# DETAILED DESIGN REVIEW

## TEAM 4 – ROBOSUB

February 11, 2015

# COMPETITION OVERVIEW

- Hosted by The Association for Unmanned Vehicle Systems International (AUVSI)
- Located in San Diego, CA at the TRANSDEC pool
- Last year's competition had 7 tasks that require the sub to have various abilities

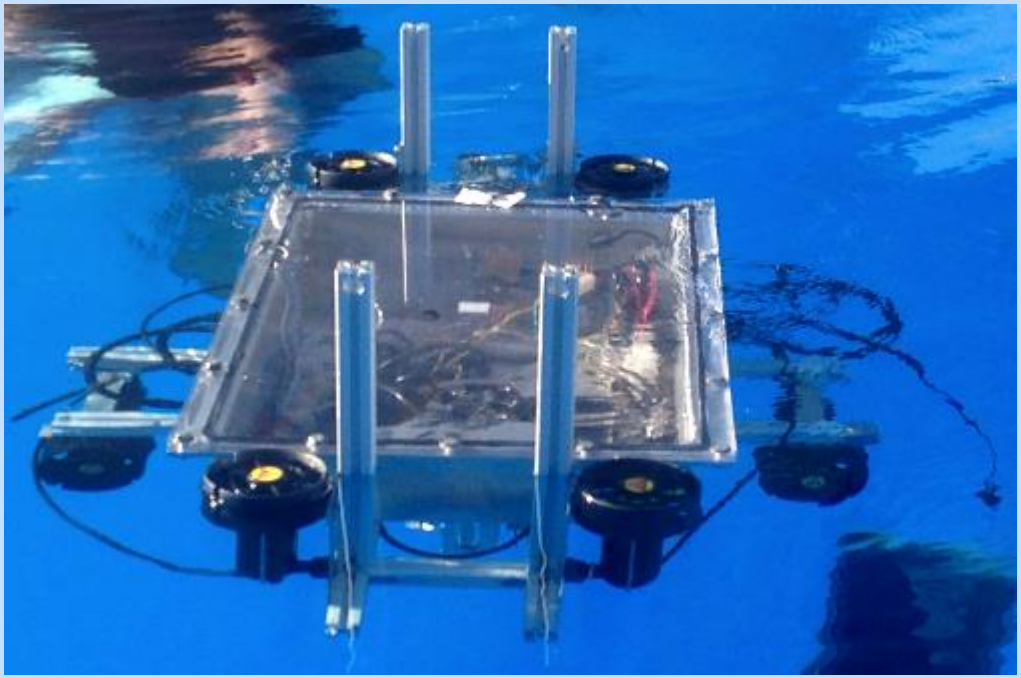
# INTRODUCTION

Key Requirements

- Run autonomously without any attachments
- Change depth, direction, and speed
- Pass through and around PVC structures
- Recognize colors

Key Limitations

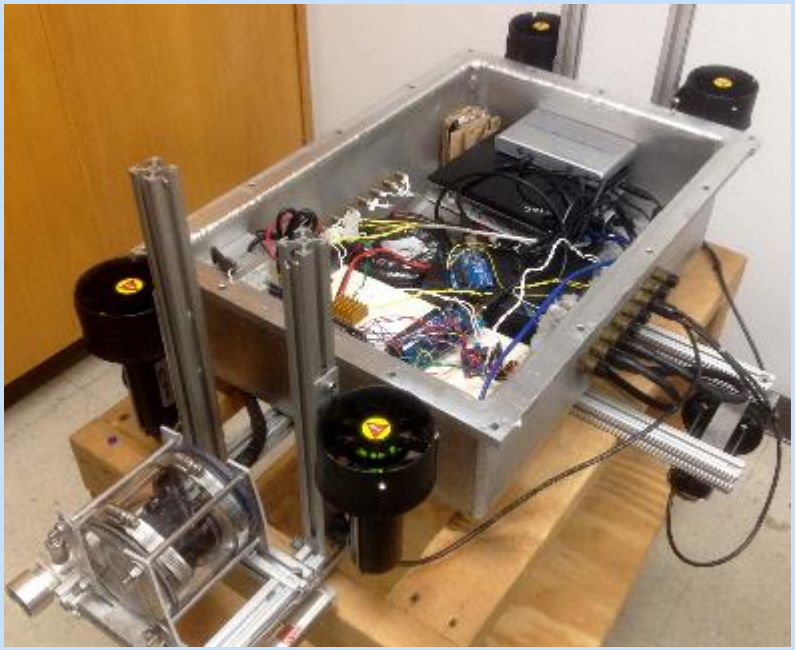
- Must use last year's sub
- Sub must weigh under 125 lbs

