Hydromodus

A Student-Led Multidisciplinary Project

Advanced Modular Robotics

Advisor: Thomas Katona

Alex Kost	Anu Mahinkanda	Jordan Read
602.531.3091	858.829.7122	650.799.4266
akost@calpoly.edu	amahinka@calpoly.edu	jpread@calpoly.edu

Statement of Disclaimer

Since this project is a result of a class assignment, it has been graded and accepted as fulfillment of the course requirements. Acceptance does not imply technical accuracy or reliability. Any use of information in this report is done at the risk of the user. These risks may include catastrophic failure of the device or infringement of patent or copyright laws. California Polytechnic State University at San Luis Obispo and its staff cannot be held liable for any use or misuse of the project.

Table of Contents

List of Tables

List of Figures

List of Nomenclature

A - Amp

Ah - Amp-hour

ATG - Algae Technology Group

AUV - Autonomous underwater vehicle

BRUV - Baited remote underwater video

DC - Direct current

DIY - Do-It-Yourself

ESC - Electronic speed controller

ESD - Electrostatic discharge

GPS - Global positioning system

IMU - Inertial measurement unit

lbf - Foot-pounds

LED - Light Emitting Diode

mA - Milliamp

mAh - Milliamp-hours

MESFAC - Mechanical Engineering Student Fee Allocation Committee

mV - Millivolt

OD - Outer diameter

QFD - Quality Function Deployment

TIG - Tungsten inert gas

V - Volt

Executive Summary

Hydromodus is a student-led multidisciplinary project conceived by Jordan Read designed to provide a low-cost modular hardware and software solution for researchers and scientist. For the scope of the Senior Project class, it is designed to be a baited remote underwater vehicle (BRUV), but the platform is highly modifiable and open-source.

To make this idea a reality, Alex Kost, Anu Mahinkanda, and Jordan Read researched, funded, designed, built, and tested a prototype BRUV setup for the Hydromodus. Key engineering specifications were identified at the beginning of the design and validated upon completion. Although there was not enough time to validate the Hydromodus, conclusions were reached and are discussed.

Background

Initial Inspiration

The Hydromodus project was based on a system Anu Mahinkanda and Jordan Read designed while working with the Algae Technology Group (ATG) at California Polytechnic State University, SLO in Spring 2014. ATG was experiencing issues with an antiquated water-quality monitoring system and requested assistance designing an in-house replacement. Anu and Jordan designed an electronic interface that connected to three water quality sensors. This system monitored the environment of the algae, connected to a solenoid to provide feedback control of the water's acidity/basicity (pH), stored all captured data on Exosite's cloud servers for remote observation, and was controllable via Bluetooth and PC connections. The design worked successfully, and seeing a need in underwater sensor and exploration technology, Jordan conceived the idea of an inexpensive autonomous underwater vehicle with modular sensor capabilities. The project has since been named the Hydromodus: *hydro* for its application in underwater environments, and *modus* from Latin for its focus on measurements for .

Current Solutions

Underwater vehicles have existed since scientists have wanted to explore the sea, but the concept of *inexpensive* underwater vehicles that are available to all is still new. The current market leader is OpenROV, an open-source, low-cost underwater robot for exploration and education developed by Eric Stackpole.[1] Essentially an underwater

[1] "OpenROV | Underwater Exploration Robots," OpenROV | Underwater Exploration Robots, http://www.openrov.com (23 Oct. 2014)

remote-controlled vehicle, OpenROV set the standard to beat for the Hydromodus project. The Hydromodus attempts to set itself apart from OpenROV in two key ways. First, the Hydromodus emphasises research and data collection over other do-it-yourself (DIY) applications and exploration. OpenROV caters to explorers and hobbyists almost exclusively and only provides connections to their microcontroller for which users program their own interfaces; the Hydromodus project caters more so to researchers and scientists with its comprehensive library and modular capabilities so that a multitude of sensors could be used with the Hydromodus. Second, the Hydromodus is meant to be entirely autonomous, unlike the remote-controlled and tethered OpenROV. Although more difficult to program and incorporate, autonomy addresses the desires and needs of the scientific community. Autonomy allows the Hydromodus to do much more than if remote controlled, and because the design is to be configurable and open, users can reprogram and configure the Hydromodus however they see fit.

Table 1. *Comparison of AUV prices and functionality.*

AUV Model	Price	Functionality
OpenROV	~$900	Hobbyist Remote-controlled tethered vehicle. Can record audio and video. Can add sensors, including depth and orientation sensors.
Hydromodus (BRUV)	~$2,000	Hobbyist-Researcher Autonomous and untethered; sensor modularity. Includes proximity, depth, and orientation sensors.
UC San Diego & Scripps Ranch Research Institute[2]	~$100k	Researcher Autonomous and untethered. Powerful, expansive sensor library Top-of-the-line components.

[2] Russ E Davis, "About," *Spray* (Instrument Development Group: 2015)

Most other projects are all research-based AUVs that are designed for a different audience in mind. In **Table 1** above, the standard designs seen on the market now are compared. These designs are all very expensive; meant to operate for days, weeks, or months; have technological capabilities that far exceed the average researcher or hobbyist; and are highly specific to whatever type of research is being conducted. A visual comparison can be seen in **Figure 1**.

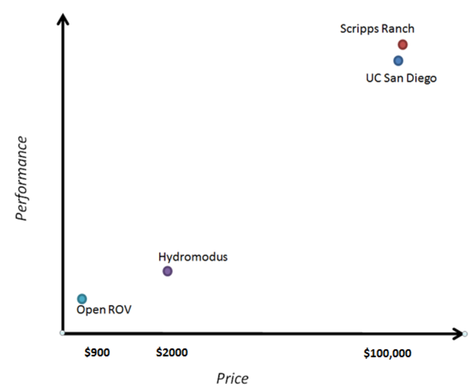


Figure 1. *Visual comparison of AUV performance against price.*

Objectives

The ocean is mostly unexplored and unknown because it is difficult and expensive to do so. Addressing the community need to further explore and research this crucial part of the world will allow people to more easily operate in marine environments and increase the ubiquity of marine operations to perform underwater jobs. The Hydromodus project was designed to provide a hardware and software platform for people to use and create automated underwater vehicles that are able to navigate underwater autonomously, collect data from connected sensors, and provide that data in a timely fashion to the user. To showcase the advantages of inexpensive AUVs and to meet the requirements of the Multidisciplinary Design Project course, the Hydromodus was configured and designed to be a BRUV AUV that could be used by researchers to survey oceanic wildlife at various depths. For this project, the Hydromodus deploys to a specified Global Positioning System (GPS) location, dives down to a specified depth, records information and footage of the area, maintains this status for a specified amount of time, rises to the surface, then repeats this procedure. This continues until all the specified locations and missions have been completed where it will then be collected by the users. To see all of the customer objectives and engineering requirements, please refer to **Appendix A** for a House of Quality diagram created with the Quality Function Deployment (QFD) technique. Using the results from this technique, a table of engineering specifications was created with their respective reasons, found in **Appendix B**.

In defining these specifications, weight was considered to be the biggest constraint. Because the final parts were selected when no funding had been awarded, majority of the components may be heavier and less efficient than costlier components. The characteristic of weight is also driven by any and all changes to the physical components of the Hydromodus, including weighting to ensure neutral buoyancy (discussed below in **Mechanical Design Development**). Thus, this specification is very dependent on other required specifications. For these reasons, meeting this requirement carries a high level of risk.

Design Development

Mechanical Design Development

Before arriving at the final design, there were many designs for each subsystem. Each mechanical subsystem required research and analysis before a final design solution could be made. Therefore, mechanically, four different subsystems were researched and developed: (1) the main chassis that protects and holds all components, (2) a waterproof, pressurized vessel that can house all electronic components safely, (3) thru hull connectors that provide a method of connecting the externally-attached components to the internal electronics housed in the vessel, and (4) a propulsion system that can drive and orient the system to the underwater locations inputted by the user.

Many ideas were discussed and analyzed when deciding the main chassis. First, the shape had to be decided. The engineering specifications dictated that the chassis have ample mounting space, not be too large in volume or weight, and be able to withstand the underwater forces. With these specifications in mind, some conceptual drawings were created to consider how these specifications would be met by various chassis shapes. These drawings, as well as a list of advantages and disadvantages, can be seen in **Figure 2.** Note that all of the ideas are symmetrical; this is so that the system is more stable when subject to hydrodynamic forces. Symmetry simplifies the design.

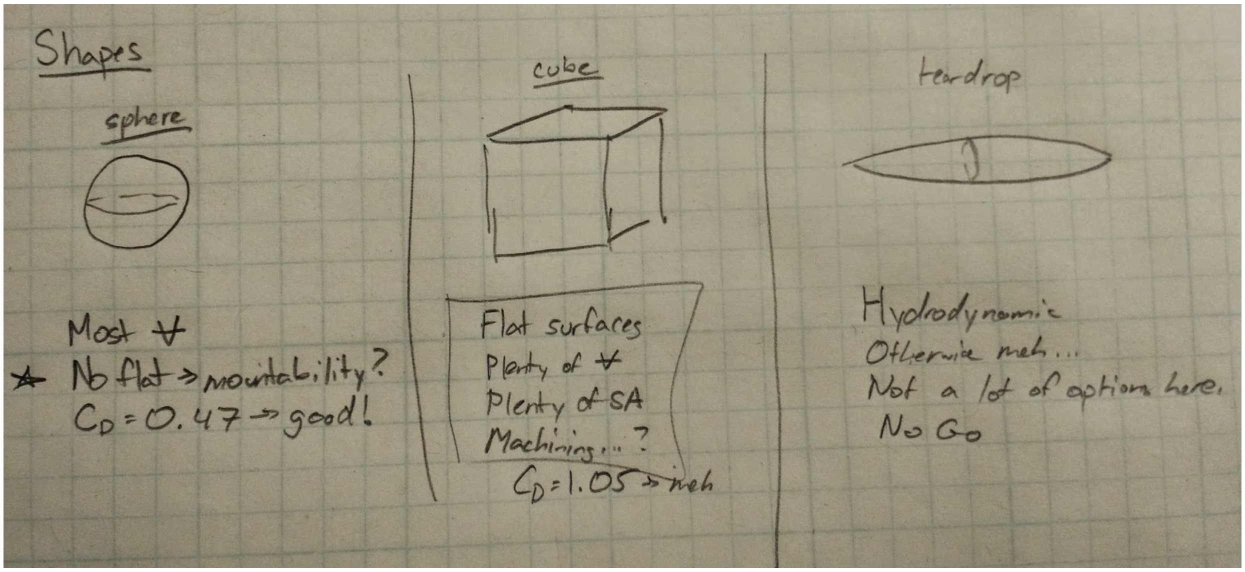


Figure 2. *Comparison of various chassis shapes.*

Some shapes were as simple as spheres or prisms, whereas others were inspired by the designs of other products in the market. This brief conceptual analysis verified the idea that a box-shaped chassis would be ideal with respects to mounting space, manufacturability, and volumetric/weight requirements. To offset the effects of drag as much as possible, it was decided that the chassis would be a box-shaped frame and not a full enclosure. This also severely reduced the weight of the design.

After selecting the shape of the chassis, the next step was to determine the material of the chassis. The chosen material had to be easy to manufacture so that mounting holes could be machined into the design; resistant against corrosion, as the Hydromodus must survive being in saltwater for extended periods of time; hold up against impact forces in case the system runs into something physical in the environment (e.g. reef, ocean floor, or animal life); and lightweight so the design is easy to carry. Aluminum felt like the natural choice for how closely its physical properties aligned with the project needs. At first, aluminum 5052 was selected as the alloy of choice for its lower cost relative to other alloys, but after weighing the importance of machinability and weldability, aluminum 6061 was chosen. The trade-off was cost, but the difference in cost was insignificant and considered negligible.

The next subsystem to determine was the pressurized, waterproof vessel. Similar to the chassis, the shape and material had to be selected before anything else. However, this time, the two were selected simultaneously. To select a shape, a few factors were decided. The vessel had to require as little machining or joining as possible to preserve the waterproof integrity that can only be achieved by having complete, whole parts. Thus, most of the vessel is made of a single hollow extrusion with endcaps compressed on both sides of the extrusion.

Simultaneously, the engineering specifications required that the vessel not interfere with any electronic components so that future configurations would not be limited due to interference from encasing the component in metal. This ruled any metals out of the material selection quickly. The next cost-effective material available was plastics. Plastics do not interfere electrical components and are available as large, inexpensive extruded cylinders. This combination of low cost and desired material properties made plastic the material of choice for the pressurized vessel.

Therefore, the vessel was designed to be an extruded plastic with plastic endcaps. First, square polycarbonate extrusion was selected to be the main vessel shape, but finding a large square extrusion (>3-inch width/height) proved to be fruitless and very expensive. This drove the change from a square vessel shape to a circular, tube vessel shape, as tubes were extruded to very large diameters.[3] An eight-inch outer-diameter polycarbonate tube was selected as the vessel because it was the first size to safely hold all of the electronics and provide ample space for other components. With the idea of “clamping” the ends with acrylic and rubber flanges, the design seemed capable of performing within specification. This was later proved with analysis (see **Final Design**).

One of the last-minute changes to the vessel design was upon discovery of polycarbonate resistance to saltwater. Research showed that after a year of use in saltwater, polycarbonate lost 9% of its mass due to corrosion.[4] This prompted the material to change from polycarbonate to acrylic. Thankfully, both plastics were similar enough that no real trade-offs occurred.

Thru hull connections are a fundamental component for the Hydromodus to operate and posed a large issue for the team. The thru hull connectors had to provide pins for every input/output required by the propulsion system and external sensors without compromising the vessel and creating pressure and water leaks. At first, third-party connectors from BIRNS and SEACON were considered as the only and best way to solve this issue. SEACON SeaMate connectors were selected after research for their lower price and ability to survive at 150 feet.[5] The connectors would be placed on one of the endcaps and sealed with silicone epoxy. However, after coordinating final parts with the electrical engineers on the project, it was calculated that 46 pins would be needed to successfully

[3] It should be noted that if the Hydromodus were ever manufactured in house, setting up a large square extrusion die would be inexpensive and easy to do. At this stage in manufacturing, having a square vessel shape could be considered. However, with the resources available now, a square plastic vessel would not work.
[4] Trishul Artham and Mukesh Doble, "Fouling and Degradation of Polycarbonate in Seawater: Field and Lab Studies," Journal of Polymers and the Environment 17.3 (2009) 170-80.
[5] SEACON, *SEACON SEA-MATE: Underwater Electrical Wet-Mate Connectors Catalog - Rev III* (2014) SM1-SM18.

interface between the propulsion system and external sensors. Although the final design ended up only requiring 31 pins, this was still too many pin connections to justify using the SEACON connector. The SEACON connectors only provided between two and 20 pins across all commercial connectors. With the team's limited budget and space, it was decided to abandon the SEACON connectors entirely and make thru hull connectors in-house using brass hex caps, stainless steel threaded rods, and PlastiDip. Each connector provides seven pins, so seven connectors were needed for the subsystem to be complete.

Lastly, the propulsion system of the Hydromodus had to be defined and selected. Many options for propulsion systems exist: for example, one system mimics how a squid moves, and other more common systems use direct-current (DC) motors with attached propellers (commonly referred to as "thrusters") to move in various directions. Other systems also incorporated the use of bladders so diving became a function of making the design negatively buoyant. Believing in the value in the thrusters' simplicity, cost-effectiveness, and proven design configurations, the team decided to use thrusters with a neutrally buoyant design. Neutral buoyancy is desirable so the system does not need an additional bladder to dive and rise while underwater--as already stated, simplicity is valued. The tradeoff is that the system must maintain neutral buoyancy through empty bottles filled with air (to increase buoyancy force if too heavy) or weighing down the chassis (to increase the weight if too light). Finding the proper weight and volume will differ for every configuration, so maintaining neutral buoyancy will provide some level of challenge for users.

The chosen thrusters had to be waterproof and survive in the corroding saltwater environment. Upon initial research, the team believed that bilge pumps, which are essentially brushed DC motors in waterproof housings, could be retrofitted as thrusters. This idea was well received because of its cost-cutting effect and the relationship between cost and thrust capabilities. However, although intended for prolonged submerged use, bilge pumps are not designed to operate at 150 feet of water, where the pressure could rupture the seal and flood the motor. Unfortunately, no specification sheets could be found to verify the quality of the housing, so the idea of bilge pumps had to be thrown out. Another idea that was discovered was from a scientific article discussing water-jet thrusters and how vectored thrusters can be modified to create low-turbulence and low noise in the underwater environment.[6] Similar to the bilge pumps, no specifications or further information could be found on the design, and the team continued to search for solutions.

Two discoveries facilitated in choosing the final thruster design. First, it was discovered that brushless DC thrusters have completely sealed electronics and work underwater. Many DIY enthusiasts have gone a step further to protect their thrusters and potted the coils inside the brushless thrusters in epoxy or resin. Although effective, it was decided not to select brushless thrusters and pot or epoxy them because for such a costly part of the design, only a few (if any) of the brushless thrusters found specified the depths they were rated to go. The second discovery was a Kickstarter campaign and website of Blue Robotics, a company creating low-cost, high-power thrusters for DIYers.[7] Their two

[6] Shuxiang Guo and Xichuan Lin, *Development of a Vectored Water-Jet-Based Spherical Underwater Vehicle*, (INTECH Open Access: 2011)
[7] "Home - Blue Robotics," *Blue Robotics*.

products, the T100 thruster and M100 DC brushless motor (the motor driving the T100), were competitive with all other design ideas. After comparison and discussion, these two products were selected for the final design for their general performance, resilience in marine environments, and inexpensive price.

Electrical Design Development

Electrically, there were several designs and options for each subsystem: power, controller, sensors, and video recording. At first, the electrical subsystem of the Hydromodus was designed with modularity and open-access as priorities; however, after narrowing the scope of the Hydromodus project to the BRUV configuration, the electrical subsystem was also refocused around providing reliable high-definition video recording capability with modular sensing as a second priority. Independent of configuration, there were many subsystems that were required for the Hydromodus to operate. These subsystems are addressed first.

The internal positioning system was decided first. Initially, it was believed that a gyroscope or accelerometer could perform all of the needed functions for positioning. However, it was found during researching that the two components would not work effectively independently. Gyroscopes use gravity as a reference to determine its current orientation. Alone this would be enough to determine its position but it would not be able to determine if an external force acted upon it in the horizontal direction, allowing a gyroscope to be effective only in determining the pitch and roll of its orientation. An accelerometer uses the accelerations of external forces to determine orientation. If it is moved suddenly, the forces acting upon it will be registered. However the ocean is full of steady currents and changes in pitch and roll may be disregarded in the software as changes induced by gravity. However yaw remains mostly unaffected leaving an accelerometer as the easy choice for yaw measurements. The sensors used in conjunction with one another would therefore produced the most accurate orientation measurements. **Figure 3** below provides a visual on what directions pitch, yaw, and roll refer to.

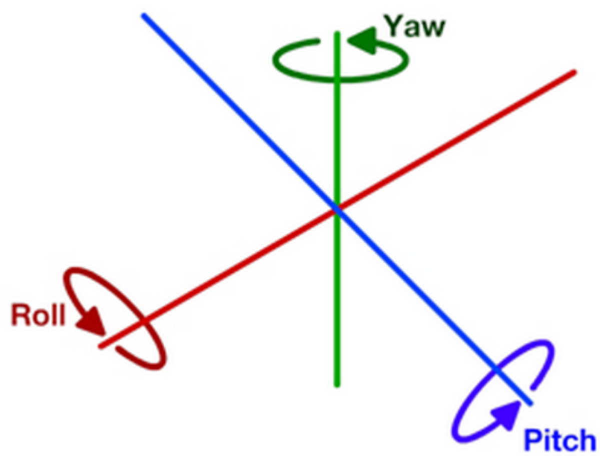


Figure 3. *Rotational axis vernacular.*

Depth measurements are critical to the Hydromodus as the whole purpose of the vehicle is centered on descending to precise depths for data collection. To get the most accurate reading, the type of measurement device to be used was restricted to three technologies: depth reader's, pressure sensors, and a hydro-acoustic navigation system. Although comparing well against the other choices in **Appendix C**, the hydro-acoustic system and depth readers were ruled out due to their high cost and depth capability. Only the pressure sensor accurately gathered depth data at a cost-effective price.

A leak detection circuit was to be employed in the AUV to prevent the electronics if the water accidentally seeps into the hull. Two circular probes mounted near the endcaps of the hull would monitor the intrusion of water inside the hull. The feedback obtained in the form of current due to a water-caused short-circuit would provide an input to the microcontroller.

Several options were considered to control the system, including an Arduino, Texas Instrument C-Series microcontroller, and a Nexys 3 FPGA. Despite the increased performance of the FPGA, the Arduino has a significantly simpler coding environment that would allow for a more robust and modular platform. From there, the question was which one; after some more research, the Arduino Mega 2560 r3 was chosen due to its increased processor power, GPIO capabilities and ubiquity in the market. An Arduino Uno was chosen for the second microprocessor in order to properly interface with the sensors; the Arduino Uno is more commonly used for DIY projects than any other microprocessor on the market, so it was chosen due to the fact that there are more references in interfacing the Arduino Uno with a peripheral than any other board. The Arduino microcontrollers were also chosen due to their high performance capabilities: they run on a 16MHz processor with a variety of integrated memories, and 2560 r3 offers more programmable GPIO pins than any other board within the price range. In addition, a second Arduino Uno microprocessor was selected in order to properly interface with the sensors chosen. a comprehensive library of software tools and sample code for all the sensors are also available to use.

To power the electronics, several types of power supplies came into consideration; lithium based, sealed lead-acid, and nickel-cadmium. Sealed lead-acid batteries were eliminated as an option due to their inherent toxicity if a leak occurs; any form of toxic by-product is a direct violation of the safety specification of the project. Between nickel-cadmium and lithium-based batteries, lithium-ion and lithium-polymer batteries were chosen over nickel-cadmium because of their smaller size and lighter weight. In addition, research showed that lithium-ion and lithium-polymer batteries have virtually no self-discharge.

Sizing the batteries was the next step. **Appendix D** shows the power analysis of each electrical component along with the needed power supply capacity. It was decided that powering different electrical components with different power supplies would maximize efficiency and provide double redundancy for supplying power for our most important components. The three batteries are separated as follows: the first battery is used exclusively for powering the microcontroller interfacing with the sensor peripherals; the second battery is used for powering the microcontroller controlling the thrusters; and the third battery is used for powering the thrusters themselves. The main reason the thrusters are separated from the other two components were due to its high power consumption and voltage needs. Two other batteries were chosen for the individual microcontrollers so as to keep the systems independent of one another. If one microcontroller was to fail, the other one would still be able to function independently.

Final Design Concept

The final design of the Hydromodus was a square-shaped aluminum frame with an acrylic central pressurized vessel and end-caps to house all of the main electronics. The configuration presented should have a camera and lighting system that should be designed to be easy to manufacture, safe to handle, and perform to desired specifications. Inside the vessel, two micro-processing boards handle the autonomous controlling of the Hydromodus. An inertial measurement unit (IMU) with an accelerometer, gyroscope, magnetometer, and depth/temperature sensors, as well as sonar sensors and three battery packs were placed outside of the vessel.

The final hardware design was revised midway through Spring 2015 due to the design's lack of compliance to the engineering specifications. It should be noted that there are two revisions to the mechanical design: Revision 1 and Revision 2.

Mechanical Functional Decomposition: Revision 1

Mechanically, the design was based off the fundamental design requirements of being safe, modular, durable, and cost-effective. The aluminum frame was to be assembled and welded together from 6061 right-angle aluminum metal (⅛-inch thick, 1-inch x 1-inch legs) due to its strength, light weight, low cost, machining qualities, weldability, and forming capabilities. The watertight vessel is an 8-inch outer diameter (OD) acrylic cylindrical tube with ⅛-inch thick walls and with a ⅛-inch thick Buna-N rubber flange and acrylic endcaps (also ⅛-inch thick). A solidworks design of Revision 1 is shown below in **Figure 4**.

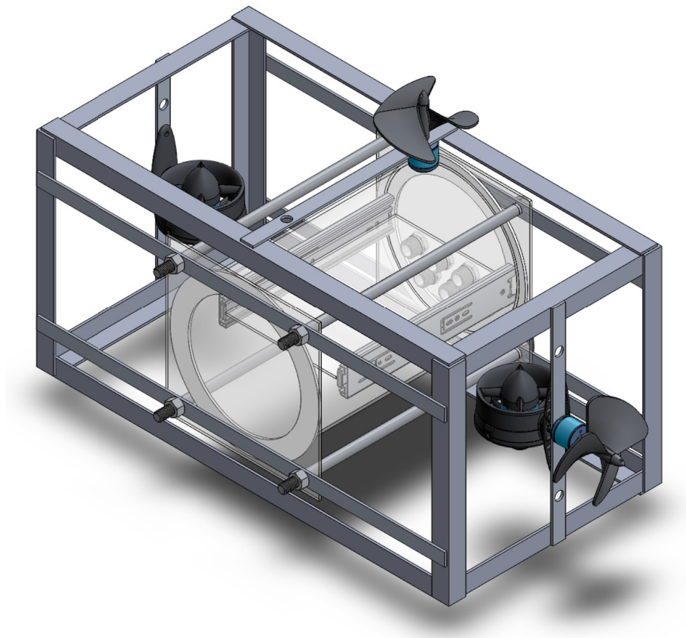


Figure 4. *Revision 1 of the mechanical subassembly for Hydromodus*

In one of the endcaps, seven homemade thru hull connectors made out of brass hex caps, threaded steel rods coated in PlastiDip silicone spray, and PlastiDip provided 49 pins to external components while maintaining the integrity of the vessel. An actual picture of a fully-manufactured thru hull connector can be seen in **Figure 5**.

Figure 5. *Fully-manufactured thru hull connector.*

Four evenly-spaced, ½-inch 14-inch-long stainless steel hex screws with washers and hex nuts provide the compressive force between the endcaps and the tube to seal the vessel. The Buna-N rubber flanges are placed between the acrylic tube and acrylic endcap. For electronic mounting space and ease of use, an acrylic shelf mounted on drawer sliding racks was also been designed and included inside the vessel. An exploded view of the chassis can be seen below in **Figure 6**.

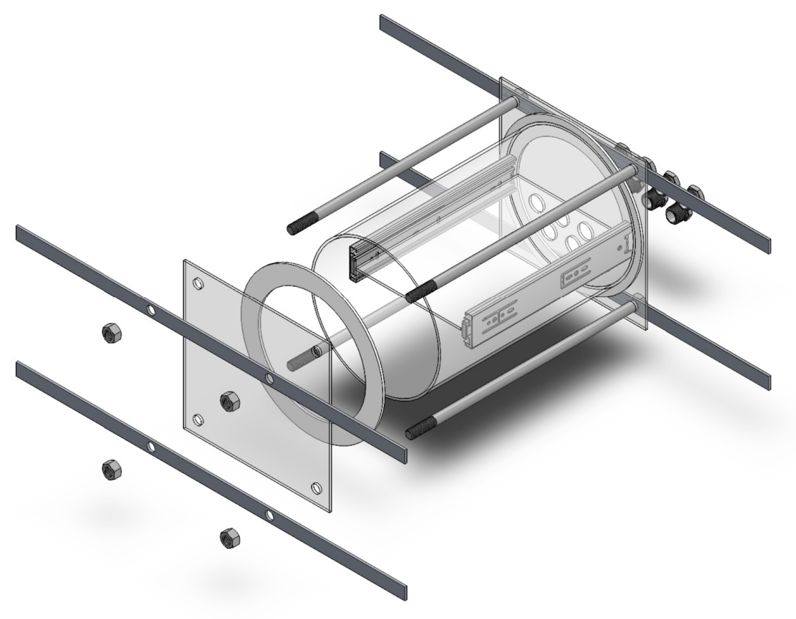


Figure 6. *Exploded view of vessel assembly. The backside is not meant to dismantle.*

The final design used two Blue Robotics T100 thrusters and three Blue Robotics M100 brushless DC thrusters to orient and drive the system. All five thrusters are controlled with Afro 30A Electronic Speed Controllers (ESCs) with custom firmware provided by Blue Robotics, seen in **Figure 7.**

Figure 7. *Afro 30A Electronic Speed Controllers.*

The final placement of the thrusters provides four degrees of freedom and ample thrusting power for the Hydromodus to dive, move, and orient itself. The Blue Robotics T100 thrusters are waterproof and corrosion-resistant thrusters with a specially designed two-blade propeller and injection-molded shroud.[8] These thrusters are powerful; each provides 5.2 lbf of forward thrust and 4 lbf of reverse thrust. Unfortunately, they are also expensive. To offset the costs of these thrusters, the other three thrusters are Blue Robotics M100 thrusters--the same thrusters driving the T100 thrusters. However, while the T100 thrusters are $109 each, the thrusters themselves are $65. By selecting other propellers and not using shrouds for these thrusters, the system was made significantly less expensive. The trade-off here was power; these thrusters would not provide the impressive thrust forces created by the T100 thrusters. Therefore, these three thrusters on the system were designed to be attitude thrusters to help orient and rotate the system rather than drive the system. If the user rotates the system with the attitude thrusters, the system

[8] "Home - Blue Robotics."

could orient itself so the T100 thrusters can propel the system in whatever direction is required.

To mount the T100s and M100s, provided brackets and additional brackets from Hobby King were chosen. For the M100 attitude thrusters, 3-blade 40x52mm boat propellers from Hobby King would be attached with 4mm shaft propeller adapters. These adapters would have allowed the propeller to be firmly attached to the motor shaft and would have made slippage a non-issue. With the combination of the T100s and M100s, the Hydromodus should have been able to go wherever the user wishes.

Mechanical Functional Decomposition: Revision 2

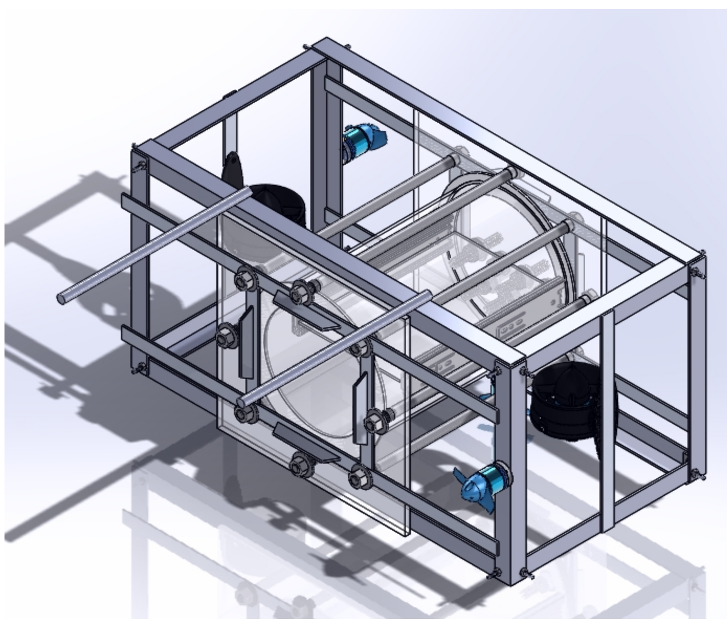


Figure 8. *Revision 2 of the mechanical subassembly for Hydromodus.*

After testing Revision 1 and determining that the design did not meet the engineering specifications, a few major modifications were done to the watertight vessel and thru hull connectors. An acrylic mount for the three M100 thrusters was also designed and fabricated because no store-bought mount could be found that satisfied the needs of the design. Lastly, it was decided to assemble the aluminum chassis with screws and nuts instead of welding it together. The rest of the mechanical subassembly remained unchanged. The updated mechanical subassembly can be seen in **Figure 8**.

Originally, it was decided to weld the chassis together so that each joint would be over designed and safe. However, this idea was dismissed after it was realized welding would warp the shape and make it difficult to fix or replace any components. Instead of welding the joints together, it was decided to buy small machine screws with locknuts and using two screw per joint. This solution was far easier to assemble and provided much more repairability in the system.

The watertight vessel saw the most of the modifications. Because four bolts did not evenly distribute the pressure across the endcap, four additional bolts were included in the new design for a total of eight stainless steel bolts. These bolts were placed through two newly-designed endcaps. The endcaps, now 5/16-inch thick (a 60% increase in thickness from the original ⅛-inch thickness), were designed with eight loose-fitting holes for these bolts, four of which are in a circular pattern. The other four bolts are still in the same pattern as before so that the original aluminum bars can still be used to attach the vessel to the chassis. However, the aluminum bars are now part of an entirely new assembly to safely hold the vessel mount and prevent any deflection of the endcaps. This new assembly can be seen below as a SolidWorks rendering in **Figure 9**.

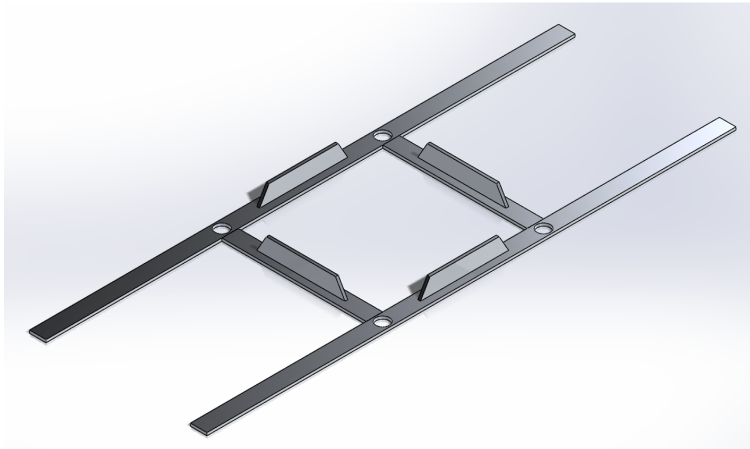


Figure 9. *SolidWorks rendering of the vessel mount.*

Dubbed as a vessel mount, extra aluminum bars were used as struts between the 2' aluminum bars to create a square-shaped mount. Once welded together, additional aluminum was welded to create a T-shape across the distances between holes. The T-shaped structure is much stiffer than a standard aluminum bar and is much more resistant to deflection. It was believed that this vessel mount could reinforce the endcap to fight deflection and also to protect the vessel from external forces, such as an impact from a marine animal.

The M100 thrusters underwent two minor changes: first, the blades were replaced with 3D-printed propellers sold by Blue Robotics for their M100 thrusters, and acrylic motor mounts were made to attach the brushless thrusters to the aluminum chassis. The first decision was made based on the increasing complexity of accurate propeller analysis. As analysis on the thrusters began, it became quickly evident how complicated and difficult propeller selection is. From various videos online, the size and blade configurations did not create any notable change in the thrust capabilities of the M100 brushless thrusters.[9]

[9] "ROV Thruster Testing," (*YouTube*: YouTube, 2011).

However, when Blue Robotics began to sell their personally-designed propeller for the M100 thrusters, it made sense to use their propellers as they were designed specifically for the M100. This had the additional benefit of reducing the number of components required for the motor assemblies due to its replacing of the propeller, shaft adapter, and propeller shroud. An acrylic mount was also incorporated into the M100 thruster assembly to provide easy mountability of the thrusters. All online solutions were expensive and did not mount easily with the aluminum chassis, so making a motor mount in-house specifically for the Hydromodus seemed best. A comparison between the old and new M100 thruster assembly, including the acrylic motor mount, can be seen below in **Figure 10**.

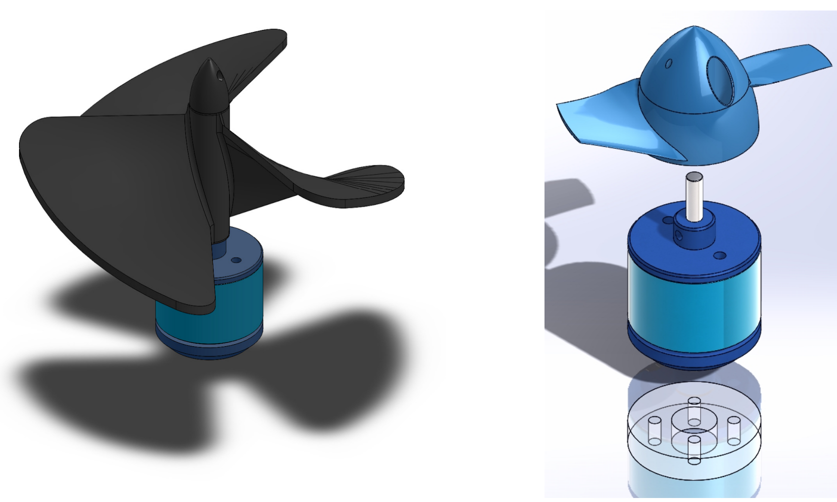


Figure 10. *Revision 1 (left) and Revision 2 (right) of the the M100 thruster assembly. Screws not shown.*

Finally, the thru hull connectors were changed. The brass hex caps were filled with marine epoxy instead of PlastiDip, the threaded rods were replaced with standard steel rods, and additional flat rubber washers were used to create an easily-replaceable, watertight connector. Because the flat rubber washers did not seal the vessel completely, additional marine epoxy and plumber's tape was used between the thru hull connectors and the acrylic endcap. With all of the changes made, minor changes in dimensioning were also made to accommodate the additional bolts and rubber washers.

All parts as well as specification sheets and layout drawings of Revision 2 for the mechanical subassembly of Hydromodus can be found in **Appendices E** and **F** respectively.

Mechanical Analysis: Methodology

All components of all mechanical subsystems were analyzed for potentially fatal issues and to determine if the mechanical subassembly meets the engineering requirements. The following analyses were conducted:

- Stress analysis on the acrylic watertight vessel to ensure the pressure of water does not subject the acrylic tube to a failure scenario.
- Bending analysis on vessel endcaps to ensure seal is maintained and water does not penetrate the vessel
- Thrust analysis to determine thrust capabilities and required force to propel the Hydromodus two knots after 10 seconds (as defined by engineering specifications).

The thrusters ability to operate underwater at 150 feet and the thru hull connector's performance were not tested because they were already proven by other parties. Blue Robotics tested their M100 and T100 thrusters for endurance, sand and debris, and depth rating, and found the following conclusions:

- Endurance: 300+ hours at max power with no noticeable wear.
- Sand and Debris: Ran in heavy sand with continuous exposure for several hours. No damage or wear. See video: https://www.youtube.com/watch?v=0X0EncNR8l8.

- Depth: Static depth tests (two cycles for 10 minutes each) to 3,000m with no damage.

The thru hull connectors were already tested by Doug Jackson and found to operate at over 1,000 psi, well above the required 66.75 psi. The video of the test is available here: https://www.youtube.com/watch?v=hy6ZCOmZSEo.

Mechanical Analysis: Conclusions

To design the system to be neutrally buoyant, the Hydromodus' mass must be equal to the mass of water offset by it's placement in water. To check if the system is buoyant, a Matlab code calculating the volume of the cylindrical vessel and the volume of the remainder of the system was used. From these calculations, it was found that the system would need an additional 18 lbf of buoyancy force before being neutrally buoyant. This could be added with air-filled containers symmetrically attached to the chassis.

From the endcap analysis, theoretical and conservative calculations verified that the endcaps do not bend outwards from the compressive force created by the screws and the forces from the acrylic tube. At lower depths, the water exerts a stronger force on the endcap and helps maintain the seal; likewise, the air inside the vessel contracts and also helps maintain the seal. Therefore, if the vessel was watertight upon being submerged, the vessel would remain watertight down to 150 feet. As shown in the analysis, the maximum deflection calculated was -.5E-5 inches, which is incredibly conservative and shows that not only does the endcap deflect such a small degree, but it deflects inwards towards the vessel and maintains the seal as depth increases.

The vessel analysis verified that the pressurized cylinder can survive the stresses presented by pressures 150 feet deep underwater. The vessel actually has a factor of safety of 4.8, showing how resilient the acrylic is in the underwater environment. This also verifies that the chassis--made out of the stronger material aluminum--would survive the marine pressures present at 150 feet.

Lastly, the thrust analysis provided shows that Hydromodus is capable of moving underwater at two knots. Unfortunately, something is incorrect in the model, as the drag force is 50x larger than the required force to propel the Hydromodus two knots underwater. This drag force inhibits the Hydromodus greatly and makes one believe that there is no way to drive the Hydromodus. However, the results do not correspond with common sense. The OpenROV is 5.7 lb. and swiftly moves with two weaker thrusters.[10] The completed Hydromodus was weighed on June 1, 2015 and found to weigh 34.3 lbs. Although it is much heavier than the OpenROV, the Hydromodus will be neutrally buoyant and will not expected to accelerate as swiftly as OpenROV. Therefore, mechanical intuition dictates that the Blue Robotics thrusters should successfully propel the vehicle. Thus, this team believes that the thruster analysis is incorrect and needs revision. Nonetheless, the code is included.

To see all mechanical analysis code written in Matlab, refer to **Appendix G**.

[10] "OpenROV | Underwater Exploration Robots."

Electrical Functional Decomposition

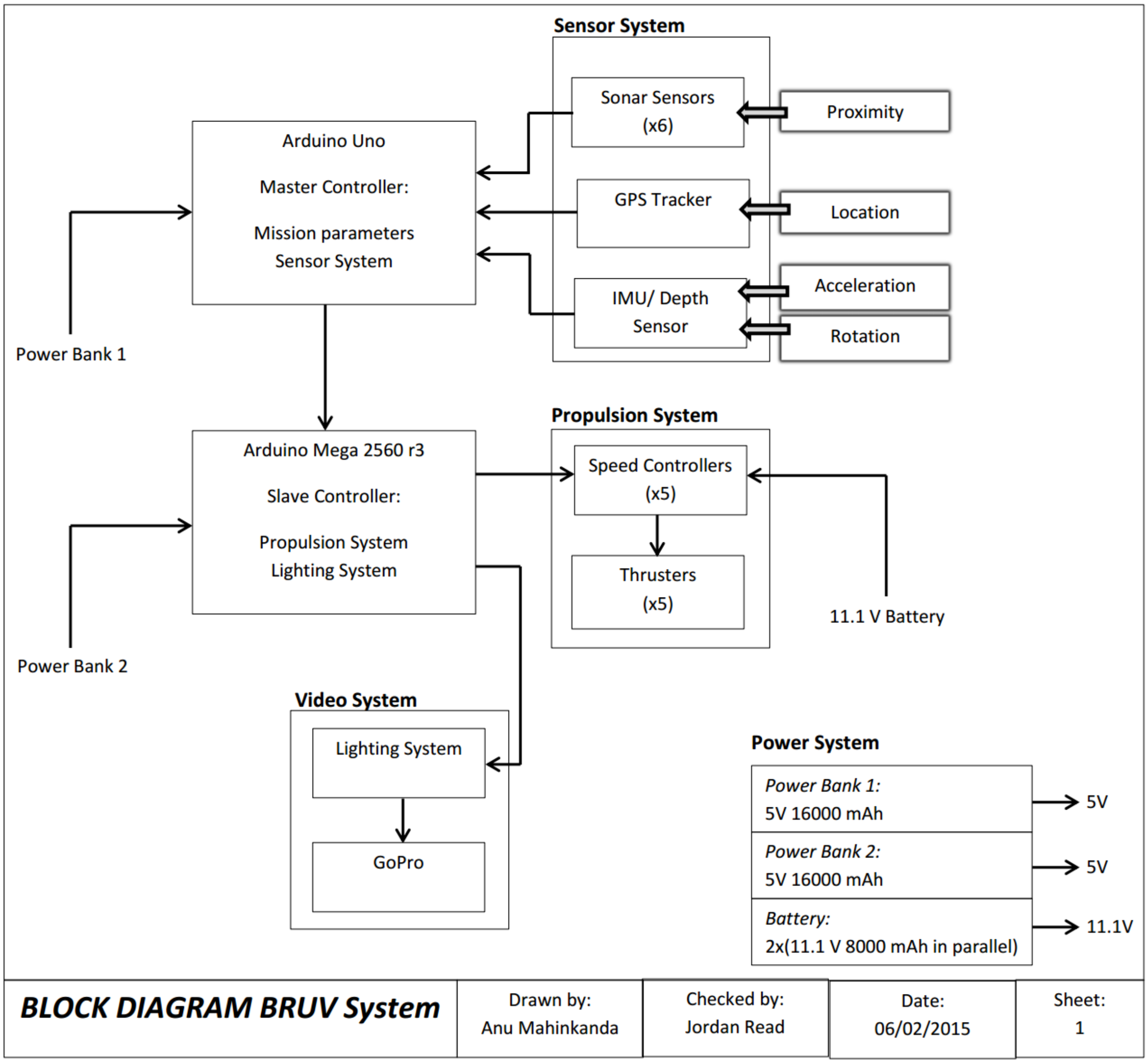


Figure 11. *System-level block diagram.*

The Hydromodus is controlled by two microcontrollers, an Arduino Mega 2560 r3 and an Arduino Uno. The Arduino Uno serves as the master controller, receiving mission parameters from the user and communicating with the slave controller, the Arduino Mega 2560 r3. The master controller also controls and analyzes data from the sonar, IMU/depth, and GPS peripherals. The slave receives three varying signals per thruster from the master controller, where each signal represents a different action it needs to perform with the individual thrusters it controls. With this information, the slave controls the attitude thrusters to get to the desired GPS location while on the surface of the water, and then descends to the specified depth. Once the desired depth has been reached, the master controller checks to make sure that the orientation of the vessel is correct. The slave controller is intended to be used by the end-user to implement their projects. In our BRUV case study, the slave controller controls the lighting equipment used on the Hydromodus. The system-level block diagram for the BRUV system can be seen in **Figure 11**. This lighting system was designed to measure the light intensity of the surrounding area and power on the lights if it is below a certain threshold. The GoPro camera remains powered on and recording throughout the entire mission. The underwater lighting has a minimum requirement of 200 lumens based on the diffusion analysis in **Appendix H**, however the actual light used produces 600 lumens. Next, the master controller begins the pre-programmed timer as part of the mission parameters. Once the timer has counted all the way down, the master instructs the slave controller to ascend to the surface. Upon receiving this signal, the slave controls the thrusters to rise to the surface. Upon completion of its ascension the Hydromodus will repeat this procedure in conjunction with the user-defined mission parameters.

In regards to connections, the Hydromodus will receive its mission parameters through a COM-port USB connection. Please refer to the user manual in **Appendix I** for specifics on how this process is done. The user will pre-program the microcontrollers directly before any missions. The two microcontrollers will communicate to each other using an I2C interface. For the user peripherals, the GoPro will continuously be running throughout an individual mission. The lighting's power will be controlled via digitally controlled switches. The master controller connects to a Sparkfun GPS Receiver and shield module with an I2C interface, and as it does not need to be exposed externally it will be housed within the pressurized vessel, with a five wire interface. The six proximity sonar sensors need to be housed externally so their cables will need to penetrate the hull and will use GPIO to communicate with the microcontroller. A single IMU/Depth sensor will also use an I2C interface and will also penetrate the vessel. Finally, the five thrusters will be controlled by ESCs using pulse width modulation signals from microcontroller. The motor will be connected to an ESC which has an input directly connected to the power supply, as well as three additional pins which will connect to our controller. There are two connections per sonar module and six modules for a total of 12 pins. Also, there are four connections on the OpenROV IMU/Depth sensor module, and three connections per motor for a total of 19 connections there, leaving us a grand total of 31 penetrating connections. External pin count requirements can be found in **Table 2**. Sample code for the master and slave microcontrollers, as well as flowcharts for the controllers and peripherals can be found in **Appendices J, K, L,** and **M**.

Table 2. *External pin count requirement.*

External Device	Necessary Pin Count Per Device	Number of Devices	Total Pin Requirement
T100 Thruster	3	2	6
M100 Thruster	3	3	9
Sonar Sensor	2	6	12
IMU/Depth Sensor	4	1	4
Total Pin Requirement			31

Each of the devices will be powered by three separate batteries. One 5-volt 16-amp-hour lithium-ion battery will be used to power the master controller (Arduino Uno) and the associated sensors it is attached to the IMU sensor, depth sensor, and sonar sensor. Another identical battery will be used to power the second slave microcontroller (Arduino Mega2560 r3). Two additional 11.1-volt 8-amp-hour batteries will be used in parallel to provide an 11.1-volt 16-amp-hour power supply to drive the thrusters. This power supply will then be connected directly to the electronic speed controller. The power analysis can be seen in **Appendix D**.

The Arduino Uno and Arduino Mega 2560 r3 microprocessors were used as a master and slave controllers respectively. The Arduino Uno collected depth and directional data, analyzed and controlled positional actuaries, and controlled the sensors. The Arduino Mega 2560 r3 only controlled the thrusters. To accurately understand the purpose and functionality of the sensors, please reference **Appendix L** for a general process diagram, and **Appendix M** for a high-level software flow. Electrical specification sheets can be found in **Appendix N**.

Product Realization

Team Members & Expertise

The project is divided in two disciplines. Alex Kost acted as the principal mechanical engineer and was responsible for the development of the chassis, pressurized vessel, propulsion system, and thru-hull connectors. He utilized his CAD experience and knowledge of material properties, fluid mechanics, stress/strain analysis, manufacturing knowledge, and general mechanical skills to complete these objectives. Alex was also responsible for overseeing all document submissions, external file organization, funding allocation, reimbursement planning, communications with all external persons, and design and material for the poster for the Senior Design Exposition on May 29, 2015. Jordan Read and Anu Mahinkanda, the principal electrical engineers, developed the code for all of the following electrical systems: sensor systems, recording system, and power systems. Jordan and Anu also used their skills to design the necessary circuits.

Scheduling

To mitigate any delays and provide time for building and testing the Hydromodus, an incomplete Gantt chart was made so the team could visually see the project scheduling. Seen in **Figure 12**, the chart did not provide enough accurate dates and deadlines and was too vague to be truly useful.

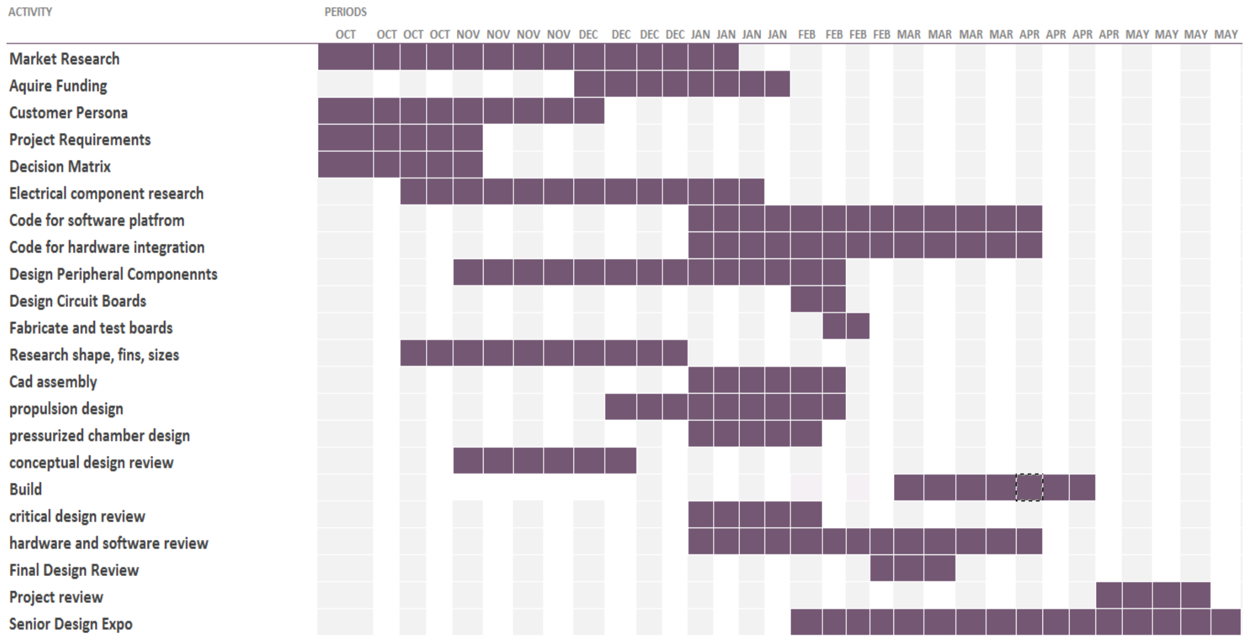


Figure 12. *Gantt Chart, dated March 13, 2015.*

After completing the project and reviewing the last three quarters, an updated Gantt chart with key milestones and stages of development was made. Found in **Appendix O**, the updated Gantt chart shows that building and testing were pushed far too late into Spring 2015 and should have been completed earlier. More conclusions regarding scheduling and timing can be found under **Risk Analysis** in **Conclusions and Recommendations**.

Budget and Funding

The Hydromodus project budget was split almost evenly between mechanical and electrical components. From the Bill of Materials seen in **Appendix P**, the Hydromodus project cost $1,120.58 and $1,217.47 between the mechanical and electrical subassemblies. In total, the project cost $2,338.05. Unfortunately, this means the Hydromodus did not meet the engineering specification of costing $850 for the base platform and $1,150 for the additional peripherals. As testing continued and additional parts were purchased, the

Hydromodus project could not compete against the OpenROV with respect to cost, and the engineering specification was largely abandoned.

The Hydromodus project was fully funding with the help of CPConnect, the Mechanical Engineering Student Fee Allocation Committee (MESFAC), and Cal Poly's Electrical Engineering Department providing $2,000, $300, and $400 respectively.

