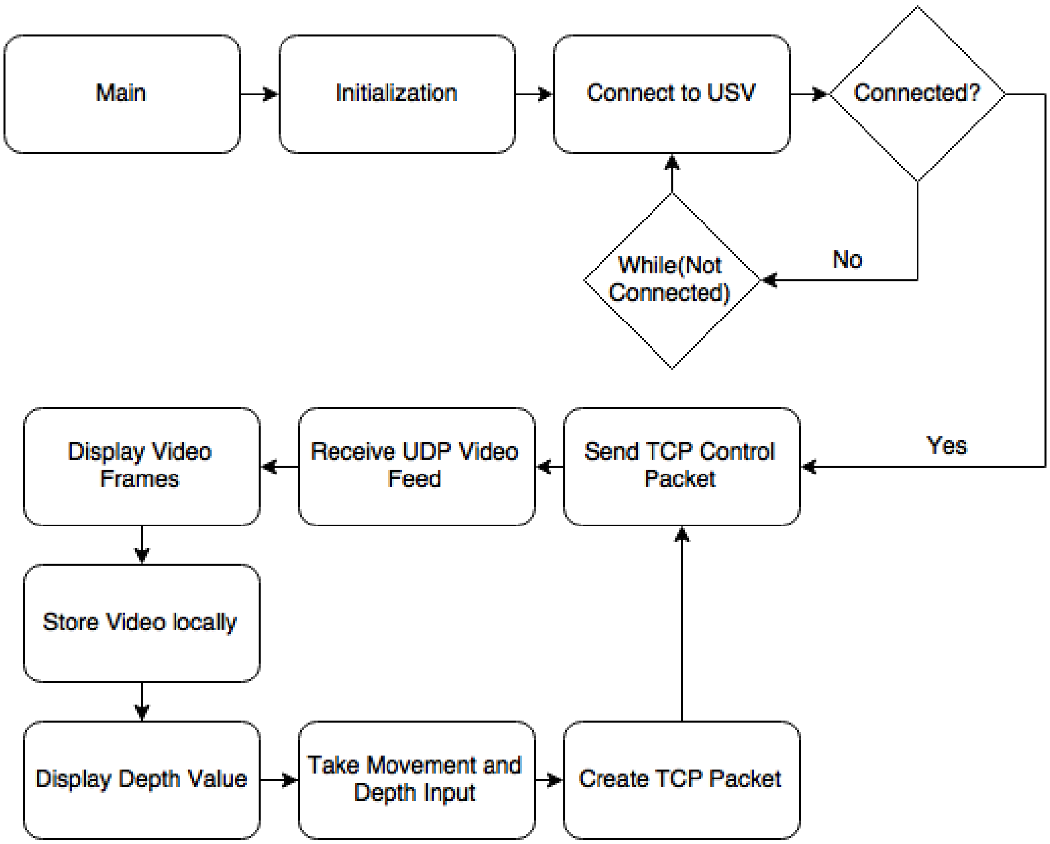


Figure 12: GUI Pseudo Code Flowchart

[JEB]

M. ROV Software

The ROV software will follow the below flow chart. Upon booting the ROV will attempt to connect wirelessly to the ground station. A state variable will be used to determine whether the ROV is connected wirelessly. If the state variable is ever set to false the ROV will attempt to reconnect to the ground system. The software will use threads to monitor the I/O buffer of the wireless communication system sending and receiving data to/from the transceiver. A second thread will be used to collect and process data from the camera. What follows the flowchart is a rough outline of the program with methods to be ran on the ROV.