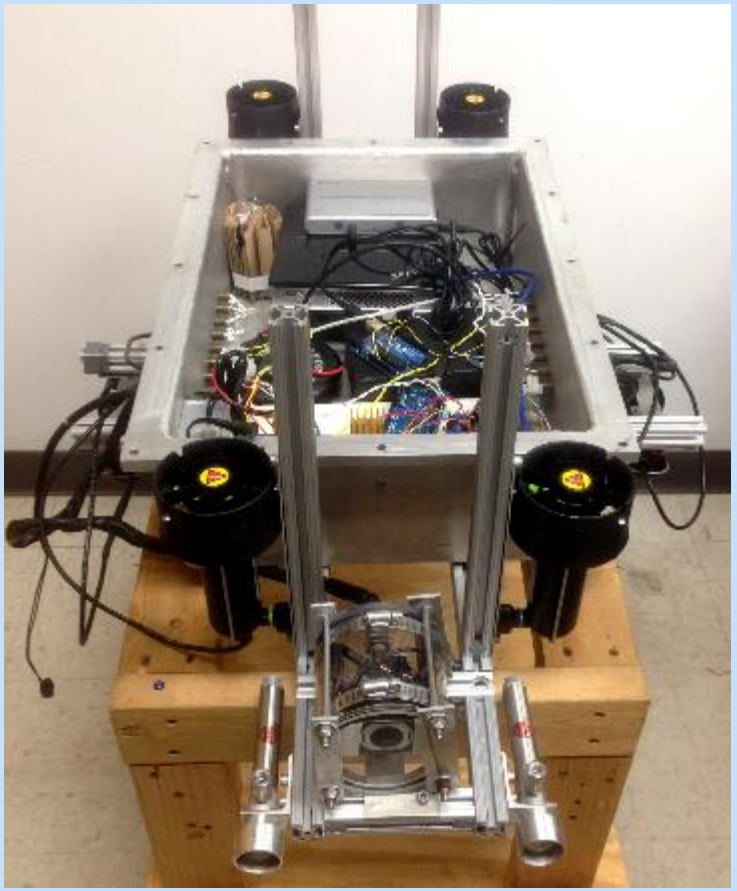


# CONCEPT GENERATION & SELECTION

# UPDATES TO MECHANICAL DESIGN

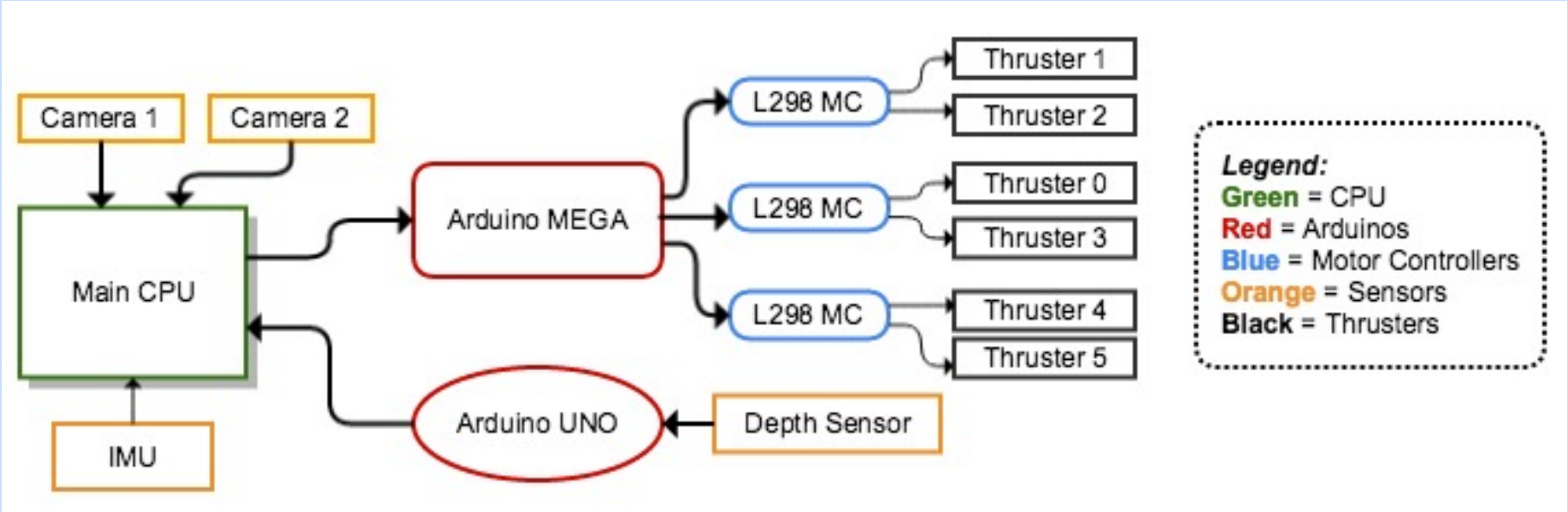
- Inherited previous year hull and frame structure
- Thruster Placement Changed
- Actuators Removed
- Gripper Removed





# ELECTRICAL DESIGN

- Cameras interface directly with the main CPU
- Arduino UNO interfaces with the depth sensor
- Arduino MEGA interfaces with the smaller controllers for the thrusters
- IMU also interfaces with CPU directly.

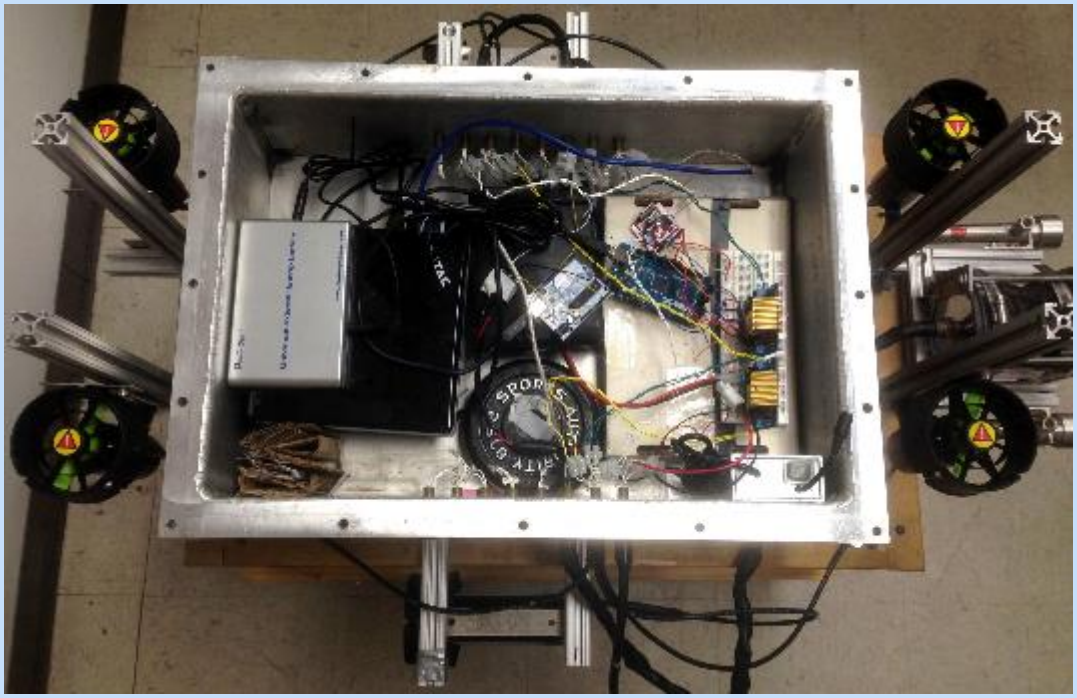


# DETAILED SYSTEM DESIGN

## COMPONENT INTERFACE

# POWER SYSTEMS

- Power Supplies
  - Universal Laptop Battery
  - Lithium Ion Battery Pack
  - Extra Battery Pack



| Components | Max Current (A) | Ave. Current (A) | Voltage Required (V) |
|---|---|---|---|
| Zotac PC Board | 3.5 | 1.5 | 19.0 |
| Arduino UNO | 0.75 | 0.5 | 7.0 - 12.0 |
| Arduino Mega | 0.75 | 0.5 | 7.0 - 12.0 |
| Motor Controllers | 2.0 | 1.5 | 5.0 |
| IMU | 0.075 | 0.060 | 3.5 - 16.0 |
| Thrusters | 12.0 | 3.0 | 19.1 |
| Depth Sensor | 0.020 | 0.012 | 8.0 - 11.0 |