Hardware Manufacturing

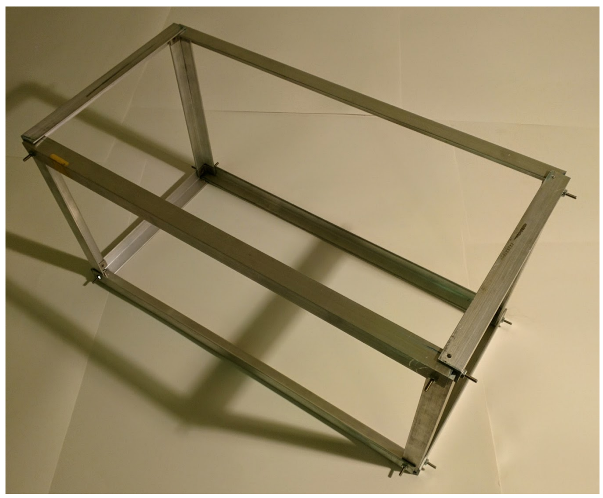


Figure 13. *Assembled aluminum chassis.*

Seen assembled and completed above in **Figure 13**, the chassis was built over two weeks in the Bonderson and Mustang '60 Machine Shop. As stated previously, the welded assembly was replaced with a screwable assembly by using two #8-32 screws per corner with locknuts. After ordering screws that were too large and could not fit in the design, smaller screws were bought and used for the chassis. Holes were drilled and deburred using a drill press, a 5/32-inch fractional drill bit, and a deburring tool. The completed chassis, although not perfectly box-shaped, satisfies the engineering requirements with a high factor of safety.

The endcaps, vessel shelf, and motor mounts were cut out of acrylic sheet (5/16-inch, ⅛-inch, and ¼-inch respectively) with the laser cutter in Mustang '60. The endcaps were cut with the Laser Cutter Bonderson Machine by converting SolidWorks drawing files to Adobe Illustrator files. All laser-cut parts are within .001-inch tolerance. Pictures of the laser cutter cutting acrylic for the endcaps and motor mounts can be seen below in **Figure 14**.

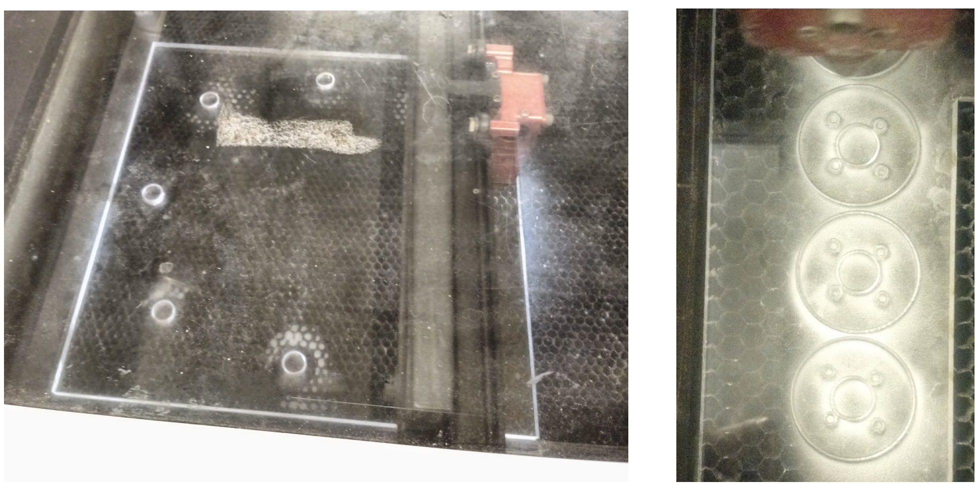


Figure 14. *Fabrication of endcaps and motor mounts using the laser cutter.*

The holes for the thru hull connectors were slightly too small on the endcap, so a ¾-inch NPT tap was used with a tap holder to tap the holes and make them snugly fit with the thru hull connectors. **Figure 15** shows the tapping procedure taking place.

Figure 15. *Thru hull connector holes being tapped.*

The thru hull connectors were tricky to construct, even though supporting documentation already existed. On the first batch of brass hex caps, a ⅝-inch end-mill bit was used with a digital mill to create a ⅛-inch-deep recess for epoxy on the top of the caps. Next, seven pin holes were drilled completely through the plug with a 3/32-inch fractional drill bit. However, at this point of fabrication, it was discovered that the wall between the connectors was too thin and gave out as the pin holes were drilled. These were thrown out, and the next revision of thru hull connectors had a 1/10-inch recess and pin holes drilled first. The beginning of the second fabrication attempt can be seen in **Figure 16**. This time around, the holes and recesses were successfully made.

Figure 16. *Second attempt at fabricating thru hull connectors.*

The next step in fabricating the thru hull connectors was to coat the threaded rods in silicone and cut them into size. Using PlastiDip silicone spray, the threaded rods were repeatedly coated, but the threadings made 100% coverage difficult. It was surmised that the space between the threads was so small and difficult for the paint to get into. A second attempt simply used two-foot-long steel rods purchased locally. These rods were cut down to about three-inch-long pieces and used for the final thru hull connectors.

To install the rods in the brass hex caps, the caps were filled with marine epoxy and with the rods were pushed through the holes. In doing so, epoxy coated the rods and was pushed through the small pin holes. Once all pins were successfully installed, additional epoxy was added to the recessed part of the caps. Epoxy was also applied along where the endcap and brass hex caps met, but this step occurred after the sandblasting, seen in **Figure 17**, shows the rods inserted in the brass hex caps and the epoxy drying.

Figure 17. *Epoxy drying on the thru hull connectors.*

To remove the silicone epoxy from the ends of the rods so that an electrical connection could be made, a sandblaster was used to remove all excess epoxy. **Figure 18** shows this.

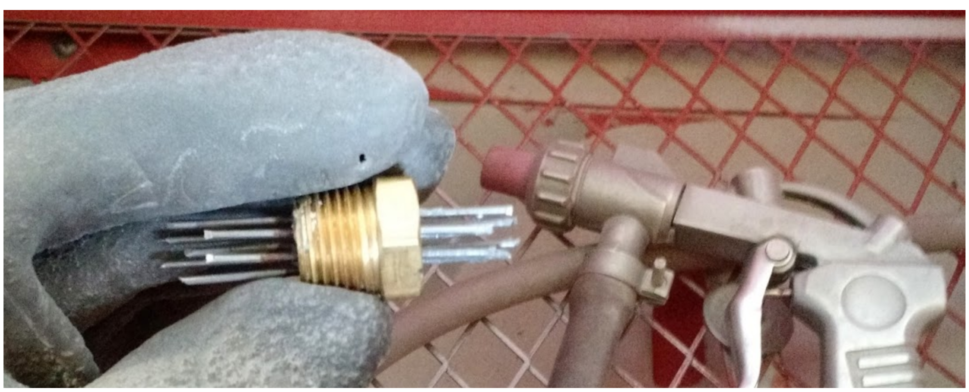


Figure 18. *A thru hull connector after being sandblasted.*

Lastly, the rods were grinded down on a grinding wheel to create a uniform rod length across all connections. For the final iteration of the thru hull connectors, plumber's tape was wrapped around the hex cap threads and rubber union washers were placed between the hex cap of the thru hull connector and the endcap before applying epoxy in the hopes that the compression between the thru hull connectors and endcaps could crush the rubber washers and prevent water from leaking. Unfortunately, this design did not work, and the entire assembly was covered in marine epoxy so no water would leak.

Overall, the thru hull connectors were difficult and frustrating to manufacture. The silicone epoxy was messy and difficult; the rods would not remain straight and parallel to one another upon being inserted into the hex cap; and sealing the connection between the endcaps and the thru hull connectors was frustrating and impossible to determine if a seal was successfully made. Because of numerous mistakes and missteps, more brass hex caps and threaded rods needed to be purchased to replace poorly fabricated prototypes. The design of these connectors, however, made using the seventh middle pin impractical. Furthermore, the inherent difficulty in making electrical thru-hole connectors waterproof became apparent when multiple iterations of the connectors failed.

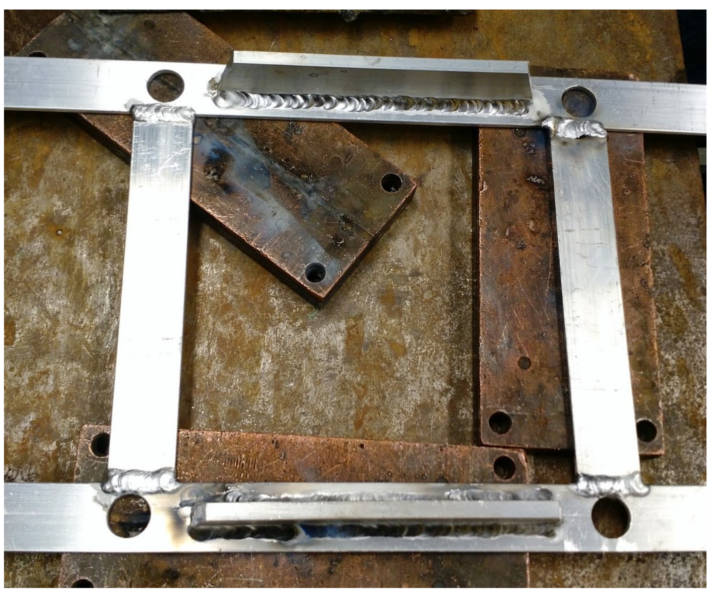


Figure 19. *Fully fabricated vessel mount.*

As described earlier, Revision 2 came with a modified vessel mount used to attach the vessel to the chassis. Seen previously in **Figure 19**, the vessel mount was fabricated with the guidance of Professor Kevin Williams. Using a wet saw and a tungsten inert gas (TIG) welder, Professor Williams successfully cut aluminum bar with 30° ends to create a T-shape across parts of the mount. The final mounts were sanded with a belt sander to ensure the flatness of the mounts were within tolerance.

The last step to create the watertight vessel was to install the shelf. Because the shelf did not need to be reinforced or tightly toleranced, they were installed one at a time with gorilla glue. Shown below in **Figure 20** is a shelf seen from the outside of the vessel. The yellow, bubbly material is the dried gorilla glue.

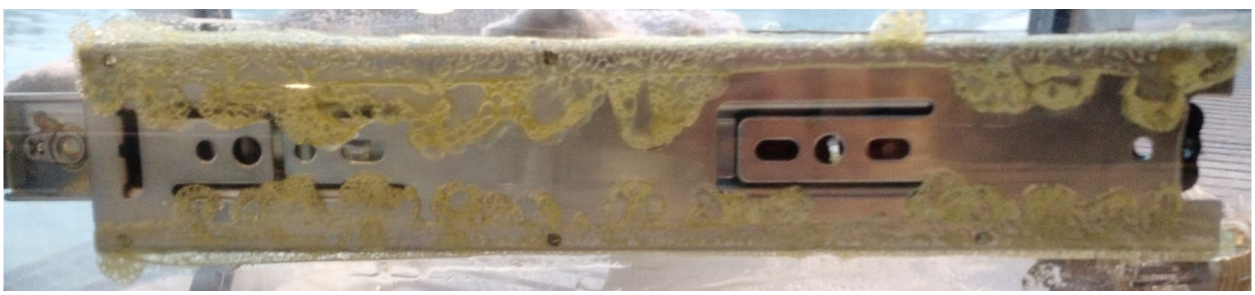


Figure 20. *Installed vessel shelf.*

With all components finally ready, the vessel could be assembled. By sliding the bolts into the endcap holes, then stacking the first vessel mount, front endcap, Buna-N flange gasket, vessel tube, second Buna-N flange, second endcap (with thru hull connectors installed), and second vessel mount (in this order), washers and hex nuts can be tightened onto the bolts and create the compressive force required to seal the vessel. A picture of the fully assembled vessel for Revision 1 can be seen in **Figure 21**. The step-by-step assembly is the same for both Revision 1 and Revision 2.

Figure 21. *Revision 1 of the watertight vessel, fully assembled.*

From here, holes were screwed with a M5 drill bit and #8-32 machine screws with hex locknuts were used to attach the vessel assembly to the chassis. The same-size drill bit and machine screws were used to attach the T100 thrusters to one-foot aluminum bars and these aluminum bars to the chassis. The M100 thrusters were attached with an M3 drill bit and M3 x 0.5 screws.

Electrically, all thru hull connectors were soldered to wires, then covered in liquid electrical tape and marine epoxy as sealant. All spliced wires were soldered together, then covered in either heat-shrink tubing or liquid electrical tape. External peripherals were never attached to thru hull connectors due to time constraints. All internal components were fixed to the vessel shelf with velcro for easy removal.

The final Hydromodus with its poster can be seen in **Figure 22** being presented at the Senior Design Exposition. Although difficult to see, all pieces of hardware are inside the watertight vessel or attached to the chassis itself.

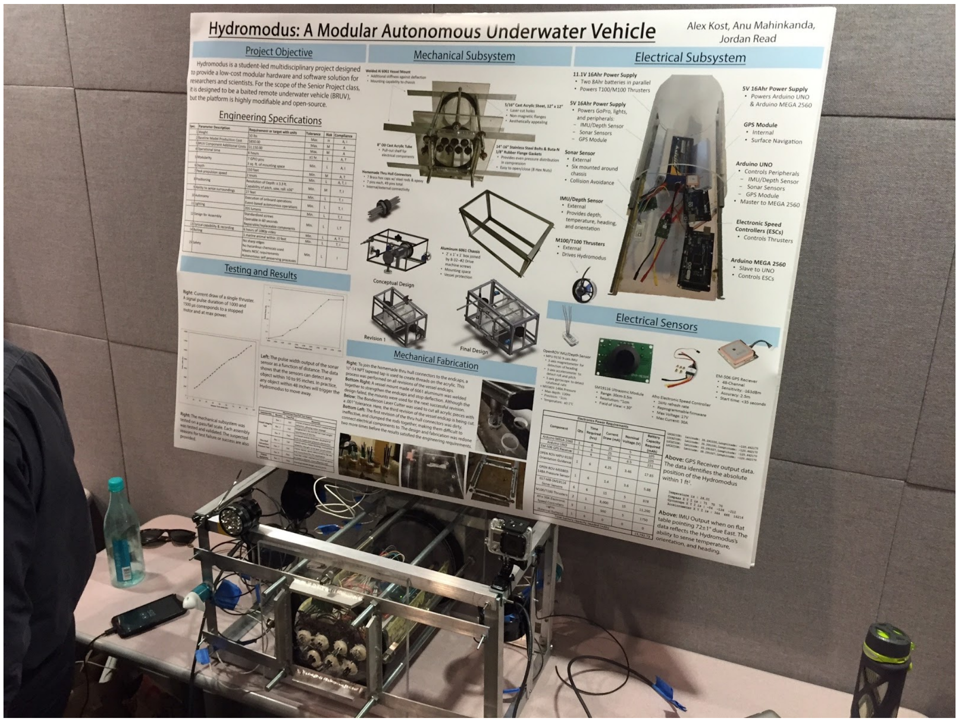


Figure 22. *The Hydromodus being presented at the Senior Design Exposition.*

Design Verification

To validate if the Hydromodus met the engineering specifications, numerous tests were performed on the Hydromodus and its subassemblies. There are ten total tests to validate that the Hydromodus meets the engineering specifications. All test procedures as well as their purposes, materials required, locations, and safety analyses can be found in **Appendix Q**. A Design Verification Plan and Report (DVPnR) was also created to organize the testing and validation procedure. The DVPnR can be seen under **Appendix R.**

Mechanical Hardware Validation

Table 3. Mechanical pass/fail testing results.

Subassembly & Relevant Test	**Revision**	**Pass/Fail**	**Reason(s)**
Subassembly: Watertight Vessel **Test:** Vessel Integrity Test	Rev. 1	Fail	Uneven pressure distribution created endcap deflection, allowing water seepage
	Rev. 2	Fail	Weighed 63.2 lbs and did not meet weight requirement of 50.0 lbs
	Rev. 3	N/A	Increased endcap thickness, doubled # of bolts to create even pressure distribution
Subassembly: Thru Hull Connectors **Test:** Vessel Integrity Test	Rev. 1	Fail	Uneven epoxy, messy caulking, inconsistent reliability; thought to be poor construction, but later determined that epoxy is too inconsistent for this application
	Rev. 2	Pass	Used flat rubber washers and minimal epoxy to create a cleaner, more effective seal
Subassembly: Aluminum Chassis **Test:** Chassis Reliability	Rev. 1	Pass	Over-engineered design, secure joints

Excluding propulsion system tests, two tests applied to the mechanical subassembly: the Chassis Reliability and Vessel Integrity tests. Both of these tests were pass/fail tests that resulted in major revisions in the mechanical design. The pass/fail results and reasoning behind the pass/fail is presented above in **Table 3**.

The Chassis Reliability test was made to determine whether the Hydromodus could withstand the impact forces from a marine animal or from hitting the ocean floor in shallow waters. Although impact strength was not defined in the formal engineering requirements, it was determined that the test was worthwhile to determine how safe the Hydromodus is to users launching the device. The procedure is outlined in the **Appendix Q**, and the test passed, verifying that the chassis was strong enough for most underwater situations.

The Vessel Integrity test was made to determine whether the Hydromodus was truly waterproof. Upon completion of Revision 1 of the mechanical subassembly, the team met at the ASI Recreation Center to test the integrity of the watertight vessel. This test, also visible in **Appendix Q**, showed that the Revision 1 design was not completely watertight. Although the vessel was waterproof the first time submerged, these results could not be repeated after placing electronics inside the vessel. Furthermore, air bubbles were seen to come from the connection between the thru hull connectors and the acrylic endcaps. **Figure 23** below shows the air bubbles circled in red. It was determined that the epoxy was a messy and difficult solution to waterproofing because of how unreliable epoxy was upon application. It was very difficult to be consistent when applying it, so while most thru hull connectors works, two did leak at a slow rate (two-four drops of water/min.).

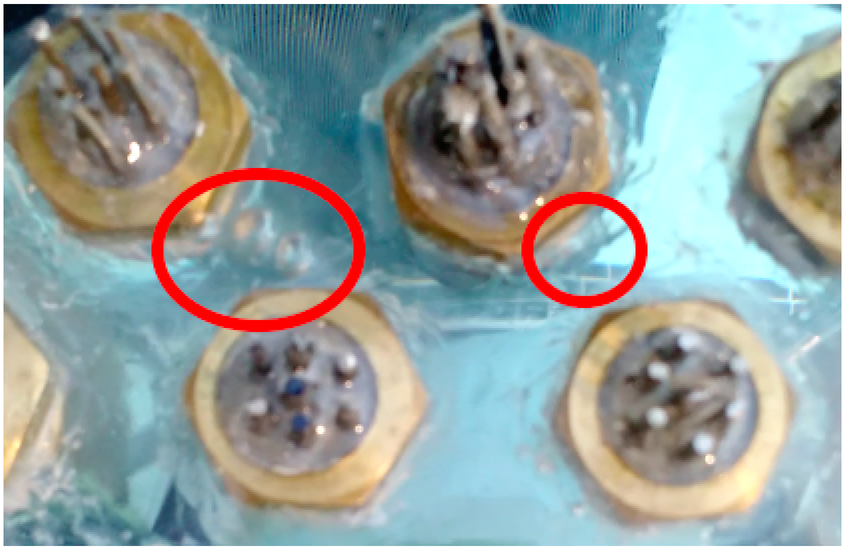


Figure 23. *Air bubbles leaking from thru hull connectors on Revision 1.*

Water also leaked between the edge of the tube and the Buna-N rubber flange. This may be attributed to the applying too much torque when fastening the nuts and bolts, thus creating a larger bending force across the endcap and a “mouth” to exist across the tube.. The root cause for the test failure was never defined because it was determined that the system had to be user-friendly, and if the design could fail simply fastening the bolts too securely, the design was not acceptable. It was determined that to resolve this issue, a more evenly-distributed pressure force would be needed on the endcap. Therefore, Revision 2 was made with more bolts to create a more flange-like system. Before Revision 2 could be tested, however, the Buna-N rubber flange was torn and the subassembly could not be tested.

For each failed design, a revision was undergone in hopes of solving the issue and passing the test when retaken. For example, when the watertight vessel failed after the first test, the vessel was redesigned to use low-pressure forged steel flanges with compressible

Buna-N flange gaskets instead of acrylic endcaps. However, this redesign resulted in a system that was too heavy. Revision 1 of the endcap is the final design of the endcap mentioned in Revision 2 of the final mechanical design. This endcap is thicker than the original and incorporates the design of the steel flanges in a much lighter, thinner material. Unfortunately, the design was not tested before the Buna-N rubber flange was torn, rendering the vessel unusable.

The thru hull connectors, incorporated in the vessel and tested in parallel with the endcap seal, were revised after it was determined that applying more silicone epoxy was more messy and less effective. To counter this, the idea to use rubber washers to seal the connection between thru hull connectors and endcaps was proposed. Once installed and tested, however, water leaked through. Marine epoxy and plumbers tape was used to seal the thru hull connectors completely and permanently to the endcap in a last-chance attempt. This design survived initial water coverage, but the test could not be completed before the Buna-N rubber flange was torn.

Electrical Hardware and Software Validation

The propulsion control test involved understanding how to connect and control the Blue Robotics Thrusters. The suppliers recommend to power the thrusters while they are submerged in water as the driving motors are lubricated with the water. Therefore all thruster related tests were performed in deep containers filled with water to roughly one foot in depth. Even at the lowest power level, the thrusters have a minimum depth of one foot to operate properly. The propulsion test involved learning how the ESCs interacted with the thrusters themselves. It was later determined that the ESCs needed to have their firmware reprogrammed in order to allow the thrusters to have both forward and reverse capabilities. This test was completed with success for three out of the five ESCs. It was with this test we found that two of our ESCs were not functional and had to be replaced. The testing apparatus can be seen in **Figure 24**.

Figure 24. *Thruster testing apparatus.*

The second test designed for the thrusters were to compare and confirm the current draw specifications provided by the supplier. A multimeter rated for ten Amps was used to measure the current drawn from the ESCs and the motor attached, at varying RPM. The test concluded at half the thrusters max capability since the current draw at that level began to melt a cable used for the procedure and burn the surface beneath it. The data is shown in **Figure 25** provides a baseline for calibrating the motors output thrust.

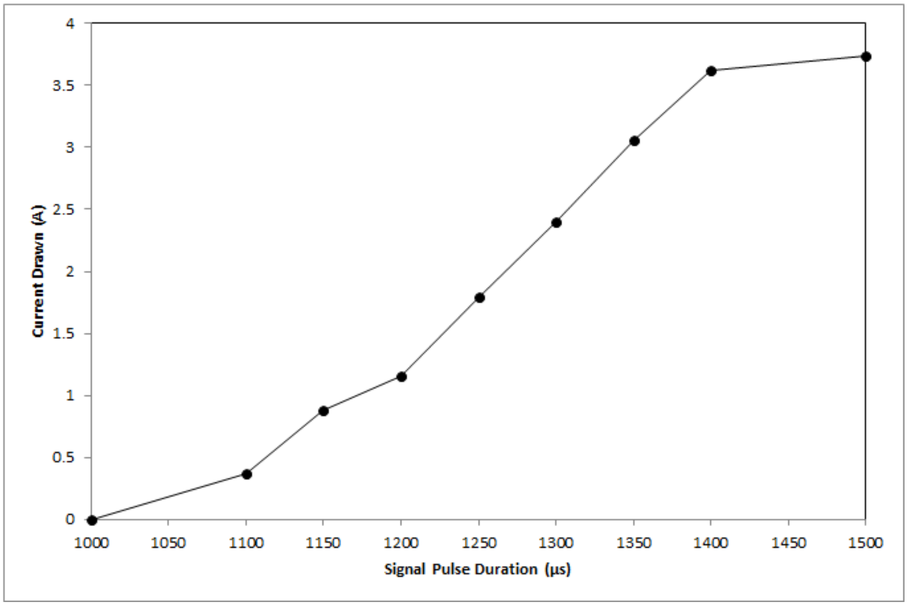


Figure 25. *Graphical representation of current draw against signal pulse duration.*

The GPS acquisition test took place at the ASI Recreation Center Pool, pictured below in **Figure 26**. This involved submerging the vessel with the GPS module inside it running, and visually determining when the GPS lost its connection. It was determined that as soon as the vessel was fully submerged underwater, the GPS lost its data connection. Similarly, as soon as the vessel ascended above the surface of the water, the GPS reconnected and acquired its position. It was noted that when the GPS module attempted to connect or reconnect after an hour or longer of inactivity, the GPS module would take three to four minutes before reestablishing a connection. This was factored into the programming by including a delay command when attempting to establish a connection with the GPS module.

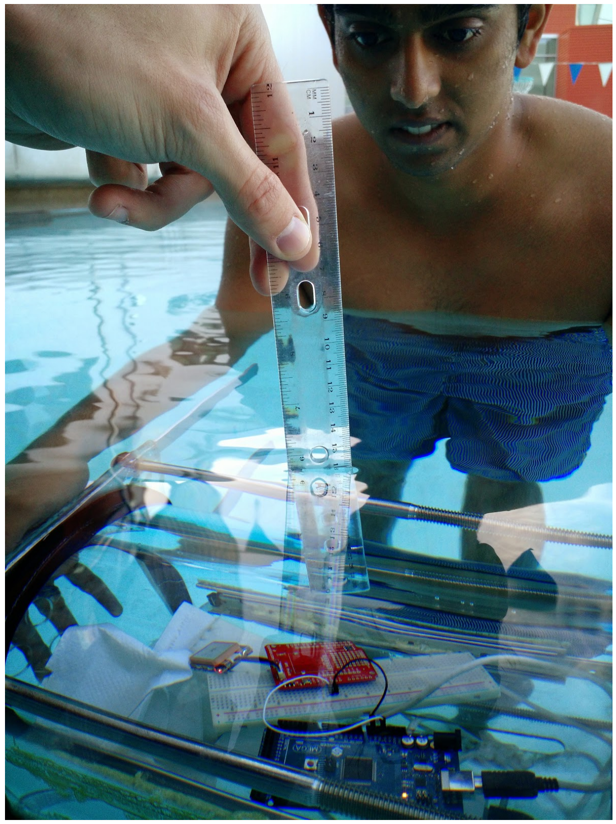


Figure 26. *GPS test at ASI Recreation Center pool*.

The Sonar Distancing test was performed to test whether sensors reliably and accurately measured objects in water. The sensor was first tested outside of water. The data collected with the in-air testing is shown below in **Figure 27**. This shows the output of the sensor and what it represents in terms of distance. The test was then performed in water. In the water, the sonar sensor range was greatly amplified. The length of the pool that the sensors were tested in limited the ability to test the maximum range of detection, but the test proved that the sensors could identify an object at least 12 feet away. The test also confirmed that the sensors were fully operational and that they produced a signal that is linearly proportional to the distance.

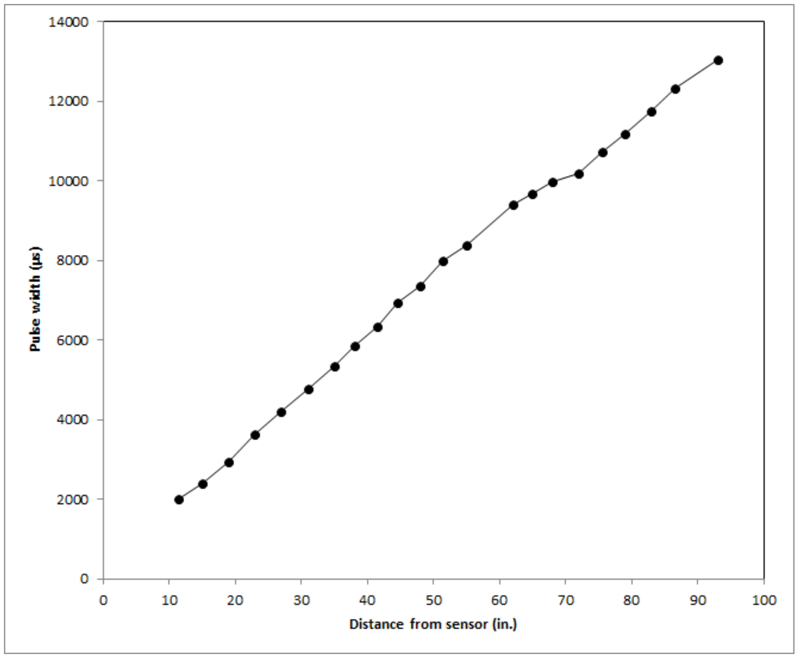


Figure 27. *Graphical representation of pulse width against distance from sensor.*

The IMU/Depth tracking test was done to determine whether the OpenROV IMU/depth sensor could reliably and accurately measure the depth of the current position as well as the horizontal orientation of the Hydromodus. The test was successful and confirmed that the sensor could output highly-accurate pitch, yaw, roll, accelerometer, and compass heading data. The compass heading data was the only data that could be validated with an actual compass, and it was determined that the compass heading was accurate up to 1°. The gyroscopic data produced the same results every time the sensor was angled to a certain orientation, proving reliability and consistency in the data. With so much positive data, the IMU/depth sensor was proven to be completely accurate and operational.

The Depth Sensor test took place in a water container a little over a foot in depth. The sensor was able to detect a difference in depth with a resolution of half an inch. Once this was determined to be accurate with a ruler, the test was deemed successful.

The Autonomous Lighting test determined if a light sensor reliably and accurately turned a LED on and off based off of ambient light conditions. Using the ambient light sensor to determine the light intensity of the environment, the test successfully showed that the sensor could determine the level of light and by proxy could send a signal to the Arduino to turn a light on and off. The test also provided a calibration value of 300 for the autonomous lighting functionality found in the slave microcontroller code. This can be seen in **Appendix K**.

Conclusions and Recommendations

Risk Analysis

As expected from the beginning, the mechanical subassembly of the Hydromodus was much more time consuming and carried more risk into testing and final project delivery. After the first revision of the endcap failed the Vessel Integrity test on April 23, 2015, redesigning and resolving the issue became the primary focus for Alex. Although all other electrical components were still undergoing individual testing and code development, time was limited so that testing would not delay. After another failed revision, it was evident the old testing schedule could no longer be followed. Detailed processes for risk analysis for the mechanical components can be found in **Appendix S**.

On the electrical side of the design, testing was pushed back repeatedly for more time to correctly interface with the sensors. All electrical components took longer than expected to interface with one another. It was also discovered on May 23 that two ESCs were dead-on-arrival and needed to be replaced, pushing testing back further as build continued. Although most of the work associated with the electrical subassembly was completed by the end of the project, the final system was not soldered together and was not tested in a real-world scenario. More time should have been provided for testing instead of building, and more organization should have completed beforehand so that testing and building would not conflict with one another. Similarly, contingency plans for failed tests should have been drafted so a plan existed in case a test failed--something that occurred repeatedly when testing the Hydromodus. Detailed processes for risk analysis for the electrical components can be found in **Appendix T**.

International Compatibility

The Hydromodus is a very versatile device in terms of global use and adaptation. The very core ideals of the Hydromodus is variety and adaptability. The programming language used to run the Hydromodus is C, one of the world's most widely used programming language. Anyone with programming experience will be able to operate and modify the Hydromodus to whatever desired functions and capabilities.

The Hydromodus is also global because of its primary function to assist researchers learn and study about marine environments. Oceans are everywhere and have a countless amount of different ecosystems ranging throughout. Outside of oceans, there are millions of lakes, wells, caves, and underwater locations. The opportunity to go underneath the surface is just an invention away; Hydromodus hopes to be this invention. This is why Hydromodus, by definition, is so internationally applicable: it addresses a topic that many people from any nation is interested in.

Technically, the Hydromodus is powered with onboard batteries. The only significant technical compatibility issues an international user might encounter with these are charging the onboard power supplies. The provided power supply chargers are designed for the standard NEMA 5–15R electricity outlets found in the united states. These chargers would require an international adapter plug, like the Targus Travel Adapter APK01US found on Amazon for ~$22, for widespread compatibility. Another technical issue is that the Hydromodus was mechanically designed in English units. Although it is not difficult to calculate the sizes in SI units, it could be more difficult to find the exact components internationally. This was not considered in the design because similar to how Americans can purchase metric-sized components, international vendors too can purchase English-sized components. It should be noted that if ever manufactured on a large scale, all components would be sourced in the same unit system.

Final Thoughts

With the conclusion of the Multidisciplinary Senior Design course, work on Hydromodus has come to a close as well. Although the system is incomplete and broken in its current state, all components theoretically work and interface with one another. Nonetheless, the system is incomplete for a number of reasons. Between design development and final design fabrication, many mistakes and assumptions were made. First, the production cost and cost of additional peripherals to make the Hydromodus a BRUV were above the expected and required amounts. This was due to the final design being a prototype, rather than a realized product, with all of the development costs associated with testing and troubleshooting. The current costs account for replaced and unused components, and if the Hydromodus were to be further developed and refined, a mass-produced product would likely be less than the target of $2,000.

Other specifications that were not met include peak propulsion speed, positioning, and the ability to sense surroundings. These requirements were not tested with a completed system because the system was not completed, and as a result, there is no collected data with all components interacting together in a realistic scenario. Although many of the specifications were not met, many lessons were learned and a proof-of-concept of the Hydromodus was made. The team would like to thank Professor Katona and Professor Laiho for making the Hydromodus more than just an idea.

Acknowledgements

Alex Kost would like to thank the following people for all of their help and support:

- Thomas Katona for advising us and doing so much more.
- James Patton, John Ridgely, Russ Crow, and Bridget Benson for their invaluable advices and contributions.
- Junior Enriquez, Pamela Lee, Quint Stevens, and Erin Ambrose for their last-minute support in my time of need.
- Austin Abernathy and Han Tran for handling my frustration and stress wonderfully.
- My grandfather Boris Anufriev for repeatedly asking if he can fit in the Hydromodus.

Anu Mahinkanda would like to thank the following people for all of their help and support:

- Thomas Katona for supporting and helping us at times when we didn't even know we needed it.
- My parents Ananda and Asoka Mahinkanda for their reassurance and support throughout the experience.
- My friends Anibal Hernandez, Andrew Ferrell, Jordan Dykstra, Raymond Quicho, Robert Prosser, AJ Lunardi, and Caitlin Cocuzzo for allowing me to rant at any time of the day.
- My sister Piyumi Mahinkanda-Lin for keeping me sane.

Jordan Read would like to thank thank the following people for all of their help and support:

- Thomas Katona for his invaluable advice and support throughout the year.
- My parents, Matthew and Paula Gregory, for their unfailing encouragement and priceless advice.

References

Artham, Trishul, and Mukesh Doble. "Fouling and Degradation of Polycarbonate in Seawater: Field and Lab Studies." *Journal of Polymers and the Environment* 17.3 (2009): 170-80. *Springer Link*. Springer US. Web.

Budynas, Richard G., J. Keith. Nisbett, and Joseph Edward. Shigley. *Shigley's Mechanical Engineering Design*. New York: McGraw-Hill, 2011. Print.

Christ, Robert D., and Robert L. Wernli. *The ROV Manual: A User Guide for Remotely Operated Vehicles.* N.p.: Butterworth-Heinemann, 2013. Print.

Davis, Russ E. "About." *Spray*. Instrument Development Group, n.d. Web. 13 Mar. 2015.

"Espadaenet." *Bibliographic Data*. N.p., n.d. Web. 24 Oct. 2014.

Goodrich, Ryan. "Accelerometer vs. Gyroscope: What's the Difference?" *LiveScience*. TechMedia Network, 01 Oct. 2013. Web. 08 Dec. 2014.

Guo, Shuxiang, and Xichuan Lin. *Development of a Vectored Water-Jet-Based Spherical Underwater Vehicle*. N.p.: INTECH Open Access, 2011. Print.

"Home - Blue Robotics." *Blue Robotics*. N.p., n.d. Web. 08 Feb. 2015.

"Is Lithium-ion the Ideal Battery?" *Advantages & Limitations of the Lithium-ion Battery*. Battery University, n.d. Web. 07 Dec. 2014.

"McMaster-Carr." *McMaster-Carr*. N.p., n.d. Web. 08 Feb. 2015.

"OpenROV | Underwater Exploration Robots." *OpenROV | Underwater Exploration Robots.* N.p., n.d. Web. 23 Oct. 2014.

"OSHA Technical Manual (OTM) - Section VII: Chapter 1: Back Disorders and Injuries." *OSHA Technical Manual (OTM)*. N.p., n.d. Web. 23 Oct. 2014.

"ROV Thruster Testing." *YouTube*. YouTube, 3 Jan. 2011. Web. 10 Jan. 2015.

SEACON*. SEACON SEA-MATE: Underwater Electrical Wet-Mate Connectors Catalog - Rev III*. 2014. SM1-SM18. Web.

Appendices

Appendix A: House of Quality

Customer Requirements (WHATS)	Weighting (1 to 5)	Weight	Baseline Model Production Cost	BRUV Component Additional Costs	Operational time	Modularity	Depth	Peak propulsion speed	Positioning	Ability to sense surroundings	Autonomy	Lighting	Design for Assembly	Optical capability & recording	Baiting	Safety	OpenROV	UC San Diego SIO Spray	Autonomous Benthic Explorer (ABE)
		Engineering Requirements (HOWS)															**Benchmarks**		
Intuitive interface for layman operator	3	3			6	3	9	6		6	3		9	6		6	6	3	3
Intuitive maintenance for layman operator	2	3	6	3	3	3	6	6		9	9		6	6		6	9		
Safe for layman operator to use	5	3	6	6			9	6	3	3	9						9	9	9
Reliable operation for two years	4	9	9	9	9	9	6	9				6	6	9	9	9	9	9	9
Reliable operation at depth (<150')	4	9	9	9	9	9	9	9				9	9	9	9	9	3	9	9
Reliable operation for two continuous hours	4	9	9	9	9	9	9	9				9	9	9	9	9	9	9	9
Capable of transporting cargo (<10lbs)	3		9	6			3		3	3		9			6				
Capable of accommodating additional interfaces	5	6					9			3	6		9				3		
Able to attach baited cannister	5		3	3	3	6	6		3	3		9			3	3			
Source code available for development	4												9	3		3	9		
Collision avoidance system	4		3			3								6	6		3		
Maintain position (depth, pitch yaw, roll, etc) without operator	4		6	9					3					9	9			9	9
Portable, lightweight (<50lbs)	3	3	3	3					9							3	9		
Rechargeable	2							9			6						9		
Able to see in low light	4				9	9										9	3		9
Aesthetics	1	6	3	6	3						3		3				3	3	3
Cost (<$5,000)	2	6	6	6	6	6	3	9	3			6	6	6		6	6		
Safe to transport	3	6	6	6				3	6		9		3				9	6	
Quiet operation	5		9									3	3	3	3	6	6	9	
On-site data storage	5															9		9	
Stereo audio/video recording	5				9	6										9	6	9	9
Units																			
Targets																			
Benchmark #1																			
Benchmark #2																			
Importance Scoring		213	315	258	243	243	270	213	102	90	144	195	255	237	216	342			
Importance Rating (%)		62	92	75	71	71	79	62	30	26	42	57	75	69	63	100			

Appendix B: Engineering Specifications

Spec.	Parameter Description	Requirement or target with units	Reason	Tolerance	Risk	Compliance
1	Weight	50 lbs	Safe lifting weight according to OSHA labor requirements	Max.	H	A, I
2	Baseline Model Production Cost	$850.00	Competitive with unassembled OpenROV ($500 mech, $350 elec)	Max.	M	A
3	BRUV Component Additional Costs	$1,150.00	All Research AUVs are costly for additional components like sensors and peripherals; we hope to be competitive here. $2000 total set because funding	Max.	M	A
4	Operational time	6 hours	Standard BRUV mission length, Jordan Goetze (1/12/15)	±1 hr.	L	A, T
5	Modularity	7 GPIO pins Easy mounting w/ space	49 pins provided for space reasons, 42 needed by system, 7 available Chassis must provide space for additional components	Min.	L	A, I
6	Depth	150 feet	Limit for recreational divers, coral reefs External pressure is max. pressure at 150-ft depth of salt water	Min.	M	A, T
7	Peak propulsion speed	2 knots	Competitive with OpenROV (2 knots)	Min.	L	A, T, I
8	Positioning	Resolution of Depth: ± 3.3 ft. Capability of pitch, yaw, roll: ±36°	Must accurately identify position/ orientation for autonomous behavior to function correctly	Min.	M	T, I
9	Ability to sense surroundings	27 feet	Maximum capability for most BRUV systems. More conservative than required sensing distance to avoid collisions (email)	Min.	L	T, I
10	Autonomy	Execution of onboard operations control procedures. Event-based autonomous operations.	Must not require human operation; must be able to follow predetermined routines and respond to external inputs	Min.	L	T, I
11	Lighting	205 lumens	Competitive with OpenROV & capable to see up to 27 ft (email)	Min.	L	T, I
12	Design for Assembly	Standardized screws Openable in 60 seconds Repairable/replaceable components	Maximizes ease of use, reparability, and modularity	Min.	L	I, T
13	Optical capability & recording	6 hours of 1080p video	Must be able to record video of wildlife and landscapes; 1080p for lower depths with low light so objects still detected	Min.	L	A, T, I
14	Baiting	1 marine animal within 10 feet	Attract fish and other marine animals to the BRUV with baiting system	Min.	L	T, I
15	Safety	No sharp edges No hazardous chemicals used Meets NESC requirements Autonomous self-preserving processes	Must be safe to transport, handle, and operate at all times	Min.	L	I

Compliance: Analysis (A), Testing (T), Inspection (I)

Appendix C: Decision Matrix

			Engineering Requirements																		
System Functions	Potential Solutions (From Convergent Thinking Exercise)	Best Benchmark from QFD	Weight	Baseline Model Production Cost	BRUV Component Additional Costs	Operational time	Modularity	Depth	Peak propulsion speed	Positioning	Ability to sense surroundings	Autonomy	Lighting	Design for Assembly	Optical capability & recording	Baiting	Safety	Weighted Sum +	Weighted Sum -	Weighted Sum S	Total Score
Subsystem	Specification Weight		7	9	8	7	7	8	6	4	4	4	6	7	7	6	10				
	Mechanical Subsystem																				
Chassis Material	5052 Aluminum	S	+	-			S									+	+	23	9	7	16.1
	6061 Aluminum		+	S			+									+	+	30	0	9	32.7
Vessel Material	Acrylic	S	+				+	+						+	S		+	39	0	7	41.1
	Polycarbonate		+				+	+						+	S		S	29	0	17	34.1
Chassis Shape & Weight Distribution	Centered Payload (main large compartment)	S	+	S				S		S	S	+		S	+		+	28	0	32	37.6
	Balanced Payloads (smaller compartments)		-	-			+										-	7	26	0	-19
	Screwed Mounting		+				+				+		+	+	+	+	-	44	10	0	34
Internal/External Connections	SEACON All-Wet	S		-			+				+			-				11	16	0	-5
	SEACON Sea-Mate			-			+				+			-				11	16	0	-5
	Homemade Thru Hull Connectors			+			+				+			-				20	7	0	13
	Swagelok Fittings			-			+				+			-				11	16	0	-5
Propulsion	Vector Thrusters	S	S	-		+			+					+			-	20	19	7	3.1
	Brushless motors (T100/M100 Thrusters)		S	S		+			+					+			-	20	10	16	14.8
	Squid-Jet		S	-		+			+					+			-	20	19	7	3.1
Bait holder	Telescoping arm	S	-		-		S									+	-	6	25	7	-16.9
	Hook		-		-		S									+	-	6	25	7	-16.9
	Gripper		-		-		S									+	-	6	25	7	-16.9
	Screwed Mounting		S		-		+							+		+	S	20	8	17	17.1
	Electrical Subsystem																				
Orientation	Accelerometer & Gyroscope	S		+		S				+								13	0	7	15.1
	Accelerometer			S		+				S								7	0	13	10.9
	Gyroscope			S		+				S								7	0	13	10.9
Positioning	Pressure Sensor	S		S		+		S		S								7	0	21	13.3
	Depth Reader			S		+		S		S								7	0	21	13.3
	Hydroacoustic Aided Inertial Navigation			-		-		+		+								12	16	0	-4
Proximity Sensors	Acoustic Echolocation	S		S		S				+	+							8	0	16	12.8
	Laser Based Distance Meter			S		S				+	S							4	0	20	10
	Visual inspection			+		+				-	-							16	8	0	8
Power	Lithium-ion	S	+	-		+											+	24	9	0	15
	Sealed lead acid		-	+		S											-	9	17	7	-5.9
	Nickel-cadmium		-	S		+											S	7	7	19	5.7
	Super-capacitor		S	+		-											S	9	7	17	7.1
Microprocessor	TM4C123GXL	S		S		S	S					S		S				0	0	34	10.2
	TM4C1294XL			S		S	S					S		S				0	0	34	10.2
	Arduino Mega 2560 r3			S		S	+					+		+				18	0	16	22.8
	Arduino Uno			S		S	+					+		+				18	0	16	22.8
Communication	Acoustic	S		-		-												0	16	0	-16
	Satellite			+		-												9	7	0	2
	Light			-		-												0	16	0	-16
	Tether			-		S												0	9	7	-6.9

Appendix D: Power Analysis

Device	Qty. Used Simultaneously	Hours Used (hr)	Current Draw (mA)	Voltage Needed (V)	Needed Battery Capacity (mAh)
Arduino MEGA 2560 r3	1	6	25	5	105
Arduino UNO	1	6	25	5	105
GPS Receiver (EM-506)	1	6	55	6.5	231
IMU (OpenROV-MPU-9 150)	1	6	4.25	3.46	17.85
Pressure (OpenROV-MS580 3-14BA)	1	6	1.4	3.6	5.88
Relay Module (SainSmart Eight Channel)	1	6	120	5	504
Sonar (017-MB-SM1911 6)	6	6	15	5	378
Thrusters (Blue Robotics M100/T100)	2	1	8,000	15	11,200
Afro 30A Electronic Speed Controllers	5	1	500	5	1750

Appendix E: Mechanical Component Specification Sheets

See attached.

4-Ounce Bottle of Gorilla Glue

In stock
$8.28 Each
7454A22

Size	4 oz.
Begins to Harden	10 min.
Reaches Full Strength	24 hrs.
Color	Tan
Additional Specifications	MSDS
RoHS	Compliant

Stronger than other glue when bonding ceramic, glass, metal, plastic, stone, and wood. Glue is waterproof, paintable, stainable, and sandable at full strength.

McMASTER-CARR. OVER 555,000 PRODUCTS
(562) 692-5911
(562) 695-2323 (fax)
la.sales@mcmaster.com
Text 75930

Multipurpose 6061 Aluminum

8982K4

90 Degree Angle, 1/8" Thick, 1" x 1" Legs

Alloy	6061
Shape	90° Angle
Finish	Unpolished
Thickness	1/8"
Thickness Tolerance	±0.007"
Outside Leg Lengths	1" × 1"
Leg Length Tolerance	±0.014"
Yield Strength	35,000 psi
Hardness	Soft (80 Brinell)
Material Condition	Heat Treated
Temper	T6
Specifications Met	ASTM B221
Material Composition	
Silicon	0.4-0.8%
Iron	0-0.7%
Copper	0.05-0.4%
Manganese	0-0.15%
Magnesium	0.8-1.2%
Chromium	0.4-0.8%
Nickel	0-0.05%
Zinc	0-0.25%
Titanium	0-0.15%
Zirconium	0-0.25%
Other	0.15%
Aluminum	95.1-98.2%
Nominal Density	0.097-0.1 lbs./cu. in.
Modulus of Elasticity	10.0 ksi × 10^3
Elongation	8-17%
Melting Range	1,080° to 1,205° F
Thermal Conductivity	1390 Btu/hr × in./sq.ft. @ 75° to 77° F
Electrical Resistivity	24 Ohm-Cir. Mil/ft. @ 68° F
Length Tolerance	±1/2"
Length	1 ft., 2 ft., 3 ft., 4 ft., 8 ft.

The most widely used aluminum, Alloy 6061 is a popular choice for vehicle parts and pipe fittings. It has better corrosion resistance and weldability than Alloys 2024 and 7075, but it's not as strong. It is nonmagnetic, heat treatable, and resists stress cracking. Temperature range is -320° to 300° F.

Yield strength is approximate and may vary based on size and shape.

Inside corner and leg edges are rounded. Outside corner is squared. Thickness tolerance for 1/8" to 1/2" thick angles is ±0.009". Leg length tolerance for 1/4" to 1 3/4" leg lengths is ±0.014". Length tolerance is ±1/2".

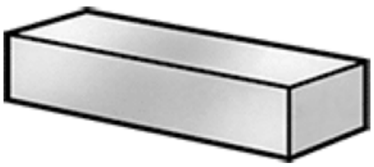
McMASTER-CARR® OVER 555,000 PRODUCTS

(562) 692-5911
(562) 695-2323 (fax)
la.sales@mcmaster.com
Text 75930

8975K578

Multipurpose 6061 Aluminum

Rectangular Bar, 1/8" x 1"



Alloy	6061
Shape	Rectangular Bar
Finish	Unpolished
Thickness	1/8"
Thickness Tolerance	±0.007"
Width	1"
Width Tolerance	±0.014"
Yield Strength	35,000 psi
Hardness	Soft (80 Brinell)
Material Condition	Heat Treated
Temper	T6511
Specifications Met	ASTM B221
Material Composition	
Silicon	0.4-0.8%
Iron	0-0.7%
Copper	0.05-0.4%
Manganese	0-0.15%
Magnesium	0.8-1.2%
Chromium	0.4-0.8%
Nickel	0-0.05%
Zinc	0-0.25%
Titanium	0-0.15%
Zirconium	0-0.25%
Other	0.15%
Aluminum	95.1-98.2%
Nominal Density	0.097-0.1 lbs./cu. in.
Modulus of Elasticity	10.0 ksi × 10^3
Elongation	8-17%
Melting Range	1,080° to 1,205° F
Thermal Conductivity	1390 Btu/hr × in./sq.ft. @ 75° to 77° F
Electrical Resistivity	24 Ohm-Cir. Mil/ft. @ 68° F
Length Tolerance	±1"
Length	1/2 ft., 1 ft., 2 ft., 3 ft., 6 ft.

The most widely used aluminum, Alloy 6061 is a popular choice for vehicle parts and pipe fittings. It has better corrosion resistance and weldability than Alloys 2024 and 7075, but it's not as strong. It is nonmagnetic, heat treatable, and resists stress cracking. Temperature range is -320° to 300° F.

Yield strength is approximate and may vary based on size and shape.

Width tolerance for 1/4" to 1 3/4" wide bars is ±0.014". Length tolerance is ±1".

1. IDENTIFICATION OF PRODUCT

Product Name: Gorilla Glue®
Product Type: Polyurethane adhesive

Distributor: The Gorilla Glue Company
4550 Red Bank Expressway
Cincinnati, OH 45227
Tel: (513) 271-3300
Fax: (513) 527-3742

2. HAZARDS IDENTIFICATION

Harmful by inhalation. Irritating to eyes, respiratory system and skin. May cause sensitization by inhalation and skin contact.

NFPA: Health – 2, Flammability – 1, Reactivity – 1
0=Insignificant 1=Slight 2=Moderate 3=High 4=Extreme

HMIS: Health – 2*, Flammability – 1, Reactivity – 1
0=Minimal 1=Slight 2=Moderate 3=Serious 4=Severe *=Chronic Health Hazard

3. COMPOSITION/INFORMATION ON INGREDIENTS

Chemical name	CAS No.	% content
Prepolymer based on aromatic polyisocyanate	67815-87-6	44
Polymeric Diphenylmethane Diisocyanate (pMDI)	9016-87-9	26
4,4'-Diphenylmethane Diisocyanate (MDI)	101-68-8	25
Diphenylmethane Diisocyanate (MDI) Mixed Isomers	26447-40-5	5

4. FIRST AID MEASURES

Inhalation If aerosol or vapor is inhaled in high concentrations: Move affected individual to fresh air and keep him warm, let him rest. If there is difficulty in breathing; call a doctor.

Eye contact Flush eyes for at least 10 minutes while holding eyelids open. Contact a doctor.

Skin contact Remove contaminated clothes immediately, and wash skin with a cleanser based on polyethylene glycol or with plenty of water and soap. Consult a doctor in the event of a skin reaction.

Ingestion Product is not intended to be ingested or eaten. If this product is ingested, it may cause gastro-intestinal blockage. If ingested, it may cause severe irritation of the gastrointestinal tract, and should be treated symptomatically. Do not induce the patient or animal to vomit. Call a doctor, ambulance or seek veterinarian assistance immediately.

5. FIRE FIGHTING MEASURES

Upper flammable limit (UFL): Not determined
Lower flammable limit (LFL): Not determined
General fire hazards
Down-wind personnel must be evacuated. Do not reseal contaminated containers; a chemical reaction generating carbon dioxide gas pressure may occur resulting in rupture of the container. Dense smoke is emitted when product is burned without sufficient oxygen. When using water spray, boil-over may occur when product temperature reaches the boiling point of water, and the reaction forming carbon dioxide will accelerate. MDI vapor and other gases may be generated by thermal decomposition.

Special hazards in fire
In case of fire, formation of carbon monoxide, carbon dioxide, nitrogen oxide, isocyanate vapor, and traces of hydrogen cyanide is possible.
Extinguishing Media
Carbon dioxide, dry powder, and foam. In cases of large scale fires, alcohol-resistant foams are preferred. If water is used, it should be used in very large quantities. The reaction between water and isocyanate may be vigorous.
Required special protective equipment for fire-fighters
Fire fighters should wear full-face, self-contained breathing apparatus and impervious protective clothing. Fire fighters should avoid inhaling any combustion products.

6. ACCIDENTAL RELEASE MEASURES

Personal precautions
Wear full-protective clothing and respiratory protection as required maintaining exposures during clean-up below the applicable exposure limits.
Environmental precautions
Do not discharge spillage into drains.
Clean-up procedures
Remove mechanically; cover remainders with wet absorbent material (e. g. sand, earth, sawdust). After approx. 15 min. transfer to waste container and do not seal (evolution of CO_2). Keep damp in a safe ventilated area for several days.