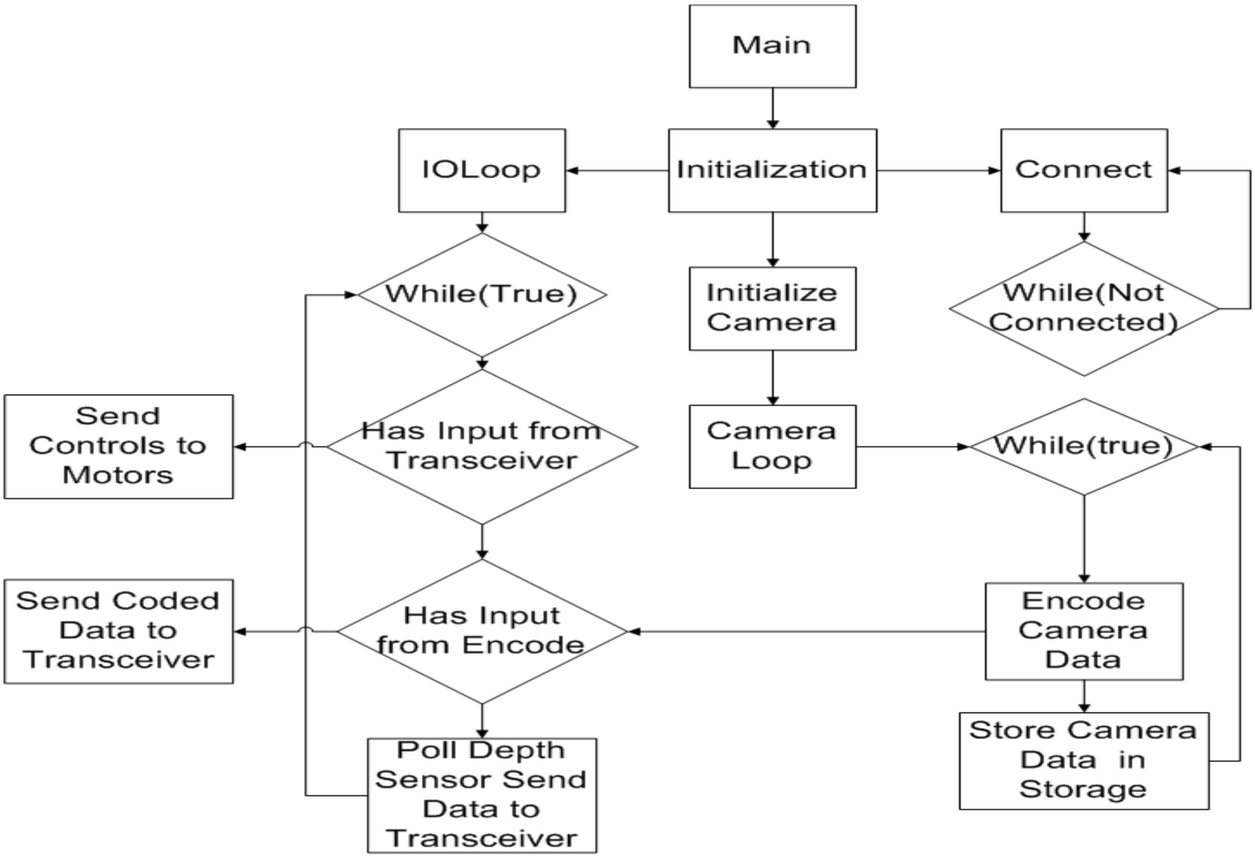


Figure 13: ROV Pseudo Code Flowchart

```
/*
Design Team 09 Underwater Robot
Pseudo Code
Peyton Lucas
*/

void Main[args]{
	//Main Method will call the initialization method
}

void initialization{
	//Will initialize threads to watch IO buffer
	//Will initialize variables
	//Will ensure device connectivity by calling while loop
}

void BaseConnect{
	/*
	while(notconnected){
		will continue to connect to base station
		will be called if the ROV ever loses device connection
```

```
        }
        */
}

void IOLoop{
                /*
                while(true){
                //Will use a thread to continually watch Hardware IO buffer
                //Will use threads to pass values to controls and the output buffer
                //Will regularly pole depth sensor
                //Will use produced/consumed tags to send data through transceiver
                //Will take data from encoder and place in empty output buffer
                }
                */
}

void CameraLoop{
        /*
        while(true){
        Will use a thread to gather camera data from buffer and produce values to be encoded
        Will store raw camera data in storage
        }
        */
}

void Encode{
        //Encodes values from camera using H.264 to be sent over IO
        //Will flag a produced flag when data is ready to be sent over IO
        //IO loop will clear produced tag and flag consumed tag when data is consumed by the
IOLoop
}

void Controls(arg){
                //Called by IOLoop when Input buffer has data
                //Takes value from IOLoop and translates it to desired protocol to actuate motors.
}
```