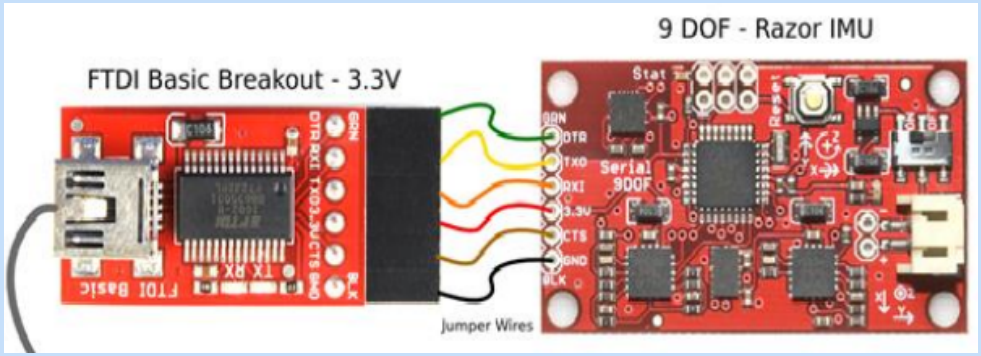
| Power Supply | Voltage (V) | Max Voltage (V) | Cut Off (V) | Max Discharge Current (A) | Capacity |
|---|---|---|---|---|---|
| Lithium Ion Battery Pack | 14.8 | 16.8 | 11.0 | 30.0 | 20 Ah or 296 Wh |
| Universal Laptop Battery | 16 or 19 | 19.0 | 13.0 | 3.0 | 4000 mAh |

# MAIN PROCESSING UNIT

- Zotac
  - CPU Intel Core i3-2330M Processor (2.2 GHz, Dual-Core)
  - Has 4 X USB 2.0 Ports, and 2 X USB 3.0 Ports
  - Max Capacity of 16GB
- Provides power for the Arduino MEGA, Arduino UNO and the Inertial Measurement Unit (IMU)

# POWERED FROM THE ZOTAC

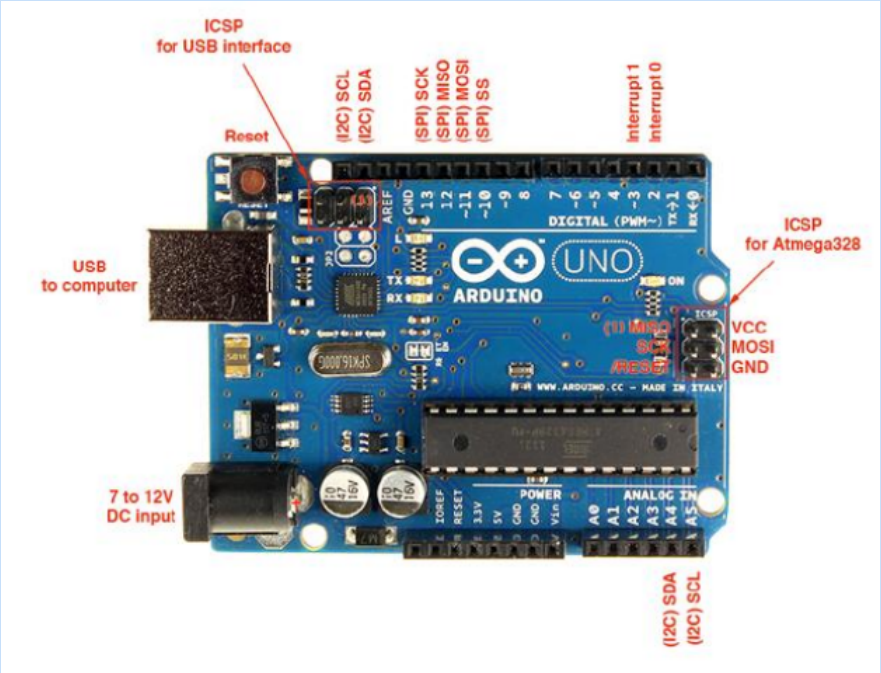
## IMU
**3.5 – 16V input**




## Logitech Webcams
**5 – 12V input**

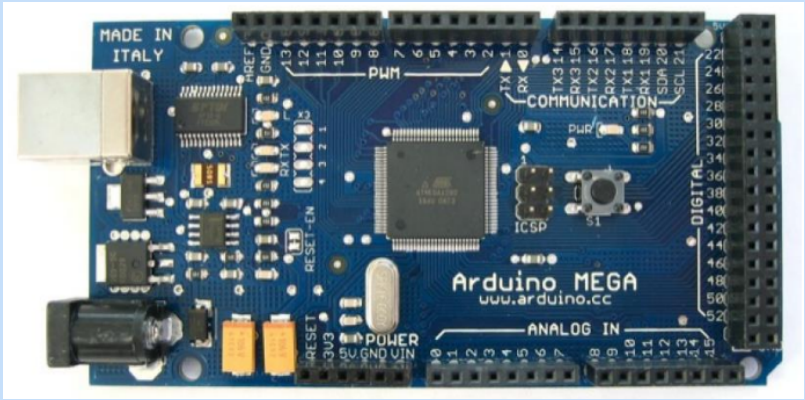
# ARDUINO UNO

- Operating Voltage : 5V
- Input Voltage : 7-12V
- Digital I/O Pins : 14 (6 Provide PWM output)
- Analog Input Pins: 6
- DC Current per I/O pin: 40mA
- Flash Memory: 32KB
- Clock Speed: 16MHz



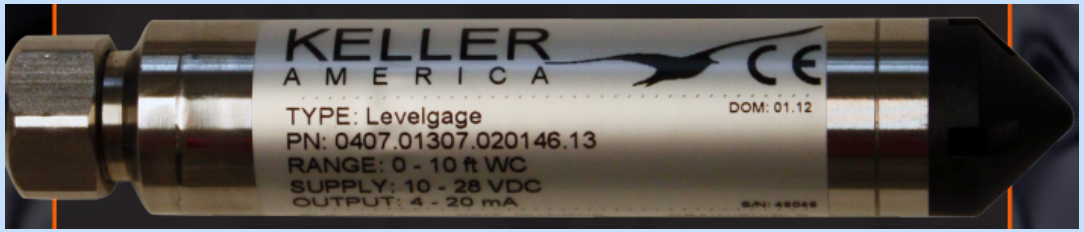
# ARDUINO MEGA

- Operating Voltage : 5V
- Input Voltage : 7-12V
- Digital I/O Pins : 54 (15 Provide PWM output)
- Analog Input Pins: 16
- DC Current per I/O pin: 40mA
- DC Current for 3.3V Pin: 50mA
- Flash Memory: 256KB
- SRAM: 8KB
- EEPROM: 4KB
- Clock Speed: 16MHz