7. HANDLING AND STORAGE

Handling
Avoid contact with skin and eye. Do not smoke, eat and drink at the work-place.
Ventilation: If vapor or mist is generated during processing or use, local exhaust ventilation should be provided to maintain exposures below the applicable limits.
Personal protection: see Section 8.
Storage
Keep product away from sources of alcohols, amines, or other materials that react with isocyanates. Avoid prolonged heating above 160°C/320°F. Store the product in tightly closed containers in a well-ventilated place and in accordance with national regulations. Keep out of reach of children and animals.

8. EXPOSURE CONTROLS/ PERSONAL PROTECTION

For exposure controls see Section 15.
Component exposure limits

	CAS no.	Type	ppm	mg/m^3
4,4’-Diphenylmethane diisocyanate	101-68-8	OSHA PEL	0.02	0.2
		ACGIH (TLV-TWA)	0.005	

Personal protection equipment

General: Wear suitable protective clothing, protective gloves and protective goggles/mask.

Suitable materials for safety gloves:

Natural rubber/natural latex – NR (>= 0.5 mm)
Polychloroprene – CR (>= 0.5mm)
Nitrile rubber – NBR (>= 0.35mm)
Butyl rubber – IIR (>= 0.5 mm)
Fluorinated rubber – FKM (>= 0.4 mm)

Personal protection equipment (continued)

Respiratory protection Required in insufficiently ventilated working areas and during spraying. An air-fed mask, or for short periods of work, a combination of charcoal filter and particulate filter is recommended.

Eyes protection Chemical goggles or full face shields are recommended. An eyewash fountain and safety shower should be available in the work area. Contact lenses should not be worn when working with this product.

Skin protection Wear special gloves and working clothes to avoid skin irritation or sensitization. Depending on operation, chemical resistant boots, overshoes, and apron may also be required.

Suitable materials for clothing: Polyethylene/ethylene vinyl alcohol laminates (PE/VAL) has been reported as an effective material of construction for chemical protective clothing for MDI.

9. PHYSICAL AND CHEMICAL PROPERTIES

Physical form. . . . . Liquid
Color . . . . Dark-Brown
Odor . . . . Earthy,musty
Boiling point . . . . >300°C
Flash point . . . . >250°C
Vapour pressure. . . . . <0,00001 mbar at 20° C (diphenyl-methane-diisocyanate)
Specific gravity . . . . Approx. 1,14 g/cm3 at 20° C
Viscosity . . . . 4,000 – 7,000 mPa.s at 25°C (Brookfield sp. 6/20 rpm)
Solubility in water . . . . . . reacts
Percent VOC . . . . 1% does not contain solvents. (12 g/L SCAQMD compliant)
Pour point…………………Approx -12°C (10.4 °F)

10. STABILITY AND REACTIVITY

Stability
The product is stable under the recommended handling and storage conditions (see section 7).

Hazardous decomposition products
By exposure to high temperature, hazardous decomposition products may develop, such as isocyanate vapour and mist, carbon dioxide, carbon monoxide, nitrogen oxide, and traces of hydrogen cyanide.

Hazardous reaction
Exothermic reaction with amines and alcohols; reacts with water forming heat,CO2, and insoluble polyurea. The combined effect of CO2 and heat can produce enough pressure to rupture a closed container.

11. TOXICOLOGICAL INFORMATION

Acute Oral Toxicity LD50 rat: > 2,000 mg/kg
Acute Inhalation Toxicity LC50 rat: 490 mg/m , aerosol, 4 h
Skin Irritation rabbit, slight irritant
Inhalation Over-exposure may cause irritating effects on nose throat and respiratory tract.
Skin contact Prolonged or repeated contact may result in tanning and irritating effects.
Eye contact Over-exposure may cause irritating effects on eyes.

12. ECOLOGICAL INFORMATION

Do not allow the product to escape into waters, wastewater or soil.

Biodegradability	0% after 28 days
Bioaccumulation	Does not bioaccumulate.
Acute toxicity to fish	LC0 > 1,000 mg/l (Zebra fish, Brachydanio rerio) 96 hrs.
Toxicity for daphnia	EC 50 > 1,000 mg/l (24 hrs.)
Acute toxicity to bacteria	EC 50 > 100 mg/l (3 hrs.)

13. DISPOSAL CONSIDERATIONS

The product remnants are classified as chemical waste. Dispose of waste according to Local, State, Federal, and Provincial Environmental Regulations.

14. TRANSPORTATION INFORMATION

No classification assigned to: Land transport (DOT) / Sea transport (IMDG) / Air transport (ICAO/IATA)

15. REGULATION INFORMATION

This product and its components are listed on the TSCA 8(b) inventory.

United States Federal Regulations
OSHA Hazcom Standard Rating: Hazardous
US. Toxic Substances Control Act: Listed on the TSCA Inventory.
US. EPA CERCLA Hazardous Substances (40 CFR 302):

Components
4,4'-Diphenylmethane Diisocyanate (MDI)

SARA Section 311/312 Hazard Categories: Acute Health Hazard, Chronic Health Hazard
US. EPA Emergency Planning and Community Right-To-Know Act (EPCRA) SARA Title III Section 302 Extremely Hazardous Substance (40 CFR 355, Appendix A):

Components
None

US. EPA Emergency Planning and Community Right-To-Know Act (EPCRA) SARA Title III Section 313 Toxic Chemicals (40 CFR 372.65) - Supplier Notification Required:

Components
Polymeric Diphenylmethane Diisocyanate (pMDI)
4,4'-Diphenylmethane Diisocyanate (MDI)

US. EPA Resource Conservation and Recovery Act (RCRA) Composite List of Hazardous Wastes and Appendix VIII Hazardous Constituents (40 CFR 261):
If discarded in its purchased form, this product would not be a hazardous waste either by listing or by characteristic. However, under RCRA, it is the responsibility of the product user to determine at the time of disposal, whether a material containing the product or derived from the product should be classified as a hazardous waste (40 CFR 261.20-24).

State Right-To-Know Information
The following chemicals are specifically listed by individual states; other product specific health and safety data in other sections of the MSDS may also be applicable for state requirements. For details on your regulatory requirements you should contact the appropriate agency in your state.
This product contains a trace (ppm) amount of phenyl isocyanate (CAS# 103-71-9) and monochlorobenzene (CAS# 108-90-7), present below the maximum concentration for D021toxicity.

Massachusetts, New Jersey or Pennsylvania Right to Know Substance Lists:

Weight %	Components	CAS-No.
40 - 50%	Polyisocyanate Prepolymer based on MDI	CAS# is a trade secret
25 - 35%	Polymeric Diphenylmethane Diisocyanate (pMDI)	9016-87-9
20 - 30%	4,4'-Diphenylmethane Diisocyanate (MDI)	101-68-8
1 - 5%	Diphenylmethane Diisocyanate (MDI) Mixed Isomers	26447-40-5

New Jersey Environmental Hazardous Substances List and/or New Jersey RTK Special Hazardous Substances Lists:

Weight %	Components	CAS-No.
25 - 35%	Polymeric Diphenylmethane Diisocyanate (pMDI)	9016-87-9
20 - 30%	4,4'-Diphenylmethane Diisocyanate (MDI)	101-68-8

California Prop. 65:
Warning! This product contains chemical(s) known to the State of California to be - Developmental toxin.

Weight %	Components	CAS-No.
10 ppm	Toluene	108-88-3

16. OTHER INFORMATION

The information herein is presented in good faith and believed to be accurate as of the effective date given. However, no warranty, expressed or implied, is given. It is the buyer's responsibility to ensure that its activities comply with Federal, State or Provincial, and Local laws.

The Gorilla Glue Company does not test on animals, nor do we require our suppliers to test on animals. Any information provided in this MSDS is based on existing scientific testing of the various raw materials, and is not commissioned by this Company.

More About Plastics

Tensile Strength—The amount of stretching a material can withstand before breaking. It is usually measured in pounds per square inch (psi). A larger number indicates a stronger material.

Impact Strength—The ability to withstand shock loading. Determined by the notched Izod test, which measures the effect on a material when it is struck by a swinging pendulum. A larger number signifies greater impact resistance. "No Break" means the material was not broken during testing.

Coefficient of Friction—The ratio of the frictional force between two surfaces and the force that keeps those surfaces in contact. A lower value indicates a material that moves more easily, or with less friction, than a material with a higher value.

Short-Term Dielectric Strength—The maximum voltage a material can withstand without rupture, measured as volts per mil of thickness. This is an indication of how effective the material is as an electrical insulator. A higher value signifies a better insulator.

Coefficient of Thermal Expansion—The amount a material increases in volume as the temperature rises. A smaller coefficient is an indicator of less thermal expansion.

Machine With—High-Speed Steel Tooling (HSS); Tungsten Carbide Tooling (Carbide).

Warning: Physical and mechanical properties are not guaranteed and are intended only as a basis for comparison. Data is not for design purposes. It is given without obligation or liability. No warranty of fitness for a particular purpose or application is made.

Material	Product	Tensile Strength, psi	Rockwell Hardness	Impact Strength, ft.-lbs./in.	Coefficient of Friction	Dielectric Strength, volts/0.001″	Water Absorption, %	Density, lbs./in.^{3}	Thermal Expansion, in./in./° F	Machine With
ABS	ABS	5,100-6,100	R102-109	5.2-7.7	Not Rated	450-1,220	0.3-1.0	0.032-0.038	5.2×10^{-5}	HSS
ABS/PVC	Electrically Conductive ABS/PVC	4,500	R87	2	Not Rated	Not Rated	Not Rated	0.04	4.6 to 5.5×10^{-5}	Carbide
Acetal	Acetal	6,400-9,500	M51-M88	1-1.8	0.11-0.35	420-500	0.2-0.8	0.048-0.051	5.4 to 12×10^{-5}	HSS
	Delrin® Acetal Resin	9,000-11,000	M89-M94	1-2.4	0.2	435-500	0.2-0.4	0.051	4.7 to 12.2×10^{-5}	HSS
	Glass-Filled Delrin® Acetal Resin	8,700	M81	0.8	Not Rated	450	Not Rated	0.054	3.33×10^{-5}	Carbide
	PTFE-Filled Delrin® Acetal Resin	6,800-12,490	M77-M78	0.7-1.2	0.07-0.14	400-500	0.25	0.054	5.1×10^{-5}	HSS
	Turcite Acetal	5,900-7,600	M63-M81	0.54-0.57	0.22-0.3	Not Rated	0.2	0.053	5.0×10^{-5}	HSS
Acrylic	Cast Acrylic	8,000-11,250	M94-M103	0.04-0.5	Not Rated	400-430	0.2-0.8	0.043	3.5 to 4.2×10^{-5}	Carbide
	Extruded Acrylic	8,100-11,030	M68-M95	0.3-0.7	Not Rated	430-760	0.2-0.4	0.043	3.0 to 4.0×10^{-5}	Carbide
Acrylic/PVC	Kydex Acrylic/PVC	6,100	R94	15	Not Rated	Not Rated	0.05	0.049	3.8×10^{-5}	HSS
Cellulose	Acetate	4,500-8,000	R78-R120	2.0-8.5	Not Rated	250-600	2.0-7.0	0.048	5.6 to 8.3×10^{-5}	Cut with knife
	Butyrate	4,800	R78	4.5	Not Rated	300-475	1.4	0.027	6.0 to 9.0×10^{-5}	HSS
CPVC	CPVC	7,100-7,300	R116-119	8-9	Not Rated	1,250	0.03	0.053-0.056	3.9×10^{-5}	Carbide
CTFE	CTFE	4,860-5,710	Shore D85-D95	2.5-3.5	0.08	500	0	0.034-0.08	3.9 to 5.1×10^{-5}	HSS
FEP	FEP	3,000	R25	No Break	0.25	1,800	<0.01	0.078	4.6 to 5.8×10^{-5}	Carbide
HDPE	HDPE Polyethylene	4,000-4,100	Shore D60-D68	1.1	0.22-0.62	450-1,800	0	0.034	5.3 to 10×10^{-5}	HSS
LDPE	LDPE Polyethylene	3,100-6,100	Shore D42-D56	Not Rated	Not Rated	Not Rated	Not Rated	0.033	Not Rated	HSS

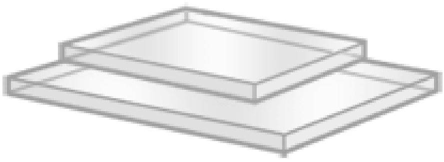
(Continued on following page)

(562) 692-5911
(562) 695-2323 (fax)
la.sales@mcmaster.com
Text 75930

Optically Clear Cast Acrylic Sheet

1/4" Thick, 12" x 12"

In stock
$16.36 Each
8560K354



Thickness	1/4"
Thickness Tolerance	+0.020", -0.040"
Color	Clear
Temperature Range	-40° to 170° F
Tensile Strength	Excellent
Impact Strength	Poor
Additional Specifications	Sheets 12" × 12"
RoHS	Not Compliant

All shapes offer excellent clarity over a wide temperature range. Comparable to Plexiglas® acrylic shapes and Lucite, this material can be used outdoors.

View detailed performance properties for plastics.

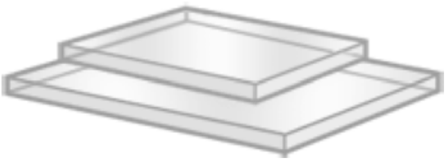
Width and length tolerances are ±1/4". When heated, sheets will uniformly shrink 0.5 to 2% in width and 1 to 2% in length. Made from FDA-compliant resins.

McMASTER-CARR. OVER 555,000 PRODUCTS

Optically Clear Cast Acrylic Sheet

1/8" Thick, 12" x 12"

In stock
$8.63 Each
8560K239



Thickness	1/8"
Thickness Tolerance	+0.015", -0.025"
Color	Clear
Temperature Range	-40° to 170° F
Tensile Strength	Excellent
Impact Strength	Poor
Additional Specifications	Sheets 12" × 12"
RoHS	Not Compliant

All shapes offer excellent clarity over a wide temperature range. Comparable to Plexiglas® acrylic shapes and Lucite, this material can be used outdoors.

View detailed performance properties for plastics.

Width and length tolerances are ±1/4". When heated, sheets will uniformly shrink 0.5 to 2% in width and 1 to 2% in length. Made from FDA-compliant resins.

McMASTER-CARR. OVER 555,000 PRODUCTS
(562) 692-5911
(562) 695-2323 (fax)
la.sales@mcmaster.com
Text 75930

Optically Clear Cast Acrylic Tube

8" OD x 7-3/4" ID

In stock
$66.27 Each
8486K735

OD	8"
OD Tolerance	±0.065"
ID	7 3/4"
Wall Thickness	1/8"
Length	1 ft.
Color	Clear
Temperature Range	0° to 150° F
Tensile Strength	Good
Impact Strength	Poor
Additional Specifications	Round Tubes
RoHS	Compliant

All shapes offer excellent clarity over a wide temperature range. Comparable to Plexiglas® acrylic shapes and Lucite, this material can be used outdoors.

View detailed performance properties for plastics.

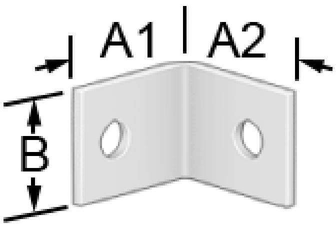
Length tolerance is ±1". Wall thickness tolerance is ±15%. Meet UL 94HB for flame retardance (except 14" OD size). Tubes up to 8" OD are also made from FDA-compliant resins.

(562) 692-5911
(562) 695-2323 (fax)
la.sales@mcmaster.com
Text 75930

Bracket

Type 304 Stainless Steel, 7/8" Length of Sides

In stock
1-49 Each $1.06
50 or more $0.79
1556A64



Material	Type 304 Stainless Steel
Length (A1), (A2)	7/8"
Width (B)	5/8"
Thickness	0.08"
Screw/Nail Size	No. 8
Number of Holes	2
RoHS	Not Compliant

Also known as angle brackets, corner brackets, and mending plates, these brackets support corners and joints. They do not include mounting fasteners.

Note: Prices are approximately 25% lower when you buy 50 or more of the same bracket.

McMASTER-CARR. OVER 555,000 PRODUCTS
(562) 692-5911
(562) 695-2323 (fax)
la.sales@mcmaster.com
Text 75930

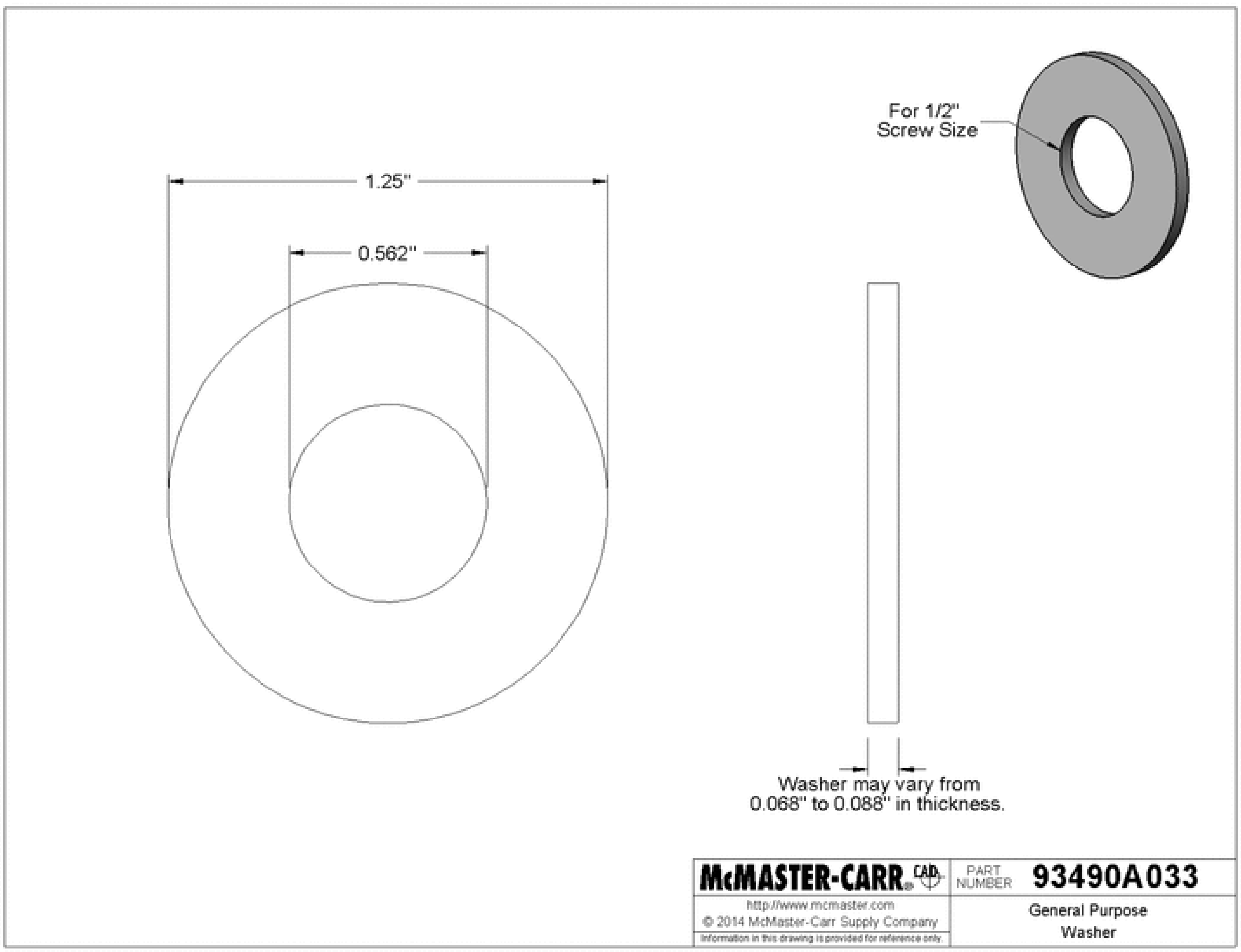
Bronze Flat Washer

1/2" Screw Size, 0.562" ID, 1.250" OD

In stock
$12.65 per pack of 10
93490A033

Screw Size	1/2"
ID	0.56"
OD	1 1/4"
Thickness, Minimum -Maximum	0.06"-0.09"
RoHS	Compliant

Silicon bronze washers are more resistant to corrosion caused by salt water and gases than brass washers. They're also nonmagnetic and not rated for hardness.



The information in this 3-D model is provided for reference only.

McMASTER-CARR. OVER 555,000 PRODUCTS
(562) 692-5911
(562) 695-2323 (fax)
la.sales@mcmaster.com
Text 75930

Compressible Buna-N Flange Gasket
for 6 Pipe Size, Class 150, Ring, 1/8" Thick

In stock
$3.64 Each
8516T22

For Pipe Size	6"
ID	6 5/8"
OD	8 3/4"
Additional Specifications	Buna-N Ring Flange Gaskets without Holes – 1/8" Thick ANSI gaskets sized to fit Class 150 flanges
RoHS	Compliant

All gaskets resist salts, water, and deformation after compression. Meet ASTM D2000. Ring flange gaskets cover the section of the flange just inside the bolt holes.

Ideal for oil applications. Also resists alkalies and detergents. Temperature range is –20° to 170° F. Maximum pressure is 1,000 psi. Durometer hardness is A60. Color is black.

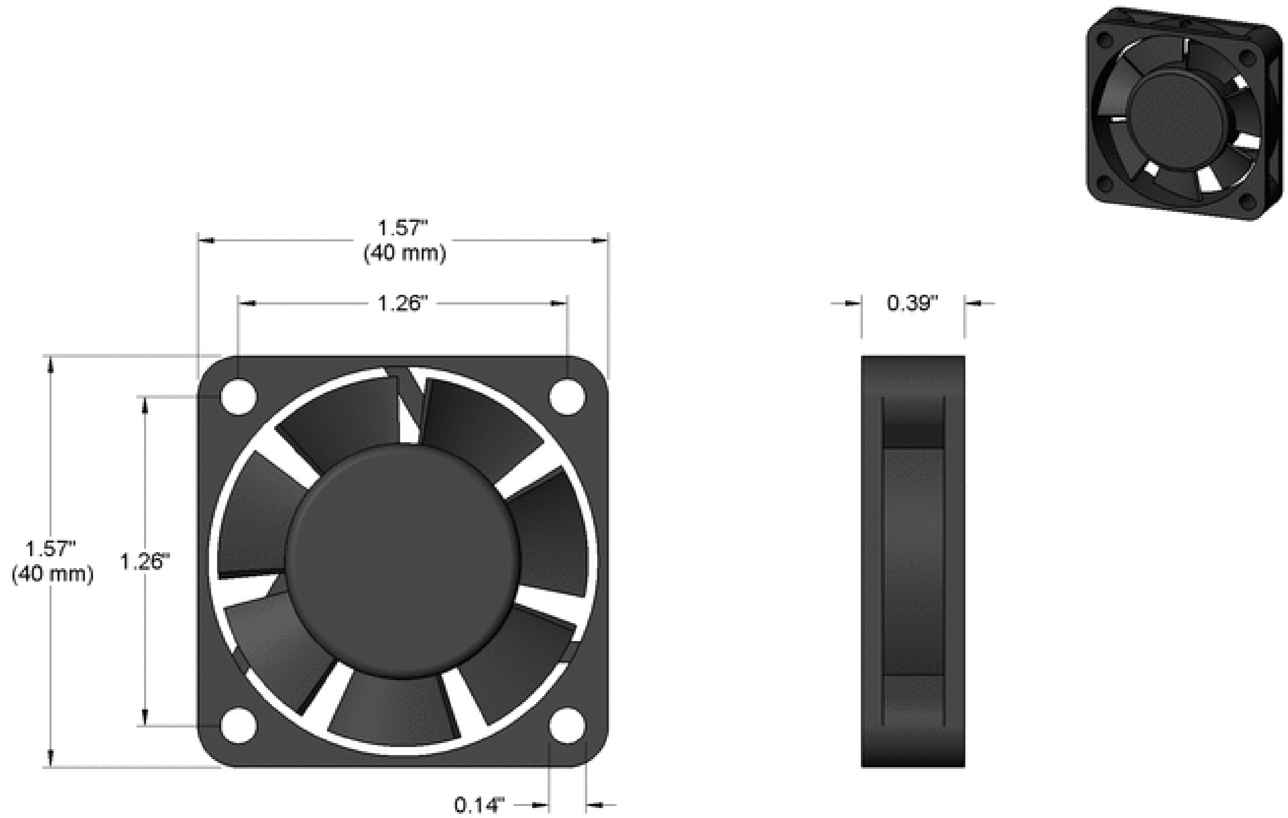
DC Equipment Cooling Fan

1.57" Square x 0.39" Depth, 3 CFM, 12 VDC

In stock
$17.25 Each
1939K17

Size	1.57" (40 mm)
Depth	0.39"
Connections	Wire Leads
Airflow	3 cfm
Volume	16 dB
Mounting Holes	0.14"
Material	
Frame	Plastic
Blade	Plastic
Amps	0.05
Additional Specifications	12 Volt DC Fans Square

Quiet and compact, these fans are the most popular choice for cooling heat-sensitive equipment. They're also known as muffin fans. All have UL recognized components and are CSA certified. Fasteners not included. For fan guards, filters, and thermostats, see Equipment-Cooling Fan Accessories.



McMASTER-CARR® CAD	PART NUMBER	1939K17
http://www.mcmaster.com © 2012 McMaster-Carr Supply Company Information in this drawing is provided for reference only.	Equipment-Cooling Fan	

The information in this 3-D model is provided for reference only.

(562) 692-5911
(562) 695-2323 (fax)
la.sales@mcmaster.com
Text 75930

General Purpose Nylon Hook and Loop

1/2" Width x 5' Length, Adhesive Back

In stock
$6.07 per length of 5
9273K11

Width	1/2"
Color	Black
Additional Specifications	Hook and Loop—Adhesive Back 5-ft. Lengths MSDS
RoHS	Compliant

A combination of moisture and chemical resistance makes this hook and loop the choice for most applications. Made of nylon. Adhesive back has acrylic adhesive.

Hook and loop temperature range is –20° to 200° F.

(562) 692-5911
(562) 695-2323 (fax)
la.sales@mcmaster.com
Text 75930

Medium-Pressure Brass Threaded Pipe Fitting

1/2 Pipe Size, Hollow Hex-Head Plug

In stock
$2.81 Each
50785K24

Pipe Size	1/2
Maximum Pressure	1,000 psi @ 72° F
Additional Specifications	Hex-Head Plugs Hollow Use with air, water, oil, natural gas
RoHS	Not Compliant

A great choice when you want brass fittings that can handle moderate pressure, but you don't want to break the bank.

Connections: NPT.

Performix BRAND

*Estimate
N/A-Not Applicable
N/R-Not Restricted
N/E-Not Established

MATERIAL SAFETY DATA SHEET

NOTE: BLANK SPACES ARE NOT PERMITTED. IF ANY ITEM IS NOT APPLICABLE, THE SPACE MUST BE MARKED TO INDICATE THAT.

IDENTITY (As shown on Label or package) **14.5 Fluid Ounce PLASTI DIP**	**PART NO. IF APPLICABLE**

SECTION I

MANUFACTURER'S NAME Plasti Dip International, Inc.	**EMERGENCY PHONE No.** 1-800-424-9300
ADDRESS (NUMBER, STREET, CITY, STATE AND ZIP CODE) 3920 Pheasant Ridge Drive	**REVISION #**
Blaine, MN 55449	**MANUFACTURER'S PHONE No. FOR INFORMATION** 1-763-785-2156
	DATE MSDS WAS PREPARED December 29, 2011

SECTION II - HAZARDOUS INGREDIENTS INFORMATION. All Health Hazards which comprise 1% or greater of the composition and all carcinogens if 0.1% of the composition or greater.

HAZARDOUS COMPONENTS CHEMICAL and IDENTITY AND COMMON NAME (S)	% Wt. (OPTIONAL)	CAS NO.	OSHA PEL	ACGIH TLV	OTHER LIMITS RECOMMEND
VM&P Naphtha	32-38	64742-89-8	300 ppm	300 ppm	None
Hexane	16-19	110-54-3	500 ppm	50 ppm	None
Toluene	13-16	108-88-3	200 ppm	50 ppm	None
Methyl Ethyl Ketone	3-7	78-93-3	200 ppm	200 ppm	None
Resins	25-27	N/A	N/A	N/A	None

SECTION III - PHYSICAL / CHEMICAL CHARACTERISTICS

BOILING POINT 149 - 285°F	**SPECIFIC GRAVITY** (H_20 =1) 0.79 - 0.83	**APPROXIMATE WEIGHT PER GALLON (LBS)** 6.60-690
VAPOR PRESSURE 125 mmHg @ 20°C	**VAPOR DENSITY (AIR = 1)** Heavier than air	**EVAPORATION RATE (BUTYL ACETATE =1)** > 1.0
SOLUBILITY IN WATER Insoluble	**% VOLATILE** 72-75% **VOC LBS./GAL** 4.8-5.2	**OTHER (IF ANY)** None

APPEARANCE AND ODOR Various Colors, Honey Like Substance - Characteristic Odor

SECTION IV-FIRE AND EXPLOSION HAZARD DATA

FLASH POINT -10.0°F **(METHOD USED)** TCC	**FLAMMABLE LIMITS** 0.9 – 11.5	**LEL** 0.9	**UEL** 11.5

EXTINGUISHING MEDIA Carbon Dioxide, Dry Chemical, or Foam

SPECIAL FIRE FIGHTING PROCEDURES Self contained breathing apparatus with a full face piece, operated in pressure demand or other positive pressure mode.

UNUSUAL FIRE AND EXPLOSION HAZARDS This material is flammable and may be ignited by heat, sparks, flame or static electricity.

HAZARDOUS PRODUCTS FORMED BY FIRE OR THERMAL DECOMPOSITION Carbon Dioxide and/or Carbon Monoxide.

EXPLOSIVE LIMITS (% BY VOLUME IN AIR) 0.9 - 11.5

SECTION V - OPTIONAL HAZARD RATINGS IDENTIFICATION

HAZARD RATING **4-EXTREME** **3-HIGH** **2-MODERATE** **1-SLIGHT** **0-INSIGNIFICANT** ****SEE SECTION IV**	**National Fire Protection Association (NFPA)** **FIRE** 3 **REACTIVITY** 0 **.** **HEALTH** 2 **SPECIAL HAZARDS** None .

PLASTI DIP.com

SECTION VI - REACTIVITY AND STABILITY DATA

STABILITY	**UNSTABLE**	**STABLE** X	**CONDITIONS TO AVOID** None

INCOMPATIBILITY (Materials to Avoid) Strong acids, bases, oxidizing agents, selected amines with alkali metals and halogens.

HAZARDOUS DECOMPOSITION OR BY PRODUCTS Carbon Monoxide, Carbon Dioxide

HAZARDOUS POLYMERIZATION	**MAY OCCUR** **WILL NOT OCCUR** X	**CONDITIONS TO AVOID** None

SECTION VII - HEALTH HAZARD DATA

ROUTES OF ENTRY **INHALATION?** Yes **SKIN?** Yes **INGESTION?** Yes **EYES?** Yes

HEALTH HAZARDS	**ACUTE**	X	See Signs and Symptoms of Exposure below.
	CHRONIC	X	Brain and Nervous System Damage (Referred to as solvents or painters syndrome). Drying or Cracking skin.

CARCINOGENICITY: No

SIGNS AND SYMPTOMS OF EXPOSURE
Headache, Dizziness, Drowsiness, Fatigue, Irregular Heartbeat, Skin and Eye Irritation.
Target Organs: CNS, CVS, PNS, Liver, Kidneys, Lungs, Respiratory System, Skin.

MEDICAL CONDITIONS GENERALLY AGGRAVATED BY EXPOSURE Pre-existing Heart, Liver, Kidney and Lung disorders.

EMERGENCY AND FIRST AID PROCEDURES
Ingestion: Contact Physician or Poison Control Immediately.
Inhalation: Remove to fresh air. Administer Oxygen or Artificial Respiration if Necessary.
Eye Contact: Flush with large amounts of water. If irritation persists, contact Physician.
Skin: Wash with soap and water.

SECTION VIII - PRECAUTIONS FOR SAFE HANDLING AND USE

STEPS TO BE TAKEN IN CASE MATERIAL IS RELEASED OR SPILLED
Wipe up with floor absorbent. Transfer to hood. Prevent run-off to sewers.
Eliminate all sources of ignition. Ventilate to maintain exposure below PEL's. Use sand or other material to dam or contain spills. If large spill, notify appropriate state and local agencies.

WASTE DISPOSAL METHODS Dispose of product in accordance with local, county, state and federal regulations.

PRECAUTIONS TO BE TAKEN IN HANDLING AND STORAGE
Keep away from sparks, flame and heat sources. Do not store above 120°F and use adequate ventilation.
Avoid inhalation of vapors and contact with liquid product. Use good personal hygiene.

OTHER PRECAUTIONS
Keep Container Closed When Not In Use. Containers should be disposed of in an environmentally safe manner in accordance with Governmental Regulations.

SECTION IX - CONTROL MEASURES

RESPIRATORY PROTECTION (SPECIFY TYPE) Depending on the Airborne concentration, use a Respirator with appropriate NIOSH approved cartridge or supplied air equipment.		**PROTECTIVE GLOVES** Impervious
VENTILATION	**LOCAL EXHAUST** Supplemental (if needed)	**SPECIAL** None
	MECHANICAL (GENERAL) To maintain exposure below PEL's	**OTHER** None
EYE PROTECTION Chemical splash goggles or approved eye protection.		**OTHER PROTECTIVE CLOTHING OR EQUIPMENT** Impervious Clothing/Boots as needed.

WORK HYGIENIC PRACTICES Wash thoroughly after handling.

SECTION X - TRANSPORTATION INFORMATION (Optional)

D.O.T. PROPER SHIPPING NAME Coating Solution (UN 1139)	**D.O.T. HAZARD CLASS** 3, PG II
IATA PROPER SHIPPING NAME Coating Solution (UN 1139)	**IATA HAZARD CLASS** 3, PG II

DOT EXCEPTION See DOT 49CFR 172.102, New Special Provision 149 AND DOT 49CFR 171.8 FOR **CONSUMER COMMODITY, ORM-D**

SECTION XI - 313 SUPPLIER NOTIFICATION

THIS PRODUCT CONTAINS THE FOLLOWING CHEMICALS SUBJECT TO REPORTING REQUIREMENTS OF SECTION 313 OF THE EMERGENCY PLANNING AND COMMUNITY RIGHT -TO-KNOW ACT OF 1986, 40 CFR 372, (see table on page 1 for CAS # and percent by weight).
Hexane, Toluene and Methyl Ethyl Ketone

WARNING: THIS PRODUCT CONTAINS A CHEMICAL KNOWN TO THE STATE OF CALIFORNIA TO CAUSE CANCER AND BIRTH DEFECTS, OR OTHER REPRODUCTIVE HARM.

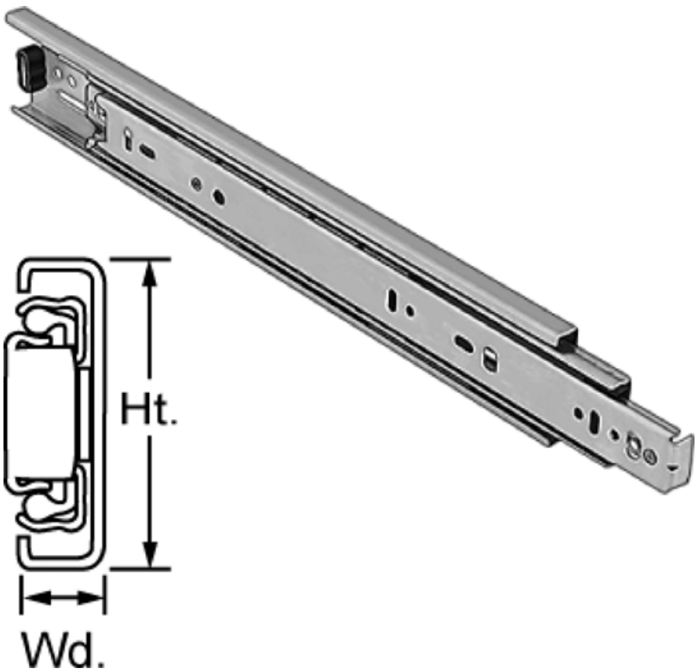
Prepared By: Mark Kenow

OVER 555,000 PRODUCTS
(562) 692-5911
(562) 695-2323 (fax)
la.sales@mcmaster.com
Text 75930

Side-Mount Drawer Slide

Full Extension, 10" Closed Length

In stock
$16.50 per pair
11435A11



Closed Length	10"
Extension	9 9/16"
Width	1/2"
Height	1 13/16"
Feature	None
Load Rating	50 lbs./pair
Drawer Release	Lever
Additional Specifications	Full Extension Standard—Steel
RoHS	Compliant

The most commonly used slides, these are sized to fit most drawer openings in desks and cabinets. Standard, stacked, low profile, and high profile slides have ball bearings for smooth, quiet operation. Steel slides include mounting screws.

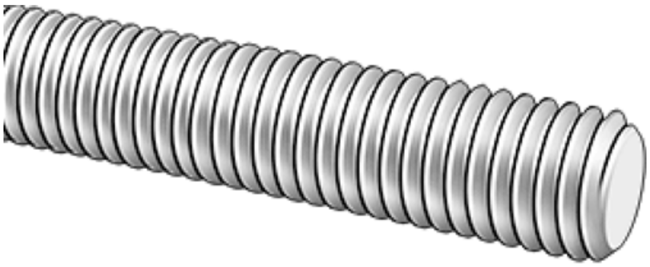
Full Extension—Allow access to items in the back of drawers.

McMASTER-CARR® OVER 555,000 PRODUCTS
(562) 692-5911
(562) 695-2323 (fax)
la.sales@mcmaster.com
Text 75930

Type 18-8 Stainless Steel Fully Threaded Rod

2-56 Thread, 3 Feet Long

In stock
$2.90 per pack of 1
98847A003



Material	18-8 Stainless Steel
Thread Size	2-56
Thread Direction	Right Hand
Length	3 ft.
RoHS	Compliant

Offering excellent resistance to chemicals and solvents, stainless steel studs and rods stand up to harsh environments. They may be mildly magnetic.

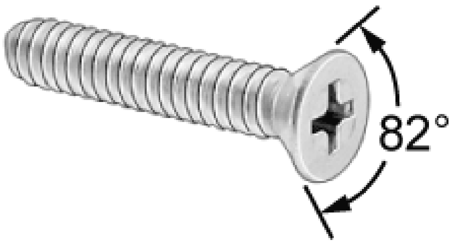
18-8 stainless steel studs and rods are corrosion resistant. They have a minimum tensile strength of 70,000 psi. Inch sizes have a have a minimum Rockwell hardness of B70. Thread fit is Class 1A.

OVER 555,000 PRODUCTS
(562) 692-5911
(562) 695-2323 (fax)
la.sales@mcmaster.com
Text 75930

Type 316 Stainless Steel Flat Head Phillips Machine Screw

8-32 Thread, 1" Length

In stock
$5.59 per pack of 25
91500A199



Length	1"
Additional Specifications	Type 316 Stainless Steel 8-32—#2 Drive
RoHS	Compliant

Screws have a standard 82° bevel under the head. Sizes noted below have an undercut head to allow more threading. Screws up to 2" long are fully threaded; those longer than 2" have at least 1 1/2" of thread. Length is measured from the top of the head.

(562) 692-5911
(562) 695-2323 (fax)
la.sales@mcmaster.com
Text 75930

Type 316 Stainless Steel Hex Head Cap Screw

1/2"-13 Thread, 14" Long

In stock
$15.16 per pack of 1
92186A754

Material	Type 316 Stainless Steel
Thread Size	1/2"-13
Head Width	3/4"
Head Height	5/16"
Screw Size	1/2" (0.500")
Length	14"
Thread Length	1 1/2" to 6"
RoHS	Compliant

Made from stainless steel, these screws are corrosion resistant in harsh environments. They may be mildly magnetic. Length is measured from under the head.

Inch screws have a minimum tensile strength of 70,000 psi, a Class 2A thread fit, and a minimum Rockwell hardness of B70. Dimensions meet ASME B18.2.1. For screws in sizes 4-40 to 12-24, see hex-head machine screws.

Type 316 Stainless Steel—Screws are more corrosion resistant than 18-8 stainless steel screws and can be used in marine environments.

(562) 692-5911
(562) 695-2323 (fax)
la.sales@mcmaster.com
Text 75930

Type 316 Stainless Steel Hex Nut

1/2"-13 Thread Size, 3/4" Wide, 7/16" High

In stock
$4.98 per pack of 10
94804A340

Material	Type 316 Stainless Steel
Thread Size	1/2"-13
Width	3/4"
Height	7/16"
Additional Specifications	Plain
RoHS	Compliant

Also known as full or finished nuts, these common nuts are also our most popular. They typically come in sizes 1/4" and larger and have a Class 2B thread fit. Sizes 1 1/2" and smaller have dimensions that meet ANSI/ASME B18.2.2.

McMASTER-CARR® OVER 555,000 PRODUCTS
(562) 692-5911
(562) 695-2323 (fax)
la.sales@mcmaster.com
Text 75930

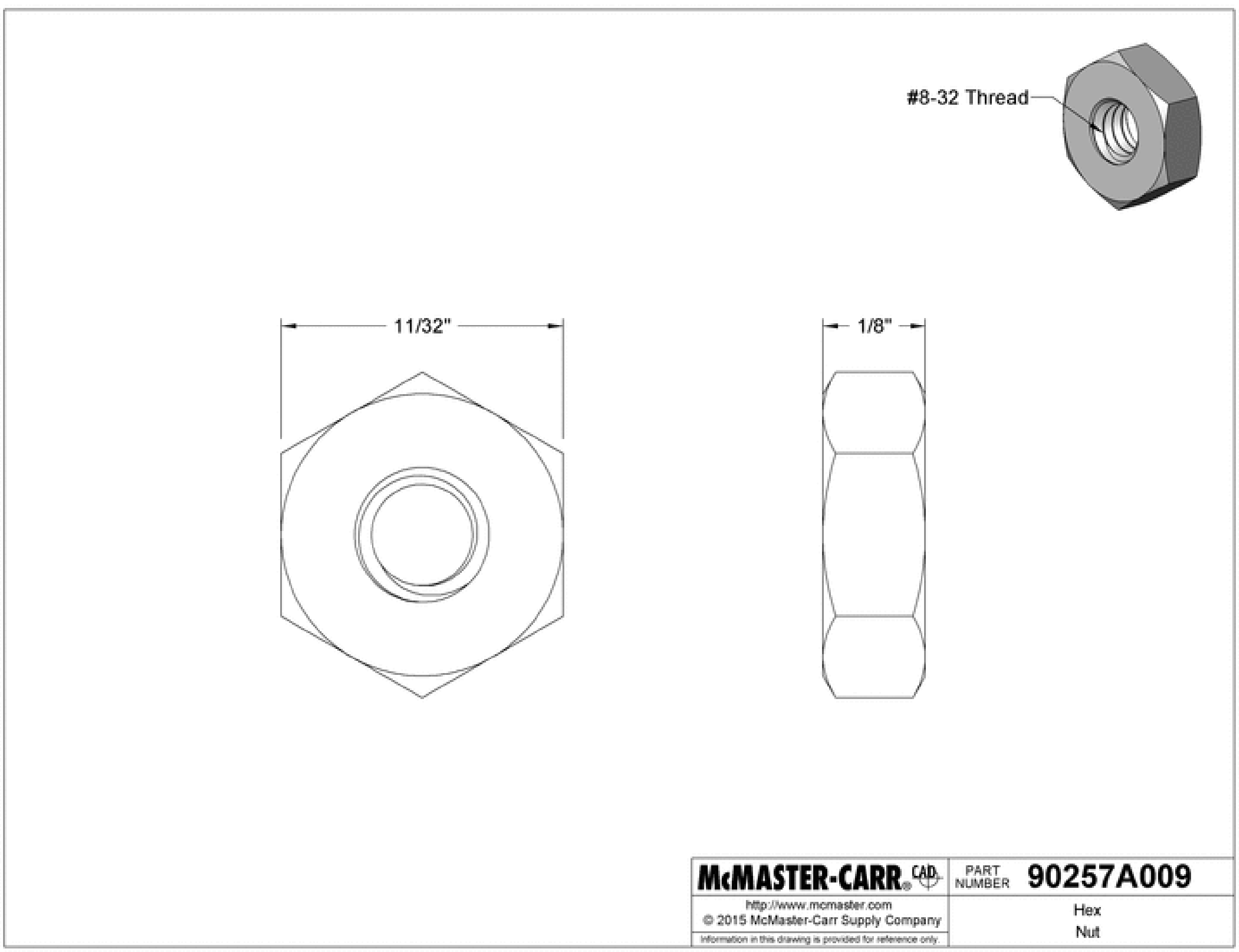
Type 316 Stainless Steel Hex Nut

8-32 Thread Size, 11/32" Wide, 1/8" High

In stock
$7.98 per pack of 100
90257A009

Thread Size	8-32
Width	11/32"
Height	1/8"
Additional Specifications	Type 316 Stainless Steel
RoHS	Compliant

Often used on machine screws and threaded rods, and great for tight spaces. Top of nut is flat and has chamfered corners; bottom may be flat or chamfered. Nuts have a Class 2B thread fit and dimensions that meet ANSI/ASME B18.6.3.



M100 Brushless Motor

$65.00

The M100 is a rugged brushless motor for use in the ocean and in extreme environments. It has the same internal components as the T100 Thruster.

Available for pre-order now, shipping by February 2015.

1 + - **Add to cart**

SKU: M100-MOTOR-R1. Category: Thrusters (https://www.bluerobotics.com/product-category/thrusters/).

(https://www.bluerobotics.com/wp-content/uploads/2014/11/top-2.png)

(https://www.bluerobotics.com/wp-content/uploads/2014/11/perspective-1.png)

Description **Resources**

Product Description

The M100 Brushless Motor is a simple motor version of the T100 Thruster.

It uses the same internal components but doesn't have the plastic nozzle or other components. You can use it for many underwater applications like:

- Waterproof brushless camera gimbals
- Underwater actuators
- Build your own thruster
- Small ROVs that don't need a full size thruster

The M100 is also used as the vertical thruster on the **OpenROV Compatible Kit**.

Key Features

- **Efficient** brushless electric motor
- **Standard mounting holes** spaced 19mm and 16mm apart
- **Rugged design** for use in extreme environments
- Operates under **extreme pressure** and **depth**
- **High-performance plastic bearings** that don't corrode
- **Solder pads** to connect and seal wires

Specifications

	Parameter	
Operating Voltage	12 Volts	
Max Current	12.5 Amps	
Max Power	135 Watts	
Length	2.12 in	54 mm
Diameter	1.26 in	32 mm
Shaft Diameter	0.157 in	4.0 mm
Mounting Hole Spacing	0.75 in & 0.63 in	19 mm & 16 mm
Mounting Holes	M3x0.5 Screws	
Weight in Air	lb	g
Weight in Water	lb	g

You may also like…

(https://www.bluerobotics.com/wp-content/uploads/2014/08/perspective-photo-1-e1407701672535.png)

T100 Thruster

$109.00–$169.00

The T100 is world's first affordable underwater thruster designed for marine robotics and the future of ocean exploration. **Available for pre-order now!**

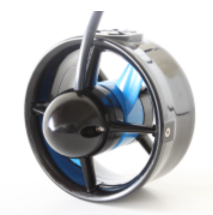
Estimated Delivery for *New* Pre-Orders:
March 2015 for T100 and Basic ESC
April 2015 for T100 w/ BlueESC

Speed Controller None
Clear selection

$109.00

1 + - **Pre-Order Now**

SKU: T100-THRUSTER-R1. Category: Thrusters (https://www.bluerobotics.com/product-category/thrusters/).



(https://www.bluerobotics.com/wp-content/uploads/2014/08/perspective-photo-1-e1407701672535.png)

(https://www.bluerobotics.com/wp-content/uploads/2014/08/[illegible]ruler.jpg)

(https://www.bluerobotics.com/wp-content/uploads/2014/08/serial-num-1.png)

(https://www.bluerobotics.com/wp-content/uploads/2014/08/package-3.png)

(https://www.bluerobotics.com/wp-content/uploads/2014/08/fin-2.png)

(https://www.bluerobotics.com/wp-content/uploads/2014/08/bracket-1.png)

Product Description

The **T100 Thruster** is a patent-pending underwater thruster designed specifically for marine robotics. It's high performing with over 5 pounds of thrust and durable enough for use in the open ocean at great depths. A variety of mounting options, simple control, and a low price tag make it the perfect thruster to use on your marine robot.

The T100 Thruster was originally launched through our **Kickstarter Campaign** (https://www.kickstarter.com/projects/847478159/the-t100-a-game-changing-underwater-thruster)!

Who is the T100 For?

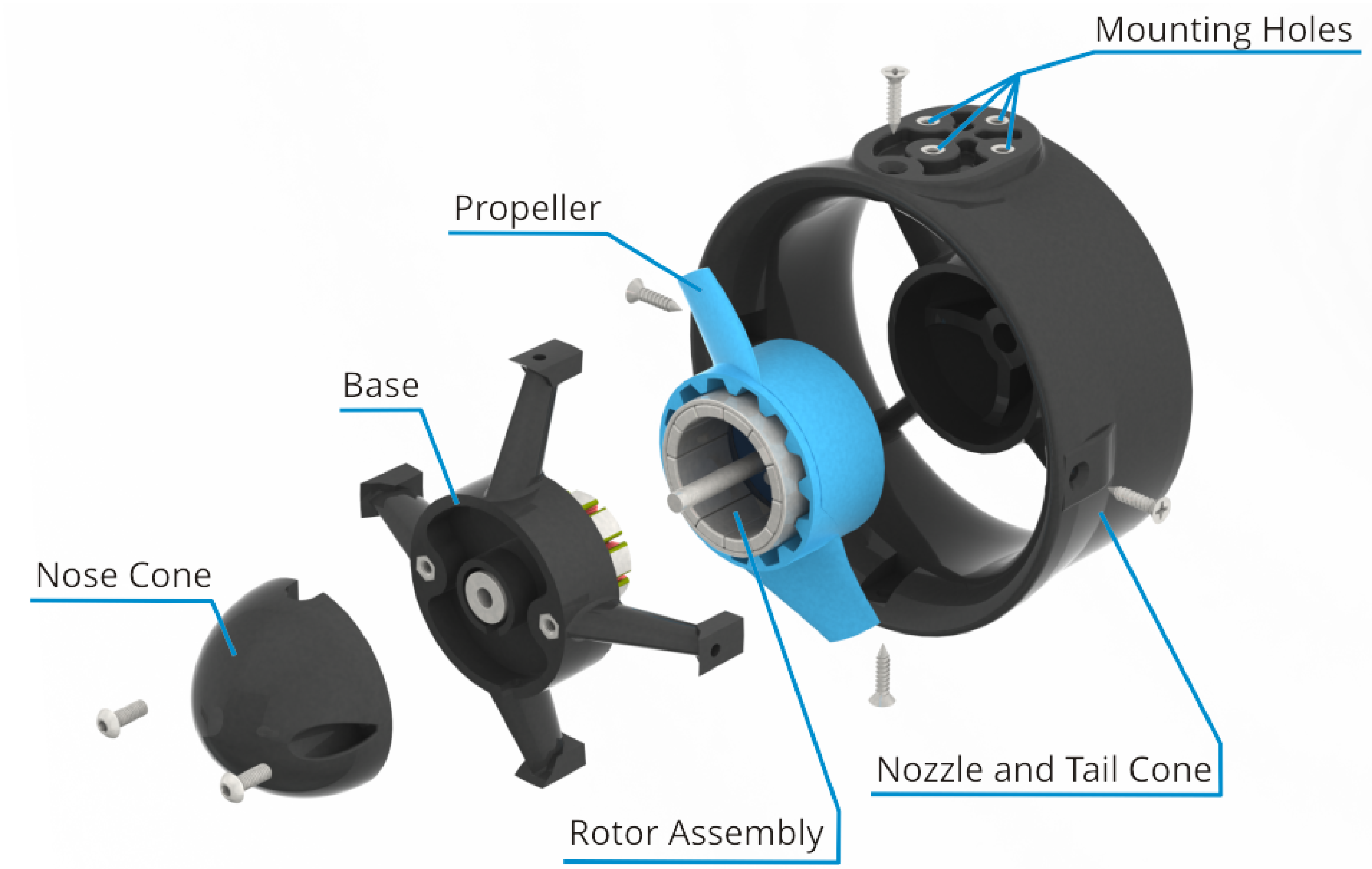
We designed the T100 for:

- **Makers and Hobbyists** who are interested in marine robotics but can't find capable, affordable hardware.
- **Students and Schools** who can use the T100 for educational projects or to compete in competitions such as the AUVSI RoboSub and RoboBoat competitions and the MATE ROV competition.
- **Professional Users** who want a high-quality thruster that is more capable than many of the high-end (expensive) alternatives.