[PAL]

IV. Project Schedules

A. Midterm Design Gantt Chart

Name	Begin date	End date	Resources
Fall Project Design	10/10/16	12/1/16	
Midterm Report	10/10/16	10/24/16	
Design requirements Specification	10/10/16	10/24/16	Joe Beck
Gantt Chart	10/10/16	10/24/16	Joe Beck
Calculations	10/10/16	10/24/16	
Electrical	10/10/16	10/24/16	
Battery requirements	10/10/16	10/24/16	Cody Bobek
Power Draw	10/10/16	10/24/16	Cody Bobek,Matt Crislip
Light requirements	10/10/16	10/24/16	Peyton Lucas
Motor requirements	10/10/16	10/24/16	Cody Bobek,Matt Crislip
Wireless transmission req...	10/10/16	10/24/16	Peyton Lucas
Software	10/10/16	10/24/16	
Minimum wireless bandwi...	10/10/16	10/24/16	Peyton Lucas
Mechanical	10/10/16	10/24/16	
Ballast:Vehicle size relatio...	10/10/16	10/24/16	Joe Beck,Peyton Lucas
Pressure vessel calculations	10/10/16	10/24/16	Joe Beck
Block Diagram Lvl 1	10/10/16	10/24/16	Matt Crislip
Block Diagram Lvl 2	10/10/16	10/24/16	Matt Crislip
Block Diagram Lvl N+1	10/10/16	10/24/16	Matt Crislip
Midterm Design Report	10/24/16	10/24/16	Joe Beck,Matt Crislip,Cody Bobek,Peyton Lucas
Midterm Design Presentation	10/21/16	10/21/16	Joe Beck,Matt Crislip,Cody Bobek,Peyton Lucas
Final Design Report	10/25/16	12/1/16	
Abstract	10/25/16	12/1/16	Joe Beck
Software Design	10/28/16	12/1/16	
Vehicle	10/28/16	12/1/16	
Psuedo Code	10/28/16	12/1/16	Peyton Lucas
GUI	10/28/16	12/1/16	
Psuedo Code	10/28/16	12/1/16	Joe Beck
Hardware Design	10/28/16	12/1/16	
Motor System	10/28/16	12/1/16	
Schematic	10/28/16	12/1/16	Cody Bobek
Simulation	10/28/16	12/1/16	Cody Bobek
Ballast System	10/28/16	12/1/16	
Schematic	10/28/16	12/1/16	Matt Crislip
Simulation	10/28/16	12/1/16	Matt Crislip
Camera	10/28/16	12/1/16	
Schematic	10/28/16	12/1/16	Matt Crislip
Depth Sensor	10/28/16	12/1/16	
Schematic	10/28/16	12/1/16	Cody Bobek
Simulation	10/28/16	12/1/16	Cody Bobek
Final Design Presentation	12/2/16	12/2/16	Joe Beck,Matt Crislip,Cody Bobek,Peyton Lucas