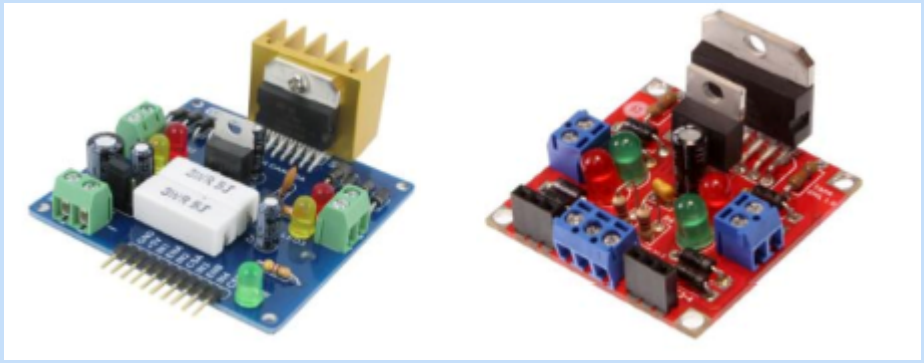
# DEPTH SENSOR

- 0-5V DC analog Output
- Produces Linear Results

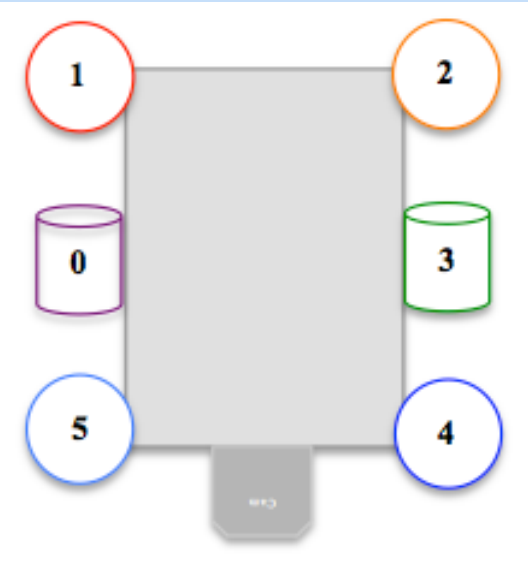



# THRUSTER SYSTEM

- Seabotix SBT150 Thrusters
  - Require 0 - 19.1 V
  - Zotac powers Arduino Mega
  - Lithium Ion Battery Pack Connected to Motor Controllers



# THRUSTER ORIENTATION



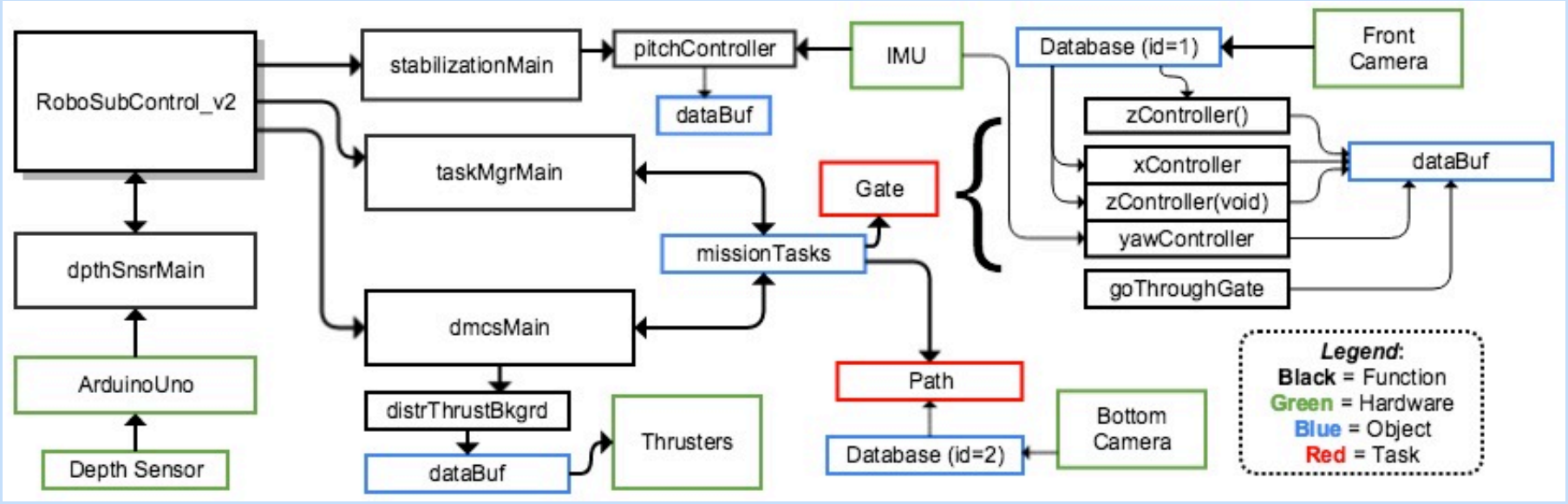

| Type of Movement | 5 | 4 | 1 | 2 | 0 | 3 |
|---|---|---|---|---|---|---|
| Forwards | Off | Off | Off | Off | + On | + On |
| Reverse | Off | Off | Off | Off | -- On | -- On |
| Rotate left | Off | Off | Off | Off | + On | Off |
| Rotate right | Off | Off | Off | Off | Off | + On |
| Ascend | -- On | -- On | -- On | -- On | Off | Off |
| Descend | + On | + On | + On | + On | Off | Off |

# DETAILED SYSTEM DESIGN

## SOFTWARE SYSTEM

# CODE STRUCTURE

- Build upon last year's code
  - Change Gate a little (simplified)
  - Add Path (using bottom camera)
  - 2 parts of Database now in use, differentiated with task_id