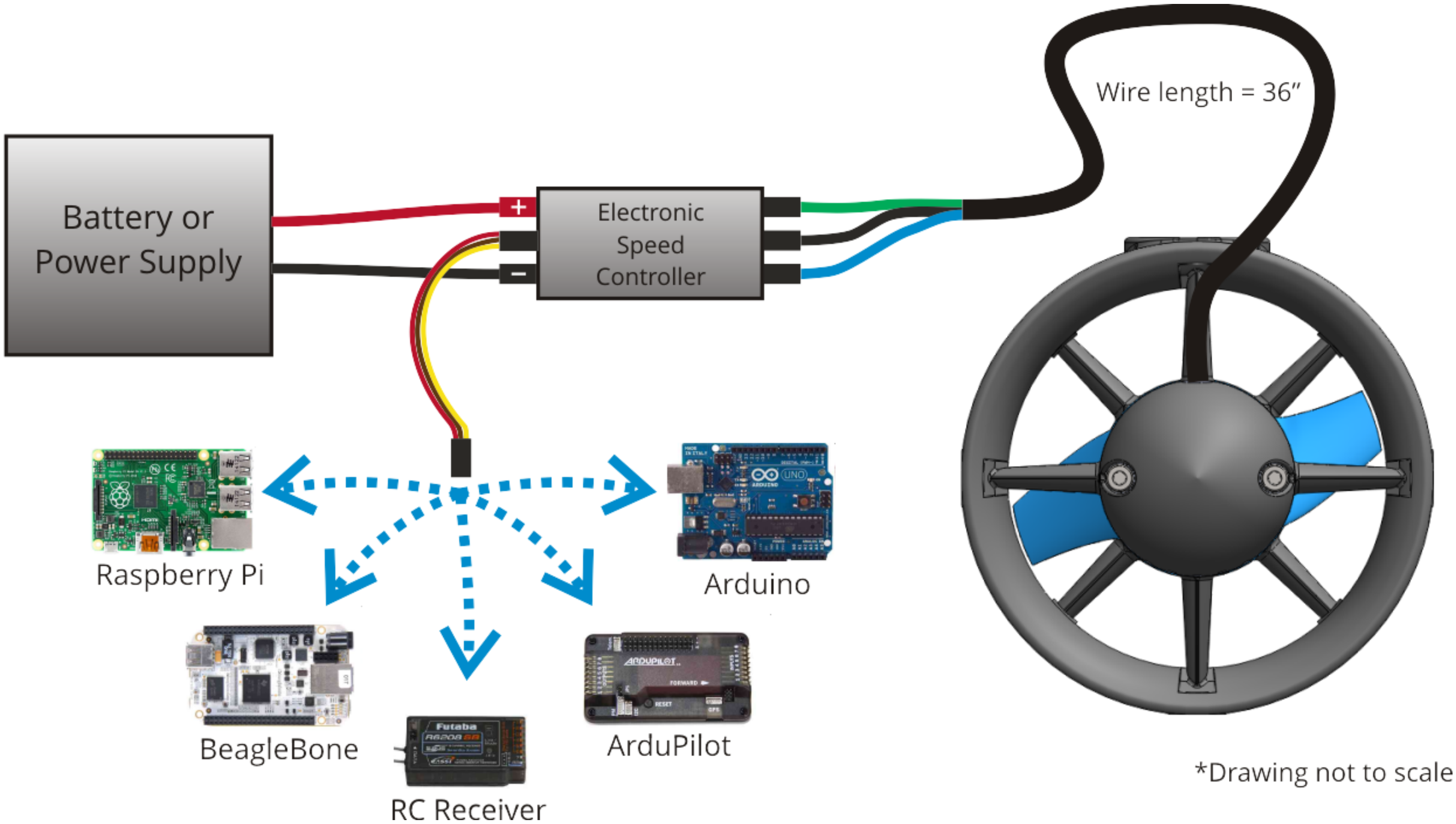
How It Works

The T100 is basically a brushless electric motor, just like you'd find on an RC airplane or a quadcopter drone. The big difference is that this motor is purpose-built for use in the ocean and was designed specifically for use on remotely operated underwater vehicles, autonomous underwater vehicles, and robotic surface vehicles. Of course you could also use it to propel your stand-up-paddleboard or cruise around while scuba diving!

The T100 is made of high-strength, UV resistant polycarbonate injection molded plastic. The core of the motor is sealed and protected with an epoxy coating and it uses high-performance plastic bearings in place of steel bearings that rust in saltwater. Everything that isn't plastic is either aluminum or high-quality stainless steel that doesn't corrode.



A specially designed propeller and nozzle provides efficient, powerful thrust while active water-cooling keeps the motor cool. Unlike other thrusters, our design doesn't have any air- or oil-filled cavities – water flows freely through all parts of the motor while it's running. That means it can go deep in the ocean and handle extreme pressures.



The thruster is easy to use: just connect the three motor wires to any brushless electronic speed controller (ESC) and you can control it with an RC radio or a microcontroller. It’s usable with Arduino, ArduPilot, Raspberry Pi, BeagleBone, and many other embedded platforms.

Key Features

- **Efficient** brushless electric motor
- **Compact design** that fits in any project
- **Strong** and **UV resistant** polycarbonate plastic
- No enclosed pressure cavities to handle **extreme pressures**
- **High-performance plastic bearings** that don’t corrode
- Comes with **clockwise** and **counter-clockwise propellers** to counter torque
- **Versatile mounting options** for use on almost anything
- **Rugged cable** with polyurethane pressure-extruded jacket

Specifications

	Parameter	
Max Thrust – Forward	5.2 lbf	2.36 kgf
Max Thrust – Reverse	4.0 lbf	1.82 kgf
Min Thrust	0.03 lbf	0.01 kgf

Operating Voltage	12 Volts	
Max Current	11.5 Amps	
Max Power	130 Watts	
Length	4.0 in	100 mm
Diameter	3.7 in	94 mm
Cable Length	36 in	915 mm
Mounting Hole Spacing	0.75 in	19 mm
Mounting Holes	M3x0.5 Screws	
Weight in Air (with 1m cable)	0.65 lb	295 g
Weight in Water (with 1m cable)	0.26 lb	120 g

You may also like…

Home (https://www.bluerobotics.com) / Store (https://www.bluerobotics.com/store/) / Parts (https://www.bluerobotics.com/product-category/parts/) / M100 Propeller

(https://www.bluerobotics.com/wp-content/uploads/2015/03/propeller-1.png)

(https://www.bluerobotics.com/wp-content/uploads/2015/03/directions-1.png)

(https://www.bluerobotics.com/wp-content/uploads/2015/03/with-m100-1.png)

M100 Propeller

$12.00

This 3D printed propeller is designed for the M100 motor and provides 3.2 lb of thrust! Counter-rotating directions available.

Rotation Direction Clockwise ▼

Clear selection

8 in stock

1 **Add to cart**

SKU: M100-P-PROPELLER-CW-REV-R1. Categories: Parts (https://www.bluerobotics.com/product-category/parts/), Thrusters (https://www.bluerobotics.com/product-category/thrusters/).

Description

Product Description

The propeller has a 68mm diameter and generates about 3.2 lb of forward thrust and 2.6 lb of reverse thrust. There are clockwise and counter-clockwise rotating versions. Thanks to the magic of 3D printing, we can build these in small batches without the need for expensive injection molded tooling. We are selling the propellers here for your convenience, but many of you have your own 3D printers and can print the propellers yourself. We are making the 3D files for the propeller open-source and licensed under CC-NC-SA 4.0 (https://creativecommons.org/licenses/by-nc-sa/4.0/).

Check out the “Resources” tab for performance charts.

Contents:

- 1 x Clockwise OR Counterclockwise Propeller
- 2 x M3 propeller screws

Resources:

The 3D files are available here for free (and also on GrabCAD (https://grabcad.com/library/3d-printable-propeller-for-the-m100-underwater-motor-1)). The files are licensed under CC-NC-SA 4.0 (https://creativecommons.org/licenses/by-nc-sa/4.0/).

- Clockwise Version: M100-P-PROPELLER-REV-CW-R1.STL (https://github.com/bluerobotics/bluerobotics.github.io/raw/master/thrusters/cad/M100-P-PROPELLER-REV-CW-R1.STL)
- Counter-clockwise Version: M100-P-PROPELLER-REV-CCW-R1.STL (https://github.com/bluerobotics/bluerobotics.github.io/raw/master/thrusters/cad/M100-P-PROPELLER-REV-CCW-R1.STL)

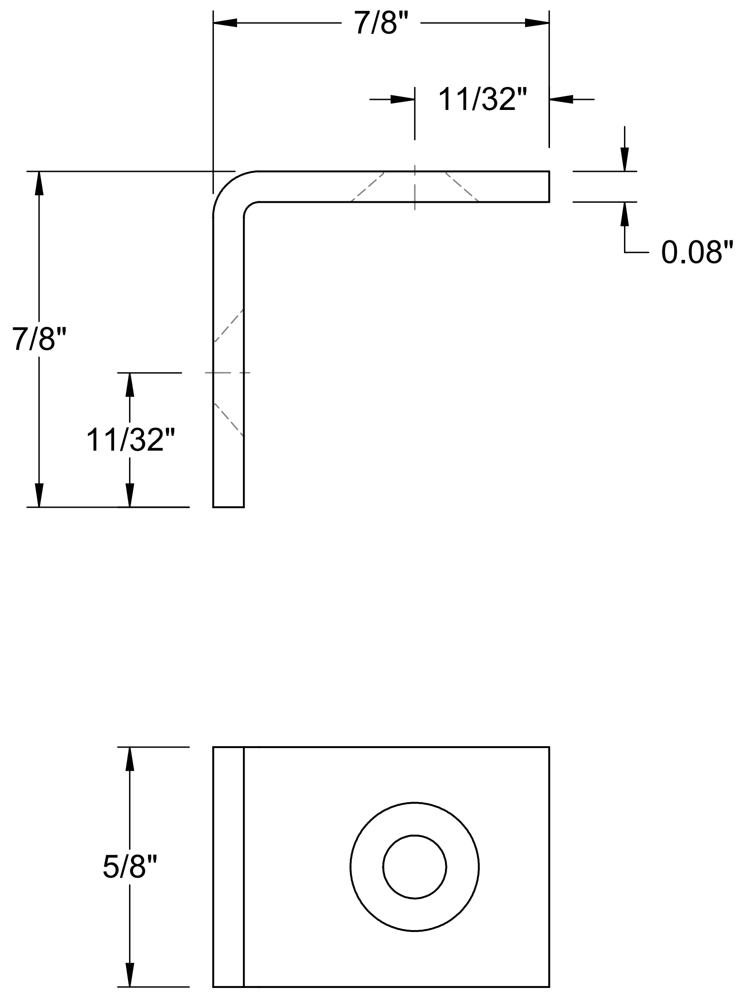
Specifications:

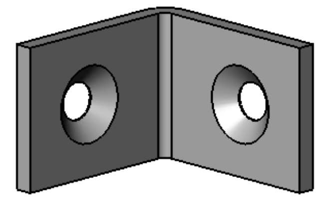
	Parameter	
Propeller Diameter	2.68 in	68 mm
Pitch (angle)	12.5 deg (at 75% radius)	
Pitch (travel)	1.4 in/rev	36 mm/rev

You may also like…

Appendix F: Mechanical Component Layout Drawings

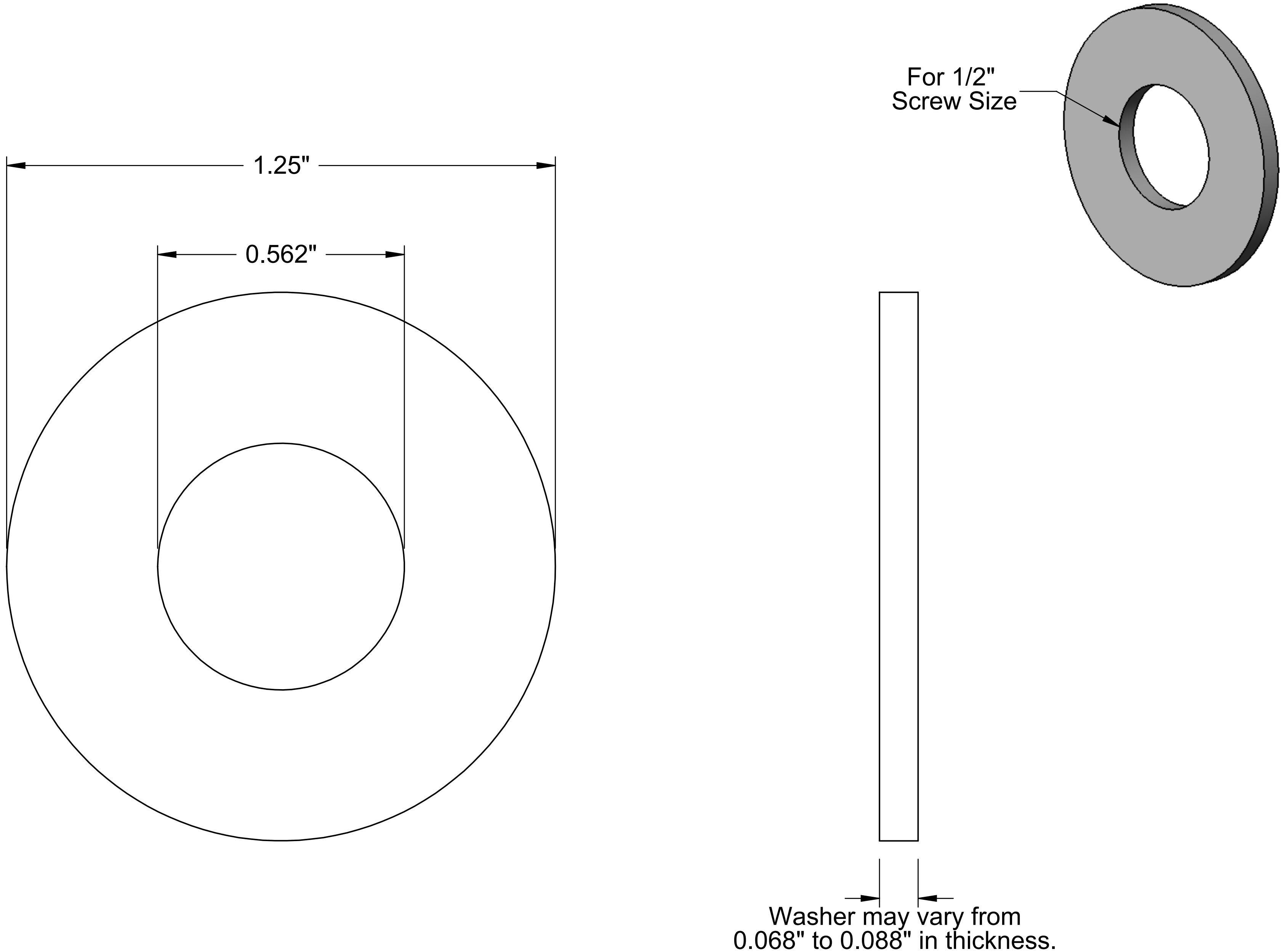
See attached. All dimensions are in inches unless otherwise stated. Tolerances are within 0.05” of dimensions.



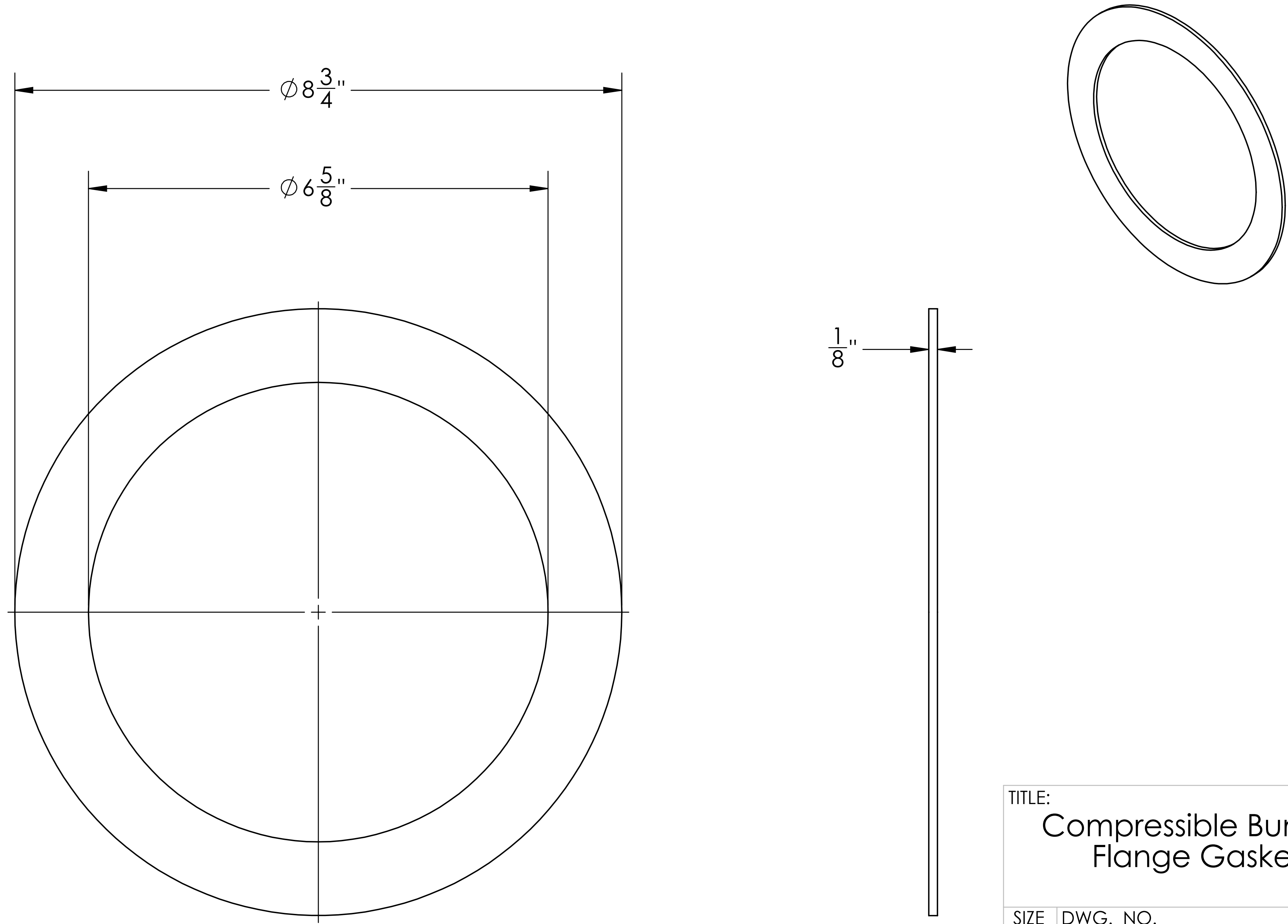


Bracket has 2 holes.
Bracket uses No. 8 screws.

McMASTER-CARR® CAD	PART NUMBER	1556A64
http://www.mcmaster.com		Bracket



McMASTER-CARR® CAD	PART NUMBER	93490A033	
http://www.mcmaster.com © 2014 McMaster-Carr Supply Company	General Purpose Washer		
Information in this drawing is provided for reference only.			

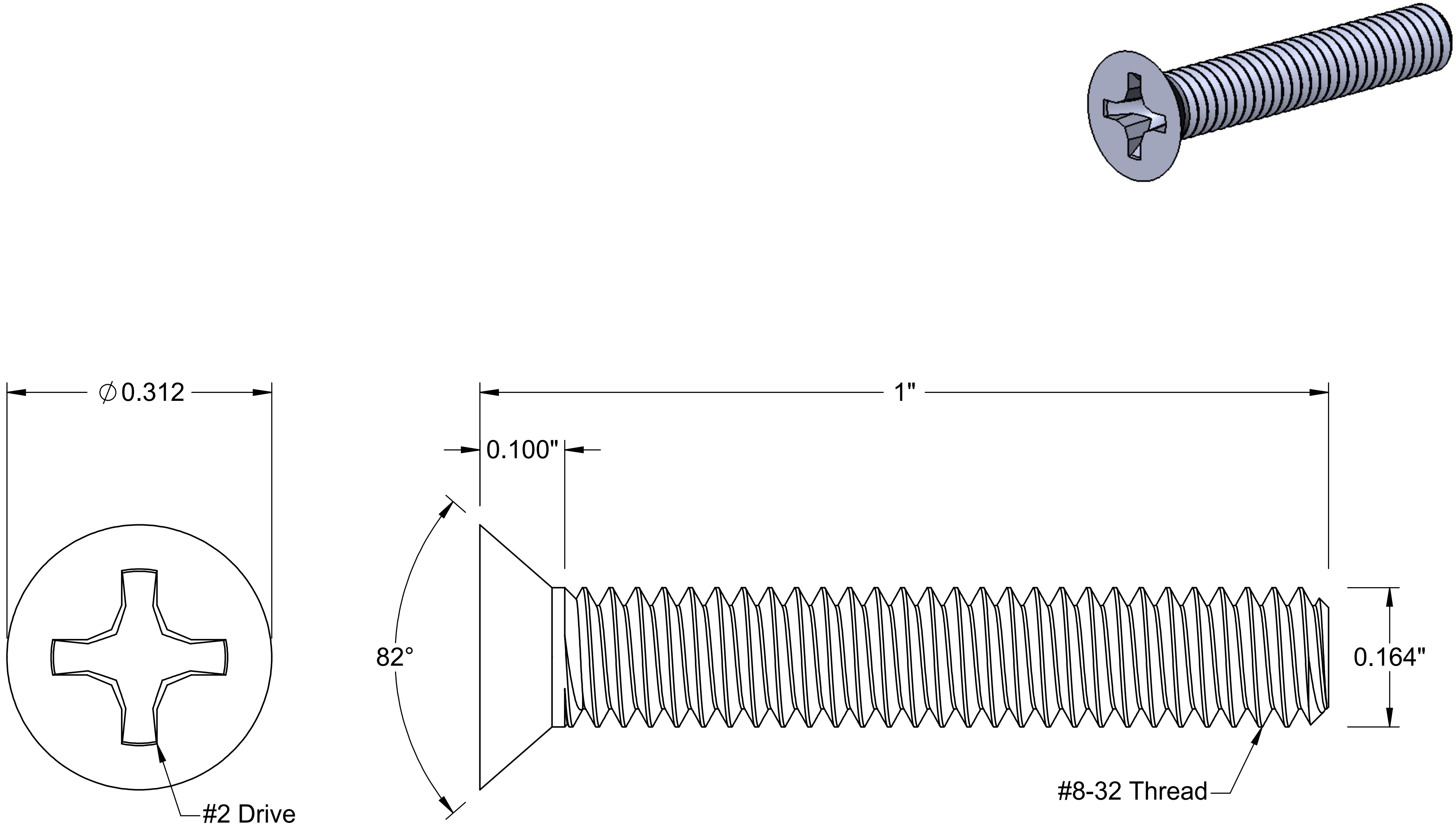


TITLE:

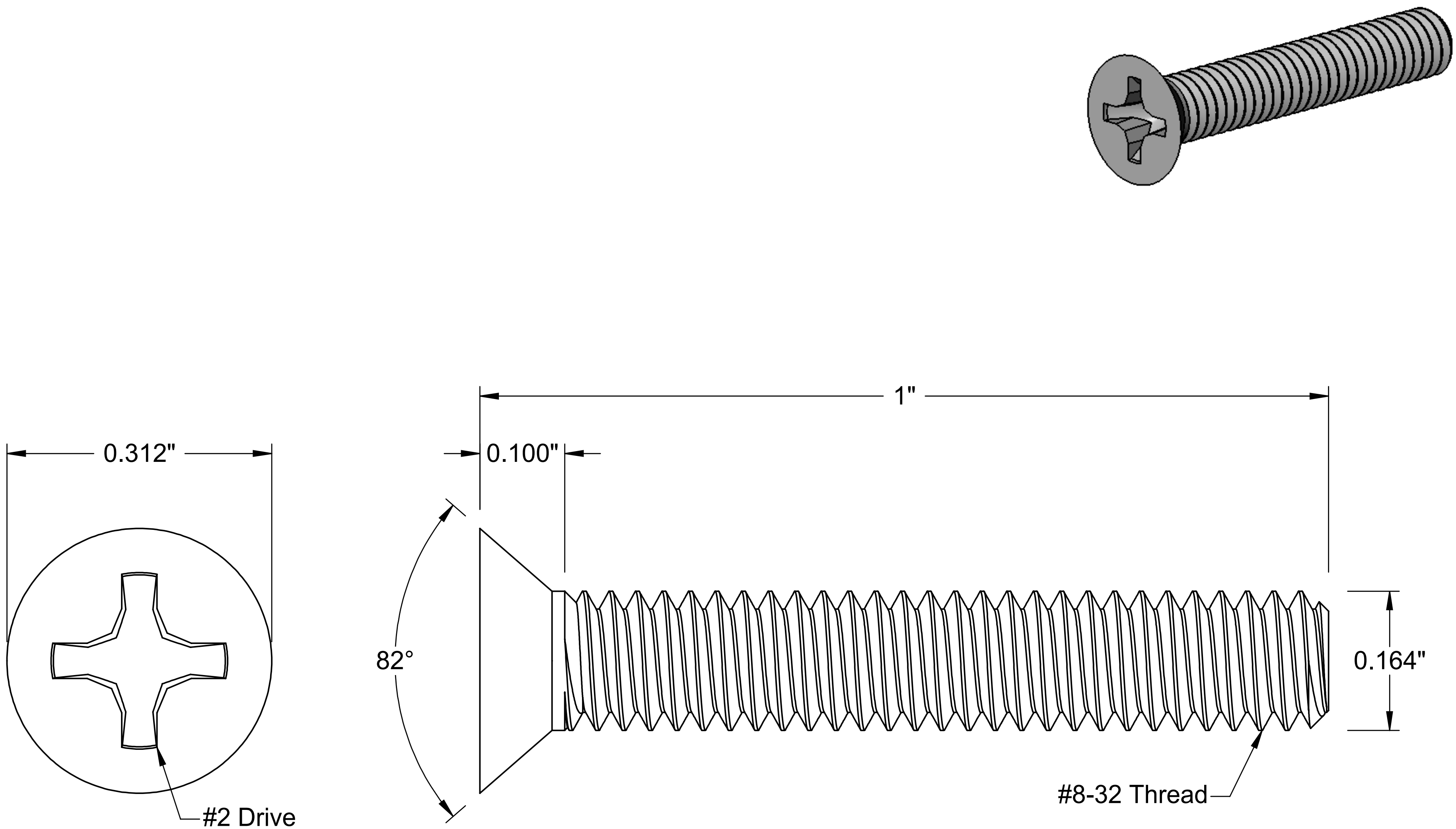
Compressible Buna-N
Flange Gasket

SIZE	DWG. NO.	REV
A	8516T22	1

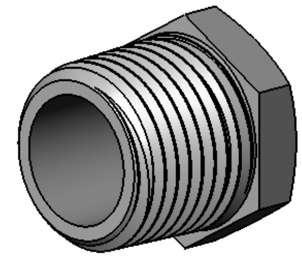
SCALE: 1:2 | WEIGHT: | SHEET 1 OF 1

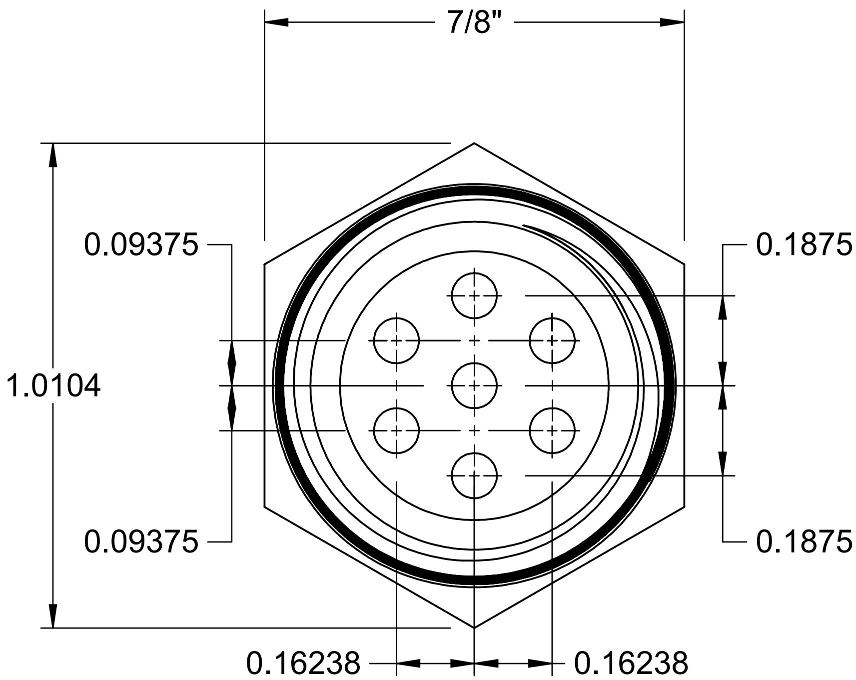


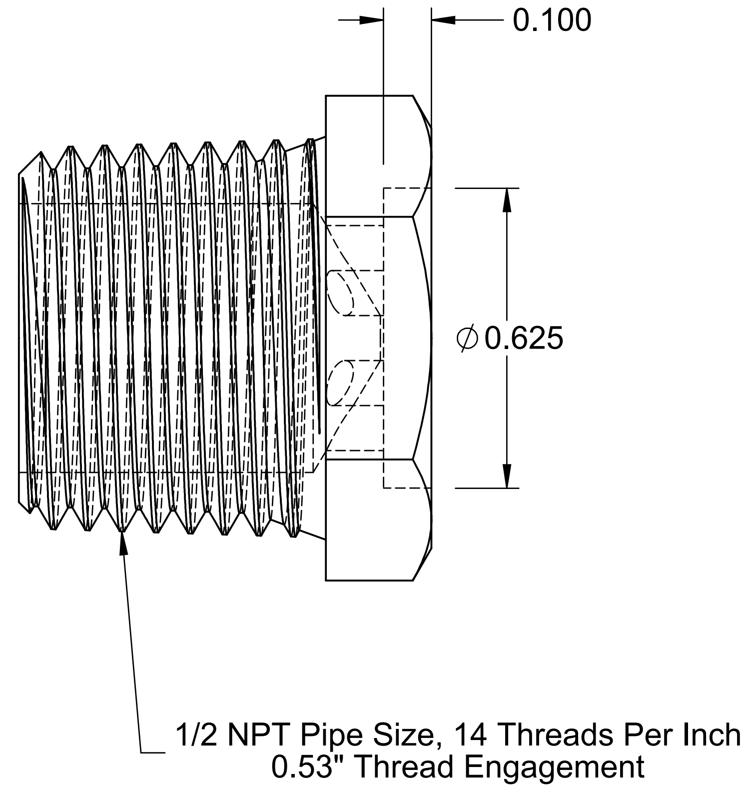
McMASTER-CARR® CAD		PART NUMBER	91500A199
http://www.mcmaster.com © 2012 McMaster-Carr Supply Company Information in this drawing is provided for reference only.		Flat Head Phillips Machine Screw	



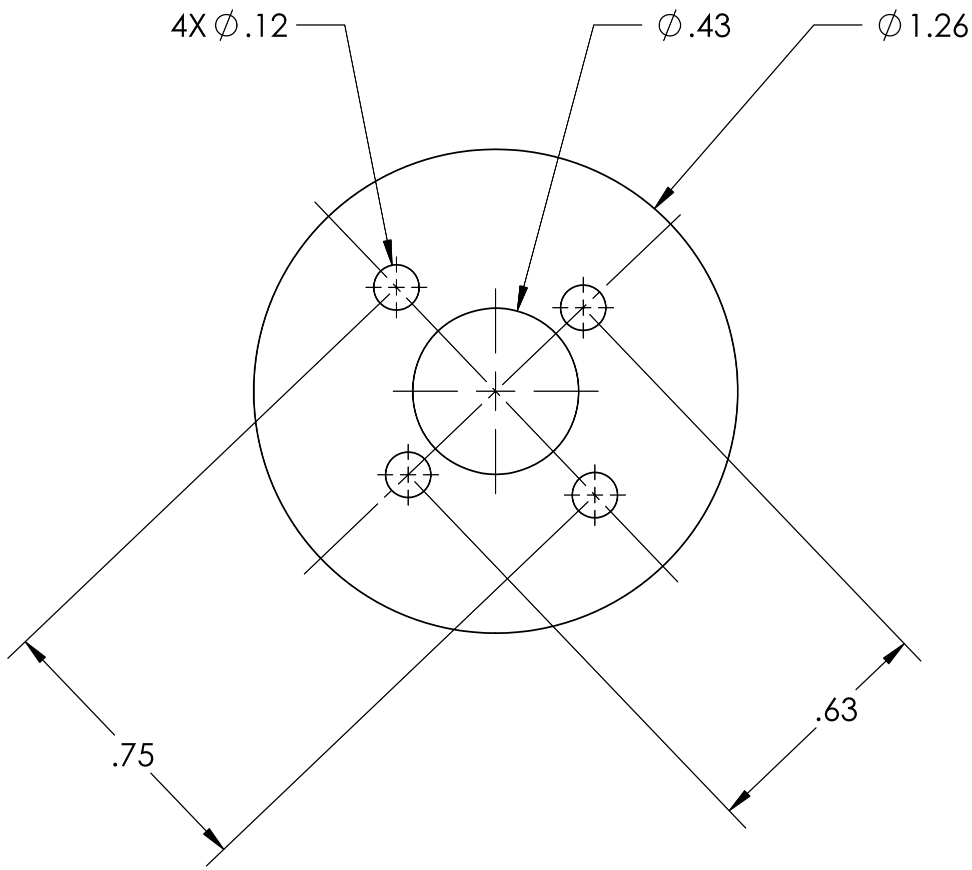
McMASTER-CARR® CAD	PART NUMBER	91500A199
http://www.mcmaster.com © 2012 McMaster-Carr Supply Company Information in this drawing is provided for reference only.	Flat Head Phillips Machine Screw	

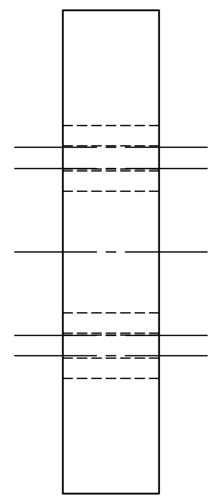


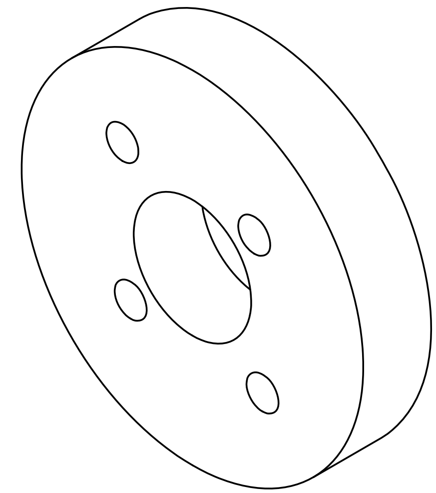




McMASTER-CARR® CAD	PART NUMBER	50785K24
http://www.mcmaster.com © 2014 McMaster-Carr Supply Company Information in this drawing is provided for reference only.	Brass Hollow Hex-Head Plug	



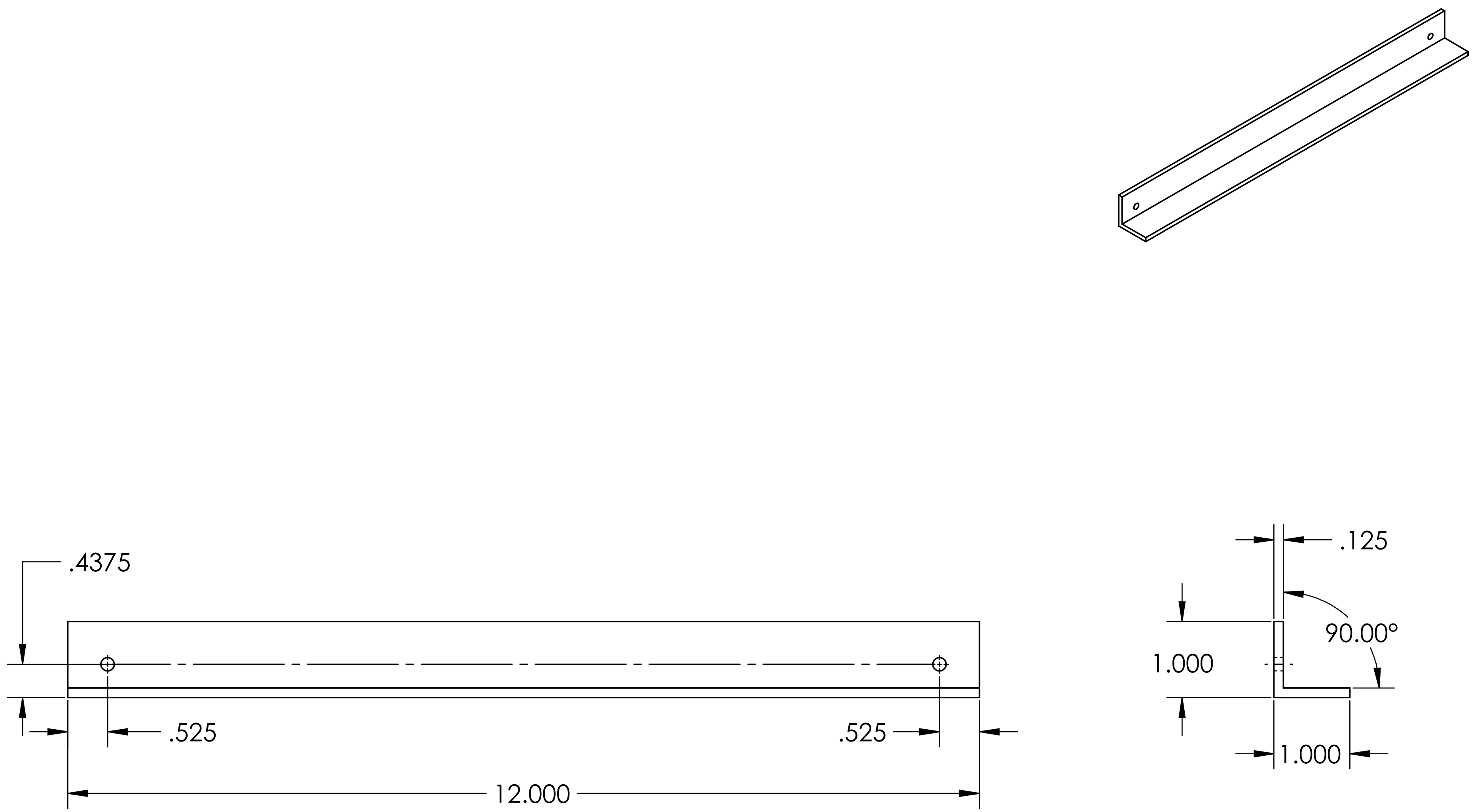




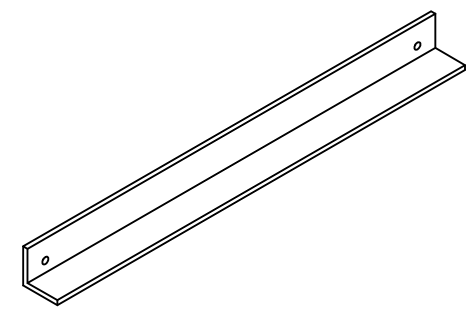
TITLE:
Acrylic Motor Mount

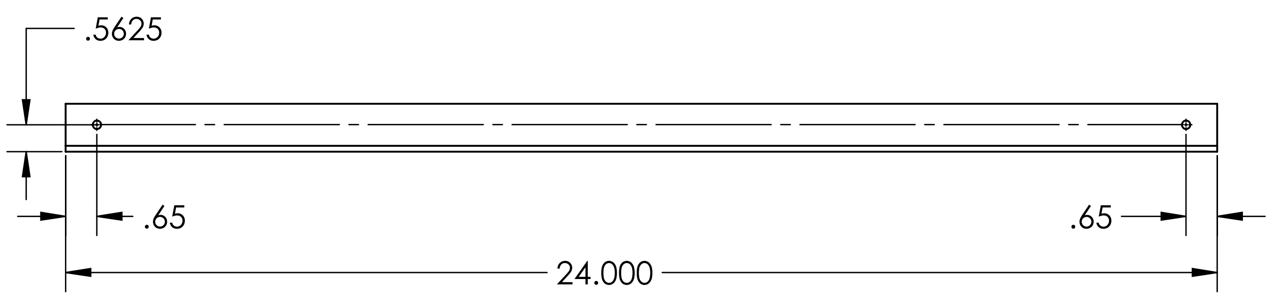
SIZE	DWG. NO.	REV
A		1

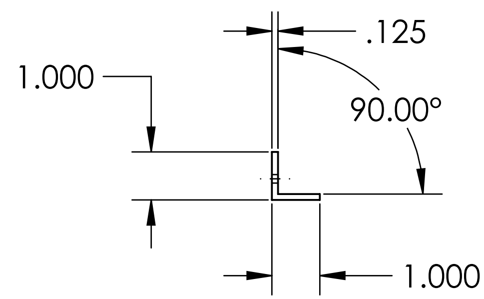
SCALE: 2:1	WEIGHT:	SHEET 1 OF 1



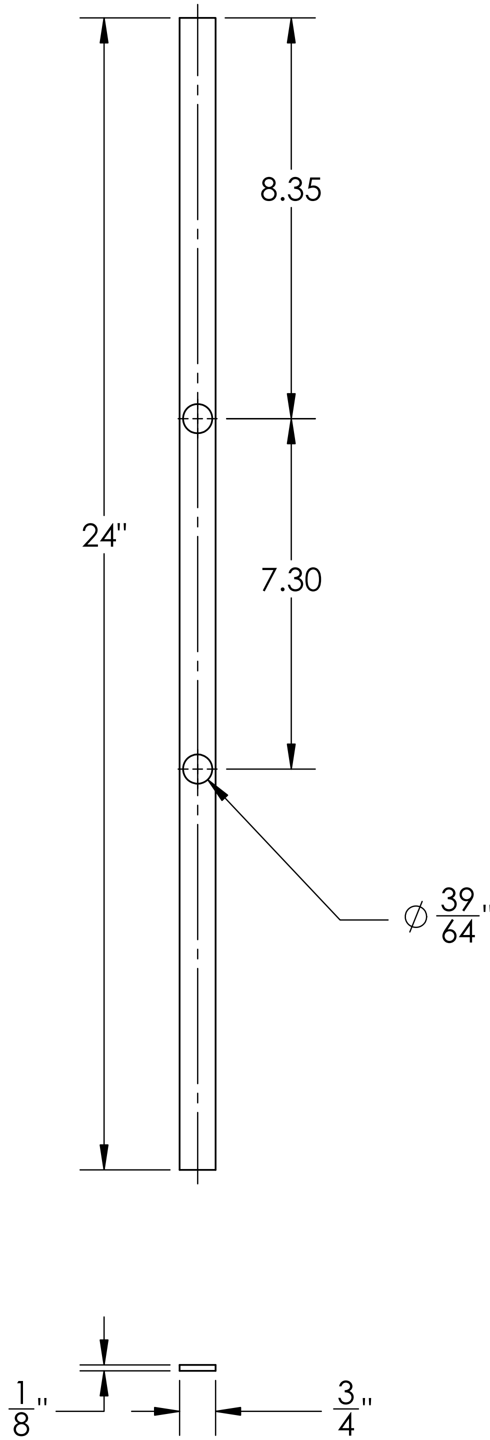
TITLE:		
6061 Aluminum 90° Angle - 1 ft.		
SIZE A	DWG. NO. 8982K4	REV 1
SCALE: 1:2	WEIGHT:	SHEET 1 OF 1

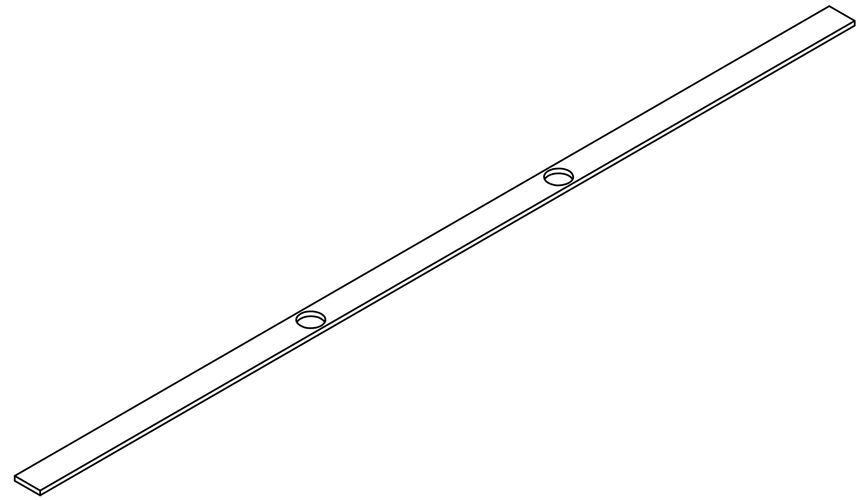






TITLE: 6061 Aluminum 90° Angle - 2 ft.		
SIZE A	DWG. NO. 8982K4	REV 1
SCALE: 1:4	WEIGHT:	SHEET 1 OF 1

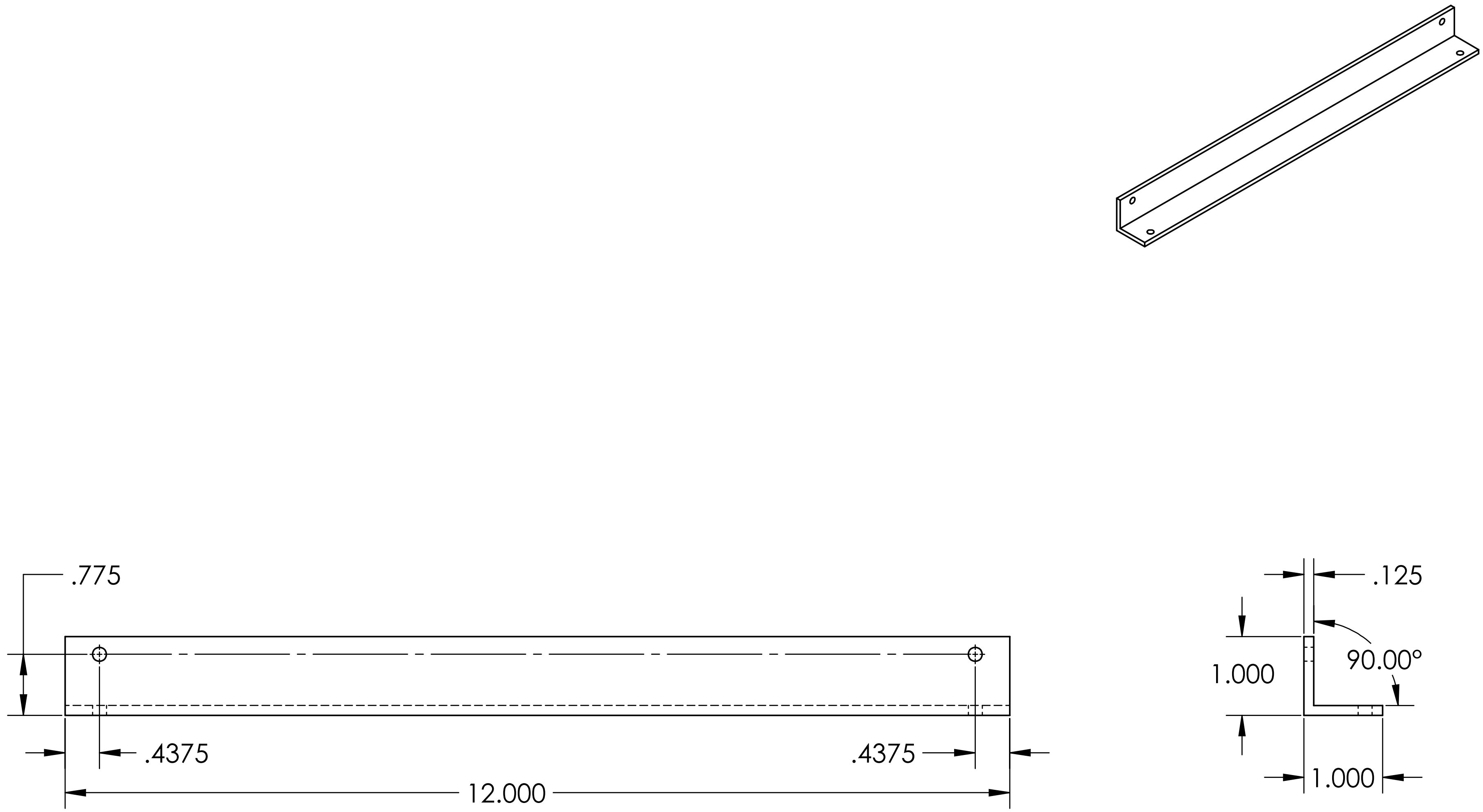




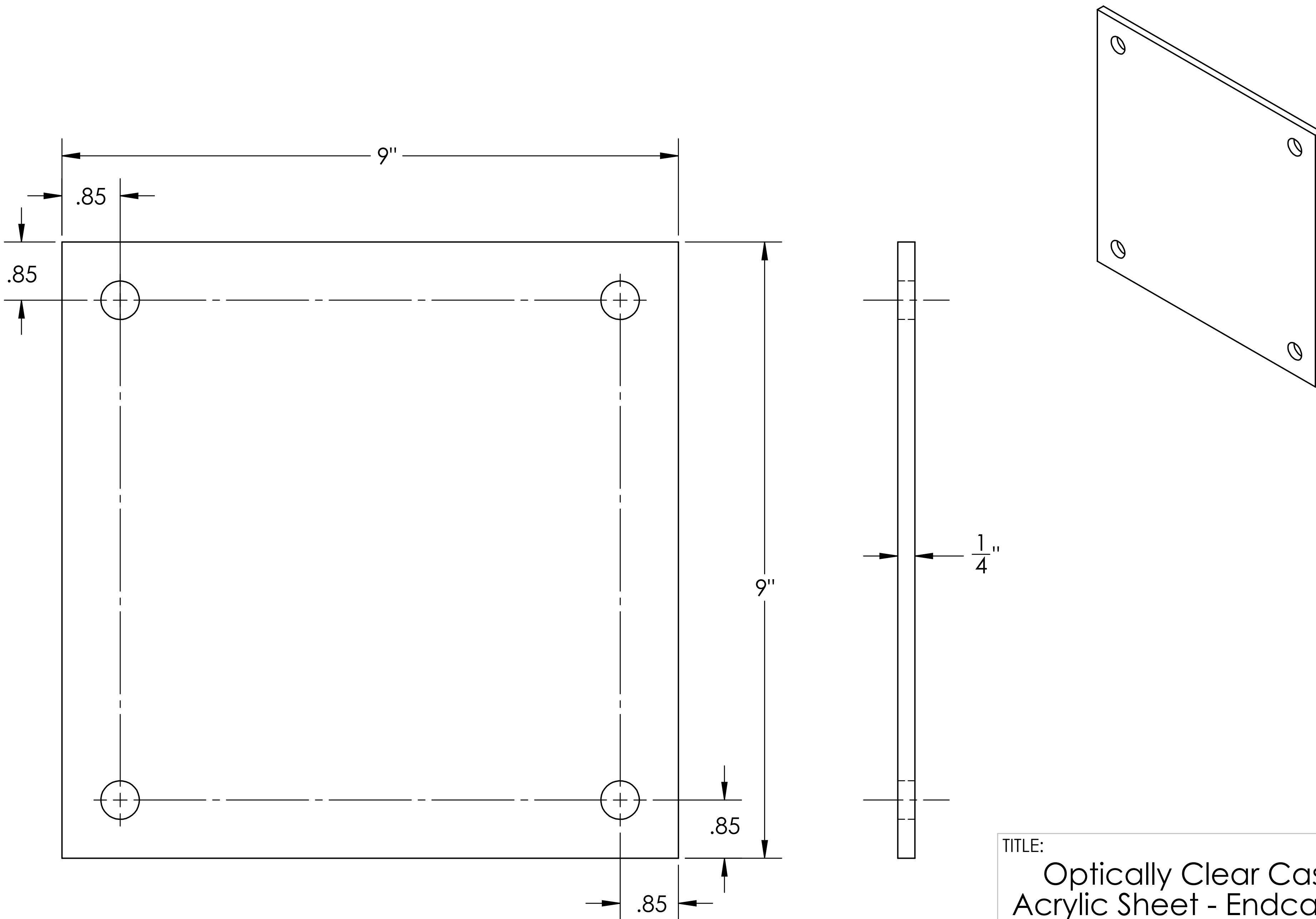
TITLE: 6061 Aluminum Rectangular Bar - Vessel

