

---

# Naiad AUV

Weronica Kovala (project manager),  
Anette Hilmersson (hardware lead),  
Jonatan Scharff Willners (software lead), Carl Ahlberg,  
Lars Asplund, Hampus Carlsson, Lennie Carlén Eriksson,  
Mikael Ekström, Andreas Gustavsson, Joakim Gustafsson,  
Ande Hana, Branko Miloradovic, Martina Öhlund

## Abstract

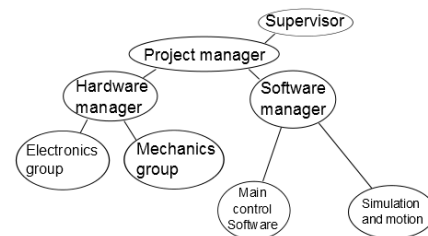
*Naiad is an AUV, designed and built during the last two years at Mälardalen University. The university has participated in RoboSub since 2012, but Naiad is a completely new robot. Naiad is built with a mono-hull which is designed to make it as hydro dynamic as possible. Modularity has been one of the aims in the designing of mechanics, electronics and software. The electronics for Naiad is mainly built on campus and it has been designed to fit the purpose and the design of Naiad to get the system as compact and modular as possible. For software, the system has been built in Ada with the aim to make a robust and modular system.*

## 1. INTRODUCTION

Naiad is an Autonomous Underwater Vehicle (AUV) designed and built almost completely by M.Sc students at Mälardalen University in Västerås, Sweden. The project started in 2013 and for the majority of the time the team has contained 18 students majoring in robotics and intelligent embedded systems. At the moment there are 8 M.Sc students, 3 PhD students and 2 seniors in the team.

The group was divided into four sub-groups which were managed by a project manager, a hardware and a software lead. The hardware group was divided into two groups, mechanical and

electronics. In a similar fashion, the software group was divided into two sub-groups, one software group and one for simulation and motion (Fig. 1).



**Figure 1:** The group was divided into four sub-groups, two for hardware and two for software.

Part of the task for the team has been to finance the project. The uni-

---

versity provided around \$ 5000 for the project and the rest has come from sponsors. The low number of team members has made it impossible to let one person focus on marketing, which has created some difficulties when it comes to financing.

The rest of this paper is structured as follows. In Section 2 the hardware, both mechanics and electronics, are described. In Section 3 the software system of Naiad is described. Section 4 describes the future works of the robot.

## 2. HARDWARE

The hardware of Naiad has been divided into two groups, mechanics and electronics.

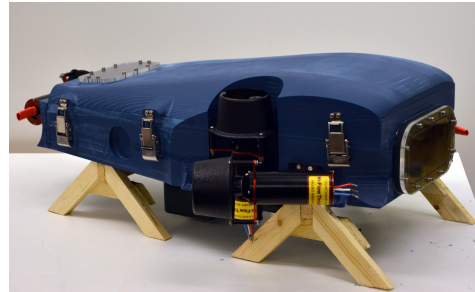
### A. Mechanics

The main hull of Naiad has got a design which is made to be hydrodynamic and modular. Compared to previous designs from Mälardalen University this AUV is completely different, both in its design and its thruster configuration. Fig. 2 shows a CAD model of Naiad. All of the mechanics has been modeled in Solid Works.



**Figure 2:** A concept image of the design of Naiad from Solid Works.

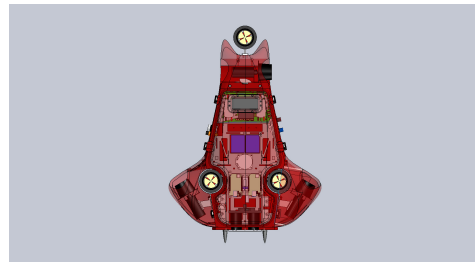
Naiad's hull has been milled completely in the university and it is made out of polyurethane plastic, which was chosen due to the fact that it is easy to work with whilst still being hard and relatively cheap. An image of the finished hull can be seen in fig. 3.



**Figure 3:** One of the first images of the main hull when it was first put together.

### A.1 Thruster configuration

The thrusters used on Naiad are Crustcrawler 400HFS thrusters. The thrusters are configured using the configuration seen in figure 4. This type of configuration was chosen to make the AUV more agile and responsive than the configuration-type previously used in the university's AUVs.