# PREVIOUS CODE

- DMCS (Decision Making Control System)
  - Run as a thread
  - Essentially decides what sub does based on current task
  - Tightly coupled with task manager for this reason
  - This is where new tasks are added
- Vision
  - Using OpenCV
  - Written to use the front camera to find orange
  - Locates center of orange, storing these values in the database
    - At task_id = 1
  - Modified to use the bottom camera as well
    - Places values in database at task_id = 2

```
case Types::GATE_COMMANDS::ALIGN_WITH_GATE:
{
    std::cout << "Aligning with the gate...\n";

    zController();

    std::cout << "aligned !\n";
    pthread_cond_signal(&cmdComplete);
    mySleep();
    break;
}
case Types::GATE_COMMANDS::APPROACH_GATE:
{
    std::cout << "Approachinxg the gate...\n";
    // spawn Z controller thread to maintain Z position
    pthread_create(&zCtrlr, NULL, &zController, NULL);
    // spawn X controller thread to approach target
    pthread_create(&xCtrlr, NULL, &xController, NULL);

    pthread_join(xCtrlr, NULL);
    pthread_cancel(zCtrlr);

    std::cout << "Gate distance threshold reached !\n";

    pthread_cond_signal(&cmdComplete);
    mySleep();
    break;
}
case Types::GATE_COMMANDS::GO_THROUGH_GATE:
{
    std::cout << "Going through gate...\n";
    goThroughGate();

    pthread_cancel(distrThrustBkgrd);
    std::cout << "Task completed !\n";

    pthread_cond_signal(&cmdComplete);
    myGate.setState(Types::TASK_STATE::COMPLETED);
    mySleep();
    break;
}
```

# SIMPLIFIED CODE SNIPPET

- Current version
  - Modified some from last year's
- Similar procedure planned for Path
- Note: zControllers here use the camera
- Other tasks may use the depth sensor to maintain z position (zControllerDepth)

# DEPTH SENSOR CODE

```
double feetUnderwater = 0.0;
int sensorOutput = 0;
double feetperVolt = 8.78; // calculated reative to the Morcom max depth

void setup()
{
  Serial.begin(9600);
}

void loop()
{
  sensorOutput = analogRead(0); // Analog Pin 0
  feetUnderwater = calcFeetUnderwater(sensorOutput);
  Serial.println(feetUnderwater);

  delay(500);
}

double calcFeetUnderwater(int sensorOutput)
{
  // sensorOutput will be between 0:1023, so map it to a voltage between 0:5
  double outputVoltage = sensorOutput *(5.0 / 1023.0) ;
  double feet = outputVoltage * feetperVolt;
  //Serial.println(feet); //for debugging

  return feet;
}
```

# BOTTOM CAMERA CODE

```
//video capture object to acquire webcam feed
VideoCapture capture;
VideoCapture camDwn;   // added for bottom camera
//open capture object at location zero (default location for webcam)
capture.open(0);
camDwn.open(1);        //added for bottom camera
```

- Added a new parameter to trackingFilteredObject

```
void trackFilteredObject(RoboSubColors theRoboSubColors,Mat threshold,Mat HSV, Mat &feedClone, string task)
```

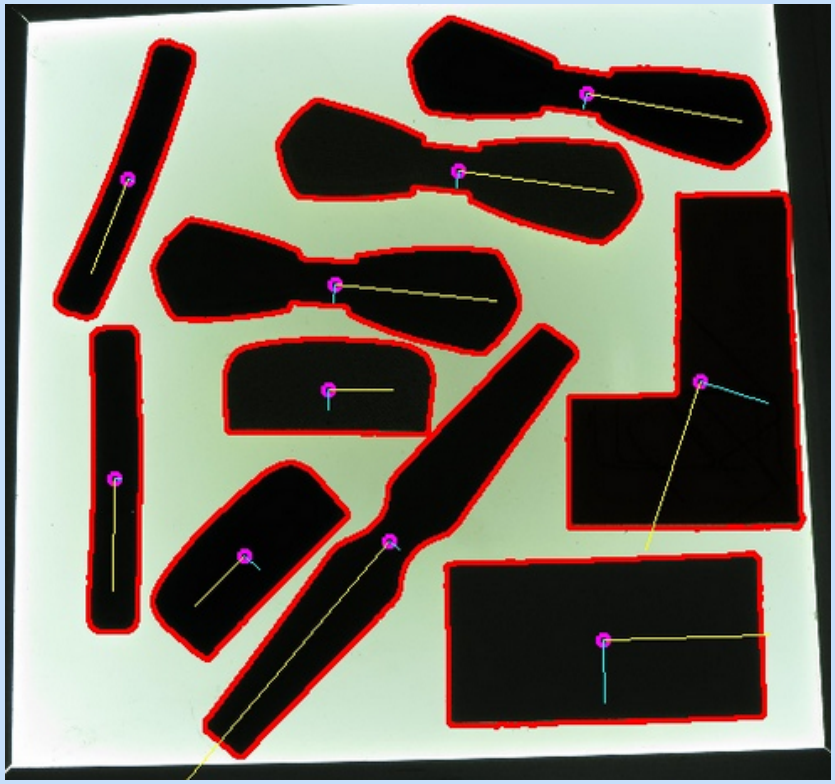
```
  //track orange objects
  cvtColor(cameraFeed,HSV,COLOR_BGR2HSV);
  inRange(HSV,orange.getHSVmin(),orange.getHSVmax(),threshold);
  morphOps(threshold);
  trackFilteredObject(orange,threshold,HSV,feedClone,"1"); // front camera

  cvtColor(camFeedDwn,HSVDwn,COLOR_BGR2HSV);
  inRange(HSVDwn,orange.getHSVmin(),orange.getHSVmax(),threshDwn);
  morphOps(threshDwn);
  trackFilteredObject(orange,threshDwn,HSVDwn,feedClnDwn,"2") //bottom camera
```

- Allows both bottom/front cam to update the database

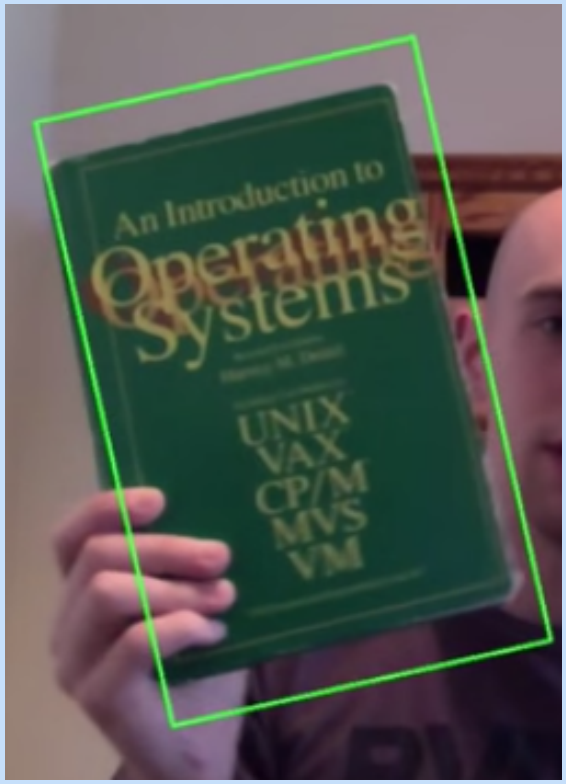
```
//changed Where task_id from 1 to +task to allow bottom camera to update the DB aswell
db.updateQuery("UPDATE Tasks_List SET task_detected = 1, distance = "+dis+", y_center = "+ycenter+", z_center = "+zcenter+" WHERE task_id = "+task);
```