SIZE	DWG. NO.	REV
A	8975K578	1

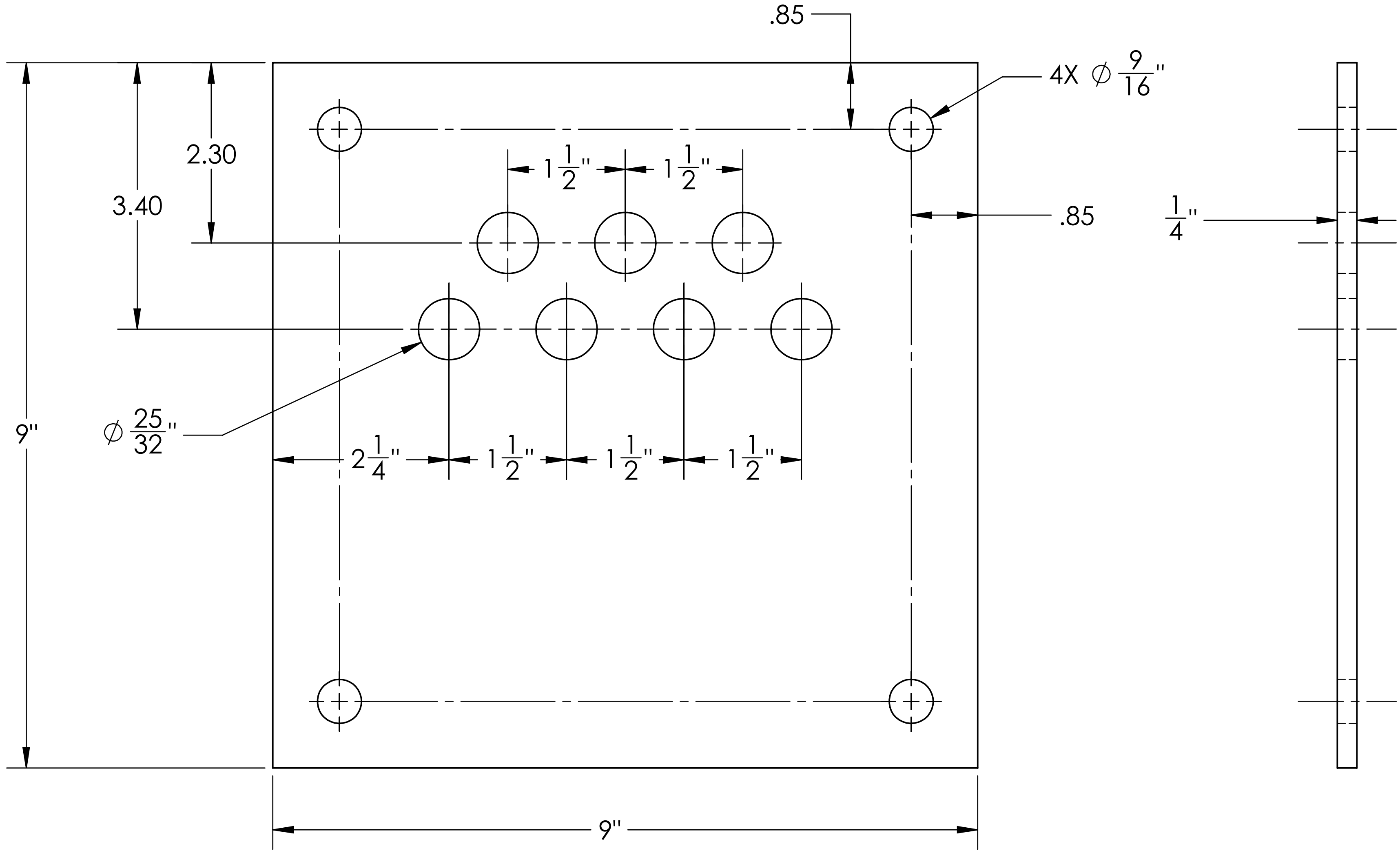
SCALE: 1:4	WEIGHT:	SHEET 1 OF 1



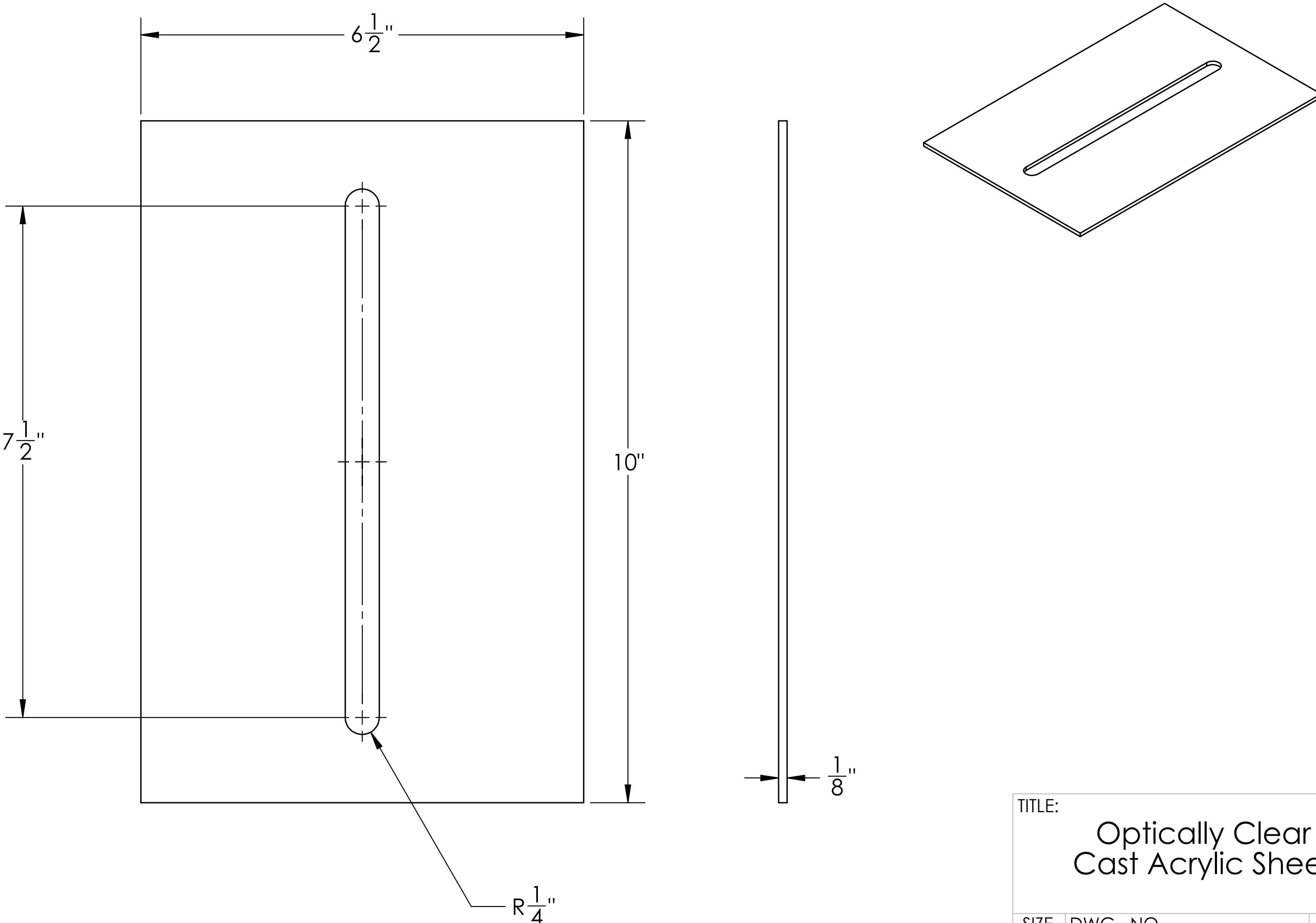
TITLE: 6061 Aluminum 90° Angle - Supports		
SIZE A	DWG. NO. 8982K4	REV 1
SCALE: 1:2	WEIGHT:	SHEET 1 OF 1



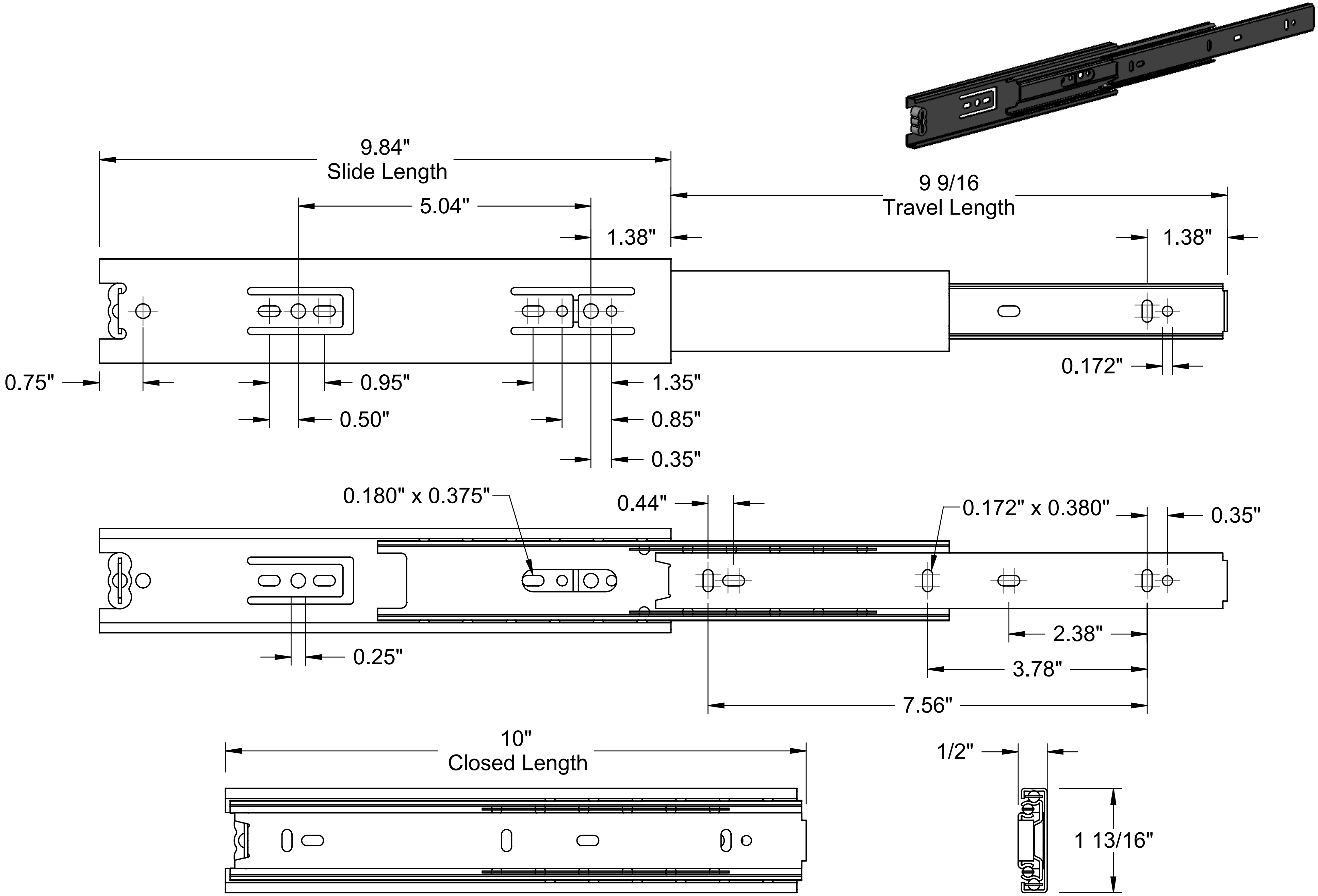
TITLE:		
Optically Clear Cast Acrylic Sheet - Endcap 1		
SIZE A	DWG. NO. 8560K239	REV 1
SCALE: 1:4	WEIGHT:	SHEET 1 OF 1



TITLE: Optically Clear Cast Acrylic Sheet - Endcap 2		
SIZE A	DWG. NO. 8560K239	REV 1
SCALE: 1:2	WEIGHT:	SHEET 1 OF 1



TITLE: Optically Clear Cast Acrylic Sheet		
SIZE A	DWG. NO. 8560K239	REV 1
SCALE: 1:2	WEIGHT:	SHEET 1 OF 1

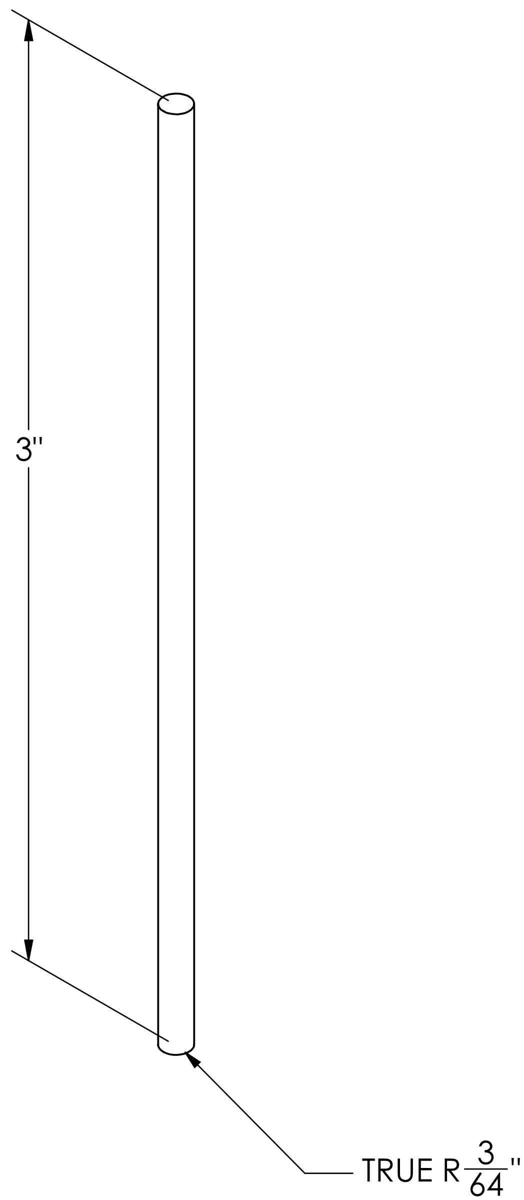


McMASTER-CARR® CAD

PART NUMBER 11435A110

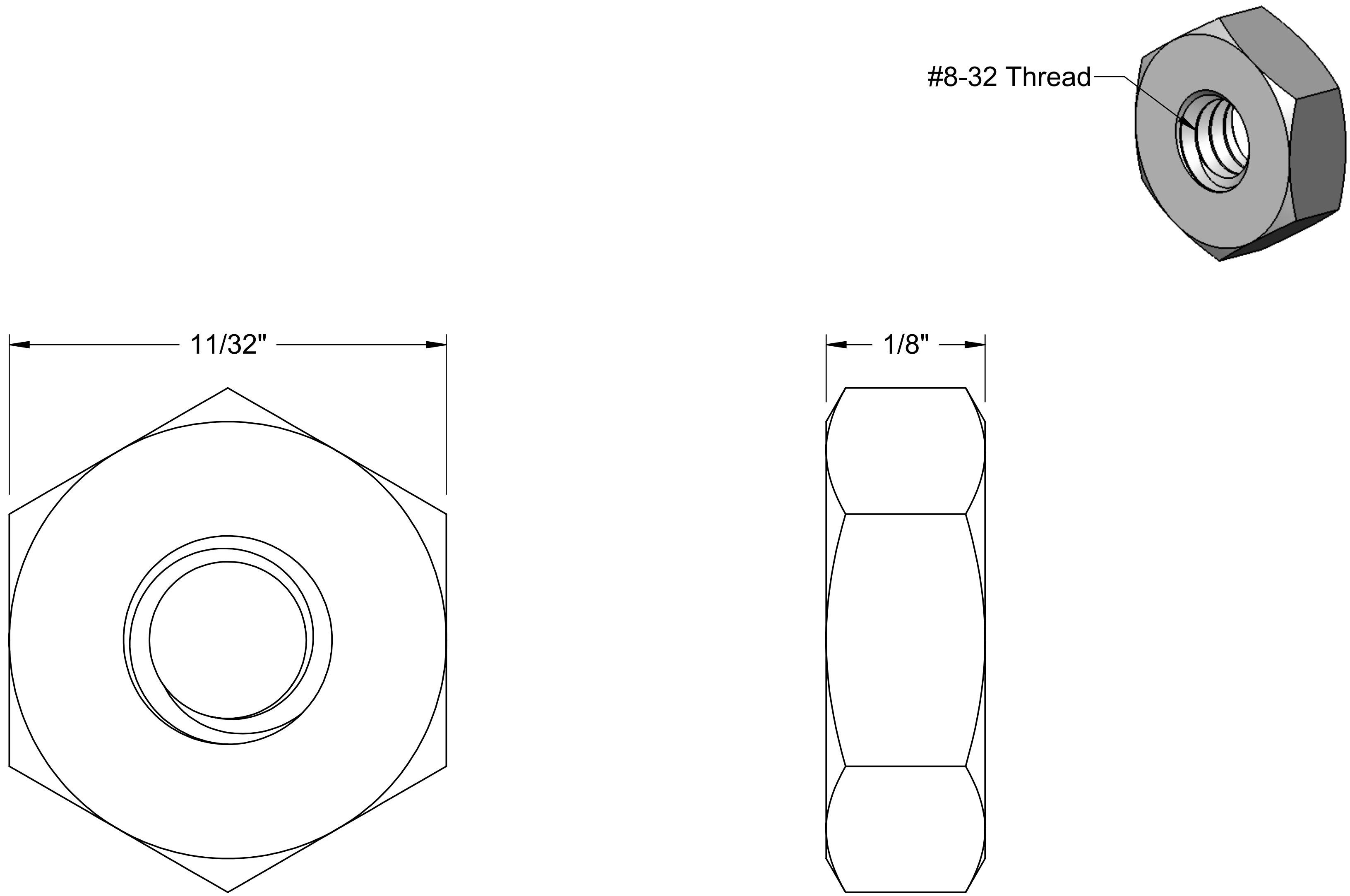
http://www.mcmaster.com

Side-Mount Drawer Slide



Threads not shown.

TITLE: Type 18-8 Stainless Steel Fully Threaded Rods		
SIZE A	DWG. NO. 98847A003	REV 1
SCALE: 2:1	WEIGHT:	SHEET 1 OF 1

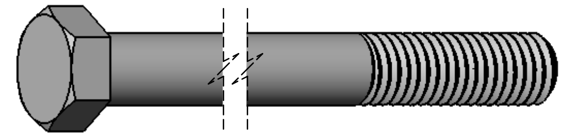


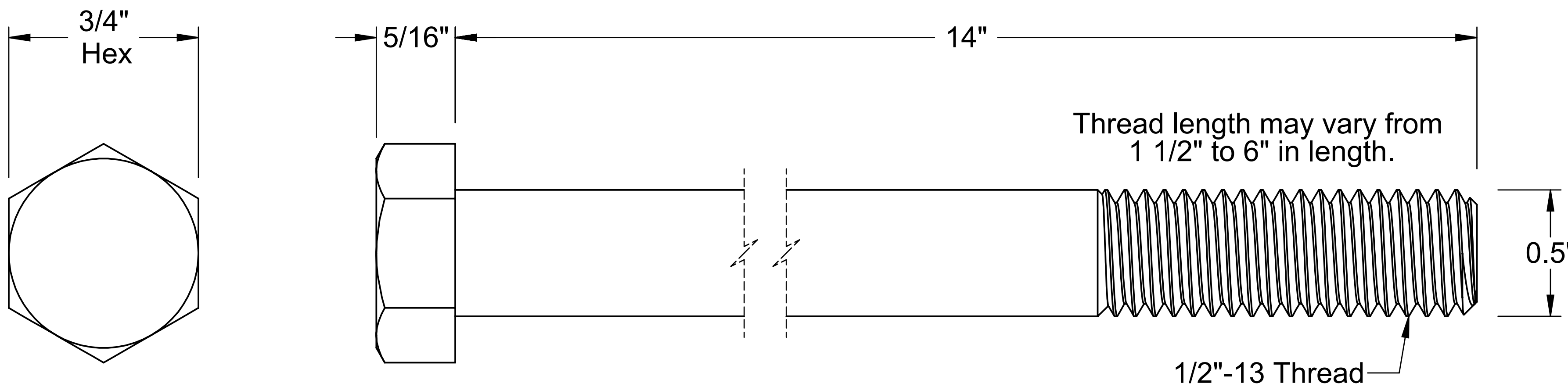
McMASTER-CARR® CAD

PART NUMBER 91841A009

http://www.mcmaster.com

Hex Nut



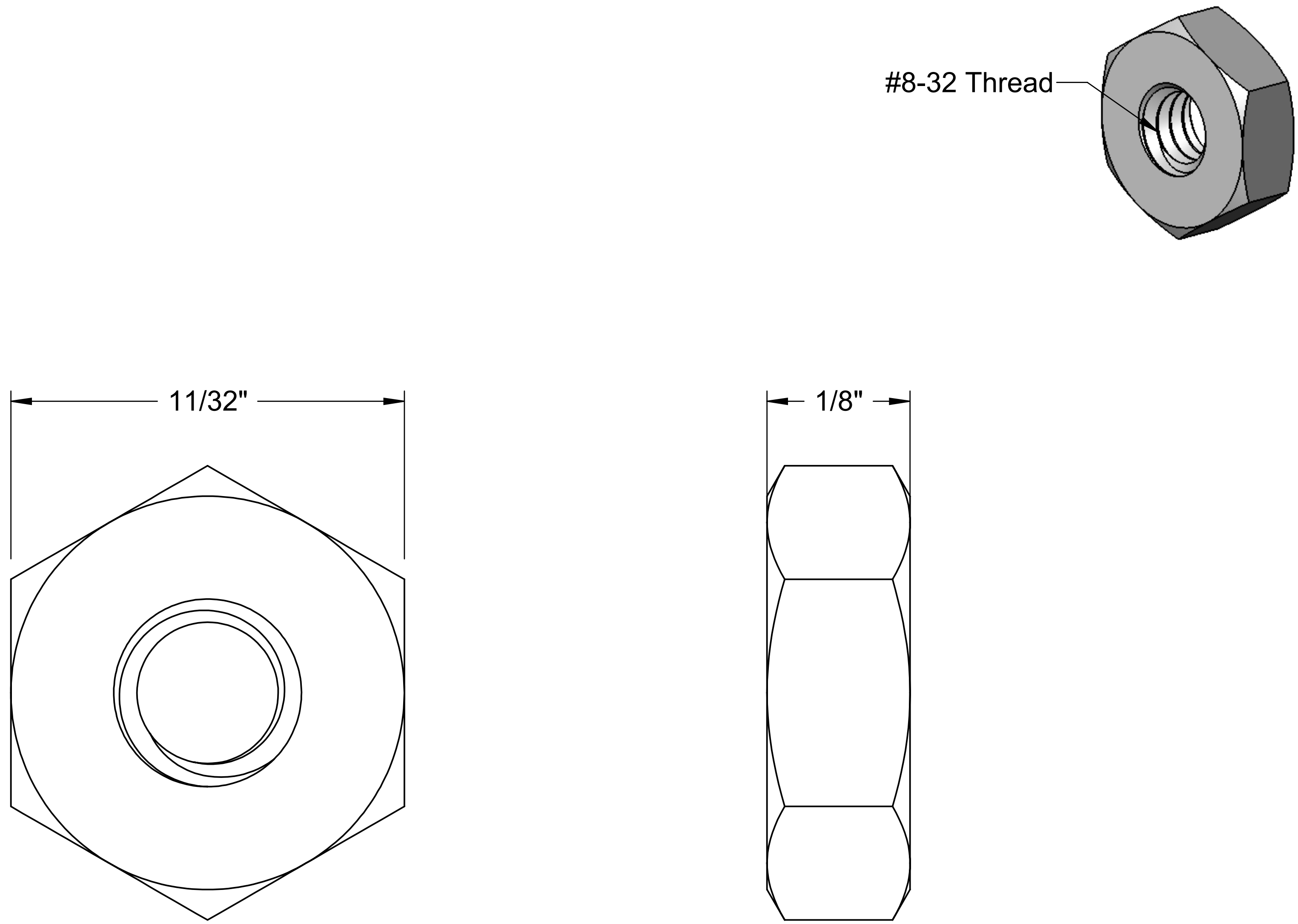


McMASTER-CARR® CAD

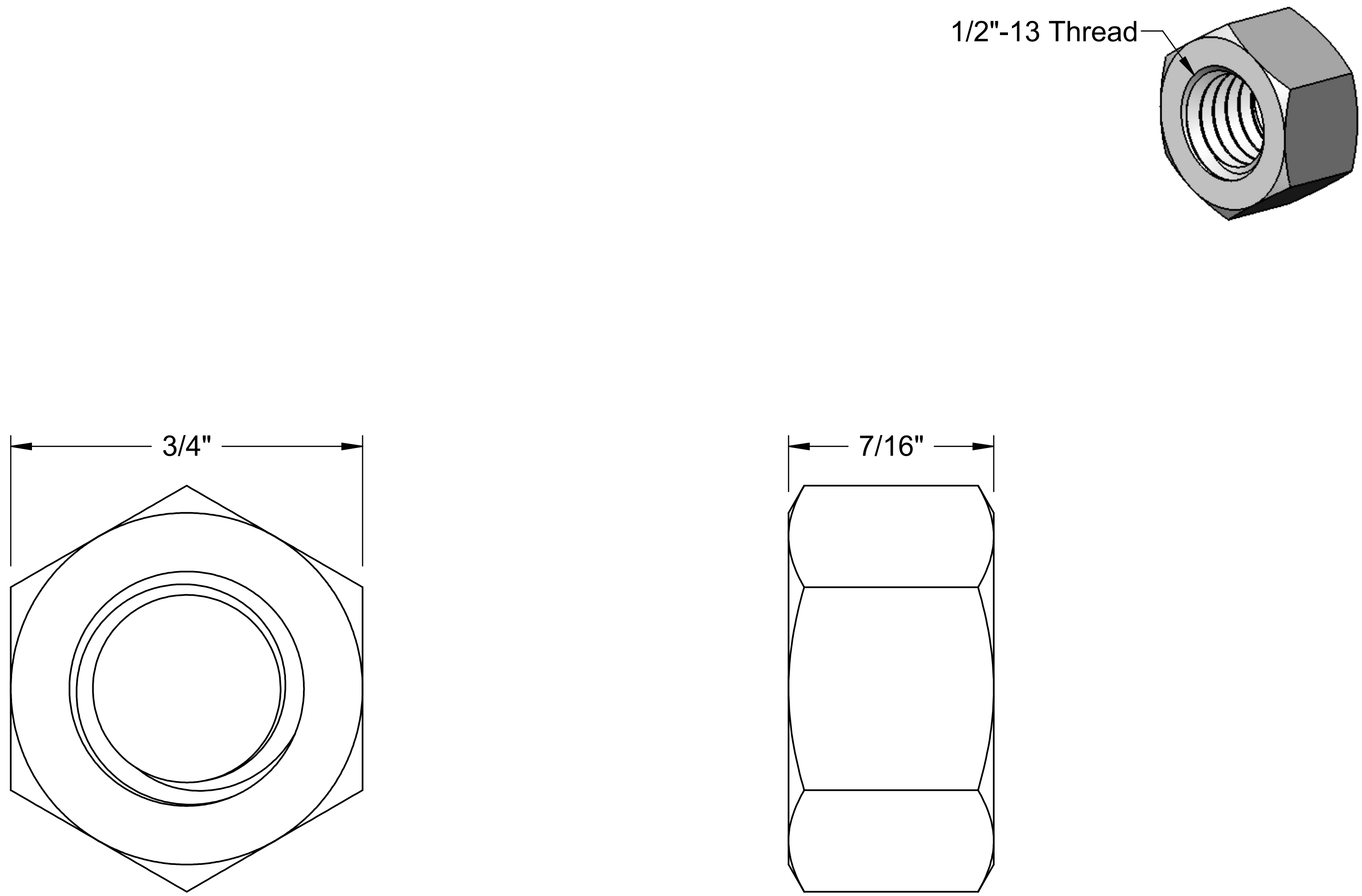
PART NUMBER **92186A754**

http://www.mcmaster.com

Stainless Steel
Cap Screw



McMASTER-CARR® CAD	PART NUMBER 90257A009
http://www.mcmaster.com © 2015 McMaster-Carr Supply Company	Hex Nut
Information in this drawing is provided for reference only.	



McMASTER-CARR® CAD	PART NUMBER **94804A340**
http://www.mcmaster.com Information in this drawing is provided for reference only.	Hex Nut

Appendix G: Mechanical Analysis

ThrusterAnalysis.m

```
% Alex Kost
% Multidisciplinary Senior Design Project
% Thruster Analysis

% Reset
clc;
clear all;
close all;

% Fluid characteristics
rho = 1.99;                    % density of saltwater, slugs/ft^3
g = 32.2;                      % gravity, ft/s^2

% Pump/blade characteristics
d = 2;                     % diameter of proeller, inches
A_in2 = .25*pi*d^2;            % area of propeller, in^2
A_ft2 = A_in2*(1/12)^2;        % area of propeller, ft^2

% ---------------------------------------------------------------------------

% WANT: 2 knots of nominal speed
v_fps = 2*(1.68780/1);          % nominal speed, in ft/sec

% Hydromodus specs
m = 9.3/g;                     % mass of Hydromodus, lbm or slug*ft/sec^2
t = 10;                        % time to achieve thrust, sec
CD = 1.05;                     % drag coefficient for cube (conservative)
A_ref = 2;                     % reference area, ft^2 (2x2x1 box, assume orientation
                               % w/ T100s for thrust
% F = m*a = m*(dv/dt)
% mdv = Fdt
% dv = (F/m)dt, assume F/m is constant over time
% v = (F/m)*t, solve for F

Fdesired = v_fps*m/t;        % desired thrust force, lbf
Fdrag = .5*rho*v_fps^2*CD*A_ref;  % drag force, lbf

% Fthrust = .5*rho_sea*A_prop*speed_fps^2;    % thrust force, lbf

% sum(Fx) = Fthrust - Fdrag = Fdesired
% Fthrust = Fdesired + Fdrag
Fthrust = Fdesired + Fdrag;

% From Fluids Ch 3, momentum CV analysis, assume steady state
F_req = rho*v_fps^2*A_ft2;
```

EndCapAnalysis.m

```
% Alex Kost
% Multidisciplinary Senior Design Project
% Endcap Analysis

% This analysis is to determine if the endcap bends from the force of the
% hex screws. If it bends, then the seal is no longer even across the
% rubber flange and could fail. The endcap is being modeled as a"beam" and
% being analyzed as such. Consider a cross-sectional view.

% Reset
clc;
clear all;
close all;

% System parameters
rho = 1.99;          % density of saltwater, slugs/ft.^3
g = 32.2;            % gravity, ft./sec.^2
h = 150;             % height, ft. - assume max depth of 150 ft.

% Endcap properties
l = 9;               % length of beam/endcap, in.
b = 9;               % width of beam/endcap, in.
A = l*b;             % area of endcap, in.^2
a = 1.7637;          % distance between bolt force and tube wall
                     % calculated with geometry, pythagorean

h = .25;             % height/thickness of beam/endcap, in.
I = (b*h^3)/12;      % Area moment of inertia, in.^4
E = 0.20*10^6;       % Modulus of elasticity for acrylic, Shigley's Table 2.2

% Aluminum T-bar properties
Ixx = 0.047;         % Area moment of inertia, in.^4
% http://www.amesweb.info/SectionalPropertiesTabs/SectionalPropertiesTbeam.aspx

E_al = 10.4*10^6;    % Modulus of elasticity for Al, Shigley's Table A-5

% Forces and pressures
p_o = (rho*g*h)/144;% outside pressure, psi
w = p_o/A;           % uniform load created by pressure, psi
p_i = 14.7;          % inner pressure, psi

% Screw force analysis
T = 10;              % in-lbs.
d = .5;              % nominal inch diameter, in.
k = .18;             % nut factor, aka Friction factor, dimensionless
       % taken from http://www.repairengineering.com/bolt-torque-chart.html
F = T/(k*d);         % desired axial bolt force desired, lbs.
                     % assumed to be at point location

% Superimpose types of loads, Shigley's Table A-9.9
ymax_acr = (F*a/(24*E*I))*((4*a^2)-(3*l^2));     % Simple supports--twin loads


ymax_al = (F*a/(24*E_al*Ixx))*((4*a^2)-(3*(l-1.7)^2));  % Simple supports--twin loads
ymax_water = (-5*w*l^4)/(384*E*I);              % Simple supports--uniform load

disp(F);
disp(ymax_acr);
disp(ymax_al);
disp(ymax_water);
```

BuoyancyAnalysis.m

```
% Alex Kost
% Multidisciplinary Senior Design Project
% Buoyancy Analysis

% Reset
clc;
clear all;
close all;

rho_sea = 1.99;         % density of saltwater, slugs/ft^3
g = 32.2;               % gravity, ft/s^2
l = 1;                  % length of cylinder, ft
d = 8/12;               % outer diameter of cylinder, ft

V_cyl = l*pi*(d/2)^2;   % volume of cylinder, ft^3
V_oth = 96.73/(12^3);   % volume displaced by system w/o vessel, ft^3 - from SolidWorks
V = V_cyl + V_oth;      % total volume, ft^3

Fb = (rho_sea*V*g);     % Bouyancy force, lbf
Fw = 34.3;              % Weight of Hydromodus, lbf

sumFy = Fb - Fw;        % sum of forces in Y direction

disp(Fb);
disp(Fw);
disp(sumFy);

% This found that the current system feels a net force of 15.3737 lbf down.
% To accomodate for this, empty, rigid bottles must be attached to the
% design to increase the bouyancy force and create a neutrally bouyant
% system.

% If the value were positively bouyant, the design would have to be weighed
% down with water bottles filled with sand or with scrap material. The
% weights would have to be placed so that symmetry is maintained.
```

VesselAnalysis.m

```
% Alex Kost
% Multidisciplinary Senior Design Project
% Vessel Analysis

% Reset
clc;
clear all;
close all;

% System parameters
rho = 1.99;         % density of saltwater, slugs/ft^3
g = 32.2;           % gravity, ft/sec^2
h = 150;            % height, ft. - assume max depth of 150 ft.

% Tube thickness
p_i = 14.7;             % internal pressure, atmospheric (psi)
p_o = rho*g*h/144;  % external pressure, pressure of water, psi
d_o = 8;            % outer diameter, in.
r_o = d_o/2;        % outer radius, in.
t = .125;           % wall thickness, in.
d_i = d_o - 2*t;    % inner diameter, in.
r_i = d_i/2;        % inner radius, in.
r = r_i;            % where max stresses lie, in.

% Shigley's 9th ed, Section 3-14, page 113
sigma_t = (p_i*(r_i^2) - p_o*(r_o^2) - (r_i^2)*(r_o^2)*(p_o - p_i)/(r^2))/((r_o^2) -↙
(r_i^2));
sigma_r = (p_i*(r_i^2) - p_o*(r_o^2) + (r_i^2)*(r_o^2)*(p_o - p_i)/(r^2))/((r_o^2) -↙
(r_i^2));

tensile_strength = 8000; % tensile strength, psi - McMaster Carr Plastics Data Sheet

disp('sigma_t = ');
disp(sigma_t);
disp('sigma_r = ');
disp(sigma_r);

if(sigma_t > tensile_strength)
    disp('Vessel will fail!');
else
    disp('Vessel passes!');
    disp('Factor of safety = ');
    disp(-1*tensile_strength/sigma_t);
end
```

Appendix H: Light Diffusion Analysis

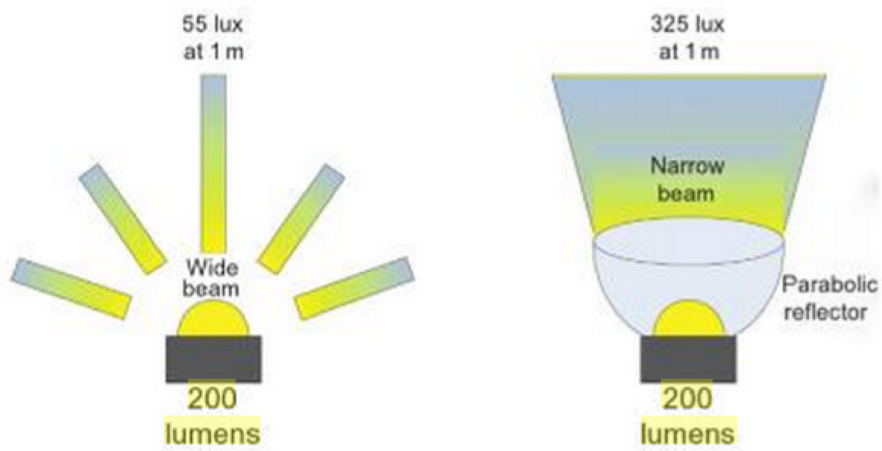


FIGURE 11.2

Lumens to lux.

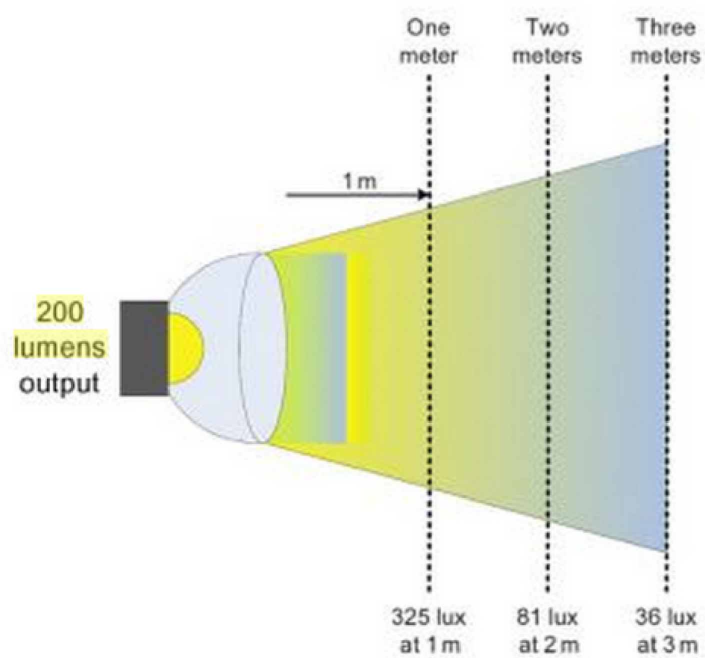


FIGURE 11.3

Spreading losses per unit distance.

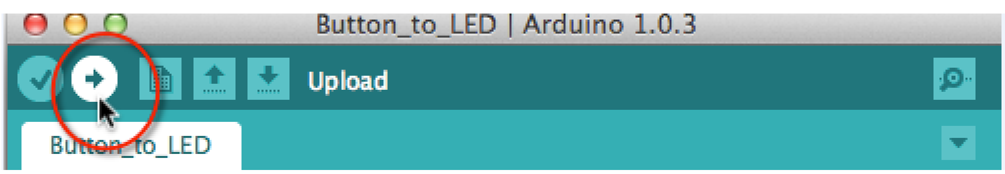
Appendix I. User Manual

A. Programming Mission Parameters into the Arduino Uno

1. Open the Hydromodus by unscrewing the hex nuts on the eight ½-inch bolts.
2. Connect the computer to the Arduino Uno USB port with a USB A/B cable.
3. Using a computer with the Arduino software, open the Hydromodus.ino sketch.
4. A long list of variable definitions will be seen. Scroll past these definitions until the following code is found.

```
//////////////////////////////////////////////////////////////////////////////
////// MISSION PARAMETER USER INPUT FOR DESTINATION, TIME, AND DEPTH /////////
//////////////////////////////////////////////////////////////////////////////

long destinationLatitude = 35.291336;
long destinationLongtitude = -120.662368;
long destinationTime = 1;
int destinationDepth = 8;
```

5. The user will see four lines of code referencing latitude, longitude, time, and depth. Replace the orange values with the user-defined or mission-specific values in decimal coordinates, decimal coordinates, minutes, and feet respectively.
6. Once completed, click the button with the arrow pointing right in the top-left corner of the window (indicated below) to upload the modified code to the Arduino Uno.



7. Once successfully uploaded, disconnect the USB A/B cable from the Arduino Uno and seal the vessel by aligning the Buna-N rubber flange with the cylindrical tube and endcap and refastening the hex nuts.

B. Charging Microcontroller Battery Packs ("PowerAdd" displayed on side)

1. Plug in a micro-USB cable to the the micro-USB ports located on the thin side of the batteries.
2. Visually confirm the batteries are charging by examining the four LEDs on the battery pack. Four LEDs will illuminate when fully charged.

C. Charging Thruster Battery Packs ("8000 mAh" displayed on side)

1. Plug in the included lithium-polymer battery charger into a wall outlet and to the charging port of the battery.
2. Visually confirm the batteries are charging by examining the three LEDs on the charger to see if they are red. When all three LEDs on the charger are green, then the batteries are fully charged.

NOTE: Always supervise the lithium-polymer battery packs when charging, as they may ignite.

D. Launching the Hydromodus

1. Submerge the Hydromodus wherever the mission is to take begin.

NOTE: If the Hydromodus is being launched out to sea, it will operate better if launched further away from the shoreline.

Appendix J: Final Source Code: Master Controller

```
#include <Wire.h>
#include <Math.h>

#include <TinyGPS++.h>   // Library for GPS Module
#include <SoftwareSerial.h>

////////////////////////////////////////////////////////////////////////////////
/////////////////// Variable Definitions for IMU sensor //////////////////////
////////////////////////////////////////////////////////////////////////////////
#define MPU9150_SELF_TEST_X    0x0D // R/W
#define MPU9150_SELF_TEST_Y    0x0E // R/W
#define MPU9150_SELF_TEST_X    0x0F // R/W
#define MPU9150_SELF_TEST_A    0x10 // R/W
#define MPU9150_SMPLRT_DIV     0x19 // R/W
#define MPU9150_CONFIG   0x1A // R/W
#define MPU9150_GYRO_CONFIG    0x1B // R/W
#define MPU9150_ACCEL_CONFIG   0x1C // R/W
#define MPU9150_FF_THR   0x1D // R/W
#define MPU9150_FF_DUR   0x1E // R/W
#define MPU9150_MOT_THR  0x1F // R/W
#define MPU9150_MOT_DUR  0x20 // R/W
#define MPU9150_ZRMOT_THR      0x21 // R/W
#define MPU9150_ZRMOT_DUR      0x22 // R/W
#define MPU9150_FIFO_EN  0x23 // R/W
#define MPU9150_I2C_MST_CTRL   0x24 // R/W
#define MPU9150_I2C_SLV0_ADDR  0x25 // R/W
#define MPU9150_I2C_SLV0_REG   0x26 // R/W
#define MPU9150_I2C_SLV0_CTRL  0x27 // R/W
#define MPU9150_I2C_SLV1_ADDR  0x28 // R/W
#define MPU9150_I2C_SLV1_REG   0x29 // R/W
#define MPU9150_I2C_SLV1_CTRL  0x2A // R/W
#define MPU9150_I2C_SLV2_ADDR  0x2B // R/W
#define MPU9150_I2C_SLV2_REG   0x2C // R/W
#define MPU9150_I2C_SLV2_CTRL  0x2D // R/W
#define MPU9150_I2C_SLV3_ADDR  0x2E // R/W
#define MPU9150_I2C_SLV3_REG   0x2F // R/W
#define MPU9150_I2C_SLV3_CTRL  0x30 // R/W
#define MPU9150_I2C_SLV4_ADDR  0x31 // R/W
#define MPU9150_I2C_SLV4_REG   0x32 // R/W
#define MPU9150_I2C_SLV4_DO    0x33 // R/W
#define MPU9150_I2C_SLV4_CTRL  0x34 // R/W
#define MPU9150_I2C_SLV4_DI    0x35 // R
#define MPU9150_I2C_MST_STATUS       0x36 // R
#define MPU9150_INT_PIN_CFG    0x37 // R/W
#define MPU9150_INT_ENABLE     0x38 // R/W
#define MPU9150_INT_STATUS     0x3A // R
#define MPU9150_ACCEL_XOUT_H   0x3B // R
#define MPU9150_ACCEL_XOUT_L   0x3C // R
#define MPU9150_ACCEL_YOUT_H   0x3D // R
#define MPU9150_ACCEL_YOUT_L   0x3E // R
```

```
#define MPU9150_ACCEL_ZOUT_H   0x3F // R
#define MPU9150_ACCEL_ZOUT_L   0x40 // R
#define MPU9150_TEMP_OUT_H     0x41 // R
#define MPU9150_TEMP_OUT_L     0x42 // R
#define MPU9150_GYRO_XOUT_H    0x43 // R
#define MPU9150_GYRO_XOUT_L    0x44 // R
#define MPU9150_GYRO_YOUT_H    0x45 // R
#define MPU9150_GYRO_YOUT_L    0x46 // R
#define MPU9150_GYRO_ZOUT_H    0x47 // R
#define MPU9150_GYRO_ZOUT_L    0x48 // R
#define MPU9150_EXT_SENS_DATA_00 0x49 // R
#define MPU9150_EXT_SENS_DATA_01 0x4A // R
#define MPU9150_EXT_SENS_DATA_02 0x4B // R
#define MPU9150_EXT_SENS_DATA_03 0x4C // R
#define MPU9150_EXT_SENS_DATA_04 0x4D // R
#define MPU9150_EXT_SENS_DATA_05 0x4E // R
#define MPU9150_EXT_SENS_DATA_06 0x4F // R
#define MPU9150_EXT_SENS_DATA_07 0x50 // R
#define MPU9150_EXT_SENS_DATA_08 0x51 // R
#define MPU9150_EXT_SENS_DATA_09 0x52 // R
#define MPU9150_EXT_SENS_DATA_10 0x53 // R
#define MPU9150_EXT_SENS_DATA_11 0x54 // R
#define MPU9150_EXT_SENS_DATA_12 0x55 // R
#define MPU9150_EXT_SENS_DATA_13 0x56 // R
#define MPU9150_EXT_SENS_DATA_14 0x57 // R
#define MPU9150_EXT_SENS_DATA_15 0x58 // R
#define MPU9150_EXT_SENS_DATA_16 0x59 // R
#define MPU9150_EXT_SENS_DATA_17 0x5A // R
#define MPU9150_EXT_SENS_DATA_18 0x5B // R
#define MPU9150_EXT_SENS_DATA_19 0x5C // R
#define MPU9150_EXT_SENS_DATA_20 0x5D // R
#define MPU9150_EXT_SENS_DATA_21 0x5E // R
#define MPU9150_EXT_SENS_DATA_22 0x5F // R
#define MPU9150_EXT_SENS_DATA_23 0x60 // R
#define MPU9150_MOT_DETECT_STATUS 0x61 // R
#define MPU9150_I2C_SLV0_DO    0x63 // R/W
#define MPU9150_I2C_SLV1_DO    0x64 // R/W
#define MPU9150_I2C_SLV2_DO    0x65 // R/W
#define MPU9150_I2C_SLV3_DO    0x66 // R/W
#define MPU9150_I2C_MST_DELAY_CTRL 0x67 // R/W
#define MPU9150_SIGNAL_PATH_RESET 0x68 // R/W
#define MPU9150_MOT_DETECT_CTRL      0x69 // R/W
#define MPU9150_USER_CTRL      0x6A // R/W
#define MPU9150_PWR_MGMT_1     0x6B // R/W
#define MPU9150_PWR_MGMT_2     0x6C // R/W
#define MPU9150_FIFO_COUNTH    0x72 // R/W
#define MPU9150_FIFO_COUNTL    0x73 // R/W
#define MPU9150_FIFO_R_W       0x74 // R/W
#define MPU9150_WHO_AM_I       0x75 // R

//MPU9150 Compass
#define MPU9150_CMPS_XOUT_L    0x4A // R
#define MPU9150_CMPS_XOUT_H    0x4B // R
#define MPU9150_CMPS_YOUT_L    0x4C // R
```

```
#define MPU9150_CMPS_YOUT_H    0x4D // R
#define MPU9150_CMPS_ZOUT_L    0x4E // R
#define MPU9150_CMPS_ZOUT_H    0x4F // R

//////////////////////////////////////////////////////////////////////////////
////// MISSION PARAMETER USER INPUT FOR DESTINATION, TIME, AND DEPTH /////////
//////////////////////////////////////////////////////////////////////////////

long destinationLatitude = 35.291336;
long destinationLongtitude = -120.662368;
long destinationTime = 1;
int destinationDepth = 8;

//////////////////////////////////////////////////////////////////////////////
//////////////////////////////////////////////////////////////////////////////
//////////////////////////////////////////////////////////////////////////////

int MPU9150_I2C_ADDRESS = 0x69;

int compassx;
int compassy;
int compassz;
int finalCompass;
int cmps[3];
int accl[3];
int gyro[3];
int temp;
int RXPin = 2;
int TXPin = 3;

long currentLatitude;
long currentLongtitude;
long calcLongtitude;
long calcLatitude;
long x = 0;
long y = 0;
long xp = 0;
long yp = 0;
long xorig = 0;
long yorig = 0;
int lmotor = 13;
int rmotor = 12;
long time1 = 0;
long time2 = 0;
long time3 = 0;
long calcHypotenuse;

int GPSBaud = 4800;


TinyGPSPlus gps;
SoftwareSerial gpsSerial(RXPin, TXPin);
const int DevAddress = 0x76;
const byte Reset = 0x1E;
```

```
const byte D1_256 = 0x40;
const byte D1_512 = 0x42;
const byte D1_1024 = 0x44;
const byte D1_2048 = 0x46;
const byte D1_4096 = 0x48;
const byte D2_256 = 0x50;
const byte D2_512 = 0x52;
const byte D2_1024 = 0x54;
const byte D2_2048 = 0x56;
const byte D2_4096 = 0x58;
const byte AdcRead = 0x00;
const byte PromBaseAddress = 0xA0;

int angle;
int destinationAngle;
unsigned int CalConstant[8];
long AdcTemperature, AdcPressure;
float Temperature, Pressure, TempDifference, Offset, Sensitivity, depth,
currentDepth;
float T2, Off2, Sens2;
byte ByteHigh, ByteMiddle, ByteLow;
int i = 0;
int b = 0;
unsigned long pstart;
unsigned long pwidth;
unsigned long distance;
int s = 0;

int sonar1 = 0; // bottom sensor 1
int sonar2 = 0; // bottom sensor 2
int sonar3 = 0; // left sensor
int sonar4 = 0; // right sensor
int sonar5 = 0; // forward sensor
int sonar6 = 0; // top sensor

////////////////////////////////////////////////////////////////////////////////
/// the setup function runs once when you press reset or power the vessel ////
////////////////////////////////////////////////////////////////////////////////

void setup() {
 Serial.begin(9600);
 delay(1000);
 Wire.begin();
 delay(10);
 delay(1000);
 MPU9150_writeSensor(MPU9150_PWR_MGMT_1, 0);

 MPU9150_setupCompass();

 gpsSerial.begin(GPSBaud);

// Initializing user input destination and determining where it is in respect
to international date line //
 if (destinationLongtitude < 0)
```

```
 {
      destinationLongtitude = 360000000 + (destinationLongtitude*1000000);
 }
 else
 {
 destinationLongtitude = destinationLongtitude*1000000;
 }

 destinationLatitude = 90000000 + (destinationLatitude*1000000);


 sendCommand(Reset);
 delay(10);
 delay(1000);
 Serial.println("Device is reset");

 // Get the calibration constants and store in array

 for (byte i = 0; i < 8; i++)
 {
      sendCommand(PromBaseAddress + (2*i));
      Wire.requestFrom(DevAddress, 2);
      while(Wire.available()){
      ByteHigh = Wire.read();
      ByteLow = Wire.read();
      }
      CalConstant[i] = (((unsigned int)ByteHigh << 8) + ByteLow);
 }


 Serial.begin(9600);

 pinMode(5, OUTPUT); //Motor 1(pin 2 on receiving side)
 pinMode(6, OUTPUT); //Motor 2(pin 3 on receiving side)
 pinMode(9, OUTPUT); //Motor 3(pin 4 on receiving side)
 pinMode(10, OUTPUT); //Motor 4(pin 5 on receiving side)
 pinMode(11, OUTPUT); //Motor 5(pin 6 on receiving side)

 pinMode(4, OUTPUT); //Sonar 1 - 6 Trigger

 pinMode(7, INPUT); // Sonar1
 pinMode(8, INPUT); // Sonar2
 pinMode(12, INPUT); // Sonar3
 pinMode(A0, INPUT); // Sonar4
 pinMode(A1, INPUT); // Sonar5
 pinMode(A2, INPUT); // Sonar6
}

/////////////// the loop function runs over and over again forever
/////////////////
void loop()
{
 if (b == 0)
 {
```

```
//calls a function to determine current gps location and receives angle to
rotate to
      destinationAngle = gpsLocation();
//receives current compass heading in degrees
      finalCompass = compassRead();
      //rotates vessel to match compass heading with destination heading
            if (destinationAngle != 500)
            {
                  stopMotor();
                  if ((destinationAngle - finalCompass) > 180)
                  {
                        while(compassRead() > (destinationAngle + 2))
                        {
                              turnLeft();
                        }
                        stopMotor();
                  }
            else
            {
                  while(compassRead() < (destinationAngle - 2))
                  {
                        turnRight();
                  }
                  stopMotor();
            }
            moveForward();
}
else   // dive down to the user input depth
{
            stopMotor();
      while ((depthRead() < destinationDepth) || (sonar1 < 3000)||(sonar2 <
3000))
            {
                  descend();
                  sonar1 = sonar(7);
                  sonar2 = sonar(8);
                  sonar3 = sonar(12);
                  sonar4 = sonar(A0);
                  sonar5 = sonar(A1);
                  sonar6 = sonar(A2);

                  sonarAction(sonar1, sonar2, sonar3, sonar4, sonar5, sonar6);
            }
            stopMotor();
            b = 1;
            time1 = millis();
}
}
else if ( b == 1)  //Start timer and comapre to user input time
{
      destinationTime = destinationTime * 60000;
      while ((millis() - time1) < destinationTime)
      {
            sonar1 = sonar(7);
```

```
            sonar2 = sonar(8);
            sonar3 = sonar(12);
            sonar4 = sonar(A0);
            sonar5 = sonar(A1);
            sonar6 = sonar(A2);

            sonarAction(sonar1, sonar2, sonar3, sonar4, sonar5, sonar6);
      }
      while(depthRead() > 0.5)
      {
            ascend();
            sonar1 = sonar(7);
            sonar2 = sonar(8);
            sonar3 = sonar(12);
            sonar4 = sonar(A0);
            sonar5 = sonar(A1);
            sonar6 = sonar(A2);

            sonarAction(sonar1, sonar2, sonar3, sonar4, sonar5, sonar6);
      }
      stopMotor();
 }
 delay(1000);

}

int compassRead() // Read from compass
{
       int fCompass;
 compassx = MPU9150_readSensor(MPU9150_CMPS_XOUT_L,MPU9150_CMPS_XOUT_H);
       compassy = MPU9150_readSensor(MPU9150_CMPS_YOUT_L,MPU9150_CMPS_YOUT_H);
       compassz = MPU9150_readSensor(MPU9150_CMPS_ZOUT_L,MPU9150_CMPS_ZOUT_H);

       if ((abs(compassx - compassy) <= 2) && (abs(compassx - compassz) <= 2))
      {
      fCompass = compassx;
      }
      else
      {
      fCompass = compassRead();
      }
      return fCompass;
}

int gpsLocation() // confirm connection to GPS module
{
int ang;
      while (gpsSerial.available() > 0)
if (gps.encode(gpsSerial.read()))
      {
            ang = getLocation();
      }

      if (millis() > 5000 && gps.charsProcessed() < 10)
```

```
      {
      Serial.println(F("No GPS detected"));
      while(true);
      }
      return ang;
}

int getLocation() // get location from GPS module
{
      int ang;
      Serial.print(F("LOCATION: "));
      if (gps.location.isValid())
      {

      currentLatitude = 1000000*gps.location.lat();
      currentLongtitude = 1000000*gps.location.lng();

      if (currentLongtitude < 0)
      {
                  currentLongtitude = 360000000 + currentLongtitude;
            }
            else
            {
                  currentLongtitude = currentLongtitude;
            }

      currentLatitude = 90000000 + currentLatitude;
      ang = determinePath(currentLatitude, currentLongtitude);
      }
      else
      {
            Serial.print(F("INVALID"));
      }

      return ang;
      delay(3000);
      Serial.println();
}

//////////////////////////////////////////////////////////////////////////////
// calculate angle device needs to be facing in order to get to destination //
//////////////////////////////////////////////////////////////////////////////

int determinePath(int cLat, int cLong)
{
calcLatitude = destinationLatitude - cLat;
      calcLongtitude = destinationLongtitude - cLong;
      if((abs(calcLatitude) <= 5) && (abs(calcLongtitude) <= 5))
      {
      angle = 500;
      }
      else
      {
      calcHypotenuse = sqrt(sq(calcLatitude) + sq(calcLongtitude));
```

```
            if ((cLat > destinationLatitude) && (cLong >
destinationLongtitude))
            {
                  angle = 180 + asin((abs(calcLongtitude))/calcHypotenuse);
            }
            else if ((cLat>destinationLatitude)&& (cLong <
destinationLongtitude))
            {
                  angle = 180 - asin((abs(calcLongtitude))/calcHypotenuse);
            }
            else if ((cLat<destinationLatitude) &&(cLong >
destinationLongtitude))
            {
                  angle = 360 - asin((abs(calcLongtitude))/calcHypotenuse);
            }
            else if ((cLat<destinationLatitude) &&(cLong <
destinationLongtitude))
            {
                  angle = asin((abs(calcLongtitude))/calcHypotenuse);
            }
      }
      return angle;
}

///////////////////////////////////////////////////////////
/////// Get the current depth of the vessel ///////////////
///////////////////////////////////////////////////////////
float depthRead()
{
      sendCommand(D1_512);
      delay(10);
      sendCommand(AdcRead);
      Wire.requestFrom(DevAddress, 3);
      while(Wire.available())
      {
            ByteHigh = Wire.read();
            ByteMiddle = Wire.read();
            ByteLow = Wire.read();
 }
      AdcPressure=((long)ByteHigh << 16)+ ((long)ByteMiddle << 8) +
(long)ByteLow;

sendCommand(D2_512);
      delay(10);
      sendCommand(AdcRead);
      Wire.requestFrom(DevAddress, 3);
      while(Wire.available())
      {
      ByteHigh = Wire.read();
      ByteMiddle = Wire.read();
      ByteLow = Wire.read();
      }
      AdcTemperature = ((long)ByteHigh<<16)+((long)ByteMiddle<<8) +
(long)ByteLow;
```

```
Temperature = Temperature / 100; // Convert to degrees C

Offset = (float)CalConstant[2] * pow(2,16);
      Offset = Offset + ((float)CalConstant[4] * TempDifference / pow(2, 7));

Sensitivity = (float)CalConstant[1] * pow(2, 15);
      Sensitivity = Sensitivity+((float)CalConstant[3]*TempDifference / pow(2,
8));

      Offset = Offset - Off2;
      Sensitivity = Sensitivity - Sens2;



Pressure = (float)AdcPressure * Sensitivity / pow(2, 21);
      Pressure = Pressure - Offset;
      Pressure = Pressure / pow(2, 15);
      Pressure = Pressure / 10000; // Set output to bars;

      Pressure = Pressure - 1.015;   // Convert to gauge pressure
      // subtract atmospheric pressure
      Pressure = Pressure * 14.50377; // Convert bars to psi
      depth = Pressure * 2.23;

      return depth;
}

void sendCommand(byte command)
{
      Wire.beginTransmission(DevAddress);
      Wire.write(command);
      Wire.endTransmission();
}

//////////////////////////////////////////////////////////////////////////////
///////// initialize sonar sensor and retrieve data from the sensor /////////
//////////////////////////////////////////////////////////////////////////////
int sonar(int k)
{
      digitalWrite(4, HIGH);
      delayMicroseconds(15);
      digitalWrite(4, LOW);

      s = pulseIn(k, HIGH);

      return s;
}

///////////////////////////////////////////////////////////////////////////////
////// control thrusters based off sonar information to avoid collisions /////
///////////////////////////////////////////////////////////////////////////////
void sonarAction(int s1, int s2, int s3, int s4, int s5, int s6)
{
```

```
if ((s1 < 4000) || (s2 < 4000))
{
stopMotor();
}
if ((s1 < 2500) || (s2 < 2500))
{
ascend();
delay(3000);
stopMotor();
}
if (s3 < 4000)
{
stopMotor();
}
if (s3 < 2500)
{
moveForward();
delay(3000);
stopMotor();
}
if (s4 < 4000)
{
stopMotor();
}
if (s4 < 2500)
{
moveForward();
delay(3000);
stopMotor();
}
if (s5 < 4000)
{
stopMotor();
}
if (s5 < 2500)
{
moveBack();
delay(3000);
stopMotor();
}
if (s6 < 4000)
{
stopMotor();
}
if (s6 < 2500)
{
if ((s1 < 4000) || (s2 < 4000))
{
            moveForward();
            delay(3000);
            stopMotor();
      }
else
{
```

```
                  descend();
                  delay(3000);
                  stopMotor();
            }
      }
}

void stopMotor() // Power down all thrusters
{
      digitalWrite(5, LOW);
      digitalWrite(6, LOW);
      digitalWrite(9, LOW);
      digitalWrite(10, LOW);
      digitalWrite(11, LOW);
}

void ascend() // Instruct depth thrusters to ascend
{
analogWrite(5, 100);
      analogWrite(6, 100);
}

void descend() // Instruct depth thrusters(1 and 2)to descend
{
      analogWrite(5, 200);
      analogWrite(6, 200);
}

void moveForward() // Instruct xy thrusters(1 and 2) to move forward
{
      analogWrite(9, 100);
      analogWrite(10, 200);
}

void moveBack() // Instruct xy thrusters(3 and 4) to move backward
{
      analogWrite(9, 200);
      analogWrite(10, 100);
}

void turnRight() // instruct xy thrusters(3 and 4) to turn right
{
      analogWrite(9, 100);
      analogWrite(10, 100);
}

void turnLeft() // instruct xy thrusters(3 and 4) to turn left
{
      analogWrite(9, 200);
      analogWrite(10, 200);
}

void tiltBack() // instruct atuary thruster(5) to tilt back
{
```

```
        analogWrite(11, 200);
}

void tiltForward() // instruct atuary thruster(5) to tilt forward
{
        analogWrite(11, 100);
}

void tiltRight() // instruct left depth thruster(1) to tilt right
{
        analogWrite(5, 100);
}

void tiltLeft() // instruct right depth thruster(2) to tilt left
{
        analogWrite(6, 100);
}

void MPU9150_setupCompass()
{
        MPU9150_I2C_ADDRESS = 0x0C;    //change Address to Compass

        MPU9150_writeSensor(0x0A, 0x00); //PowerDownMode
        MPU9150_writeSensor(0x0A, 0x0F); //SelfTest
        MPU9150_writeSensor(0x0A, 0x00); //PowerDownMode

        MPU9150_I2C_ADDRESS = 0x69;    //change Adress to MPU

        MPU9150_writeSensor(0x24, 0x40); //Wait for Data at Slave0
        MPU9150_writeSensor(0x25, 0x8C); //Set i2c address at slave0 at 0x0C
        MPU9150_writeSensor(0x26, 0x02); //Set where reading at slave 0 starts
        MPU9150_writeSensor(0x27, 0x88); //set offset at start reading and enable
        MPU9150_writeSensor(0x28, 0x0C); //set i2c address at slv1 at 0x0C
        MPU9150_writeSensor(0x29, 0x0A); //Set where reading at slave 1 starts
        MPU9150_writeSensor(0x2A, 0x81); //Enable at set length to 1
        MPU9150_writeSensor(0x64, 0x01); //override register
        MPU9150_writeSensor(0x67, 0x03); //set delay rate
        MPU9150_writeSensor(0x01, 0x80);

        MPU9150_writeSensor(0x34, 0x04); //set i2c slv4 delay
        MPU9150_writeSensor(0x64, 0x00); //override register
        MPU9150_writeSensor(0x6A, 0x00); //clear usr setting
        MPU9150_writeSensor(0x64, 0x01); //override register
        MPU9150_writeSensor(0x6A, 0x20); //enable master i2c mode
        MPU9150_writeSensor(0x34, 0x13); //disable slv4
}

////////////////////////////////////////////////////////////
///////// I2C functions to get easier all values ///////////
////////////////////////////////////////////////////////////

int MPU9150_readSensor(int addrL, int addrH)
{
        Wire.beginTransmission(MPU9150_I2C_ADDRESS);
```

```
	Wire.write(addrL);
	Wire.endTransmission(false);

	Wire.requestFrom(MPU9150_I2C_ADDRESS, 1, true);
	byte L = Wire.read();

Wire.beginTransmission(MPU9150_I2C_ADDRESS);
	Wire.write(addrH);
	Wire.endTransmission(false);

	Wire.requestFrom(MPU9150_I2C_ADDRESS, 1, true);
	byte H = Wire.read();

	return (int16_t)((H<<8)+L);
}

int MPU9150_readSensor(int addr)
{
	Wire.beginTransmission(MPU9150_I2C_ADDRESS);
	Wire.write(addr);
	Wire.endTransmission(false);

	Wire.requestFrom(MPU9150_I2C_ADDRESS, 1, true);
	return Wire.read();
}

int MPU9150_writeSensor(int addr,int data)
{
	Wire.beginTransmission(MPU9150_I2C_ADDRESS);
	Wire.write(addr);
	Wire.write(data);
	Wire.endTransmission(true);

	return 1;
}
```