Figure 14: Midterm Design Gantt Chart

B. Final Design Gantt

Name	Begin date	End date	Resources
Fall Project Design	10/10/16	12/7/16	
Midterm Report	10/10/16	10/24/16	
Design requirements Specification	10/10/16	10/24/16	Joe Beck
Gantt Chart	10/10/16	10/24/16	Joe Beck
Calculations	10/10/16	10/24/16	
Electrical	10/10/16	10/24/16	
Battery requirements	10/10/16	10/24/16	Cody Bobek
Power Draw	10/10/16	10/24/16	Matt Crislip,Cody Bobek
Light requirements	10/10/16	10/24/16	Peyton Lucas
Motor requirements	10/10/16	10/24/16	Matt Crislip,Cody Bobek
Wireless transmission req...	10/10/16	10/24/16	Peyton Lucas
Software	10/10/16	10/24/16	
Minimum wireless bandwi...	10/10/16	10/24/16	Peyton Lucas
Mechanical	10/10/16	10/24/16	
Ballast:Vehicle size relatio...	10/10/16	10/24/16	Joe Beck,Peyton Lucas
Pressure vessel calculations	10/10/16	10/24/16	Joe Beck
Block Diagram Lvl 1	10/10/16	10/24/16	Matt Crislip
Block Diagram Lvl 2	10/10/16	10/24/16	Matt Crislip
Block Diagram Lvl N+1	10/10/16	10/24/16	Matt Crislip
Midterm Design Report	10/24/16	10/24/16	Joe Beck,Matt Crislip,Cody Bobek,Peyton Lucas
Midterm Design Presentation	10/21/16	10/21/16	Joe Beck,Matt Crislip,Cody Bobek,Peyton Lucas
Final Design Report	10/25/16	12/7/16	
Abstract	10/25/16	12/1/16	Joe Beck
Software Design	10/28/16	12/1/16	
Vehicle	10/28/16	12/1/16	
Psuedo Code	10/28/16	12/1/16	Peyton Lucas
GUI	10/28/16	12/1/16	
Psuedo Code	10/28/16	12/1/16	Joe Beck
Part Choices	10/25/16	10/25/16	Joe Beck,Matt Crislip,Cody Bobek,Peyton Lucas
Hardware Design	10/25/16	12/1/16	
Motor System	10/28/16	12/1/16	
Schematic	10/28/16	12/1/16	Cody Bobek
Simulation	10/28/16	12/1/16	Cody Bobek
Ballast System	10/28/16	12/1/16	
Schematic	10/28/16	12/1/16	Matt Crislip
Simulation	10/28/16	12/1/16	Matt Crislip
Camera	10/28/16	12/1/16	
Schematic	10/28/16	12/1/16	Matt Crislip
Depth Sensor	10/28/16	12/1/16	
Schematic	10/28/16	12/1/16	Cody Bobek
Simulation	10/28/16	12/1/16	Cody Bobek
Vehicle Structure	10/25/16	10/25/16	
3D Model	10/25/16	10/25/16	Joe Beck
Final Design Presentation	12/8/16	12/8/16	Joe Beck,Matt Crislip,Cody Bobek,Peyton Lucas

Figure 15: Final Design Gantt Chart

V. Design Team Information

Joseph Beck, Computer Engineer.

Cody Bobek, Electrical Engineer.

Matthew Crislip, Electrical Engineer.

Peyton Lucas, Computer Engineer.

VI. References

[1] C.F. Motley, "Extended range undersea communication system," U.S. Patent 9 154 234, October 9, 2013.

[2] N.E. Farr, "Systems and methods for underwater optical communication," U.S. Patent 7 935 326, February 6, 2006.

[3] B. Wolfe, "Mobile device with an underwater communications system and method," U.S. Patent 9 270 387. February 23, 2016.

[4] N.Faidhon et al. "UNEXAR – Mini AUV Design & Measurement," in 2015 international Conference on Control, Electronics, Renewable Energy and Communications (ICCEREC), IEEE, August 27, 2015.

[5] F.M. Raimondi, "A Innovative Semi-Immergible USV (SI-USV) Drone for Marine and Lakes Operations with Instrumental Telemetry and Acoustic Data Acquisition Capability," in OCEANS 2015, Genova, IEEE, May 18, 2015

[6] Clark, J. (2007). Absorption spectra - the Beer-Lambert Law - Chemguide. Retrieved October 24, 2016, from http://chemguide.co.uk/analysis/uvvisible/beerlambert.html

[7] Kopansky, A., DR, Isnardi, M., Haudhry, A., Chai, I., & Xiong, P. (2010, May 19). H.264 Format/Bitrate/Quality Tradeoff Study. Retrieved October 24, 2016, from https://www.sri.com/sites/default/files/publications/3_07_h264_format_bitrate_quality_tradeoff_study.pdf

[JEB, PAL, CJB, MKC]