# LINE FOLLOWING APPROACH

## Object Orientation Principal Component Analysis



## Camshift



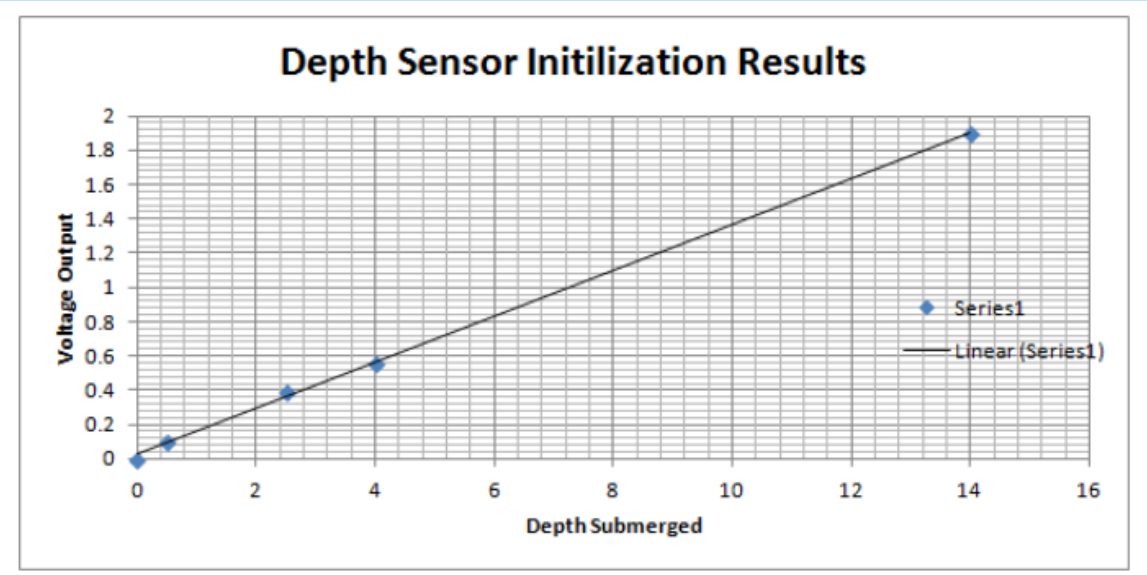
# TESTING PLAN STATUS

# SUMMARY

| Test | Status |
|---|---|
| **Power Systems** | Pass |
| **Ascend and Submerge Sub Using Thrusters** | TBD |
| **Maintain Depth** | TBD |
| **Underwater Object Recognition** | TBD |
| **Gate Traversal** | TBD |
| **Line Following** | TBD |
| **Gate and Path Combined Traversal** | TBD |
| **Waterproofing** | Pass |
| **Initial Depth Sensor Test** | Pass |
| **Gate Construction** | Pass |
| **Killswitch** | TBD |

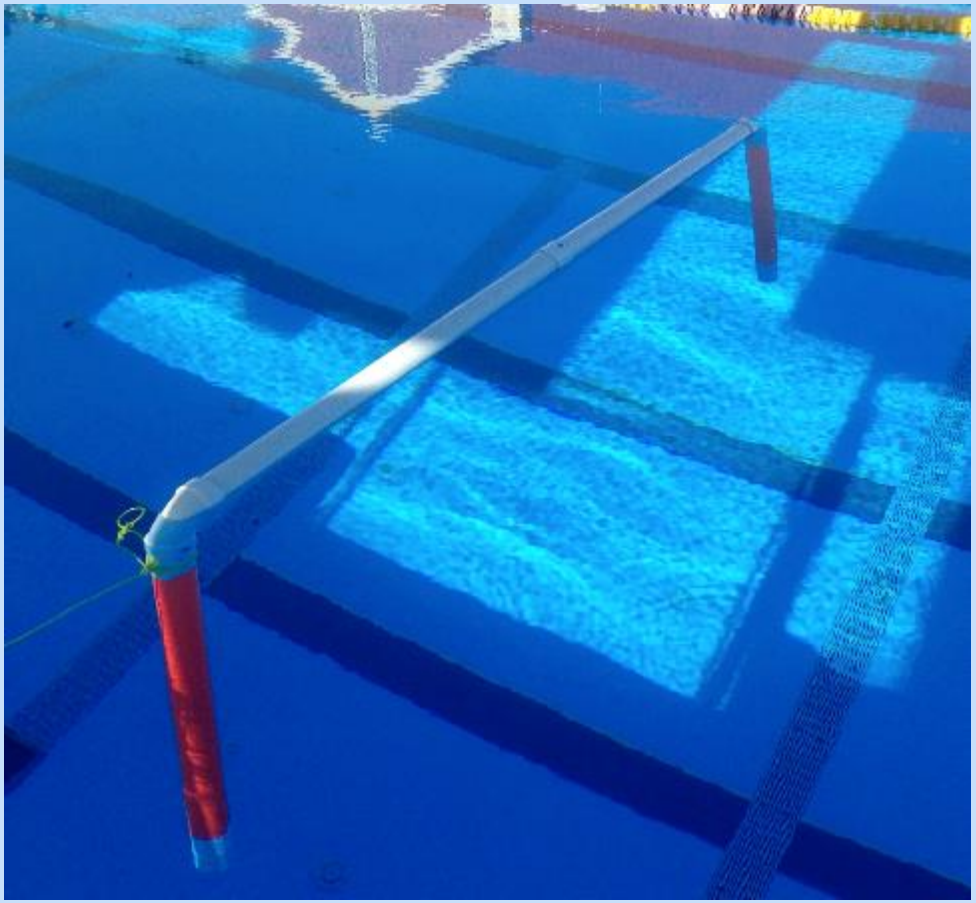
# COMPLETED TESTS

- Power Systems Test
  - Objective: Fully functional power distribution
  - Replaced universal laptop battery
- Initial Depth Sensor Test
  - Objective: Find max voltage output by depth sensor; waterproof cable
  - Results produce linear voltages corresponding to depth