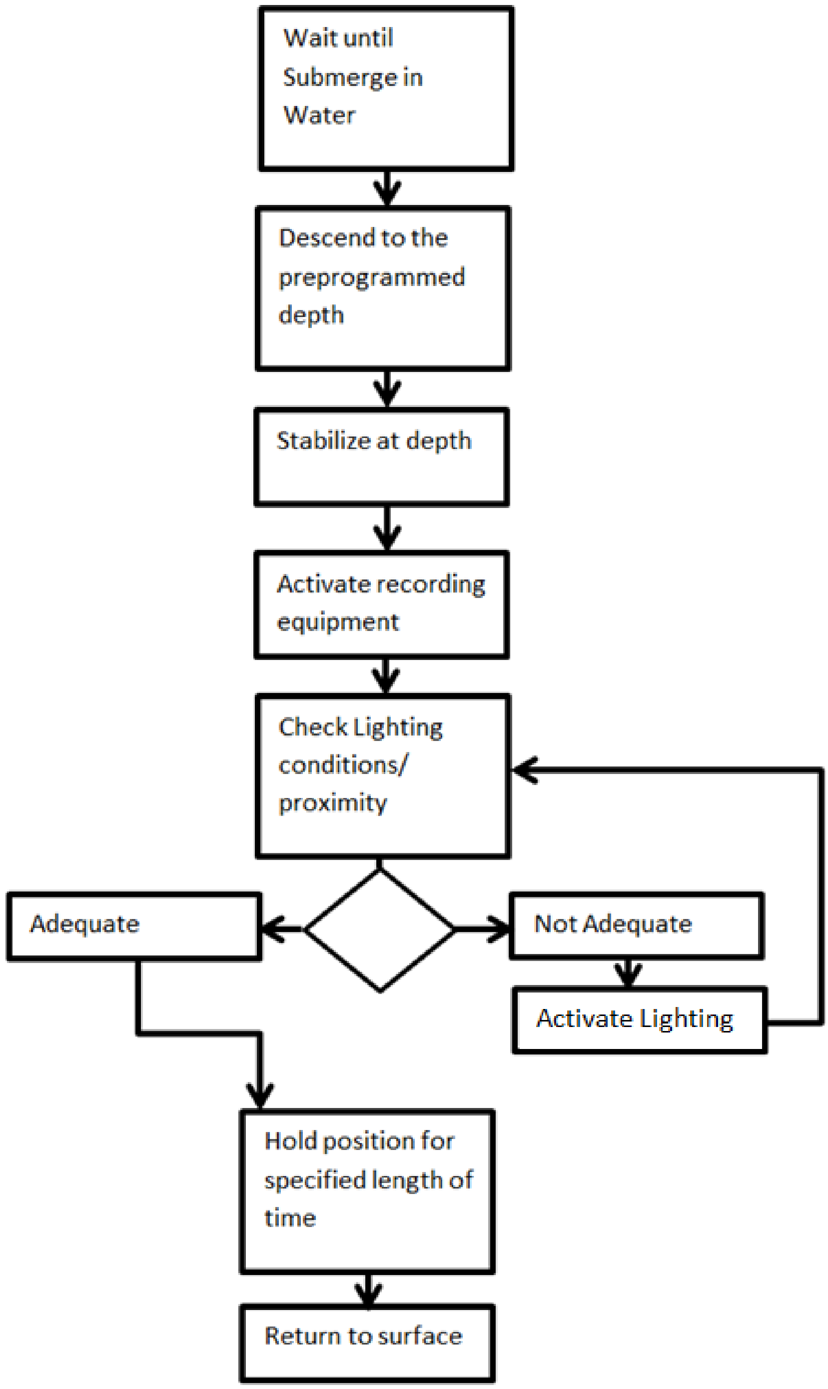
Appendix K: Final Source Code: Slave Controller

```
#include <Servo.h>

byte servoPin = 10;
Servo servo;
int i = 0;
int t1 = 0;
int t2 = 0;
int t3 = 0;
int t4 = 0;
int t5 = 0;
int times = 0;
int light = 0;
void setup()
{
      Serial.begin(9600);

pinMode(A0, INPUT);      // Light sensor input
      pinMode(22, OUTPUT);     // Light on/off output

pinMode(8, OUTPUT);      // Thruster 1 output
      pinMode(9, OUTPUT);      // Thruster 2 output
      pinMode(10, OUTPUT);     // Thruster 3 output
      pinMode(11, OUTPUT);     // Thruster 4 output
      pinMode(12, OUTPUT);     // Thruster 5 output

pinMode(2, INPUT);       // Thruster 1 command input
      pinMode(3, INPUT);       // Thruster 2 command input
      pinMode(4, INPUT);       // Thruster 3 command input
      pinMode(5, INPUT);       // Thruster 4 command input
      pinMode(6, INPUT);       // Thruster 5 command input

/////////////////////////////////////////////////////
////////// send "stop" signal to ESC.////////////////
/////////////////////////////////////////////////////

      digitalWrite(8, HIGH);
      digitalWrite(9, HIGH);
      digitalWrite(10, HIGH);
      digitalWrite(11, HIGH);
      digitalWrite(12, HIGH);

// delay to allow the ESC to recognize the stopped signal
      delay(1000);

      digitalWrite(8, LOW);
      digitalWrite(9, LOW);
      digitalWrite(10, LOW);
      digitalWrite(11, LOW);
      digitalWrite(12, LOW);
      delay(4000);
```

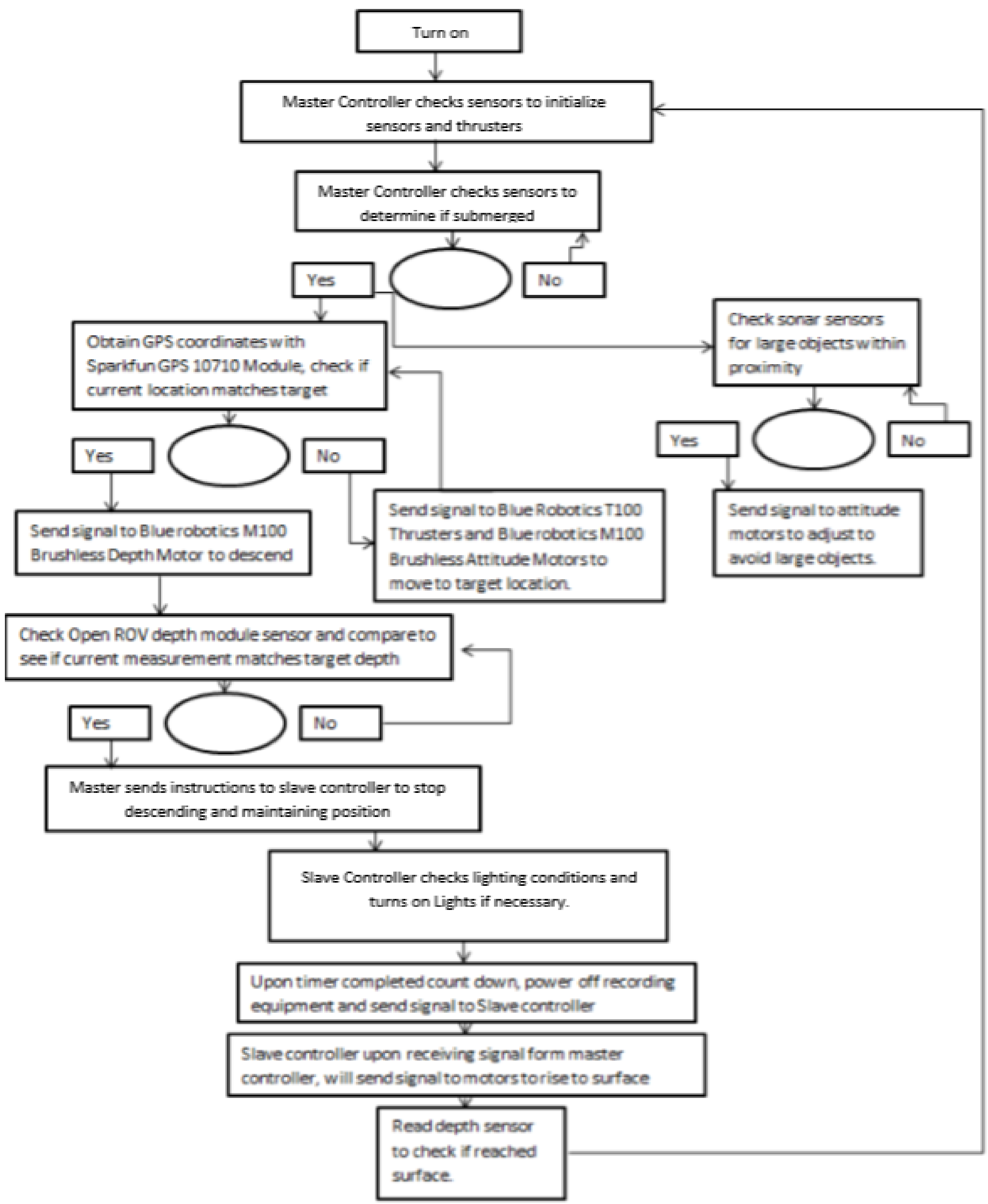
```
}

void loop()
{
      motorInitialize(8);      // Initialize Thruster 1
      motorInitialize(9);      // Initialize Thruster 2
      motorInitialize(10);     // Initialize Thruster 3
      motorInitialize(11);     // Initialize Thruster 4
      motorInitialize(12);     // Initialize Thruster 5
      t1 = signalRead(2);      // Read intructions for thruster 1
      t2 = signalRead(3);      // Read intructions for thruster 2
      t3 = signalRead(4);      // Read intructions for thruster 3
      t4 = signalRead(5);      // Read intructions for thruster 4
      t5 = signalRead(6);      // Read intructions for thruster 5

      motor(8, t1);
      motor(9, t2);
      motor(10, t3);
      motor(11, t4);
      motor(12, t5);

      if (determineLighting() == 0)
      {
      digitalWrite(22, LOW);
      }
      else if (determineLighting() == 1)
      {
      digitalWrite(22, HIGH);
      }


}

void motor(int l, int m)
{
      if (m == 1)
      {
      analogWrite(l, 125); // Stopped Signal
      }
      else if (m == 2)
      {
      analogWrite(l, 141); // Reverse Signal
      }
      else if (m == 3)
      {
      analogWrite(l, 153); // Forward signal
      }
}

void motorInitialize(int k)
{
      analogWrite(k, 125);
}
```

```
int signalRead(int j)
{
	int t = 0;
	times = pulseIn(j, HIGH);
	if (times ==0)
	{
	t = 1;
// Serial.println(t3);
	}
	else if (times > 1000)
	{
	t = 2;
// Serial.println(t3);
	}
	else if (times < 1000)
	{
	t = 3;
// Serial.println(t3);
	}
	return t;
}

int determineLighting()
{
	if (analogRead(A0) < 300)
	{
	light = 1;
	}
	else
	{
	light = 0;
	}
	return light;
}
```

Appendix L: General Process Diagram



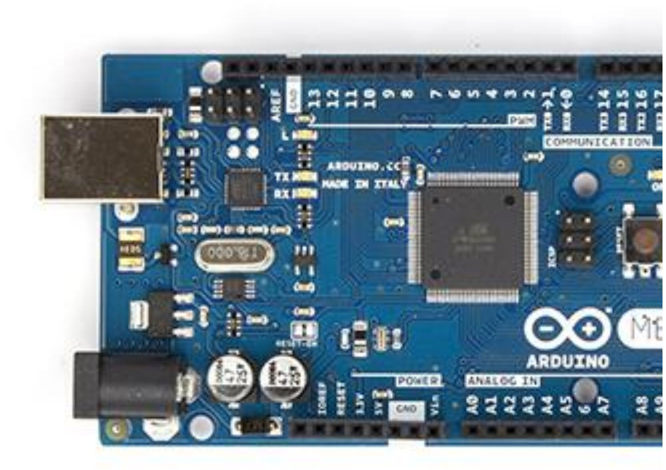
Appendix M: High-Level Software Flow

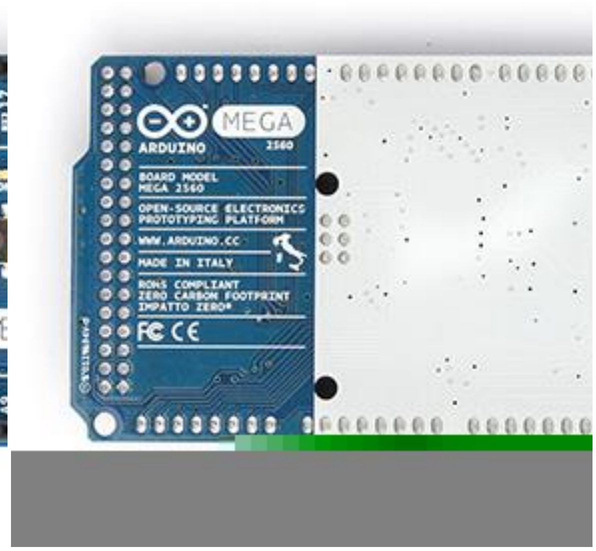


Appendix N: Electrical Specification Sheets

See attached.

Arduino Mega 2560





Arduino Mega 2560 R3 Front

Arduino Mega2560 R3 Back

Overview

The Arduino Mega 2560 is a microcontroller board based on the ATmega2560 (datasheet). It has 54 digital input/output pins (of which 15 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Mega is compatible with most shields designed for the Arduino Duemilanove or Diecimila.

The Mega 2560 is an update to the Arduino Mega, which it replaces.

The Mega2560 differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the ATmega16U2 (ATmega8U2 in the revision 1 and revision 2 boards) programmed as a USB-to-serial converter.

Revision 2 of the Mega2560 board has a resistor pulling the 8U2 HWB line to ground, making it easier to put into DFU mode.

Revision 3 of the board has the following new features:

- 1.0 pinout: added SDA and SCL pins that are near to the AREF pin and two other new pins placed near to the RESET pin, the IOREF that allow the shields to adapt to the voltage provided from the board. In future, shields will be compatible both with the board that use the AVR, which operate with 5V and with the Arduino Due that operate with 3.3V. The second one is a not connected pin, that is reserved for future purposes.
- Stronger RESET circuit.
- Atmega 16U2 replace the 8U2.

Schematic, Reference Design & Pin Mapping

EAGLE files: arduino-mega2560_R3-reference-design.zip

Schematic: arduino-mega2560_R3-schematic.pdf

Pin Mapping: PinMap2560 page

Summary

Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	54 (of which 15 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	40 mA

DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz

Power

The Arduino Mega can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

- VIN. The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.

- 5V. This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 12V), the USB connector (5V), or the VIN pin of the board (7-12V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage your board. We don't advise it.
- 3V3. A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- GND. Ground pins.
- IOREF. This pin on the Arduino board provides the voltage reference with which the microcontroller operates. A properly configured shield can read the IOREF pin voltage and select the appropriate power source or enable voltage translators on the outputs for working with the 5V or 3.3V.

Memory

The ATmega2560 has 256 KB of flash memory for storing code (of which 8 KB is used for the bootloader), 8 KB of SRAM and 4 KB of EEPROM (which can be read and written with the EEPROM library).

Input and Output

Each of the 54 digital pins on the Mega can be used as an input or output, using pinMode(), digitalWrite(), anddigitalRead() functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- Serial: 0 (RX) and 1 (TX); Serial 1: 19 (RX) and 18 (TX); Serial 2: 17 (RX) and 16 (TX); Serial 3: 15 (RX) and 14 (TX). Used to receive (RX) and transmit (TX) TTL serial data. Pins 0 and 1 are also connected to the corresponding pins of the ATmega16U2 USB-to-TTL Serial chip.
- External Interrupts: 2 (interrupt 0), 3 (interrupt 1), 18 (interrupt 5), 19 (interrupt 4), 20 (interrupt 3), and 21 (interrupt 2). These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the attachInterrupt()function for details.

- PWM: 2 to 13 and 44 to 46. Provide 8-bit PWM output with the analogWrite() function.
- SPI: 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS). These pins support SPI communication using the SPI library. The SPI pins are also broken out on the ICSP header, which is physically compatible with the Uno, Duemilanove and Diecimila.
- LED: 13. There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.
- TWI: 20 (SDA) and 21 (SCL). Support TWI communication using the Wire library. Note that these pins are not in the same location as the TWI pins on the Duemilanove or Diecimila.

The Mega2560 has 16 analog inputs, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and analogReference() function.

There are a couple of other pins on the board:

- AREF. Reference voltage for the analog inputs. Used with analogReference().
- Reset. Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

Communication

The Arduino Mega2560 has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers.
The ATmega2560 provides four hardware UARTs for TTL (5V) serial communication. An ATmega16U2(ATmega 8U2 on the revision 1 and revision 2 boards) on the board channels one of these over USB and provides a virtual com port to software on the computer (Windows machines will need a .inf file, but OSX and Linux

machines will recognize the board as a COM port automatically. The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the board. The RX and TX LEDs on the board will flash when data is being transmitted via the ATmega8U2/ATmega16U2 chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A SoftwareSerial library allows for serial communication on any of the Mega2560's digital pins.

The ATmega2560 also supports TWI and SPI communication. The Arduino software includes a Wire library to simplify use of the TWI bus; see the documentation for details. For SPI communication, use the SPI library.

Programming

The Arduino Mega can be programmed with the Arduino software (download). For details, see the reference andtutorials.

The ATmega2560 on the Arduino Mega comes preburned with a bootloader that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol (reference, C header files).

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header using Arduino ISP or similar; see these instructions for details.

The ATmega16U2 (or 8U2 in the rev1 and rev2 boards) firmware source code is available in the Arduino repository. TheATmega16U2/8U2 is loaded with a DFU bootloader, which can be activated by:

- On Rev1 boards: connecting the solder jumper on the back of the board (near the map of Italy) and then resetting the 8U2.
- On Rev2 or later boards: there is a resistor that pulling the 8U2/16U2 HWB line to ground, making it easier to put into DFU mode. You can then use Atmel's FLIP software (Windows) or the DFU programmer (Mac OS X and Linux) to load a new firmware. Or you can use the ISP header with an external programmer (overwriting the DFU bootloader). See this user-contributed tutorial for more information.

Automatic (Software) Reset

Rather then requiring a physical press of the reset button before an upload, the Arduino Mega2560 is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2 is connected to the reset line of the ATmega2560 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the bootloader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload.

This setup has other implications. When the Mega2560 is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the bootloader is running on the Mega2560. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data.

The Mega2560 contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line; see this forum thread for details.

USB Overcurrent Protection

The Arduino Mega2560 has a resettable polyfuse that protects your computer's USB ports from shorts and overcurrent. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

Physical Characteristics and Shield Compatibility

The maximum length and width of the Mega2560 PCB are 4 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Three screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 is 160 mil (0.16"), not an even multiple of the 100 mil spacing of the other pins.

The Mega2560 is designed to be compatible with most shields designed for the Uno, Diecimila or Duemilanove. Digital pins 0 to 13 (and the adjacent AREF and GND pins), analog inputs 0 to 5, the power header, and ICSP header are all in equivalent locations. Further the main UART (serial port) is located on the same pins (0 and 1), as are external interrupts 0 and 1 (pins 2 and 3 respectively). SPI is available through the ICSP header on both the Mega2560 and Duemilanove / Diecimila. *Please note that I^2C is not located on the same pins on the Mega (20 and 21) as the Duemilanove / Diecimila (analog inputs 4 and 5).*

Arduino Uno

Arduino Uno R3 Front

Arduino Uno R3 Back

Overview

The Arduino Uno is a microcontroller board based on the ATmega328 (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started.

The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.
Revision 2 of the Uno board has a resistor pulling the 8U2 HWB line to ground, making it easier to put into DFU mode.
Revision 3 of the board has the following new features:

- 1.0 pinout: added SDA and SCL pins that are near to the AREF pin and two other new pins placed near to the RESET pin, the IOREF that allow the shields to adapt to the voltage provided from the board. In future, shields will be compatible with both the board that uses the AVR, which operates with 5V and with the Arduino Due that operates with 3.3V. The second one is a not connected pin, that is reserved for future purposes.
- Stronger RESET circuit.
- Atmega 16U2 replace the 8U2.

"Uno" means one in Italian and is named to mark the upcoming release of Arduino 1.0. The Uno and version 1.0 will be the reference versions of Arduino, moving forward. The Uno is the latest in a series of USB Arduino boards, and the reference model for the Arduino platform; for a comparison with previous versions, see the index of Arduino boards.

Summary

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328)

EEPROM	1 KB (ATmega328)
Clock Speed	16 MHz
Length	68.6 mm
Width	53.4 mm
Weight	25 g

Schematic & Reference Design

EAGLE files: arduino-uno-Rev3-reference-design.zip (NOTE: works with Eagle 6.0 and newer)

Schematic: arduino-uno-Rev3-schematic.pdf

Note: The Arduino reference design can use an Atmega8, 168, or 328, Current models use an ATmega328, but an Atmega8 is shown in the schematic for reference. The pin configuration is identical on all three processors.

Power

The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

- VIN. The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- 5V.This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 12V), the USB connector (5V), or the VIN pin of the board (7-12V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage your board. We don't advise it.
- 3V3. A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- GND. Ground pins.
- IOREF. This pin on the Arduino board provides the voltage reference with which the microcontroller operates. A properly configured shield can read the IOREF pin voltage and select the appropriate power source or enable voltage translators on the outputs for working with the 5V or 3.3V.

Memory

The ATmega328 has 32 KB (with 0.5 KB used for the bootloader). It also has 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the EEPROM library).

Input and Output

Each of the 14 digital pins on the Uno can be used as an input or output, using pinMode(), digitalWrite(), anddigitalRead() functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- Serial: 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- External Interrupts: 2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the attachInterrupt() function for details.
- PWM: 3, 5, 6, 9, 10, and 11. Provide 8-bit PWM output with the analogWrite() function.
- SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication using the SPI library.
- LED: 13. There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Uno has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and the analogReference() function. Additionally, some pins have specialized functionality:

- TWI: A4 or SDA pin and A5 or SCL pin. Support TWI communication using the Wire library.

There are a couple of other pins on the board:

- AREF. Reference voltage for the analog inputs. Used with analogReference().
- Reset. Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

See also the mapping between Arduino pins and ATmega328 ports. The mapping for the Atmega8, 168, and 328 is identical.

Communication

The Arduino Uno has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers.
The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX).
An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The '16U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, on Windows, a .inf file is required. The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A SoftwareSerial library allows for serial communication on any of the Uno's digital pins.

The ATmega328 also supports I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the documentation for details. For SPI communication, use the SPI library.

Programming

The Arduino Uno can be programmed with the Arduino software (download). Select "Arduino Uno from the Tools > Board menu (according to the microcontroller on your board). For details, see the reference and tutorials.

The ATmega328 on the Arduino Uno comes preburned with a bootloader that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol (reference, C header files).

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header using Arduino ISP or similar; see these instructions for details.

The ATmega16U2 (or 8U2 in the rev1 and rev2 boards) firmware source code is available . The ATmega16U2/8U2 is loaded with a DFU bootloader, which can be activated by:

- On Rev1 boards: connecting the solder jumper on the back of the board (near the map of Italy) and then resetting the 8U2.
- On Rev2 or later boards: there is a resistor that pulling the 8U2/16U2 HWB line to ground, making it easier to put into DFU mode.

You can then use Atmel's FLIP software (Windows) or the DFU programmer (Mac OS X and Linux) to load a new firmware. Or you can use the ISP header with an external programmer (overwriting the DFU bootloader). See this user-contributed tutorial for more information.

Automatic (Software) Reset

Rather than requiring a physical press of the reset button before an upload, the Arduino Uno is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of theATmega8U2/16U2 is connected to the reset line of the ATmega328 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the bootloader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload.

This setup has other implications. When the Uno is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the bootloader is running on the Uno. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data.

The Uno contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line; see this forum thread for details.

USB Overcurrent Protection

The Arduino Uno has a resettable polyfuse that protects your computer's USB ports from shorts and overcurrent. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

Physical Characteristics

The maximum length and width of the Uno PCB are 2.7 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Four screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 is 160 mil (0.16"), not an even multiple of the 100 mil spacing of the other pins.

GLOBALSAT GPS Module

Hardware Data Sheet

Product No : EM-506

User Manual Version 1.4

Globalsat Technology Corporation
16F., No. 186, Jian-Yi Road, Chung-Ho City, Taipei Hsien 235, Taiwan
Tel: 886-2-8226-3799 Fax: 886-2-8226-3899
E-mail : service@globalsat.com.tw
Website: www.globalsat.com.tw

Issue Date	APPR	CHECK	PREPARE
2013/10/08	Ray		Mason

Product Description

Product Description

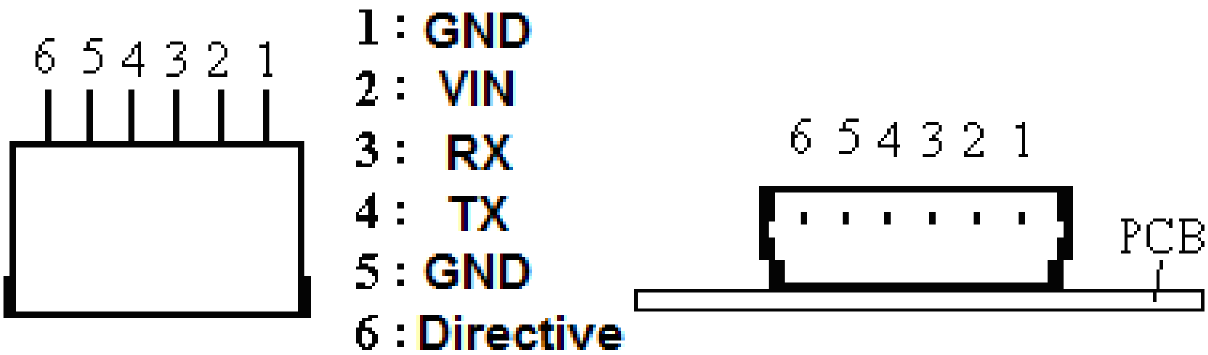
EM-506 GPS module features high sensitivity, low power and ultra small form factor. This GPS module is powered by SiRF Star IV, it can provide you with superior sensitivity and performance even in urban canyon and dense foliage environment. With SiRF CGEE (Client Generated Extended Ephemeris) technology, it predicts satellite positions for up to 3 days and delivers CGEE-start time of less than 15 seconds under most conditions, without any network assistance. Besides, MicroPower Mode allows GPS module to stay in a hot-start condition nearly continuously while consuming very little power. EM-506 is suitable for the following applications:

- Automotive navigation
- Fleet management
- Marine navigation

Product Features

- SiRF Star IV high performance GPS Chipset
- Very high sensitivity (Tracking Sensitivity: -163 dBm)
- Extremely fast TTFF (Time To First Fix) at low signal level
- Support UART interface.
- Built-in LNA(with in CHIP)
- Compact size (30.0mm x 30.0 mm x 10.7mm) suitable for space-sensitive application
- Support NMEA 0183 V3.0 (GGA, GSA, GSV, RMC, VTG, GLL, ZDA)
- Support OSP protocol
- Support SBAS (WASS, EGNOS, MSAS, GAGAN)

Product Pin Description



PIN Number(s)	Name	Type	Description	Note
1,5	GND	P	Ground.	
2	VIN	P	This is the main power supply to the engine board. (4.5Vdc to 6.5Vdc)	
3	RXD	I	This is the main receive channel for receiving software commands to the engine board from SiRFdemo software or from user written software. Baud rate based on flash memory setting.	
4	TXD	O	This is the main transmits channel for outputting navigation and measurement data to user's navigation software or user written software. Output 3.3V level.	
6	Directive	O	This pin indicates the GPS states.	

Electrical Specification

Absolute Maximums Ratings

Parameter	Min.	Typ.	Max.	Conditions	Unit
POWER Supply					
Main power supply(VCC)	4.5	5.0	6.5		V
Main power supply Current	45	50	55	GPS is not 3D Fixed.	mA
	33	34	38	GPS is 3D Fixed.	mA
RF					
Operating Frequency		1.575			Ghz

DC Electrical characteristics

Parameter	Symbol	Min.	Typ.	Max.	Conditions	Units
I/O Low Level Output Voltage	V_{OL}			0.4		V
I/O High Level Output Voltage	V_{OH}		3.3			V
I/O Low Level Input Voltage	V_{IL}	-0.4		0.45		V
I/O High Level Input Voltage	V_{IH}	1.26		3.6		V
High Level Output Current	I_{OH}		2			mA
Low Level Output Current	I_{OL}		2			mA

Receiver Performance

Sensitivity	Tracking : Autonomous acquisition :	-163dBm -160 dBm
Time-To-First-Fix[1]	Cold Start – Autonomous	< 35s
		<15s (with CGEE)
	Warm Start – Autonomous[2]	< 35s
		< 15s(with CGEE)
	Hot Start – Autonomous[3]	< 1s
Horizontal Position Accuracy[4]	Autonomous	< 2.5m
Velocity Accuracy[5]	Speed Heading	< 0.01 m/s < 0.01 degrees
Reacquisition	0.1 second, average	
Update Rate	1 Sec / 5 Sec	
Maximum Altitude	< 18,000 meter	
Maximum Velocity	< 515 meter/ second	
Maximum Acceleration	< 4G	

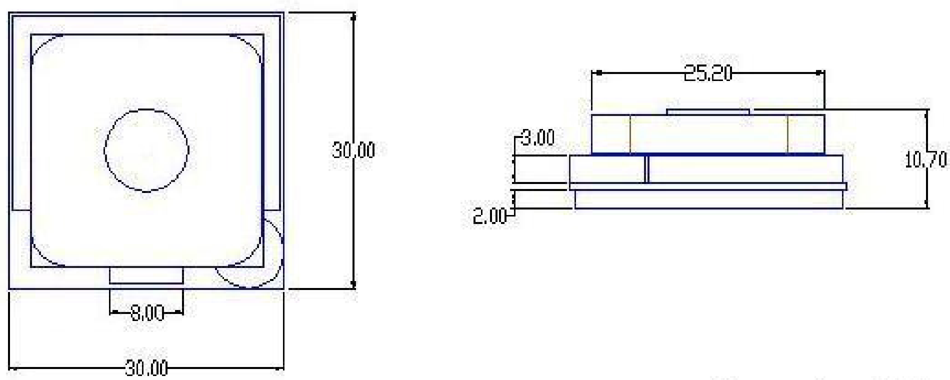
<Note>

1. 50% -130dBm Fu 0.5ppm Tu ±2s Pu 30Km
2. Commanded **Warm START**
3. Commanded **Hot START**
4. 50% 24hr static, -130dBm
5. 50% @ 30m/s

Environmental Characteristics

Parameter	Min	Typ	Max	Unit
Humidity Range	5		95	% non-condensing
Operation Temperature	-40	25	85	℃
Storage Temperature	-40		85	℃

Physical Characteristic



Dimension ±0.2mm

OPERATING Description

GND

This is Ground pin for the baseband circuit.

VIN

This is the main power supply to the engine board. (4.5Vdc to 6.5Vdc)

RXD

This is the main channel for receiving software commands from SiRFdemo software or from your proprietary software.

TXD

This is the main transmits channel for outputting navigation and measurement data to user's navigation software or user written software. Output is TTL level, 0V ~ 3.3V.

Directive

This pin exports signal to indicate the GPS states.
GPS unfix: always low level.
GPS fixed: one second high level, one second low level.

LED

LED indicator for GPS fix or not fix

LED OFF:	Receiver switch off
LED ON:	No fixed, Signal searching
LED Flashing:	Position Fixed

SOFTWARE COMMAND

NMEA Output Command

GGA - Global Positioning System Fixed Data

Note – Fields marked in italic *red* apply only to NMEA version 2.3 (and later) in this NMEA message description

Table B-2 contains the values for the following example:

$GPGGA,161229.487,3723.2475,N,12158.3416,W,1,07,1.0,9.0,M,-34.2,M,,0000*18

Table B-2 GGA Data Format

Name	Example	Units	Description
Message ID	$GPGGA		GGA protocol header
UTC Time	161229.487		hhmmss.sss
Latitude	3723.2475		ddmm.mmmm
N/S Indicator	N		N=north or S=south
Longitude	12158.3416		dddmm.mmmm
E/W Indicator	W		E=east or W=west
Position Fix Indicator	1		See Table B-3
Satellites Used	07		Range 0 to 12
HDOP	1.0		Horizontal Dilution of Precision
MSL Altitude	9.0	meters	
Units	M	meters	
Geoid Separation[1]	-34.2	meters	Geoid-to-ellipsoid separation. Ellipsoid altitude=MSL Altitude + Geoid Separation
Units	M	meters	
Age of Diff. Corr.		second	Null fields when DGPS is not used
Diff. Ref. Station ID	0000		
Checksum	*18		
<CR><LF>			End of message termination

Table B-3 Position Fix Indicator

Value	Description
0	Fix not available or invalid
1	GPS SPS Mode, fix valid
2	Differential GPS, SPS Mode , fix valid
3	Not supported
6	*Dead Reckoning Mode, fix valid*

Note:

A valid status is derived from all the parameters set in the software. This includes the minimum number of satellites required, any DOP mask setting, presence of DGPS corrections, etc. If the default or current software setting requires that a factor is met, then if that factor is not met the solution will be marked as invalid.

GLL - Geographic Position-Latitude/Longitude

Note – Fields marked in italic *red* apply only to NMEA version 2.3 (and later) in this NMEA message description

Table B-4 contains the values for the following example:

$GPGLL,3723.2475,N,12158.3416,W,161229.487,A,A*41

Table B-4 GLL Data Format

Name	Example	Units	Description
Message ID	$GPGLL		GLL protocol header
Latitude	3723.2475		ddmm.mmmm
N/S Indicator	n		N=north or S=south
Longitude	12158.3416		dddmm.mmmm
E/W Indicator	W		E=east or W=west
UTC Position	161229.487		hhmmss.sss
Status	A		A=data valid or V=data not valid
Mode	*A*		*A=Autonomous, D=DGPS,* *E=DR* *N=Output Data Not Valid* *R= Coarse Position*[1] *S=Simulator*
Checksum	*41		
<CR><LF>			End of message termination

1. Position was calculated based on one or more of the SVs having their states derived from almanac parameters, as opposed to ephemerides.

GSA - GNSS DOP and Active Satellites

Table B-5 contains the values for the following example:

$GPGSA,A,3,07,02,26,27,09,04,15,,,,,,1.8,1.0,1.5*33

Table B-5 GSA Data Format

Name	Example	Units	Description
Message ID	$GPGSA		GSA protocol header
Mode 1	A		See Table B-6
Mode 2	3		See Table B-7
Satellite Used[1]	07		Sv on Channel 1
Satellite Used[1]	02		Sv on Channel 2
…..			
Satellite Used[1]			Sv on Channel 12
PDOP[2]	1.8		Position dilution of Precision
HDOP[2]	1.0		Horizontal dilution of Precision
VDOP[2]	1.5		Vertical dilution of Precision
Checksum	*33		
<CR><LF>			End of message termination

1. Satellite used in solution.
2. Maximum DOP value reported is 50. When 50 is reported, the actual DOP may be much larger.

Table B-6 Mode1

Value	Description
M	Manual-forced to operate in 2D or 3D mode
A	2Dautomatic-allowed to automatically switch 2

Table B-7 Mode 2

Value	Description
1	Fix Not Available
2	2D (<4 SVs used)
3	3D (>3 SVs used)

GSV - GNSS Satellites in View

Table B-8 contains the values for the following example:

$GPGSV,2,1,07,07,79,048,42,02,51,062,43,26,36,256,42,27,27,138,42*71

$GPGSV,2,2,07,09,23,313,42,04,19,159,41,15,12,041,42*41

Table B-8 GSV Data Format

Name	Example	Units	Description
Message ID	$GPGSV		GSV protocol header
Number of Messages[1]	2		Range 1 to 3
Message Number[1]	1		Range 1 to 3
Satellites in View[1]	07		
Satellite ID	07		Channel 1(Range 1 to 32)
Elevation	79	degrees	Channel 1(Maximum90)
Azimuth	048	degrees	Channel 1(True, Range 0 to 359)
SNR(C/No)	42	dBHz	Range 0 to 99,null when not tracking
…….			…….
Satellite ID	27		Channel 4 (Range 1 to 32)
Elevation	27	Degrees	Channel 4(Maximum90)
Azimuth	138	Degrees	Channel 4(True, Range 0 to 359)
SNR(C/No)	42	dBHz	Range 0 to 99,null when not tracking
Checksum	*71		
<CR><LF>			End of message termination

1. Depending on the number of satellites tracked, multiple messages of GSV data may be required. In some software versions, the maximum number of satellites reported as visible is limited to 12, even though more may be visible.

RMC - Recommended Minimum Specific GNSS Data

Note – Fields marked in italic *red* apply only to NMEA version 2.3 (and later) in this NMEA message description

Table B-9 contains the values for the following example:

$GPRMC,161229.487,A,3723.2475,N,12158.3416,W,0.13,309.62,120598,,,A*10

Table B-9 RMC Data Format

Name	Example	Units	Description
Message ID	$GPRMC		RMC protocol header
UTC Time	161229.487		hhmmss.sss
Status[1]	A		A=data valid or V=data not valid
Latitude	3723.2475		ddmm.mmmm
N/S Indicator	N		N=north or S=south
Longitude	12158.3416		dddmm.mmmm
E/W Indicator	W		E=east or W=west
Speed Over Ground	0.13	knots	
Course Over Ground	309.62	degrees	True
Date	120598		ddmmyy
Magnetic Variation[2]		degrees	E=east or W=west
East/West Indicator[2]	E		E=east
Mode	*A*		*A=Autonomous, D=DGPS,* *E=DR* *N=Output Data Not Valid* *R= Coarse Position[3]* *S=Simulator*
Checksum	*10		
<CR><LF>			End of message termination

1. A valid status is derived from all the parameters set in the software. This includes the minimum number of satellites required, any DOP mask setting, presence of DGPS corrections, etc. If the default or current software setting requires that a factor is met, then if that factor is not met the solution will be marked as invalid.
2. SiRF Technology Inc. does not support magnetic declination. All “course over ground” data are geodetic WGS84 directions relative to true North.
3. Position was calculated based on one or more of the SVs having their states derived from almanac parameters, as opposed to ephemerides.

VTG - Course Over Ground and Ground Speed

Note – Fields marked in italic *red* apply only to NMEA version 2.3 (and later) in this NMEA message description

Table B-10 contains the values for the following example:

$GPVTG,309.62,T,,M,0.13,N,0.2,K,A*23

Table B-10 VTG Data Format

Name	Example	Units	Description
Message ID	$GPVTG		VTG protocol header
Course	309.62	degrees	Measured heading
Reference	T		True
Course		degrees	Measured heading
Reference	M		Magnetic[1]
Speed	0.13	knots	Measured horizontal speed
Units	N		Knots
Speed	0.2	Km/hr	Measured horizontal speed
Units	K		Kilometers per hour
Mode	*A*		*A=Autonomous, D=DGPS,* *E=DR* *N=Output Data Not Valid* *R= Coarse Position*[2] *S=Simulator*
Checksum	*23		
<CR><LF>			End of message termination

1. SiRF Technology Inc. does not support magnetic declination. All “course over ground” data are geodetic WGS84 directions.
2. Position was calculated based on one or more of the SVs having their states derived from almanac parameters, as opposed to ephemerides.

ZDA - Time and Date

This message is included only with systems which support a time-mark output pulse identified as "1PPS". Outputs the time associated with the current 1PPS pulse. Each message is output within a few hundred ms after the 1PPS pulse is output and tells the time of the pulse that just occurred.

Table B-11 contains the values for the following example:

$GPZDA,181813,14,10,2003,,*4F<CR><LF>

Table B-11: ZDA Data Format

Name	Example	Unit	Description
Message ID	$GPZDA		ZDA protocol header
UTC Time	181813	hhmmss	The UTC time units are: hh=UTC hours from 00 to 23 mm=UTC minutes from 00 to 59 ss=UTC seconds from 00 to 59 Either using valid IONO/UTC or estimated from default leap seconds
Day	14		Day of the month, range 1 to 31
Month	10		Month of the year, range 1 to 12
Year	2003		Year
Local zone hour[1]		hour	Offset from UTC (set to 00)
Local zone minutes[1]		minute	Offset from UTC (set to 00)
Checksum	*4F		
<CR><LF>			End of message termination

1. Not supported by CSR, reported as 00.

NMEA Input Command

A). Set Serial Port ID: 100 Set PORTA parameters and protocol

This command message is used to set the protocol (SiRF Binary, NMEA, or USER1) and/or the communication parameters (baud, data bits, stop bits, parity). Generally, this command would be used to switch the module back to SiRF Binary protocol mode where a more extensive command message set is available. For example, to change navigation parameters. When a valid message is received, the parameters will be stored in battery backed SRAM and then the receiver will restart using the saved parameters.

Format:

$PSRF100,<protocol>,<baud>,<DataBits>,<StopBits>,<Parity>*CKSUM<CR><LF>

<protocol>	0=SiRF Binary, 1=NMEA, 4=USER1
<baud>	1200, 2400, 4800, 9600, 19200, 38400
<DataBits>	8,7. Note that SiRF protocol is only valid f8 Data bits
<StopBits>	0,1
<Parity>	0=None, 1=Odd, 2=Even

Example 1: Switch to SiRF Binary protocol at 9600,8,N,1

$PSRF100,0,9600,8,1,0*0C<CR><LF>

Example 2: Switch to User1 protocol at 38400,8,N,1

$PSRF100,4,38400,8,1,0*38<CR><LF>

**Checksum Field: The absolute value calculated by exclusive-OR the 8 data bits of each character in the Sentence, between, but excluding “$” and “*”. The hexadecimal value of the most significant and least significant 4 bits of the result are converted to two ASCII characters (0-9, A-F) for transmission. The most significant character is transmitted first.

**<CR><LF> : Hex 0D 0A

B). Navigation Initialization ID：101 Parameters required for start

This command is used to initialize the module for a warm start, by providing current position（in X, Y, Z coordinates）,clock offset, and time. This enables the receiver to search for the correct satellite signals at the correct signal parameters. Correct initialization parameters will enable the receiver to acquire signals more quickly, and thus, produce a faster navigational solution.

When a valid Navigation Initialization command is received, the receiver will restart using the input parameters as a basis for satellite selection and acquisition.

Format：

$PSRF101,<X>,<Y>,<Z>,<ClkOffset>,<TimeOfWeek>,<WeekNo>,<chnlCount>,<ResetCfg>*CKSUM<CR><LF>

<X>	X coordinate position INT32
<Y>	Y coordinate position INT32
<Z>	Z coordinate position INT32
<ClkOffset>	Clock offset of the receiver in Hz, Use 0 for last saved value if available. If this is unavailable, a default value of 75000 for GSP1, 95000 for GSP 1/LX will be used. INT32
<TimeOf Week>	GPS Time Of Week UINT32
<WeekNo>	GPS Week Number UINT16 （ Week No and Time Of Week calculation from UTC time）
<chnlCount>	Number of channels to use.1-12. If your CPU throughput is not high enough, you could decrease needed throughput by reducing the number of active channels UBYTE
<ResetCfg>	bit mask 0×01=Data Valid warm/hot start=1 0×02=clear ephemeris warm start=1 0×04=clear memory. Cold start=1 UBYTE

Example: Start using known position and time.

$ PSRF101,-2686700,-4304200,3851624,96000,497260,921,12,3*7F

C). Set DGPS Port ID: 102 Set PORT B parameters for DGPS input

This command is used to control Serial Port B that is an input only serial port used to receive RTCM differential corrections.

Differential receivers may output corrections using different communication parameters. The default communication parameters for PORT B are 9600Baud, 8data bits, 0 stop bits, and no parity. If a DGPS receiver is used which has different communication parameters, use this command to allow the receiver to correctly decode the data. When a valid message is received, the parameters will be stored in battery backed SRAM and then the receiver will restart using the saved parameters.

Format:

＄PSRF102,<Baud>,<DataBits>,<StopBits>,<Parity>*CKSUM<CR><LF>

<baud>	1200,2400,4800,9600,19200,38400
<DataBits>	8
<StopBits>	0,1
<Parity>	0=None, Odd=1,Even=2

Example: Set DGPS Port to be 9600,8,N,1

＄PSRF102,9600,8,1.0*12

D). Query/Rate Control ID: 103 Query standard NMEA message and/or set output rate

This command is used to control the output of standard NMEA message GGA, GLL, GSA, GSV, RMC, VTG. Using this command message, standard NMEA message may be polled once, or setup for periodic output. Checksums may also be enabled or disabled depending on the needs of the receiving program. NMEA message settings are saved in battery backed memory for each entry when the message is accepted.

Format:

＄PSRF103,<msg>,<mode>,<rate>,<cksumEnable>*CKSUM<CR><LF>

<msg>	0=GGA, 1=GLL, 2=GSA, 3=GSV, 4=RMC, 5=VTG 6=MSS(if internal beacon is supported) 7=Not defined 8=ZDA(if 1PPS output supported) 9=Not defined
<mode>	0=SetRate 1=Query 2=ABP On 3=ABP Off
<rate>	Output every <rate>seconds, off=0,max=255
<cksumEnable>	0=disable Checksum,1=Enable checksum for specified message

Example 1: Query the GGA message with checksum enabled

＄PSRF103,00,01,00,01*25

Example 2: Enable VTG message for a 1Hz constant output with checksum enabled

$PSRF103,05,00,01,01*20

Example 3: Disable VTG message

$PSRF103,05,00,00,01*21

E). LLA Navigation Initialization ID: 104 Parameters required to start using Lat/Lon/Alt

This command is used to initialize the module for a warm start, by providing current position (in Latitude, Longitude, Altitude coordinates), clock offset, and time. This enables the receiver to search for the correct satellite signals at the correct signal parameters. Correct initialization parameters will enable the receiver to acquire signals more quickly, and thus, will produce a faster navigational soution.

When a valid LLA Navigation Initialization command is received, the receiver will restart using the input parameters as a basis for satellite selection and acquisition.

Format:

$PSRF104,<Lat>,<Lon>,<Alt>,<ClkOffset>,<TimeOfWeek>,<WeekNo>,<ChannelCount>,
<ResetCfg>*CKSUM<CR><LF>

<Lat>	Latitude position, assumed positive north of equator and negative south of equator float, possibly signed
<Lon>	Longitude position, it is assumed positive east of Greenwich and negative west of Greenwich Float, possibly signed
<Alt>	Altitude position float, possibly signed
<ClkOffset>	Clock Offset of the receiver in Hz, use 0 for last saved value if available. If this is unavailable, a default value of 75000 for GSP1, 95000 for GSP1/LX will be used. INT32
<TimeOfWeek>	GPS Time Of Week UINT32
<WeekNo>	GPS Week Number UINT16
<ChannelCount>	Number of channels to use. 1-12 UBYTE
<ResetCfg>	bit mask 0×01=Data Valid warm/hot starts=1 0×02=clear ephemeris warm start=1 0×04=clear memory. Cold start=1 UBYTE

Example: Start using known position and time.

$PSRF104,37.3875111,-121.97232,0,96000,237759,922,12,3*37

F). Development Data On/Off ID: 105 Switch Development Data Messages On/Off

Use this command to enable development debug information if you are having trouble getting commands accepted. Invalid commands will generate debug information that should enable the user to determine the source of the command rejection. Common reasons for input command rejection are invalid checksum or parameter out of specified range. This setting is not preserved across a module reset.

Format: $ PSRF105,<debug>*CKSUM<CR><LF>

<debug> 0=Off, 1=On

Example: Debug On $ PSRF105,1*3E

Example: Debug Off $ PSRF105,0*3F

G). Select Datum ID: 106 Selection of datum to be used for coordinate Transformations

GPS receivers perform initial position and velocity calculations using an earth-centered earth-fixed (ECEF) coordinate system. Results may be converted to an earth model (geoid) defined by the selected datum. The default datum is WGS 84 (World Geodetic System 1984) which provides a worldwide common grid system that may be translated into local coordinate systems or map datums. (Local map datums are a best fit to the local shape of the earth and not valid worldwide.)

Examples:

Datum select TOKYO_MEAN

$PSRF106,178*32

Name	Example	Unit	Description
Message ID	$PSRF106		PSRF106 protocol header
Datum	178		21=WGS84 178=TOKYO_MEAN 179=TOKYO_JAPAN 180=TOKYO_KOREA 181=TOKYO_OKINAWA Debug
Checksum	*32		
<CR><LF>			End of message termination

RoHS / Lead Free Compliance

Globalsat WorldCom Corporation
16F., No.186, Jian 1st Rd. Zhonghe Dist., New Taipei City 23553 Taiwan
Tel: 886-2-82263799 Fax: 886-2-82263899 Web: www.globalsat.com.tw

RoHS / Lead Free Compliance

Dear Sales:

This letter is intended to answer questions you may come across regarding the compliance of Globalsat WorldCom Corporation products with the following European Directive 2011/65/EU (RoHS) :

- Restriction of the Use of Certain Hazardous Substances in Electrical and Electronic Equipment (RoHS) .

This Directive aim is to avoid or limit the use of hazardous material compliant and meet the Standard by July.16,2011 of less than "0.1% by weight per homogeneous material for lead,hexavalent chromium,mercury, PBB and PBDE and 0.01% by weight and per homogeneous material for cadmium".

Globalsat has incorporated the requirement of 2011/65/EU into the product / technology development roadmaps and is committed to make lead free / RoHS fully compliant product available for shipment by July.16,2011.

Kevin Tan
Quality Assurance Manager

Prince Cheng
Chief Executive Officer (CEO)

QR8204-02A

Reversion history

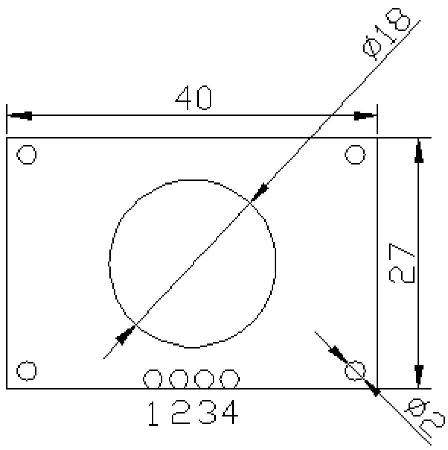
Reversion	Date	Name	Status / Comments
V1.1	20120320	Luwalk	Initial Version
V1.2	20130521	Mason	Modify VCC range
V1.3	20130724	Mason	1.Modify Pin define 2.Modify DC electrical characteristics
V1.4	20131008	Mason	Modify pin 1PPS to Directive

This module provides ultrasonic 30cm-3m of non-contact distance measurement function, range sensor in the sensor output corresponding to the objects in the high pulse width signal.

Features

- Detecting range: 30cm-3m
- Best in 30 degree angle
- Electronic brick compatible interface
- 5VDC power supply
- Breadboard friendly
- Dual transducer

Pin Definition

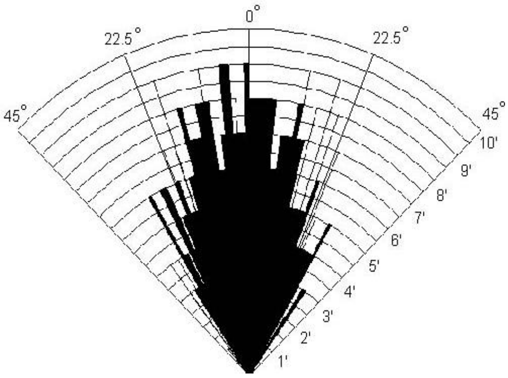


Pin	Value
1	GND
2	ECHO
3	TRIG
4	VCC

Specifications

Parameter	Value
Supply voltage	5V
Global Current Consumption	15 mA
Ultrasonic Frequency	40k Hz
Maximal Range	350 cm
Minimal Range	30 cm
Resolution	1 cm
Trigger Pulse W idth	10 µs

Practical test of performance, best in 30 degree

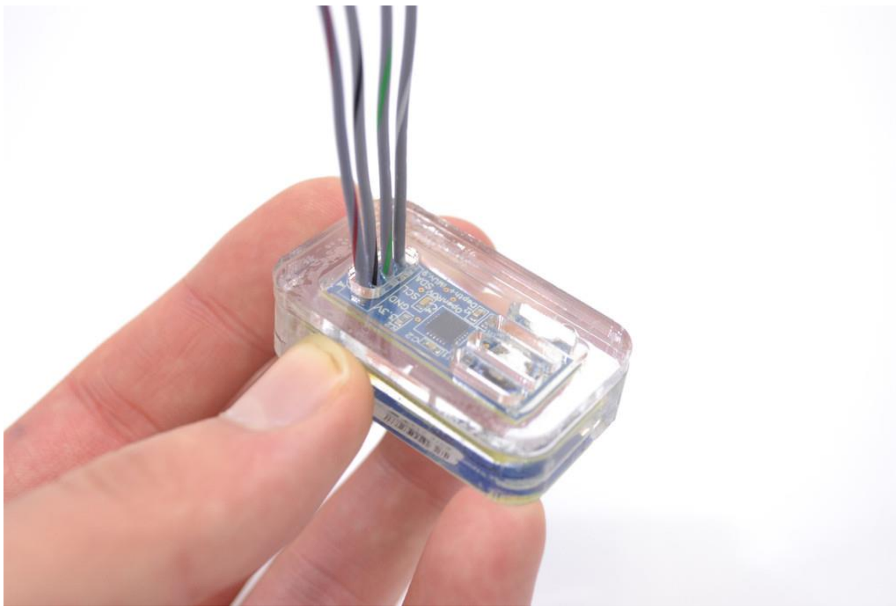


Sequence Chart

After power on, the module is waiting for the trigger signal. Triggered automatically issued within 8 40kHz cycle level, and to detect the long echo time, and through the corresponding timer output level TTL level PWM pulse width. According to the object at different distances, a corresponding proportion of the output pulse width. MCU can be used to determine the timing of pulses calculated distance. Formula: uS/5.8 = mm, or uS/148 = inches. If the object is not detected, the module output pin will output a constant pulse width of about 35MS.

Note: The module detects a minimum distance of 30cm inside the object, the signal will be inaccurate.

OpenROV IMU/Compass/Depth Module



Description

OpenROV IMU/Depth Sensor Add-On

Gives your OpenROV cockpit added telemetry including **depth**, **orientation**, and **compass heading**.

The OpenROV team and community members have been working very hard on a way to determine the heading and depth (as well as some other things) of our OpenROVs while underwater. Having this data will help with navigation, gathering relevant scientific data, and allows for closed-loop control systems that will make the OpenROV easier to fly. We've evaluated all sorts of components on the market and have found a low cost but effective way to integrate these sensors in an external module that can be added easily to new and existing OpenROVs. As many of our designs start out, this is a developer product--we've got everything working well enough to tinker with, but we're counting on our community to really explore the potential of what a sensor like this can do.

Overview:

- Makes depth, heading, water temperature and accelerometer/gyrometer telemetry available to the ROV via I2C which can displayed in OpenROV Cockpit
- Can be built with the same tools and materials as the OpenROV kit
- Ready to use on OpenROV versions 2.5, 2.6, and 2.7.

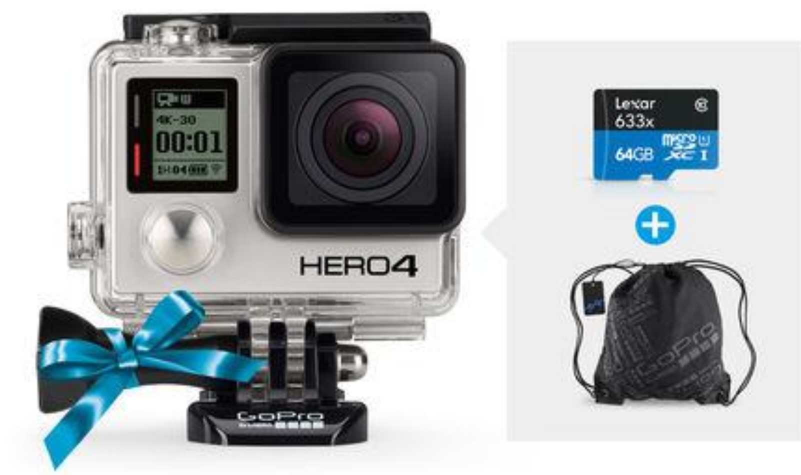
This upgrade kit comes with:

- Acrylic housing
- PCB with 9-axis inertial measurement unit (IMU) and pressure sensor
- Adhesive lined heatshrink tubing
- Hookup wire

Sensor Specifications:

- MPU-9150 9-axis IMU
- 3-axis magnetometer for detection of heading (regardless of orientation)
- 3-axis accelerometer to detect roll and pitch
- 3 axis gyroscope to detect rotational rate
- MS5803-14BA Pressure sensor
- Senses down up to 130m depth
- Precision to about 2cm of depth
- Integrated temperature sensor precise to about 0.1C

GoPro HERO4 Black



Weight
Camera: 3.1oz (89g)
Camera with housing: 5.4oz (152g)

Video Mode

Video Resolutions

Video Resolution	Frames Per Second (fps) NTSC/PAL
4K	30, 25, 24
4K SuperView	24
2.7K	60[1], 50, 48, 30, 25, 24

2.7K SuperView	30, 25
2.7K 4:3	30, 25
1440p	80, 60, 50, 48, 30, 25, 24
1080p	120, 90, 60, 50, 48, 30, 25, 24
1080p SuperView	80, 60, 50, 48, 30, 25, 24
960p	120, 60, 50
720p	240[1], 120, 60, 50, 30 , 25
720p SuperView	120, 60, 50
WVGA	240

Video Format

H.264 codec, .mp4 file format

Time Lapse Video[1]

Automatically creates video from frames captured at set intervals. Available only in 4K and 2.7K 4:3 resolutions.

Time Lapse Video Intervals

0.5, 1, 2, 5, 10, 30 and 60 seconds

Video + Photo

Record video and capture Time Lapse photos at the same time. Available intervals are 5, 10, 30 and 60 seconds.

Video Resolution	Video Frames per Second (fps)
1440p	25, 24
1080p	30, 25, 24
720p	60, 50, 30, 25

Looping

Record a continuous video loop that overwrites itself until you press the shutter button to stop it and save.

Advanced Video Capture Settings

SuperView™

SuperView video mode captures the world's most immersive wide-angle perspective. It allows you to capture more of yourself and your surroundings in the shot, and it provides full widescreen playback.

SuperView Settings

SuperView Mode
4K SuperView
2.7K SuperView
1080p SuperView
720p SuperView

Auto Low Light

Auto Low Light mode automatically adjusts frame rates according to lighting conditions for enhanced low-light performance. Frame rates are adjusted in medium- and low-light conditions. Playback occurs at the selected resolution and frame rate.

Photo Mode

Photo Resolutions

Resolution	Field of View (FOV)
12MP (Default)	Wide
7MP	Wide, Medium
5MP	Medium

Continuous Photo (up to 30 seconds)

Hold down the shutter button to continuously capture a series of photos until shutter button is released.

Continuous Interval

3 photos/1 second (Default)
5 photos/1 second
10 photos/1 second

Night Photo

Capture a single photo with a customizable exposure time of up to 30 seconds.

Shutter

Applies only to Night Photo and Night Lapse and determines the amount of time that the shutter is open.

Shutter Setting	Description
Auto (Default)	Up to 2 seconds
Fixed	2, 5, 10, 15, 20 or 30 seconds

Multi-Shot Mode

Photo Resolutions

Resolution	Field of View (FOV)

12MP (Default)	Wide
7MP	Wide, Medium
5MP	Medium

Burst

Capture up to 30 photos per second.

Burst Rate

30 photos/1 second (Default), 30 photos/2 seconds, 30 photos/3 seconds, 30 photos/6 seconds[1]
10 photos/1 second, 10 photos/2 seconds, 10 photos/3 seconds
5 photos/1 second
3 photos/1 second

Time Lapse

Automatically capture a series of photos at timed intervals.

Time Lapse Photo Intervals

0.5, 1, 2, 5, 10, 30 and 60 seconds

Night Lapse

Capture a series of photos at specific intervals and exposure times.

Night Lapse Photo Intervals

15, 20, 30 and 60 seconds
2, 5, 30 and 60 minutes

Shutter

Applies only to Night Photo and Night Lapse and determines the amount of time that the shutter is open.

Shutter Setting	Description
Auto (Default)	Up to 2 seconds
Fixed	2, 5, 10, 15, 20 or 30 seconds

Advanced Video and Photo Capture Settings

Protune™

Protune unlocks the camera's full potential, delivering minimally compressed, cinema-caliber video optimized for professional productions, and advanced manual controls for photos and video. Flex your creativity with customizable settings for Color, ISO Limit, White Balance, Sharpness and Exposure—and enjoy the greatest degree of manual control available in a GoPro camera.

- Captures images with less compression, giving content creators higher quality for professional productions.
- Protune enables manual control of **White Balance**, **Color**, **ISO Limit**, **Sharpness, Shutter** and **Exposure Value Compensation** for advanced control and customization of your videos and photos.
- Protune is available for all video and photo resolutions.

Protune Settings

White Balance

Adjusts the overall color tone of videos and photos.

White Balance Setting	Recommended Lighting Conditions
Auto (Default)	Automatically adjusts the color tone based on the environme
3000K	Warm light (incandescent or sunrise/sunset lighting)
5500K	Slightly cool light (cool fluorescent, average daylight)
6500K	Cool light (overcast conditions)
Native	Industry standardized optimized color

Color

Allows you to adjust the color profile of your video footage or photos.

Color Setting	Resulting Color Profile
GoPro Color (Default)	GoPro color-corrected profile (same great color as when Protune is turned off)
Flat	Flat, neutral color profile that can be color-corrected to better match footage ca

---

ISO Limit

Adjusts the camera's sensitivity in low-light environments, and creates a balance between brightness and resulting image noise. Image noise refers to the degree of graininess in the image.

Protune for Video ISO Limit

Setting	Resulting Video Quality
6400	Brighter video in low light, increased image noise
1600 (Default[1])	Moderately bright video in low light, moderate image noise
400	Darker video in low light, reduced image noise

Protune for Photo ISO Limit

Setting	Resulting Photo Quality
800 (Default)	Darker photo in very low light, increased image noise
400	Darker photo in low light, moderate image noise
200	Bright photo in indoor lighting, marginal image noise
100	Bright photo in outdoor daylight, minimal image noise

Sharpness

Controls the sharpness of your video footage or photos.

Sharpness Setting	Resulting Quality
High (Default)	Ultra sharp video or photo
Medium	Moderately sharp video or photo
Low	Softer video or photo that allows for more flexibility in post-produ

Spot Meter
Spot Meter is ideal for filming within a dark space with the camera pointed towards a brighter setting (such as filming the outdoors from within a car).
QuikCapture
With the press of a button, the camera automatically turns on and begins recording video or capturing Time Lapse photos.

Image Quality + Optics

- Ultra sharp image quality with all-glass lens
- Ultra wide-angle field of view with reduced distortion

Battery + Charging

- Rechargeable lithium-ion battery
- Rated at 1160mAH, 3.8V, 4.4Wh

Battery Life
The chart below indicates the approximate continuous recording time (hr:min) you can expect when shooting in various video modes using a fully charged battery.[5]

	With Wi-Fi Off	With Wi-Fi On + Using Wi-Fi Remote	With Wi-Fi On + Usin
Video Mode	**Estimated Time**	**Estimated Time**	**Estimated Time**
4K 30fps	1:05	0:55	0:50
2.7K 48fps	1:05	1:00	0:55
2.7K 30fps (4:3)	1:10	1:05	0:55
1080p 120fps	1:10	1:05	1:00
1080p 60fps	1:20	1:15	1:10
1080p 30fps SuperView	1:30	1:20	1:15
720p 120fps	1:50	1:40	1:30

Audio

- Format: 48kHz sampling rate, AAC compression
- Advanced AGC (automatic gain control) with multi-band compressor
- Internal Microphone:
  - Mono
  - Approximately 2x greater dynamic range (compared to the HERO3+ Black Edition)

- External Microphone:
  - Stereo supported with 3.5mm microphone adapter (sold separately)
  - High quality ADC (analog to digital converter) to support studio quality external microphones (compared to HERO4 Silver). See list of supported microphones.

Ports

Mini USB

- Charging
- Connecting to a computer for playback/file transfer/charging
- Supports 3.5mm stereo microphone via optional adapter (sold separately)
- Supports playback to composite TV via optional cable (sold separately)

Micro HDMI

- Supports playback to HDTV via optional cable (sold separately)
- HDMI playback is certified up to 1080p

microSD

- Memory card

Storage

- microSD memory card with a Class 10 or UHS-1 rating required. See list of recommended microSD cards
- Up to 64GB capacity supported
- Record times vary with resolutions and frame rates

Photo + Video Playback

HDTV

Micro HDMI to HDMI cabel required (sold seperately)
Note: HDMI playback depends on the resolution of the device and is certified up to 1080p.

TV

Mini USB to composite cable required (sold separately)
Direct playback may be available using the microSD card and a USB card reader (sold separately). See list of 4k TVs that support SD card playback.

LCD Touch BacPac™ (sold separately)

Attach to your camera for preview and play back of videos and photos.

GoPro App
Use your phone or tablet to preview and play back videos and photos.
Computer
Connect via mini USB to USB cable (included), or copy files from the microSD card to your computer.
Minimum system requirements for best playback on Mac® and Windows® computers

- Mac OS® X 10.8 and later / Microsoft Windows 7, 8.x
- Intel® Core 2 Duo™ or Intel® Dual Core™
- 4GB RAM
- Mac: Graphics card shipped with Intel® Dual Core™ Macs or better / Windows: Graphics card that supports OpenGL1.2 or later
- 5400 RPM internal hard drive (7200 RPM drive or SSD recommended)

- See more at: http://shop.gopro.com/hero4/hero4-black/CHDHX-401.html#sthash.rRfl8aIi.dpuf

ZIPPY Flightmax 8000mAh 3S1P 30C Lipo Pack

Zippy Flightmax batteries deliver full capacity & discharge as well as being the best value batteries in the market! No matter the application you have in mind, zippy lipoly batteries are an ideal choice.

Spec.
Capacity: **8000mAh**
Voltage: **3S1P / 3 Cell / 11.1v**
Discharge: **30C Constant / 40C Burst**
Weight: **644g (including wire, plug & case)**
Dimensions: **169x69x27mm**
Balance Plug: **JST-XH**
Discharge plug: **5.5mm Bullet-connector (without housing)**

Poweradd Pilot X5

Salient Features of Poweradd Pilot X5 16000mAh Battery Charger

1, Exquisite outward appearance, great workmanship with first-rate quality and credible performance.

2, Pilot X5 with advanced intelligent IC chip has been found particularly suitable for various devices' charging. The wide compatibility makes the charger more practical.

3, Pilot X5 provides 3.1 A (1 A and 2.1 A combined) output power for charging, which charges smart phones surprisingly fast.

4, With an enormous capacity of 16000mAh and double USB output ports, the charger affords two devices simlutaneous charging.

5, Pilot X5 has a major technological breakthrough in upgrading the energy output of mobile power. The newly achieved output has increased by 10%, which reaches 16000mAh. That is to say, Pilot X5 gives 1000mAh more than other similar products. This extra 1000mAh can almost charge a regular mobile phone for one time.

6, The charger is made of high quality PC+ABS alloy plastic which is noteworhty for its hardness and durability. Thousands of repeated tests prove the strict standards of such reliable plastic. Aside from the stiffness, the material is also non-toxic, lead-free, fireproof and anticorrosive.

7, Besides, there's a LED flashlight for illumination in case of need.

Portable Size

Pilot X5 has a size of 165.1 x 60.96 x 22.09 mm and a weight of 349 g. The lightweight rectangled charger is easy to carry and hold. It's conceivable that the portable charger makes your life more convenient, especially when you go traveling.

Well Reserved Power

Pilot X5 can support long hours' charging for a wide variety of devices on account of its large amount of power reservation by the 16000mAh capacity. So, don't worry about the power consumption, just enjoy using your devices. Pilot X5 will charge them up pretty shortly whenever they are eager for power.

Super-speed Charging

With the support of huge capacity 16000mAh and 5V/1A/2.1A output, the charging speed has been remarkably accelerated. You must be favorably impressed with the high efficiency of charging.

Product Specification:

--Battery Type: Li-ion battery

--Capacity: 16000mAh

--Life Cycle > 500 Times

--Input: DC 5V / 1000mAh (Max)

--Output: DC 5V / 1000mAh and DC 5V / 2100mAh (Max)

--Size: 165.1 x 60.96 x 22.09 mm

--Weight: 349 g

Package Contents:

--1 x Poweradd Pilot X5 16000 mAh External Battery

--1 x Micro USB cable

--3 x Connector(Apple Adapters Not Included)

--1 x User guide

Compatible Phone Models:

Apple: iPhone 5s,5c,5,4s,4,3gs,iPad,iPad 2,iPad Air,iPad 4,iPad mini,iPod,iPod Touch Nano(Apple adapters not inckuded)

Samsung: Galaxy s4、s4 mini,Galaxy s3,Galaxy note 3,Galaxy note 2,Galaxy s i9000,verizon Galaxy s3 i535,at&t Galaxy s3 i747,sprint Galaxy s3 l710,t-mobile Galaxy s3 t999,t-mobile Galaxy s2 t989,at&t Galaxy s2 i777,at&t Galaxy s2 skyrocket sgh-i727,Galaxy nexus,fouce s,infuse

HTC: HTC One, Nexus 5

Google: Nexus 4, Nexus 7, Nexus 10

LG:: nexus 4,optinus 4x,2x,v,s,t,3d,7

Nokia: 710,800,n8,n9,lumia 1020,920,820,900

Motorola: bionic,atrix 2,triumph,x phone,droid razr

Others: Amazon Kindle,Kindle fore,nexus 7,Blacknerry z10,torch curve 9900,9360,9320,Sony Ericsson xperia arc s,sony xperia z,x10,psp,nook color,mp3,mp4,mp5,gopro

Warranty Policy:

12 months' worry-free product guarantee & 90 days' refund unconditionally for every purchase from Poweradd.
Professional technical support, efficient and timely satisfactory customer service all the year around.

LUXdrive™ by LEDdynamics

3021/3023 BuckPuck™

Wide Range LED Power Module

Product Overview

The 3021 and 3023 BuckPuck LED Power Modules are a line of true current regulated drivers for powering LEDs. The BuckPuck line of LED drivers is the ideal choice for powering all types of high-brightness and high-power LED Packages and LED arrays.

The line of BuckPuck LED drivers exhibit high efficiency and require no external current limiting resistors or additional heat sinking for operation. A fast response current-sensing circuit makes the 3021 and 3023 ideal for applications where flashing or strobe operation of the LED(s) is required.

A wide range of options are available including external dc analog voltage intensity control, TTL/CMOS logic level on/off control ("E" Version), and set-and-forget internal current limiting ("I" Version). The standard units are fully potted in an extremely small form factor* and are provided with a simple 7 pin SIP connection for through-hole PCB mounting (3021) or 6" 24AWG Colored Leads (3023).

The 3021 and 3023's built-in regulated 5V reference (E and I versions) can provide output to power logic circuitry or microprocessor, eliminating the need for an additional power supply on the circuit board.

Features

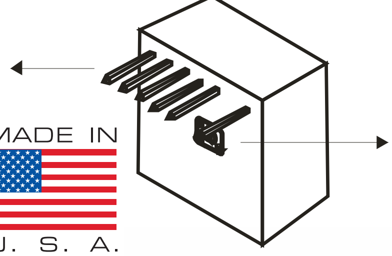
- DC or AC input voltage up to 32V ($24VAC_{RMS}$)
- 350mA, 500mA, 700mA, or 1.0A constant current output*
- Extremely small form factor* (0.83"x0.83"x0.43")
- The 3021 has a simple 7-pin SIP connection for through-hole PCB mounting or use with an optional wiring harness (3021Hx)
- The 3023 has permanently attached wires
- External analog/digital intensity control (TTL compatible)
- Optional external potentiometer intensity control (0-100%)
- Optional on-board trim adjustment (40-110%)
- Output short circuit protection up to 15 seconds
- Output open circuit protection
- Pulse and strobe capable (control input)
- Built-in 5V reference/output to power logic circuitry or µProcessor

Typical Applications

- Solar & Landscape Lighting
- Architectural Lighting
- Track Lighting
- Automotive & Marine Lighting
- Portable Lighting & Flashlights
- Point of Purchase Lighting
- Desk & Reading Lamps
- Signal & marker Lighting
- Flashing & Strobe Lighting
- Cabinet & Display Case Lighting
- Sign & Channel Letters
- Much More...

7-Pin SIP through-hole PCB mounting

MADE IN U. S. A.



Optional on-board trim adjustment (40-110%)

3021 BuckPuck with optional wiring harness (3021HEP)

RoHS Compliant 2002/95/EC

* - Custom units can be designed for OEM applications. Contact LUXdrive for more information.

LEDdynamics, Inc.
802.728.4533 P
802.728.3800 F
sales@LUXdrive.com
www.LUXdrive.com

Made in the USA

LEDdynamics™ what a bright idea

Part Number Identification Table

Table 1
Product Selection

Part Number	DC Input	AC Input	On-Board Trim	Control/ Dimming	Connection Type
3021-D-N-xxxx	5-32V	no	no	no	7-Pin SIP (4 Pins)
3021-D-E-xxxx	7-32V	no	no	yes	7-Pin SIP (6 Pins)
3021-D-I-xxxx	7-32V	no	yes	yes	7-Pin SIP (6 Pins)
3023-D-N-xxxx	5-32V	no	no	no	4 Wires
3023-D-E-xxxx	7-32V	no	no	yes	6 Wires
3021-A-N-xxxx	no	$7\text{-}24V_{RMS}$	no	no	7-Pin SIP (4 Pins)
3021-A-E-xxxx	no	$7\text{-}24V_{RMS}$	no	yes	7-Pin SIP (6 Pins)
3021-A-I-xxxx	no	$7\text{-}24V_{RMS}$	yes	yes	7-Pin SIP (6 Pins)
3023-A-N-xxxx	no	$7\text{-}24V_{RMS}$	no	no	4 Wires

XXXX - Output current rating in milliamperes (mA): 350, 500, 700, 1000 or special order factory custom rating

Absolute Maximum Ratings

Input Voltage, DC Model . . . . $32V_{DC}$
Input Voltage, AC Model . . . . $24V_{RMS}$
Output Voltage . . . . $32V_{DC}$
Control Pin Voltage 10V
Reference regulator current ($5V_{DC}$) Output 20mA

Typical Characteristics

Output tolerance (within specified temp. range) . . . . ±10%
Efficiency 95%
Input Voltage Minimum . . . . $5V_{DC}$ (N), $7V_{DC}$ (E or I), $7V_{RMS}$
Input Margin (350mA unit[1], add to LED Vf MAX). . . . . $2.5V_{DC}$, $4V_{RMS}$

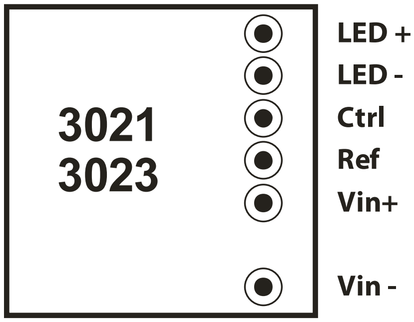


Figure 1.
Bottom view Pinout of the 3021/3023 BuckPuck

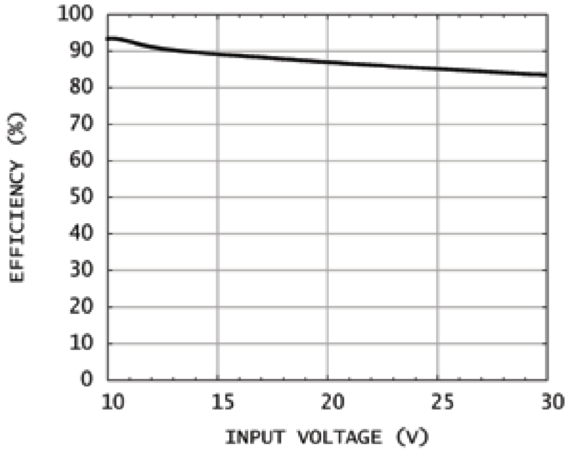


Figure 2.
Efficiency vs. Vin

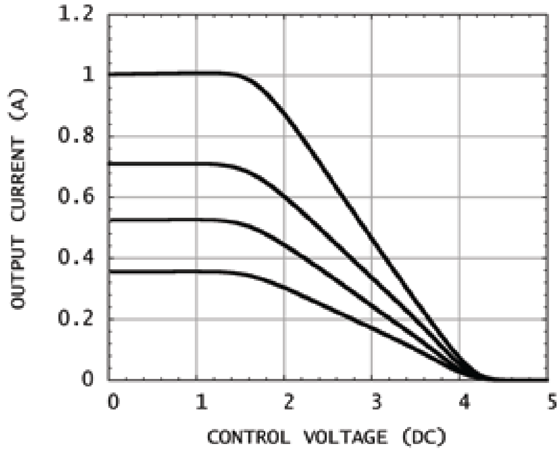


Figure 3.
Output current vs. control voltage

1 - Margin increases with higher current units.

Specifications

Output current, 3021-x-x-350, 3023-x-x-350	350mA[1]
Output current, 3021-x-x-500, 3023-x-x-500	500mA[1]
Output current, 3021-x-x-700, 3023-x-x-700	700mA[1]
Output current, 3021-x-x-1000, 3023-x-x-1000	1000mA[1]
Control Pin, adjustment threshold	1.65 V ±5%
Control Pin, shutoff threshold	4.2 V ±5%
Control Pin, propagation delay to output	<15 µs
Control Pin, input impedance	1.5k ohm
Reference voltage (V_{in}= 7V or greater)	5 V_{DC} ±5%[2]
Optional trim pot adjustment range	40%-110%
External pot adjustment range	0%,1-100%[2]
Maximum flash frequency	10 kHz
Minimum strobe pulse width	50 µs
Output rise time	<10 µs[3]
Output fall time	<350 µs[3]
Quiescent current (no load or control pin high)	<4.5 mA[3]
Operating temperature (Tcase)	-40-+80°C
Storage temperature	-40-+125°C

Application Information

Description

The 3021/3023 Wide Range LED Power Module is a high efficiency dc to dc converter which delivers a fixed output current by varying the output voltage as required to maintain the specified current . A fast response current-sensing circuit permits the unit to be used in applications where flashing or pulsing of the LEDs is required. Several options are available allowing for use with many types of LEDs and in a variety of operating modes

Fixed Current Drive

The fixed output versions of the 3021/3023 are designed to supply their rated current to one or more LED junctions. For example, a 350 mA rated unit will drive up to six white 350mA LEDs connected in series at $24V_{DC}$. Due to the nature of the buck regulator, the input voltage must always be higher than the total forward voltage drop of the LED junction(s) connected in series (2.5V for DC models, 4V for AC models). Thus, for a series string of six junctions having an average forward drop of 3.5Veach, the required minimum input voltage will be $23.5V_{DC}$. A standard $24V_{DC}$ power supply is a good choice for this application.

Figures 4 through 6 show 700mA and 1000mA rated units driving multiple LEDs. Note that parallel strings of LEDs can be driven directly with no additional circuitry required to ensure current sharing. The nature of the LEDs themselves will provide sufficient current sharing if the parallel strings comprise of 3 or more junctions each.

1 - Measured with single emitter; output current drops slightly with additional series junctions to limit maximum power dissipation.
2 - When Vin > 7VDC
3 - Actual value varies greatly based on input and/or output voltages. Value shown is for V_{in} = $24V_{DC}$ V_{OUT}=$20V_{DC}$ LED load. Actual values will be smaller in most applications.

LEDdynamics, Inc.
802.728.4533 P
802.728.3800 F
sales@LUXdrive.com
www.LUXdrive.com

Adjustable Current - On-Board Control - "I" Model

Where the ability to adjust the output current to an intermediate value is required, all output current ratings are available with an on-board potentiometer. This permits the output current to be varied from approximately 40% to 110% of the rated value. When measuring the output is required to determine a particular set point, the following method is recommended:

> Temporarily place a 0.1 ohm, 1% resistor in series with the LEDs.
> Read the voltage across the 0.1 ohm resistor.
> The voltage, in millivolts X 10, will equal the output current in mA.

Because there is a small, high-frequency component in the 3021/3023 output, many multi-meters may give an incorrect reading when used in the current mode. It has been found that the method described above yields a far more accurate measurement.

The potentiometers used for the on-board adjustable units are rated for a limited number of rotations (typically 100) and are intended for "set it and forget it" applications. Where frequent adjustments of output current are needed, the use of units with external adjustment capabilities is recommended.

Adjustable Current - External Control - "E" Model

Figures 10 and 11 show external adjustment configurations. Both use a 5Kohm, linear taper potentiometer. In Figure 10, the potentiometer is connected between the internal $5V_{DC}$ reference (Ref) output and the control (Ctrl) input. When using this configuration, it is important that Vin be $7V_{DC}$ or higher. Figure 11 shows the control potentiometer being powered by an external $5V_{DC}$ source. When using an external power source for the potentiometer, the source ground must be common to the LED- output pin. In either configuration, connect the potentiometer such that clockwise rotation increases the resistance. Note that, because the current through the potentiometer is less than 5mA, a low power potentiometer may be used.

External On/Off Control

Where a manual on/off control is desired, the potentiometer in Figures 10 and 11 may be replaced by a pushbutton or toggle switch. The output current will be zero when the switch is closed. Figures 12 and 13 show external dimming control combined with on/off control. The circuit in Figure 13 uses a 2N3906 or equivalent PNP switching transistor.

External Pulse/Strobe Control

Figures 14 and 15 show two methods for low speed pulsing or high speed flashing operation. In Figure 14, a 5V TTL/CMOS logic signal is applied directly to the control (Ctrl) input of the 3021/3023. The output current will be zero when the control signal is high. Note that the input needs to source a minimum of $4.75V_{DC}$ into a 1.5Kohm input impedance. Also, as is the case with a dc control signal, the logic input ground should to be common to the LED- output terminal.

Figure 15 shows an inverted input configuration using a 2N3906 or other PNP switching transistor. In this case, a logic high will cause the output to be "on". In either configuration, the rise time of the output will be 10µsec or less. A pulse frequency up to 10kHz may be used.

LEDdynamics, Inc.
802.728.4533 P
802.728.3800 F
sales@LUXdrive.com
www.LUXdrive.com

Microprocessor Control

Figure 16 shows a typical interface for a Microchip PIC® or similar µcontroller. The reference output provides the operating voltage for the processor (5V at up to 20mA current).

Other Control Applications

In addition to the configurations described above, the 3021/3023 may also be driven by a D to A converter. As in the cases above, the analog control signal should have its ground common to LED-. Figure 2 shows the effective control range of the analog input signal.

Connections

In all cases, the LEDs being driven should be located as close to the 3021/3023 LED output as possible. When the use of long leads is required, use heavier gauge wire. For strobe or pulse applications, a wire length not exceeding 6" should be used to maintain accurate timing.

The power input wires/traces should also be kept short. Where DC input units are located more than 18" from the source, a 220µF, 50V capacitor should be placed across the input terminals as shown in Figure 18.

For applications where the use of header pins is inconvenient, a mating connector with 6" leads is available as an accessory, or the 3023 part number may be used, which is supplied with 6" colored leads.

3021HN - Harness for "N" type (4-wire)
3021HE - Harness for "E" & "I" type (6-wire)
3021HEP - Harness for "E" & "I" w/pot (6-wire w/pot)

Application Figures

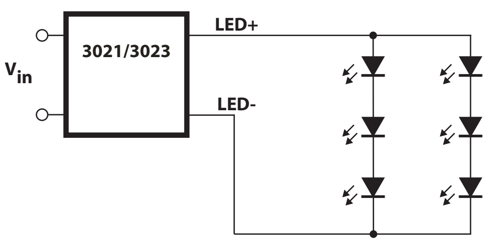


Figure 4.
700 mA unit driving 6 High Power LEDs
(VIN ≥ 12VDC)

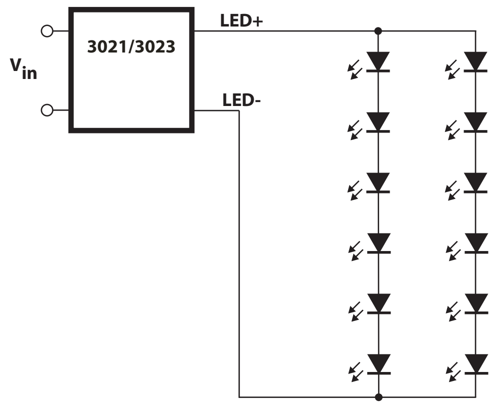


Figure 5.
700mA unit driving 12 High Power LEDs
(VIN ≥ 24VDC)

LEDdynamics, Inc.
802.728.4533 P
802.728.3800 F
sales@LUXdrive.com
www.LUXdrive.com

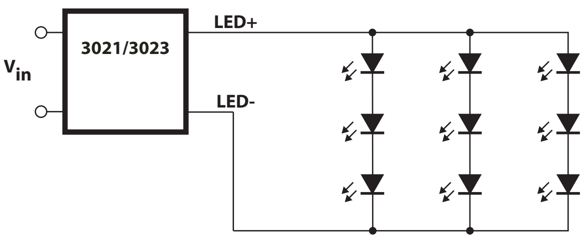


Figure 6.
1000mA unit driving nine High Power LEDs at 1W each (VIN ≥ 12VDC)

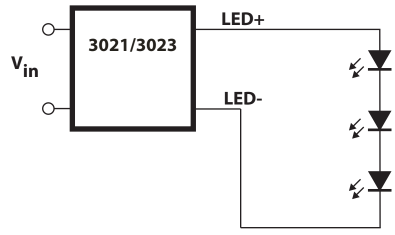


Figure 7.
1000mA unit driving three High Power LEDs at 3W each (VIN ≥ 12VDC)

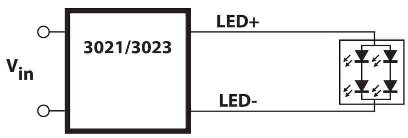


Figure 8.
700mA unit driving one Cree MC-E emitter (VIN ≥ 8VDC)

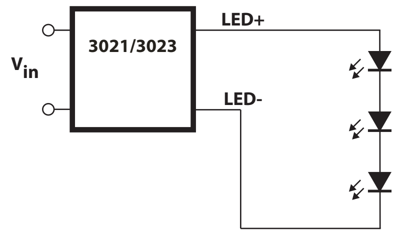


Figure 9.
700mA unit driving three High Power LEDs at 2W each (VIN ≥ 12VDC)

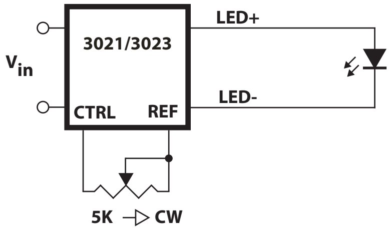


Figure 10.
External potentiometer using internal reference

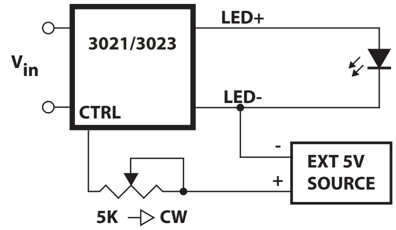


Figure 11.
External potentiometer using external voltage source

LEDdynamics, Inc.
802.728.4533 P
802.728.3800 F
sales@LUXdrive.com
www.LUXdrive.com

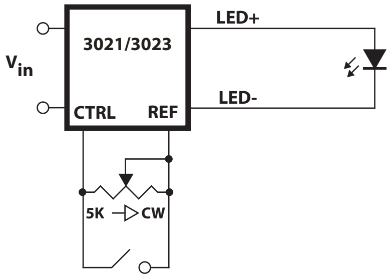


Figure 12.
External dimming plus ON/OFF control with switch closure

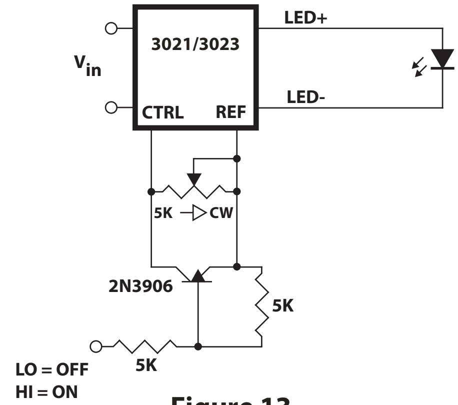


Figure 13.
External dimming plus ON/OFF control with logic level input

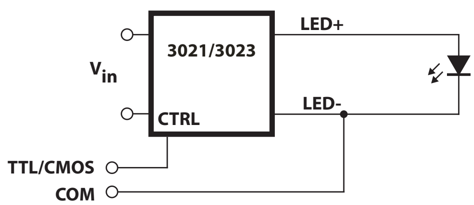


Figure 14.
Pulse/Strobe input 5V=OFF

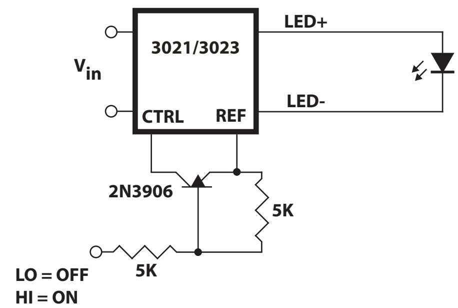


Figure 15.
Pulse/Strobe input 5V=ON

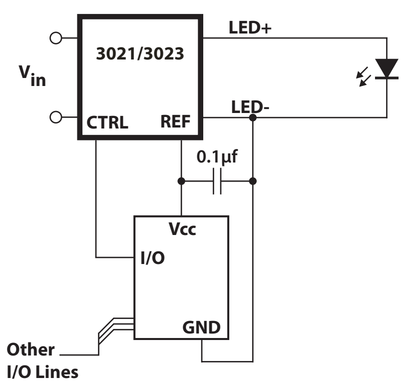


Figure 16.
Interface to PIC or other microcontroller

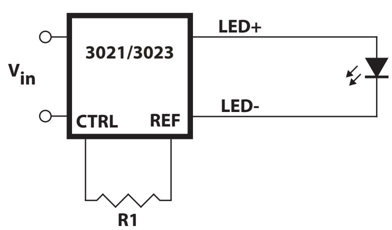


Figure 17.
Using resistor for fixed current reduction
Output is approximately: %IOUT = R1/50

LEDdynamics, Inc.
802.728.4533 P
802.728.3800 F
sales@LUXdrive.com
www.LUXdrive.com

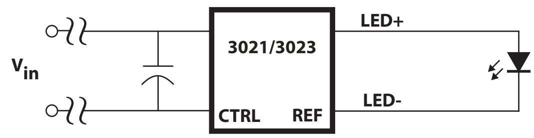
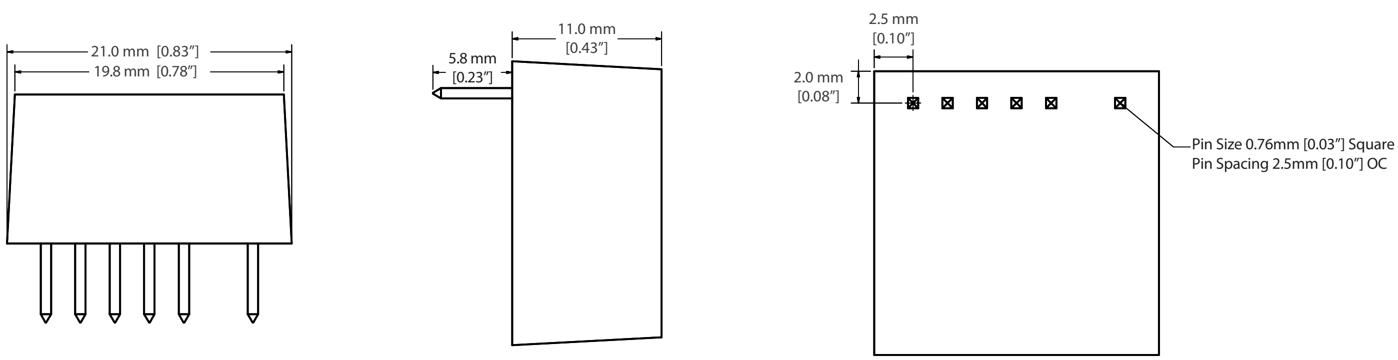
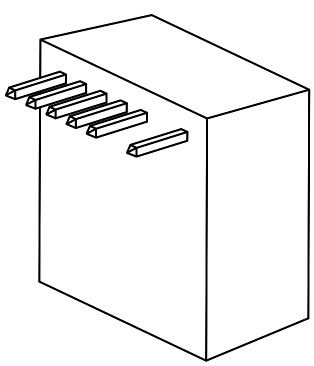


Figure 18.
Place a capacitor across the input terminals when the distance to the DC power source is greater than 18 inches

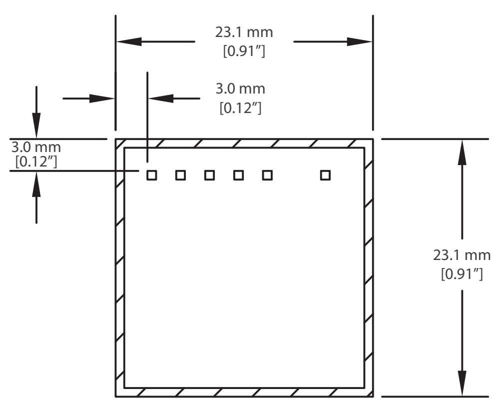
Physical Dimensions



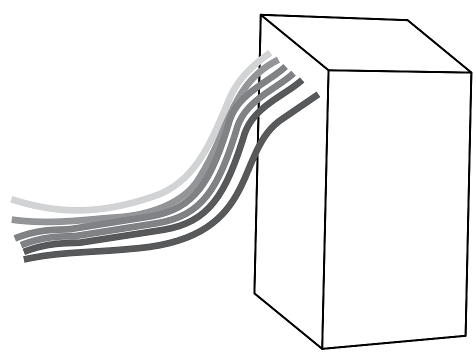
3021



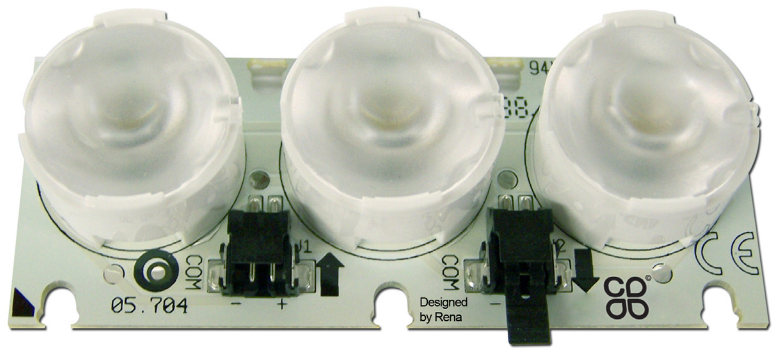
Recommended clearance envelope



3023
6" - 24AWG Colored Leads



LEDdynamics, Inc.
802.728.4533 P
802.728.3800 F
sales@LUXdrive.com
www.LUXdrive.com



DATA SHEET
05704 Series

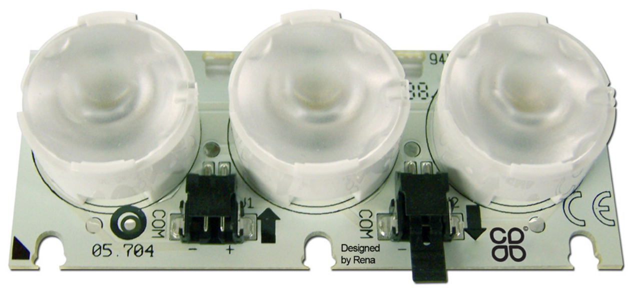
Part of the simpleLED® program

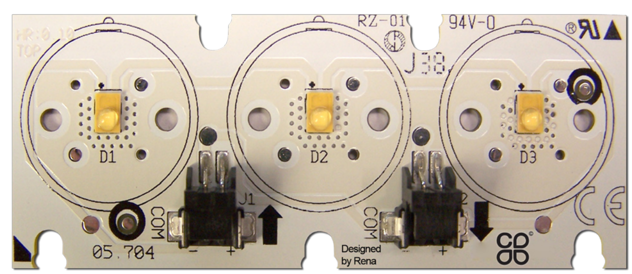
simpleLED 05704 SERIES

The light engine series consist of 3 high power LUXEON Rebel LEDs. It is engineered to provide customers with the flexibility to select the optimal light source for the applications. Customers can modify the simpleLED light source by selecting the LUXEON Rebel LED, optic and connector to best suit their needs.