# COMPLETED TESTS

- Waterproofing Test
  - Objective: Ensure sub is completely waterproof
  - No leakage
- Gate Construction Test
  - Objective: Have gate float as desired for a substantial amount of time.
  - Gate floated for 30 min; Need to get PVC glue



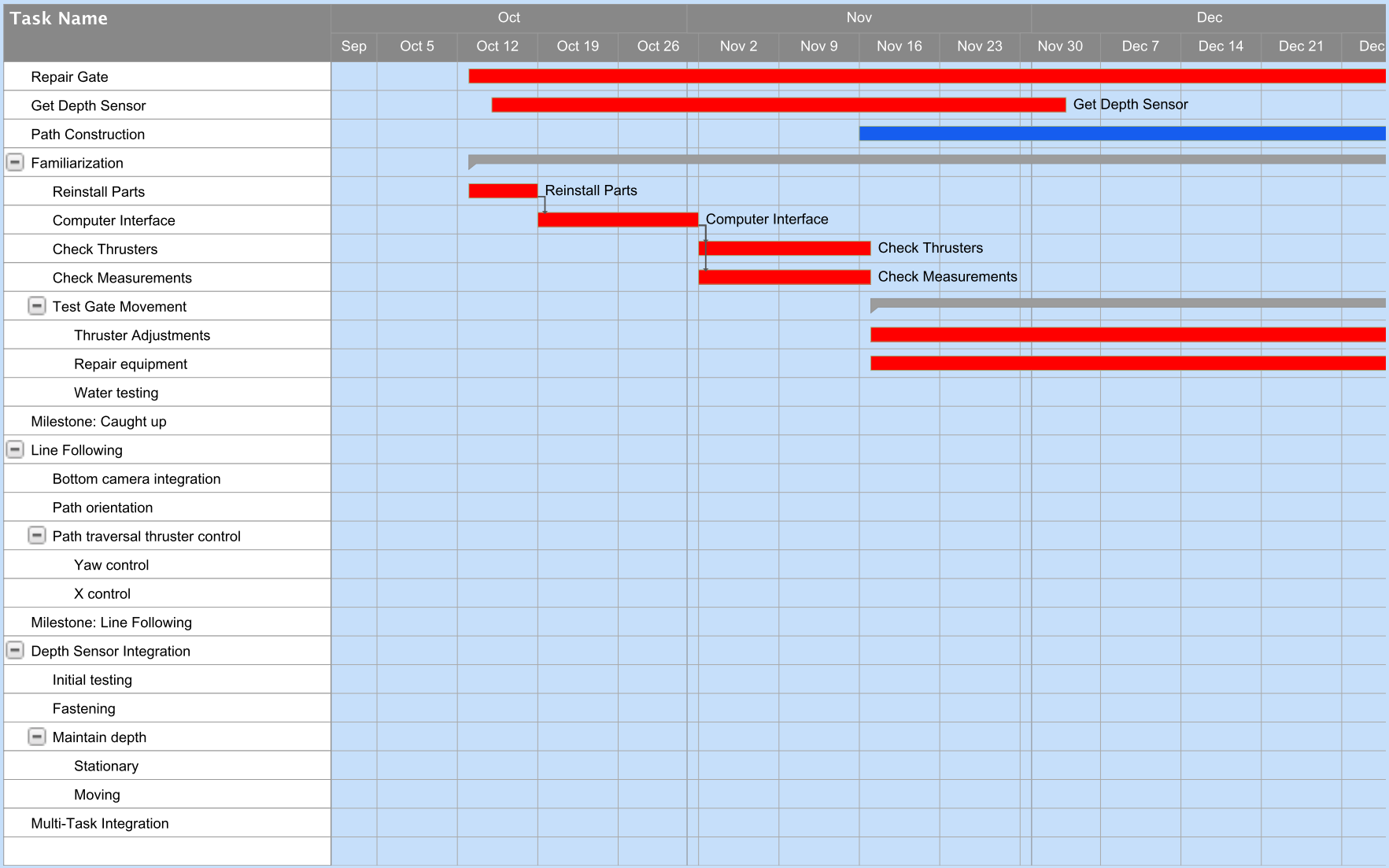
# MAINTAIN DEPTH

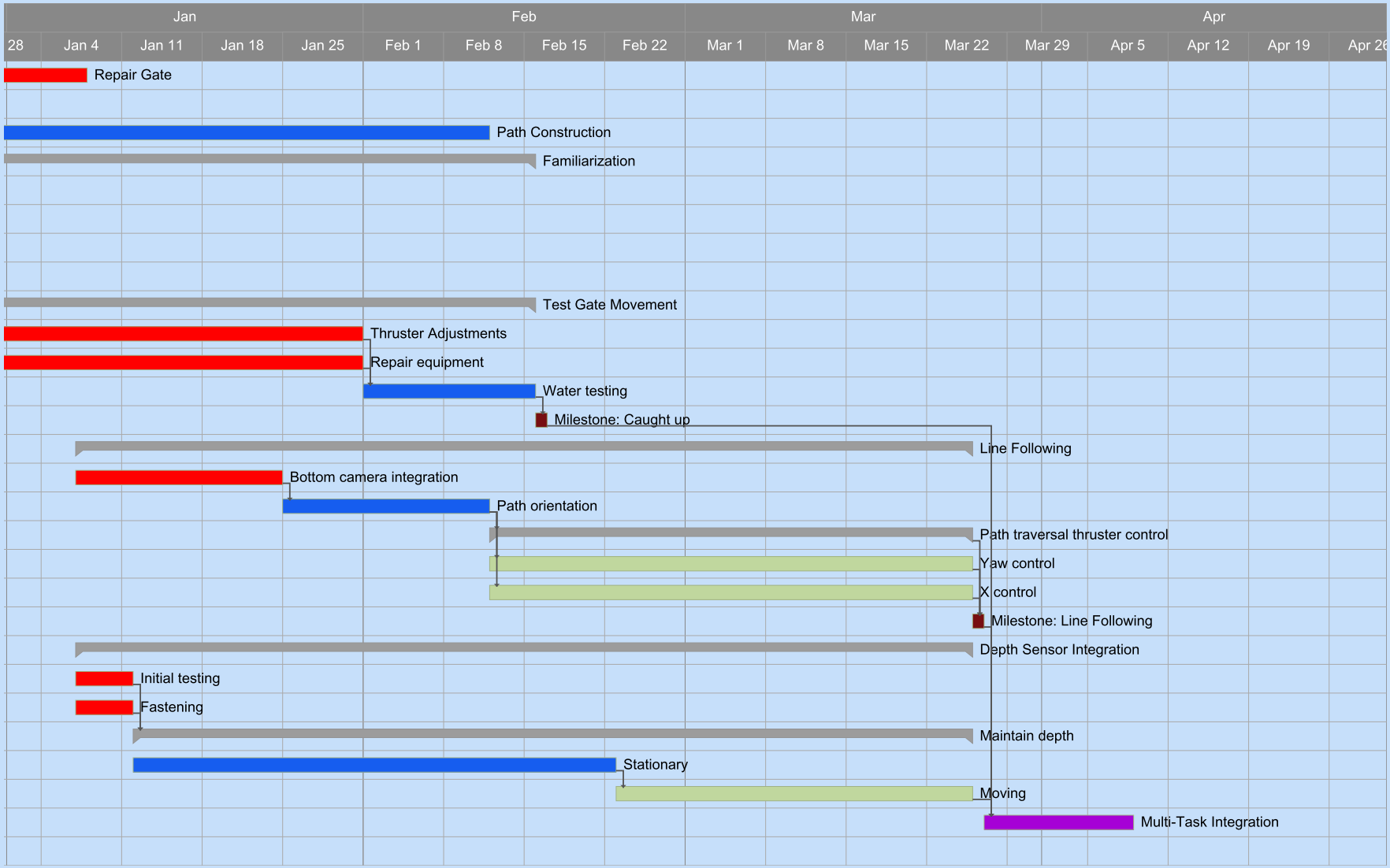
- Objective: For the sub to maintain a specific depth underwater.
- **Requirements**: Team Members (2+), Sub, Pool, External computer, 100 ft Ethernet cable
- **Process**: Submerge to an arbitrary depth and adjust thrusters until the depth can be maintained. Repeat test for various other depths.
- **Anticipated Results**: The sub will probably be in constant motion. The team will need to work with some error room above and below the depth in question.
- **Requirement for Success**: The sub needs to be able to maintain a specific depth within 6” above or below.