FEATURES & BENEFITS

3 Year Manufacturer (Rena) Warranty
High-Reliability LED Sources
Rugged Construction
Wide Operational Temperature Range
Multiple Configurable Options
Flexible Optic Options
Wide Range Drive Current
Multiple White CCT's Available
Short Lead Time
CE certified, UL recognized





TYPICAL APPLICATIONS

Under Cabinet Lighting
Cove Lighting
Accent Lighting
Display Case Lighting
Display Lighting

THE PHILIPS LED LICENSING PROGRAM

Future Lighting Solutions offers a basic light engine from the simpleLED program, marked with the Clover trademark as a qualified component under the Philips LED Licensing Program to help you qualify your finished luminaire for a 0% royalty license. For more information about the licensing program requirements and the Clover, please visit

www.ip.philips.com/licensing/clover

Note: All specifications are subject to change without notice.

MECHANICAL CHARACTERISTICS

PARAMETER	CONDITIONS
PCB	FR-4
Finish	White
Size	30 x 75 mm
Source Type	LUXEON Rebel
Connector	Tyco CT (2-292173-2)
Thermal Resistance (p-n junction to bottom of PCB)	Rth= 21 K/W

ELECTRICAL CHARACTERISTICS

PARAMETER	MIN	NOM	MAX
Forward Voltage (V) @350mA & Tj=25 °C	7.6	9.0	12.0
Power Consumption @350mA (W)	2.7	3.2	4.2

ENVIRONMENTAL CHARACTERISTICS

PARAMETER	MIN	MAX
Storage Temperature (°C)	-40	+70
PCB temperature (°C)	-20	+80

Note: All specifications are subject to change without notice.

THERMAL STATEMENT & ASSEMBLY INSTRUCTIONS

The light engines must operate under proper environmental conditions and the operating ambient air temperature must not exceed a certain maximum which cause the LEDs to exceed the maximum junction temperature as stated in Philips Lumileds datasheet. A heat sink must always be used when operating the light engines. The size of the heat sink depends on the amount of power consumed by the LEDs. The objective is to maintain the junction temperature below the maximum rating in Philips Lumileds datasheet while also not exceeding the maximum PCB temperature.

The light engine must be mounted on a flat heat sink using M3 screws. All screw holes must be used to attach the light engine to the heat sink in order to provide proper heat transfer. Also a thermal conductive interface must be used between the heat sink and light engine. This thermal conductive interface could be a thermal conductive paste such as AmasanT12 from Armack Lottechnik or a thermal interface material such as T-PCM 585 from Laird.

The light engine must not be bent to avoid damaging the LED and/or dislodging the optics. All above specifications must be met in order to qualify for the 3 year warranty.

THERMAL MANAGEMENT

The graphs below show the required thermal resistance of the heat sink based on the maximum operating ambient temperature, the drive current and the maximum allowable PCB temperature. The maximum allowable Tj is a function of the target lifetime of the light engine and the LED current. This information can be found in the Philips Lumileds reliability datasheet RD07.

For example, if the maximum ambient temperature is 40°C and the drive current is 500 mA, the heat sink should have a Rth of 6 K/W to meet the max PCB temperature requirement. This is shown in figure 1. With the known Rth of the heat sink, the delta T from junction to ambient can be determined in figure 2. A Rth of 6 K/W has a delta T of 63 °C, which means that the LED has a Tj of 103 °C.

With the same graphs the max operating ambient temperature and the junction temperature can also be determined if the thermal resistance of the chosen heat sink is known.

Note; the graphs show that not all combinations of Tj and max ambient are possible.

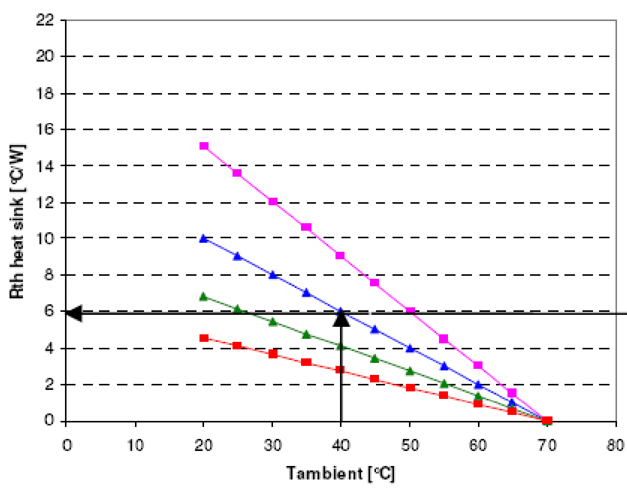


Figure 1

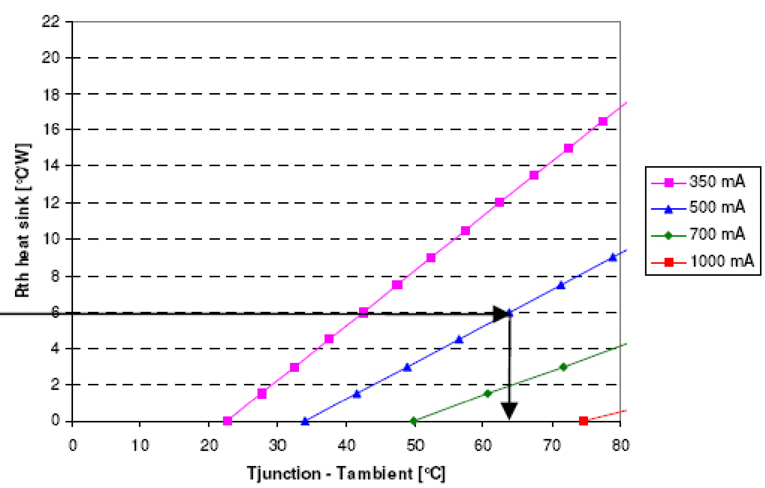
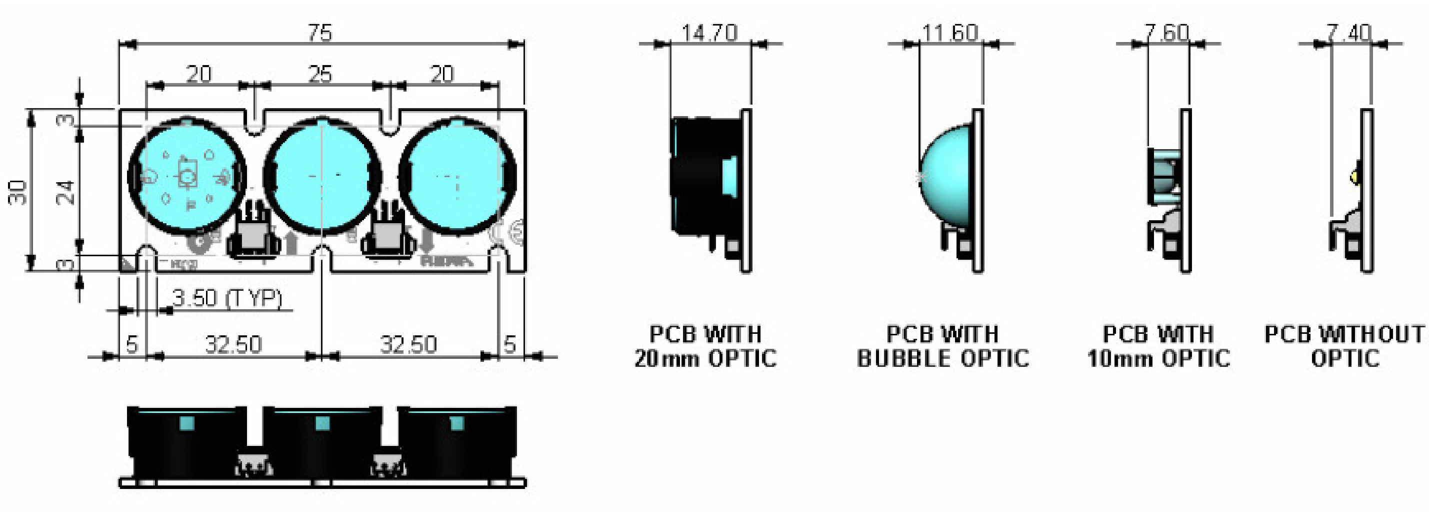


Figure 2

Note: All specifications are subject to change without notice.

MECHANICAL DRAWINGS

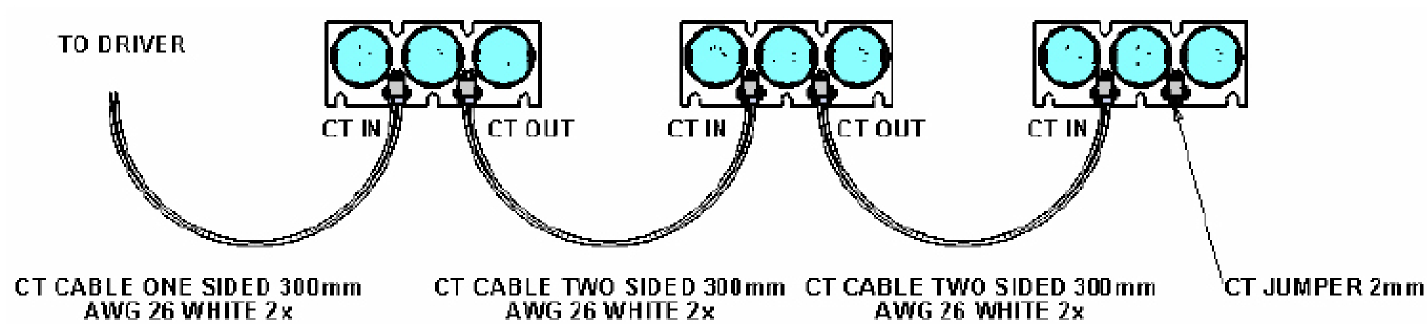
2D drawings with dimensions in mm



INTERCONNECTIVITY OPTIONS

Board-to-board wiring options and drawings

Boards connected in series:

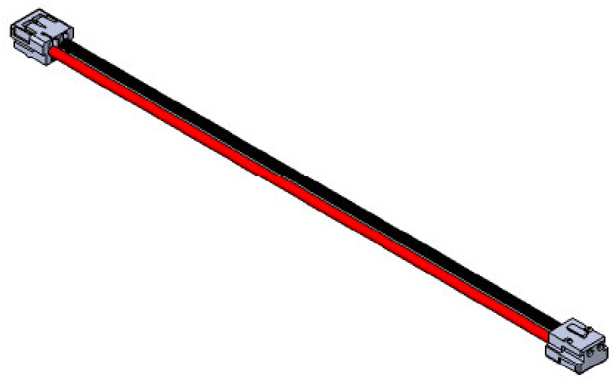


Note: All specifications are subject to change without notice.

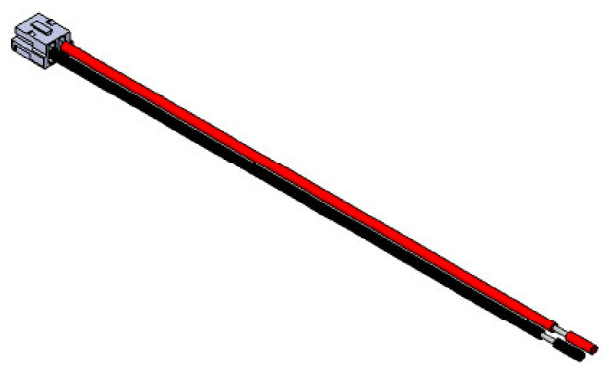
ACCESSORIES FOR INTERCONNECTIONS

Cable options for board to board connection and for driver connection (depending on selected driver)

Part number CT-CT cable	Cable length (mm)	Wire colors
1969343-6	300	white
1969343-5	150	white
1969343-4	50	white
1969343-3	300	red & black
1969343-2	150	red & black
1969343-1	50	red & black



Part number single CT cable	Cable length (mm)	Wire colors
1969336-6	300	white
1969336-5	150	white
1969336-4	50	white
1969336-3	300	red & black
1969336-2	150	red & black
1969336-1	50	red & black



* Please refer to www.FutureLightingSolutions.com for a detailed explanation on choosing the correct cable assembly.

Note: All specifications are subject to change without notice.

PART NUMBERING & ORDERING INFORMATION

1. PRODUCT SERIES (05704)

05704 = Linear Board with 3 LEDs in Series

05804 = Linear Board with 3 LEDs in Series with Clover tradmark

1. LED TYPE

R = LUXEON Rebel

2. COLOR TEMP (AAAA)

0000 = Royal-Blue
1111 = Cyan
2222 = Red
3333 = Red-Orange
4444 = Amber
5555 = Green
6666 = Blue
7777 = Neutral White
8888 = Warm White
9999 = Cool White

3. MINIMUM CRI* (BB)

XX = No Min CRI
55 = Min 55
60 = Min 60
65 = Min 65
70 = Min 70
75 = Min 75
80 = Min 80
85 = Min 85
90 = Min 90

4. MINIMUM FLUX* (LM) (CCC)

065 = Min 65
066 = Min 66
067 = Min 67
075 = Min 75
080 = Min 80
100 = Min 100
120 = Min 120
200 = Min 200
350 = Min 350mW
425 = Min 425mW
500 = Min 500mW

5. CONNECTOR (D)

C = Connector
N = No Connector

Part Number:

05704RAAAABBCCCDEFG

1 2 3 4 5 6 7 8

6. SUPPLIER COLLIMATOR (E)

X = No Optics
A = Carclo 10mm
B = Carclo 20mm
D = Carclo bubble

7. OPTIC HOLDER (F)

(Carclo 20mm)
X = No Holder
A = Carclo Single Black Holder 10235
B = Carclo Single White Holder 10236
C = Carclo Single Clear Holder 10237

8. COLLIMATOR (G)

X = No Lens
10 and 20 mm optics:
C = Narrow Beam
D = Narrow Beam Frosted
E = Medium Beam
F = Medium Beam Frosted
G = Wide Beam
H = Wide Beam Frosted
K = Elliptical Beam
L = Elliptical Beam 90°
Bubble optics:
R = Ultra Wide 120°
S = Ultra Wide 130°

* According to Lumileds datasheet
Special configurations available upon request
Contact your local sales representative

Note: All specifications are subject to change without notice.

COMPANY INFORMATION

About Future Lighting Solutions

Future Lighting Solutions (www.futurelightingsolutions.com) is a leading provider of LED lighting components and support services for solid-state lighting products and installations, including engineering expertise, concept development, full system solutions and online tools that accelerate quality application development. The company is a division of Future Electronics.

About simpleLED®

Future Lighting Solutions simpleLED program has over 500 combinations of LUXEON® LED & Optic configurations, enabling you to select the right Light Engine for your application, eliminate prototyping delays and accelerating time to market. Additional benefits include UL recognized quality and a 3 year warranty. Visit our website and start innovating.

CONTACT DETAILS

In North America:
1-888-LUXEON2
Americas@futurelightingsolutions.com

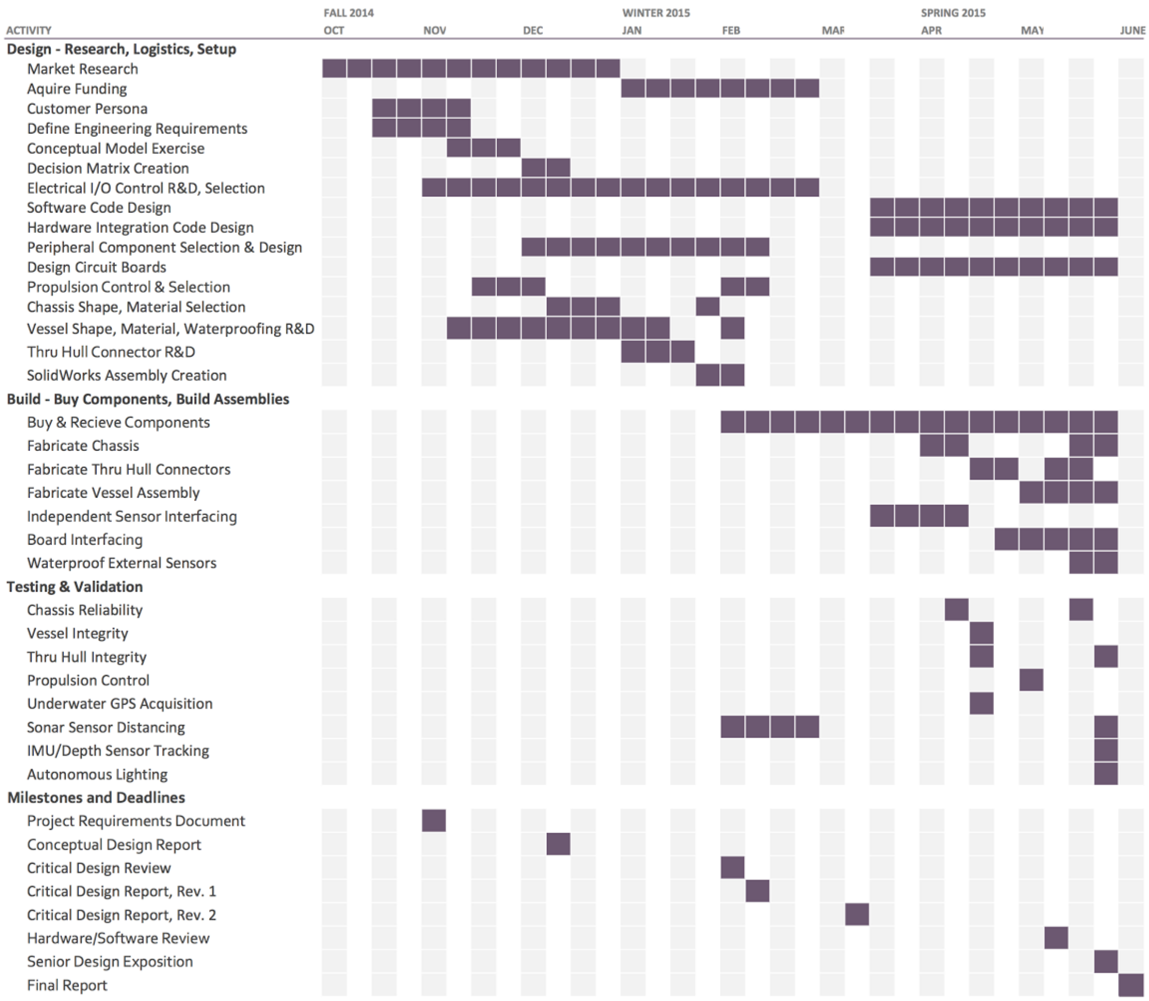
In Europe:
00-800-44FUTURE
Europe@futurelightingsolutions.com

In Asia:
+800-LUMILEDS
Asia@futurelightingsolutions.com

In Japan:
+81-0120-667-013
Japan@futurelightingsolutions.com

Note: All specifications are subject to change without notice. Warranty provided by the manufacturer, Rena Electronica BV.

Appendix O: Gantt Chart



Appendix P: Bill of Materials

Mechanical (left) and Electrical (right):

Subsystem	Vendor	Part Number	Item Description	Qty.	Unit Cost	Subtotal	Shipping	Tax (8%)	Total Cost
Thruster	Blue Robotics	M100-MOTOR-R1	Blue Robotics M100 Brushless Motor	3	$65.00	$195.00	$0.00	$15.60	$210.60
	Blue Robotics	M100-MOTOR-R1	Blue Robotics T100 Thruster	2	$109.00	$218.00	$12.00	$17.44	$247.44
	Hobby King	9192000124	Afro ESC 30Amp Multi-rotor Motor Speed Controller (SimonK Firmware)	5	$14.00	$70.00	$0.00	$5.60	$75.60
	Hobby King	9171000243	Afro ESC USB Programming Tool	1	$6.91	$6.91	$0.00	$0.55	$7.46
	Blue Robotics	T100-PROPELLER-S	T100 Propeller Set	3	$9.00	$27.00	$8.00	$2.16	$37.16
	Blue Robotics	TOOL-HEX-SET-R1	Hex Key Set	1	$6.00	$6.00	$0.00	$0.48	$6.48
	Blue Robotics	M100-P-PROPELLE	M100 Propeller - Clockwise	3	$12.00	$36.00	$0.00	$2.88	$38.88
	Ace Hardware	43130	Hillman M3 x 0.5 Screws and Locknuts	16	$0.59	$9.44	$0.00	$0.76	$10.20
Chassis	McMaster	8982K4	Multipurpose 6061 Aluminum, 90 Degree Angle, 1/8" Thick, 1" x 1" Legs, 1' Long	8	$3.22	$25.76	$100.00	$2.06	$127.82
	McMaster	8982K4	Multipurpose 6061 Aluminum, 90 Degree Angle, 1/8" Thick, 1" x 1" Legs, 2' Long	4	$6.26	$25.04	$0.00	$2.00	$27.04
	McMaster	8975K578	Multipurpose 6061 Aluminum, Rectangular Bar, 1/8" x 1", 2' Long	4	$2.68	$10.72	$0.00	$0.86	$11.58
	McMaster	8975K578	Multipurpose 6061 Aluminum, Rectangular Bar, 1/8" x 1", 1' Long	6	$1.44	$8.64	$0.00	$0.69	$9.33
	McMaster	91500A199	Type 316 Stainless Steel Flat Head Phillips Machine Screw, 8-32 Thread, 1" Length, Packs of 25	2	$5.59	$11.18	$0.00	$0.89	$12.07
	McMaster	90257A009	Type 316 Stainless Steel Hex Nut, 8-32 Thread Size, 11/32" Wide, 1/8" High, Packs of 100	1	$7.98	$7.98	$0.00	$0.64	$8.62
	McMaster	92510A303	Aluminum Unthreaded Spacer, 1/4" OD, 1/4" Length, #8 Screw Size	10	$0.23	$2.30	$0.00	$0.18	$2.48
	McMaster	1556A64	Bracket, Type 304 Stainless Steel, 7/8" Length of Sides 8	8	$1.06	$8.48	$0.00	$0.68	$9.16
	McMaster	90264A425	Grade 2 Zinc-Plated Steel Coupling Nut 10-24 Thread Size, 3/4" Length, 5/16" Width	2	$0.36	$0.72	$0.00	$0.06	$0.78
	McMaster	90575A358	Type 316 Stainless Steel Fully Threaded Stud, 10-24 Thread, 2" Long	1	$10.38	$10.38	$0.00	$0.83	$11.21
Vessel	McMaster	92186A754	Type 316 Stainless Steel Hex Head Cap Screw, 1/2"-13 Thread, 14" Long	8	$15.16	$121.28	$0.00	$9.70	$130.98
	McMaster	93490A033	Bronze Flat Washer, 1/2" Screw Size, 0.562" ID, 1.250" OD, packs of 10	1	$12.65	$12.65	$0.00	$1.01	$13.66
	McMaster	94804A340	Type 316 Stainless Steel Hex Nut, 1/2"-13 Thread Size, 3/4" Wide, 7/16" High, Packs of 10	1	$4.98	$4.98	$0.00	$0.40	$5.38
	McMaster	8560K591	Optically Clear Cast Acrylic Sheet, 5/16" Thick, 12" x 12"	2	$19.11	$38.22	$0.00	$3.06	$41.28
	McMaster	8560K239	Optically Clear Cast Acrylic Sheet, 1/8" Thick, 12" x 12"	1	$8.63	$8.63	$0.00	$0.69	$9.32
	McMaster	8486K735	Optically Clear Cast Acrylic Tube, 8" OD x 7-3/4" ID, 1' Length	1	$66.27	$66.27	$0.00	$5.30	$71.57
	McMaster	8516T22	Compressible Buna-N Flange Gasket for 6 Pipe Size, Class 150, Ring, 1/8" Thick	2	$3.64	$7.28	$0.00	$0.58	$7.86
	McMaster	11435A11	Side-Mount Drawer Slide, Full Extension, 10" Closed Length	1	$16.50	$16.50	$0.00	$1.32	$17.82
	McMaster	7454A22	4-Ounce Bottle of Gorilla Glue	1	$8.28	$8.28	$0.00	$0.66	$8.94
	McMaster	9273K113	General Purpose Nylon Hook and Loop, 1/2" Width X 5' Length, Adhesive Back, Lengths of 5 ft.	1	$6.07	$6.07	$0.00	$0.49	$6.56
Thru Hull Connectors	McMaster	50785K24	Medium-Pressure Brass Threaded Pipe Fitting, 1/2 Pipe Size, Hollow Hex-Head Plug	7	$2.81	$19.67	$0.00	$1.57	$21.24
	McMaster	98847A003	Type 18-8 Stainless Steel Fully Threaded Rod, 2-56 Thread, 3 Feet Long	3	$2.90	$8.70	$0.00	$0.70	$9.40
	Ace Hardware	12268	Dap® Auto/Marine Silicone Sealant	1	$6.99	$6.99	$0.00	$0.56	$7.55
	Ace Hardware	4028510	Union Flat Washer 1/2" Standard, 1 3/16" O.D., 3/4" I.D	7	$0.59	$4.13	$0.00	$0.33	$4.46
	Home Depot	11603-6	Plasti Dip 14.5 oz Black Rubber Coating	1	$6.98	$6.98	$0.00	$0.56	$7.54
					TOTAL	**$1,016.18**	**$120.00**	**$81.29**	**$1,217.47**

Subsystem	Vendor	Part Number	Description	Qty.	Unit Cost	Subtotal	Shipping	Tax (8%)	Total Cost
Sensors	SparkFun Electronics	EM-506	GPS Reciever	1	$39.95	$39.95	$0.00	$3.20	$43.15
	SparkFun Electronics	GPS-10710	GPS Shield	1	$14.95	$14.95	$0.00	$1.20	$16.15
	Core-Electronics	017-MB-SM19116	Sonar Module	10	$9.72	$97.20	$0.00	$7.78	$104.98
	OpenROV	---	IMU/Compass/Depth Module	1	$80.00	$80.00	$11.62	$6.40	$98.02
Recording System	GoPro	GoPro Hero Black	GoPro with 32GB Memory Card	1	$472.00	$472.00	$0.00	$37.76	$509.76
	Future Lighting Solutions	05704R777760200CXXX	Linear 3 Cool White LEDs (4100 K 200 lm)	1	$14.25	$14.25	$33.00	$0.00	$47.25
	LED Supply	03021-D-I-350	350mA Constant Current LED Driver with Trim Pot	1	$16.48	$16.48	$29.99	$0.00	$46.47
Power	Hobby King	Z80003S-30	11.1 Volt 8Ahr Battery	2	$45.39	$90.78	$0.00	$7.26	$98.04
	Amazon	RCP-8542125-S2WR	5V 16Ah Battery	2	$28.99	$57.98	$0.00	$4.64	$62.62
Controller	SparkFun Electronics	Arduino Mega 2560 V3	Microcontroller	1	$45.95	$45.95	$0.00	$3.68	$49.63
	Amazon	Arduino Uno	Microcontroller	1	$23.98	$23.98	$0.00	$1.92	$25.90
Cooling	McMaster	1939K17	DC Equipment Cooling Fan	1	$17.25	$17.25	$0.00	$1.38	$18.63
					TOTAL	**$970.77**	**$74.61**	**$75.20**	**$1,120.58**

Appendix Q: Testing and Validation Procedures

The testing and validation procedures are all provided below. Mechanical tests are presented first, followed by electrical and software tests.

Mechanical Tests	**Electrical and Software Tests**
Chassis Reliability	GPS Acquisition
Vessel Integrity	Sonar Distancing
Propulsion Control	IMU/Depth Tracking
Propulsion Strength	Autonomous Lighting
	Board Integration

Chassis Reliability Test

Determine: Whether the aluminum chassis meets the engineering specifications

Materials: Completed aluminum chassis

Location: Alex's House (Broad St. and Serrano Dr.)

Safety: Safety glasses

Procedure:

1. Align the bottom of the chassis to be parallel with the ground.
2. From the second floor, drop the chassis onto the ground.
3. Observe and record any damage, if any.
4. Repeat Steps One through Three with the chassis positioned so the sides and corners impact the ground.

Results: If chassis is undamaged, the test is a pass. If the chassis begins to yield, the test failed.

Vessel Integrity Test

Determine: Whether the waterproof vessel meets the engineering specifications and is truly waterproof.

Materials: one completed waterproof vessel subassembly, three 8.5-inch x 11-inch sized paper, tape, two gallon size ziplock bags, swimsuit

Location: ASI Recreation Pool (Cal Poly, SLO)

Safety: Safety glasses

Procedure:

1. Disassemble the endcap assembly so that the vessel is open.
2. Using the tape, attach a piece of paper to the attached endcap, shelf, and as close to the open side of the vessel as possible.
3. Reassemble the waterproof vessel, ensuring proper tightening on the screws.
4. Enter the water with the vessel. Observe and record any immediate leaks, if any.
5. Submerge the vessel entirely in the pool water. Keep it submerged for as long as possible. The buoyancy force will resist.
6. Retrieve the vessel and place above water. Check if the paper inside is wet.

Results: If the paper is dry the vessel passes. If the paper is wet, the vessel fails.

Propulsion Control Test

Determine: Whether the electronic speed controllers (ESCs) work with the Blue Robotics thrusters.

Materials: one Afro ESCs, one USB Debugger tool, one Blue Robotics T100 thruster, one Blue Robotics M100 brushless motor, one DC power supply, eight banana to banana cables, two water containers, three gallons of water

Location: Anu's apartment in a water safe area (Murray Ave.)

Safety: Safety glasses, proper grounding, anti-static wristband, insulated wire

Procedure:

1. Turn on the DC power supply and set the voltage to 12V with a current limit set to one A.
2. Using the documentation provided at http://docs.bluerobotics.com, wire the previously-programmed ESC to DC power supply and to the T100 thruster using the breadboard and jumper cables if needed.
3. With the USB Debugger tool, vary the programming so the thruster receives various voltages. Ensure the thruster motor is rotating at proportional speeds.
4. Repeat Step Three with a different thruster motor until all thruster thrusters have been tested.
5. Repeat Step Three with a different ESC until all controllers have been tested

Results: If all thrusters operate as expected, the test passes. If any complications occur, the test fails.

Propulsion Current Draw Test

Determine: The thrust of the Blue Robotics T100 thruster and M100 motor with various thrust programming

Materials: One completed wiring setup from the Propulsion Control Test, one multimeter, two water containers, three gallons of water

Location: Anu's apartment in a water safe area (Murray Ave.)

Safety: Safety glasses, proper grounding, anti-static wristband, insulated wire

Procedure:

1. Wire the propulsion control test wiring through the test rig.
2. Connect the multimeter in series between the output of the power supply and the input of the ESC and have it set to current measurement.
3. Program the ESC to produce various propulsion power starting from zero propulsion to the max.
4. Through each iteration of the different propulsion powers, record the current draw for this run.

Results: Recorded values.

GPS Acquisition Test

Determine: Whether GPS reliably and accurately connects with satellites.

Materials: One GPS module, one Arduino Uno, one swimsuit, one waterproofed container with a clear side to allow visibility to the interior.

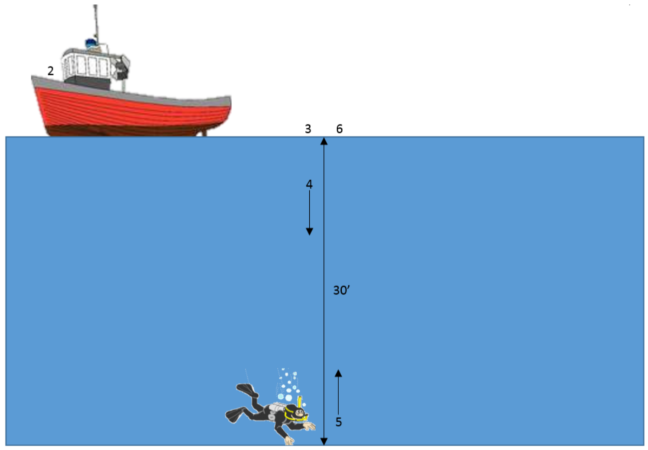
Location: ASI Recreation Center (Cal Poly, San Luis Obispo)

Safety: Insulated Electrical Components

Procedure:

1. Confirm no major obstructions in sky.
2. Turn GPS on and wait until connection is established while above water.
3. Confirm module is waterproofed in container.
4. Enter water and ensure stable connection at sea level (wait for up to two min.).
5. Submerge watertight container at least six inches until the connection fails. Take note of the depth.
6. Wait one minute and ascend to surface . Take note of when the connection reestablishes.
7. Wait on surface until GPS reconnects and locks onto a position. If the position is not available after 20 minutes, stop the test.

Results: If the GPS disconnects when submerged underwater and reconnects when brought to the surface, the test passed. Anything else and the test fails.



Sonar Distancing Test

Determine: Whether sensor reliably and accurately measures objects in water.

Materials: One 017-MB-SM19116 sensor module, one Arduino Mega 2560 r3, one USB cable, one laptop, one watersafe target.

Location: Anu's pool (Murray Ave.)

Safety: Wear bathing suit (swimmer), Electrostatic discharge (ESD) strap (tester)

Procedure:

1. Confirm water temperature 70±15˚F and no major obstructions in sky.
2. Connect Arduino to PC. Ensure an active connection exists.
3. Submerge sensor to 1.5' and target 1.5'.
4. Face sensor towards target and vary the distance from the sensor and the target.
5. Record the sensor output data.
6. Repeat this procedure for all six sonar sensors.

Results: Recorded values.

IMU/Depth Tracking Test

Determine: Whether sensor reliably/accurately measure depth of current position as well as horizontal orientation.

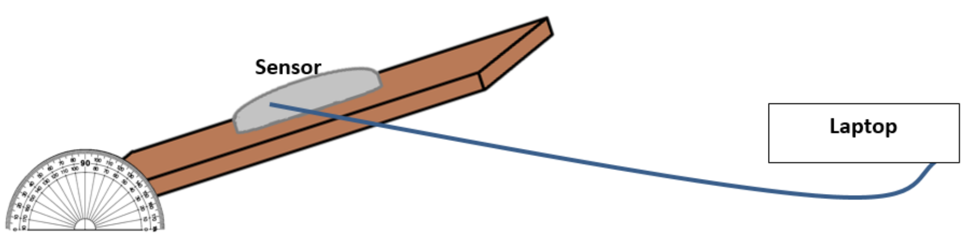
Materials: One IMU/Depth sensor module, one Arduino Mega 2560 r3, one water safe flat surface 1-inch x 2-inch, one laptop, one compass with degree heading, one water container, three gallons of water.

Location: Anu's apartment in a water safe area (Murray Ave.)

Safety: ESD strap (tester)

Procedure: Orientation

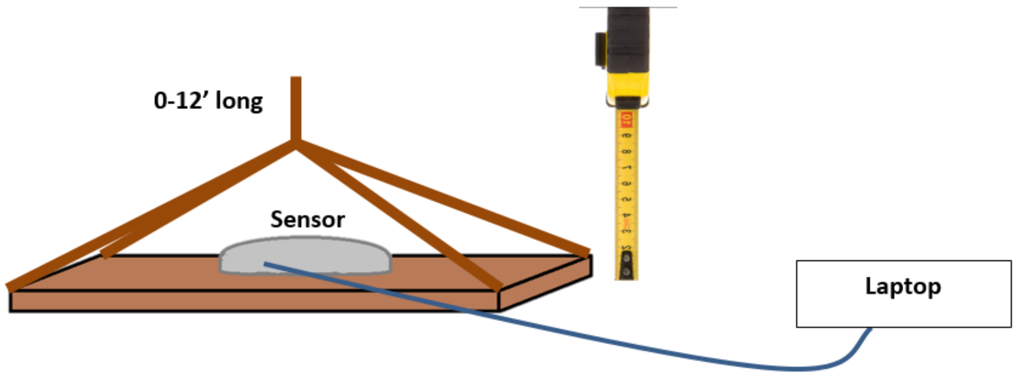
1. Attach Sensor module to flat surface platform securely.
2. Connect sensor to Arduino.
3. Connect Arduino to computer.
4. Run initialization program on module.
5. Tilt the sensor platform on various sides and angles.
6. Use protractor to measure exact angles of test and compare with received data from sensor.
7. Rotate sensors in all directions to read compass heading and compare with actual compass on hand.



Results: If compass heading from sensor matches actual compass heading, the test passes. Otherwise, the test fails.

Procedure: Depth

1. Connect sensor to Arduino.
2. Connect Arduino to PC.
3. Run initialization program on module.
4. Submerge sensor/platform assembly in water up to 1' depth.
5. Measure sensor depth and compare with output data on computer.



Results: If actual sensor depth matches output data the test passes, else test fails.

Autonomous Lighting Test

Determine: Whether sensor reliably/accurately determines adequate lighting conditions

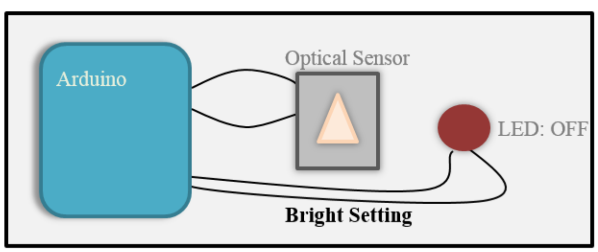
Materials: One 2760326 optical sensor, one Arduino Mega 2560 r3, one laptop, variable lighting conditions (dimmer light), led lights.

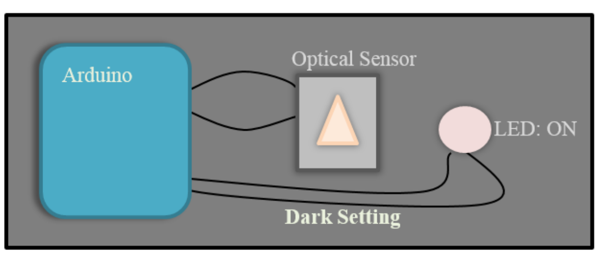
Safety: ESD strap (tester)

Location: Jordan's apartment (Cal Poly PCV - Dover)

Procedure:

1. Connect optical sensor to Arduino.
2. Connect LED to Arduino.
3. Connect Arduino to computer.
4. Run initialization program on sensor.
5. Begin with full light and slowly dim lights.
6. Obtain critical point value on optical sensor and calibrate as such.
7. When critical dimness is achieved LED, representing lighting system, should power on.





Appendix R: Design Validation Procedure and Report

Report Date: 6/03/2015		Hydromodus	Bridget Benson							REPORTING ENGINEERS: Anu Mahinkanda, Alex Kost, Jordan Read			
TEST PLAN										TEST REPORT			
No.	Specification	Test Description	Acceptance Criteria	Test Responsibilit	Test Stage	SAMPLE Qty.	SAMPLE Type	TIMING Start Date	TIMING Finish Date	TEST RESULTS Test Result	TEST RESULTS Qty. Pass	TEST RESULTS Qty. Fail	NOTES
1	Weight	*Weight the final assembly on a bathroom scale.*	*< 50 lbs.*	Team	PV	*1*	C	*6/1/2015*	*6/1/2015*	Passed	1	0	System weighed < 50 lbs.
2	Baseline Model Production Cost	*$850.00*	*< $850*	Team	CV	*1*	A	*N/A*	*N/A*	Failed	0	1	Over budget
3	BRUV Component Additional Cost	*$1,150.00*	*< $1150*	Team	CV	*1*	A	*N/A*	*N/A*	Failed	0	1	Over budget
4	Operational time	*Run the completed system with preprogrammed dive at max depth*	*> 6 hours*	EE	PV	*3*	C	*3/29/2015*	*6/4/2015*	Electronics ran for ~2 days	3	0	Motors not included in test
5	Modularity	*Install fake components on design*	*N/A*	ME	DV	*1*	B	*3/11/2015*	*5/29/2015*	Were able to attach external light, GoPro	1	0	Plenty of space available
6	Depth	*Place watertight vessel underwater with a piece of paper inside. Visually determine if water leaks into the vessel.*	*No leaks*	ME	PV	*3*	C	*4/23/2015*	*5/28/2015*	Endcaps failed, thru hull connectors passed	1	2	System was not waterproof
8	Peak propulsion speed	*Manually control thrusters to max power*	*> 2 knots*	Team	PV	*3*	C	*3/29/2015*	*5/19/2015*	Unable to perform test	0	0	System was not waterproof
9	Positioning	*Carry module while connected to microcontroller, read outputs and compare to absolute values from phone GPS.*	*Resolution of Depth: ± 3.3 ft.*	EE	DV	*5*	B	*3/11/2015*	*4/25/2015*	GPS was able to record locations ± 1 ft.	5	0	Test in Anu's Backyard
10	Ability to sense surroundings	*Test above water with measuring tape, microcontroller and computer hookup; afterwards, test in ASI Recreation Center Pool*	*Sonar identifies object 27 feet away*	EE	DV	*5*	B	*3/11/2015*	*5/28/2015*	Testing in air, the sonar sensor was able to accurately detect objects between 10 - 95 in. away	0	5	Test in Anu's Backyard, Yardstick used, ~8 feet determined to be sufficient
11	Autonomy	*Observe if Hydromodus follows procedures programmed in*	*Performs all procedures programmed*	EE	PV	*5*	B	*3/11/2015*	*5/5/2015*	Unable to perform test	0	0	System incomplete
12	Lighting	*Use GoPro and lighting in conjunction to test whether objects are identifiable beyond 27 feet underwater. Need tape measure.*	*If object is visible 27 feet away*	EE	DV	*3*	B	*3/11/2015*	*N/A*	Lighting solution found provides 3x performance over minimum requirement	3	0	600 lumen light, 205 lumen requirement
15	Design for Assembly	*Open the pressurized vessel with wrench under 60 seconds*	*< 60 seconds*	ME	DV	*5*	B	*3/11/2015*	*N/A*	Although difficult, it can be done	4	1	Could be easier with socket wrenc
16	Optical capability & recording	*Turn on GoPro for 6 hours*	*> 6 hours*	EE	CV	*1*	A	*3/11/2015*	*6/1/2015*	GoPro recorded for 6 hours	1	0	Battery BacPac, 32GB memory, 1080p 30 fps
17	Baiting	*Test bait at Avila Pier; Jordan will test on dive*	*> 1 Fish*	Jordan	DV	*1*	B	*N/A*	*N/A*	Bait never tested	0	1	System incomplete
18	Safety	*Attempt to cut paper with corners and edges; confirm no risk of electrical shorting*	*All objects undamaged; follows NESC requirements*	Team	PV	*1*	C	*6/2/2015*	*6/2/2015*	Mechanical design has inherent risk; electrical connections adhered to IPC standards	1	0	All higher voltage/current wires adequately insulated

Appendix S: Mechanical Failure Mode and Effects Analysis

FAILURE MODE AND EFFECTS ANALYSIS

Item:	Vessel & Thru Hull Connectors	Responsibility:	Alex Kost	FMEA number:	5
Model:	Current	Prepared by:	Alex Kost	Page :	1 of 1
Core Team:	Alex Kost (ME), Anu Mahikanda (EE), Jordan Read (EE)			FMEA Date (Orig):	2/19/2015 Rev: 1

Process Function	Potential Failure Mode	Potential Effect(s) of Failure	Sev	Class	Potential Cause(s)/ Mechanism(s) of Failure	Occur	Current Process Controls	Detec	RPN	Recommended Action(s)	Responsibility and Target Completion Date	Action Results: Actions Taken	Action Results: Sev	Action Results: Occ	Action Results: Det	Action Results: RPN
End Cap Assembly	*Endcaps fail*	*Water leak and electronic failure*	*10*		*Wrong size drill bit*	3	*Feedback from Mustang Hanger*	3	*90*	*Add epoxy, provide additional bracing*	*Alex Kost*		*4*	3	*1*	*12*
Thru Hull to End Cap	*Thru hull connectors fail*	*Water leak and electronic failure*	*10*		*Improper measurement*	5	*None*	*4*	*200*	*Add epoxy,*	*Alex Kost*		*6*	5	*1*	*30*
Acrylic Tube	*Vessel cracks*	*Water leak and electronic failure*	*10*		*Improper tooling setup*	5	*None*	5	*250*	*Repurchase thicker tube, replace*	*Alex Kost*		5	5	*1*	*25*

Item:	Chassis	Responsibility: Alex Kost	FMEA number:	6
Model:	Current	Prepared by: Alex Kost	Page :	1 of 1
Core Team:	Alex Kost (ME), Anu Mahikanda (EE), Jordan Read (EE)		FMEA Date (Orig):	2/18/2015 Rev: 1

Process Function	Potential Failure Mode	Potential Effect(s) of Failure	Sev	Class	Potential Cause(s)/ Mechanism(s) of Failure	Occur	Current Process Controls	Detec	RPN	Recommended Action(s)	Responsibility and Target Completion Date	Action Results: Actions Taken	Sev	Occ	Det	RPN
Machine holes	*Holes too big/small*	*Improper sizing for other components*	*4*		*Wrong size drill bit*	*3*	*Feedback from Mustang Hanger*	*3*	*36*	*Check hole size with scrap metal*	*Alex Kost*		*4*	*3*	*1*	*12*
	Holes improperly placed	*Assymetric design affects general stability*	*6*		*Improper measurement*	*5*	*None*	*4*	*120*	*Measure three times, mark twice, drill once*	*Alex Kost*		*6*	*5*	*1*	*30*
	Holes improperly placed	*Assymetric design affects general stability*	*5*		*Improper tooling setup*	*5*	*None*	*5*	*125*	*Use correct tooling; ask for help*	*Alex Kost*		*5*	*5*	*1*	*25*
Weld pieces together	*Poor weld, weak linkage*	*Failure of chassis*	10		*Bad welder*	*6*	*None*	5	300	*Ask Professor Kevin Williams for assistance*	*Alex Kost*		10	1	1	10
	Not enough welding material placed	*Failure of chassis*	10		*Bad welding resources*	*3*	*None*	1	30	*Utilize Cal Poly resources*	*Alex Kost*		10	1	1	10

Appendix T: Electrical Failure Mode and Effects Analysis

FAILURE MODE AND EFFECTS ANALYSIS

Item:	Sensor System	Responsibility:	Jordan Read	FMEA number:	1
Model:	Current	Prepared by:	Jordan Read	Page :	1 of 1
Core Tean	Jordan Read (Lead), Anu Mahinkanda (Electrical Engineering), Alex Kost (Mechanical Engineering)			FMEA Date (Orig):	2/22/2015 Rev: 1

Process Function	Potential Failure Mode	Potential Effect(s) of Failure	Sev	Class	Potential Cause(s)/ Mechanism(s) of Failure	Occur	Current Process Controls	Detec	RPN	Recommended Action(s)	Responsibility and Target Completion Date	Action Results: Actions Taken	Sev	Occ	Det	RPN
Object Detection	Object too close	Innacurate distance measurement	2	x	Foreign object moving too quickly to avoid	2	AUV moves away from any objects within its proximity	1	4	None		N/A	0	0	0	0
IMU / Depth	Sensors mis-calibrated	Unable to accurately control position	7	x	Improper machine set up	2	Operator training and instructions	4	56	Minimum: include some indicator that accurate depth is being measured (light on at surface)	Jordan/Anu: 4/1/15	Surface indicator light installed	5	2	2	20
IMU / Depth	Sensor damaged	Unable to accurately control position	7	x	Abusive marine animal	2	None	8	112	Write fall-back safety program to surface upon loss of comunication with IMU/Depth Sensor	Jordan/Anu: 4/1/15	Emergency program is implemented	6	2	2	24
GPS	Sensor unable to connect	Unable to accurately control position	5	x	Bad weather	2	Avoiding deployment in unfavorable conditions	1	10	None		N/A	0	0	0	0

FAILURE MODE AND EFFECTS ANALYSIS

Item: User System	Responsibility: Jordan Read	FMEA number:	2
Model: Current	Prepared by: Jordan Read	Page :	1 of 1
Core Tean Jordan Read (Lead), Anu Mahinkanda (Electrical Engineering), Alex Kost (Mechanical Engineering)		FMEA Date (Orig):	2/22/2015 Rev: 1

Process Function	Potential Failure Mode	Potential Effect(s) of Failure	Sev	Class	Potential Cause(s)/ Mechanism(s) of Failure	Occur	Current Process Controls	Detec	RPN	Recommended Action(s)	Responsibility and Target Completion Date	Action Results: Actions Taken	Sev	Occ	Det	RPN
Video System	*Running out of memory*	*Loss of valuable data*	*7*	*x*	*Operator forgetting to clear memory*	*3*	*Operator training and instructions*	*10*	*210*	*Ensure operator is properly instructed on clearing memory*	*Customer: TBD*	*Operator is properely instructed*	*6*	*1*	*10*	*60*
Video System	*Running out of power*	*Loss of valuable data*	*7*	*x*	*Power supply not fully charged*	*2*	*Operator training and instructions*	*2*	*28*	*Minimum: include some indicator that denotes current power levels*	*Jordan/Anu: 4/1/15*	*Power Indicator is Installed*	*5*	*2*	*1*	*10*
Bait System	*Loss of bait*	*Loss of valuable data*	*4*	*x*	*Abusive marine animal*	*2*	*None*	*10*	*80*	*Research object identification cameras*	*Jordan/Anu: 4/1/15*	*Object identification camera is installed*	*4*	*2*	*2*	*16*

FAILURE MODE AND EFFECTS ANALYSIS

Item:	Controller	Responsibility:	Anu Mahinkanda
Model:	Current	Prepared by:	Anu Mahinkanda
Core Team:	Jordan Read (Lead), Anu Mahinkanda (Electrical Engineering), Alex Kost (Mechanical Engineering)		

FMEA number:	4		
Page :	1 of 1		
FMEA Date (Orig):	2/22/2015	Rev:	1

Process Function	Potential Failure Mode	Potential Effect(s) of Failure	Sev	Class	Potential Cause(s)/ Mechanism(s) of Failure	Occur	Current Process Controls	Detec	RPN	Recommended Action(s)	Responsibility and Target Completion Date	Action Results: Actions Taken	Sev	Occ	Det	RPN
Current Draw	*The Microcontroller draws too much current*	*Potential damage to microcontroller*	*6*	*x*	*improper connections (short circuit)*	*3*	*Operator training and instructions, proper wiring*	*2*	*36*	*testing*	*Anu/Jordan: 4/1/15*	*Visual Inspection / Testing*	*4*	*6*	*2*	*48*
		Potential damage to connected peripherals	*3*	*x*	*improper connections (short circuit)*	*3*	*Operator training and instructions, proper wiring*	*2*	*18*	*testing*	*Anu/Jordan: 4/1/15*	*Visual Inspection / Testing*	*4*	*6*	*2*	*48*
			3	*x*	*peripherals current rating not accurate*	*1*	*None*	*1*	*3*	*None*		*N/A*	*0*	*0*	*0*	*0*
		battery drain	*6*	*x*	*short circuit/ innacrate information of peripherals*	*3*	*Operator training and instructions*	*3*	*54*	*short circuit detection, add peripheral testing*	*Anu/Jordan: 4/1/15*	*Short Circuit Protection / Tetsing*	*6*	*4*	*4*	*96*
Programming	*improper sensor readings*	*unable to accurately control position*	*7*	*x*	*improper calibration, error in code, damage*	*5*	*testing and calibration*	*2*	*70*	*additional testing and calibration*	*Anu/Jordan: 4/1/15*	*Calibration*	*5*	*4*	*3*	*60*
		collision with foreign objects	*7*	*x*	*calibration, error in code, damage*	*5*	*testing and calibration*	*2*	*70*	*additional testing and calibration*	*Anu/Jordan: 4/1/15*	*Calibration*	*5*	*4*	*3*	*60*

Appendix U: Vendors and Contact Information

Alibaba 400 South El Camino Real, Suite 400 San Mateo, CA 94402 1.408.785.5580 http://www.alibaba.com/	**LED Supply** 44 Hull Street Randolph, VT 05060 1.802.728.631 http://www.ledsupply.com/
Amazon 410 Terry Ave. North Seattle, WA 98109-5210 1.206.266.1000 http://www.amazon.com/	**McMaster-Carr** 9630 Norwalk Blvd. Santa Fe Springs, CA 90670-2932 1.562.692.5911 http://www.mcmaster.com/
Blue Robotics (No address) Redondo Beach, CA 90278 (No phone number) http://www.bluerobotics.com/	**Miner's Ace Hardware** 2034 Santa Barbara Ave San Luis Obispo, CA 93401 1.805.543.2191 http://www.acehardware.com/
Future Electronics, Inc 237 Boul. Hymus Pointe-Claire Quebec, Canada 1.514.694.7710 http://www.futureelectronics.com/	**OpenROV** 2222 Third St. Berkeley, CA 94710 1.510.859.3207 http://openrov.com/
Hobby King 14/F, Kerry Warehouse (Shatin) 36-42 Shan Mei Street Fotan, Hong Kong 8.52.3586.0327 http://www.hobbyking.com/	**RadioShack** 481 Madonna Rd. Ste A San Luis Obispo, CA 93405 1.805.544.5400 http://www.radioshack.com/
The Home Depot 1551 Froom Ranch Rd San Luis Obispo, CA 93405 1.805.596-0857 http://www.homedepot.com/	**Sparkfun** 6333 Drycreek Parkway Niwot, CO 80503 1.303.284.0979 http://www.sparkfun.com/