# LINE FOLLOWING

- **Objective**: Ensure that the sub can effectively follow lines under water.
- **Requirements**: Team Members (2+), Sub, Pool, External computer, Ethernet cable, Orange path.
- **Process**: Ensure that the sub can identify the line by using the front facing camera. Once the bottom camera detects the line, the sub will align itself parallel to the line and maintain depth. The forward thrusters engage and traverse the line.
- **Anticipated Results**: The camera may struggle to identify the line object in deep water. The sub may continue to drift when aligning due to the lag in thruster response.
- **Requirement for Success**: The sub will be approximately three feet above the line and should stay within 2 feet to the right/left of the line and follow the line until it ends, continue in that direction, and maintain alignment during the process.

# SCHEDULE


Task Name
Oct
Nov
Dec
Sep
Oct 5
Oct 12
Oct 19
Oct 26
Nov 2
Nov 9
Nov 16
Nov 23
Nov 30
Dec 7
Dec 14
Dec 21
Dec
Repair Gate
Get Depth Sensor
Get Depth Sensor
Path Construction
Familiarization
Reinstall Parts
Reinstall Parts
Computer Interface
Computer Interface
Check Thrusters
Check Thrusters
Check Measurements
Check Measurements
Test Gate Movement
Thruster Adjustments
Repair equipment
Water testing
Milestone: Caught up
Line Following
Bottom camera integration
Path orientation
Path traversal thruster control
Yaw control
X control
Milestone: Line Following
Depth Sensor Integration
Initial testing
Fastening
Maintain depth
Stationary
Moving
Multi-Task Integration


Jan
Feb
Mar
Apr
28
Jan 4
Jan 11
Jan 18
Jan 25
Feb 1
Feb 8
Feb 15
Feb 22
Mar 1
Mar 8
Mar 15
Mar 22
Mar 29
Apr 5
Apr 12
Apr 19
Repair Gate
Path Construction
Familiarization
Test Gate Movement
Thruster Adjustments
Repair equipment
Water testing
Milestone: Caught up
Line Following
Bottom camera integration
Path orientation
Path traversal thruster control
Yaw control
X control
Milestone: Line Following
Depth Sensor Integration
Initial testing
Fastening
Maintain depth
Stationary
Moving
Multi-Task Integration

# ESTIMATED BUDGET

# PARTS

| D. Expense | Purpose | Total |
|---|---|---|
| **3” Diameter x 10’ Long PVC** | Center Horizontal PVC Pipe, Vertical PVC Pipes for Gate | $29.96 |
| **90 Degree Elbows 3” PVC** | Connectors for the Gate | $5.14 |
| **2” Clean Out Tee PVC** | Connect center PVC of Gate | $1.37 |
| **Blaze Orange Duck Tape** | Vertical color of vertical PVC pipes | $6.74 |
| **Heat Gun** | Heat Shrink Tubing | $14.99 |
| **Heat Shrink Tubing Assorted Sized** | Waterproofing outside connection cables | $4.99 |
| **Liquid Electrical Tape** | Waterproofing outside connection cables | $10.99 |
| **Heat Shrink Tubing Large Size** | Waterproofing outside connection cables | $16.99 |
| **Push Button Switch** | Kill Switch | $6.49 |
| **Hex Keys 10pc** | Required to move thrusters | $5.35 |
| **Cable Tie** | Multiple uses including attaching components, and equipment | $6.96 |
| **Zotac Wall Charger** | Increase battery life | $11.63 |
| **1”x6” – 8 FT Weather Shield Wood** | Path Lines | $10.74 |
| **Hallow Braid Poly Rope (1/4” x50’)** | Needed for mooring lines | $5.60 |
| **2” Diameter by 6’ Long PVC** | Maneuvering parts, Horizontal and Vertical | $16.44 |
| **90 Degree Elbows 2” PVC** | Connectors for the Maneuvering Platform | $1.66 |
| **2” Clean Out Tee PVC** | Connect center PVC of Maneuvering Platform | $3.26 |
| **PVC Glue** | Need to seal maneuvering structure to become buoyant | $4.87 |
| **2” PVC Caps** | Seal vertical | $3.28 |
| **16/19 V Ah LI-Ion Universal External Battery** | Old Battery damaged, battery life is limited | $74.76 |
| **Depth Sensor + Shipping** | Required for sub | $371.24 |
| **Current Budget Spent** | | $577.83 |
| **Total Expected Budget** | Maneuvering Structure+Path | $623.68 |
| **Allotted Budget** | | $750.00 |

# QUESTIONS