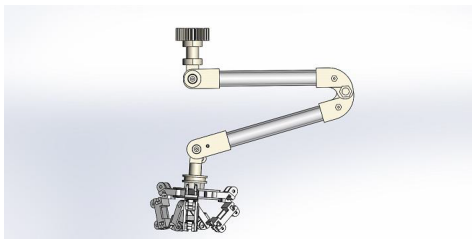
**Figure 4:** The thruster configuration of Naiad.

The configuration enables 6 degrees

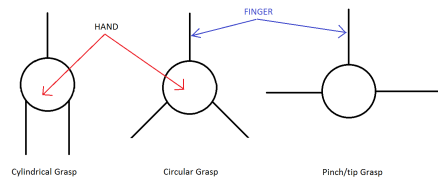
of freedom (DOF). Three linear motions and three rotational motions. The linear motions are heaving (vertical movement), sway (side to side movement), surge (forward or backward motion). The three rotational motions are yaw (rotation around vertical axis), pitch (rotation around lateral, or side-to-side, axis) and roll (rotation around the longitudinal, or front/back, axis).

## A.2 Manipulator

Naiad has a tendon driven manipulator that can be seen in fig. 5. The manipulator consists of two main parts, an arm which is able to bend in two places as well as rotate, and a hand. The hand has three fingers which can be used to grip a wide range of objects since it can achieve a cylindrical grasp, a circular grasp and a pinch/tip grasp (fig. 6).

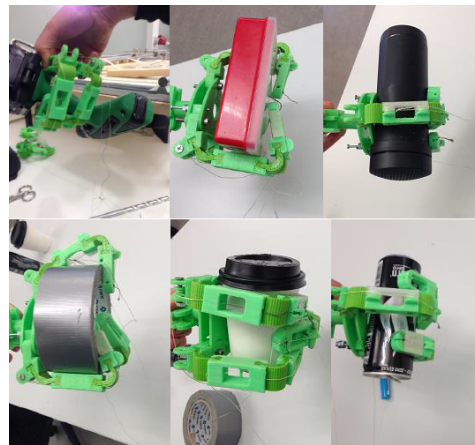


**Figure 5:** CAD-figure of complete manipulator with both arm and hand.



**Figure 6:** The three different grasps the manipulator can use.

The hand on the manipulator is illustrated in fig. 7. The gripper is naturally closed and has to be opened using a force, this ensures a stable grip.



**Figure 7:** The final version of the tendon driven hand on Naiad in a variety of gripping positions.

## A.3 Markers

The markers are released using a solenoid. The main issue with the markers has been to get them heavy enough in order for them to drop in a straight enough line from the AUV down to the bottom of the pool. In order to get this to work, the marker, which is 3D-printed is fastened onto a bolt to increase the weight.

---

The markers are mounted inside plexi-glass tubes close to the downward facing camera to improve aiming accuracy.

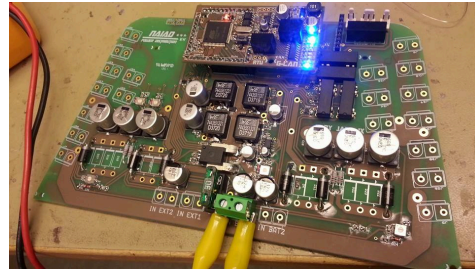
## B. Electronics

The AUV is built with modularity as a key objective, therefore it is easy to add and remove different parts of the electronics. As much as possible has been developed on campus to ensure that the electronics will fit the purpose properly. This section has been divided according to the different types of electronics in Naiad, one section regarding the Power Supply Unit, another for the generic CAN-cards and one for the different extension boards that are present inside Naiad.

### B.1 Power Supply Unit

Naiad is powered using two 22,2 V Lithium Polymer (LiPo) batteries which are connected to the Power Supply Unit (PSU). The PSU is used to power the complete system. It converts the battery voltage to the different supply voltages. To control the supply to different parts of the system, a CAN-card is mounted on the PSU. To ensure safety, there is a hardware kill switch for the motors on this card which makes it possible to stop the power supply to the motors without shutting down the whole system. The kill switch is placed on the outside of the hull and short-circuits this connection to allow the motors to run using a magnet switch.

Since the hull of Naiad is not squared in its shape, the PSU has been designed to fit the hull as closely as possible. The PSU is shown in fig. 8 where it is mounted together with the CAN-card that is used to control the power supply to the rest of the system.



**Figure 8:** *The PSU with the CAN-card controlling the power supply to the rest of the system mounted for testing.*

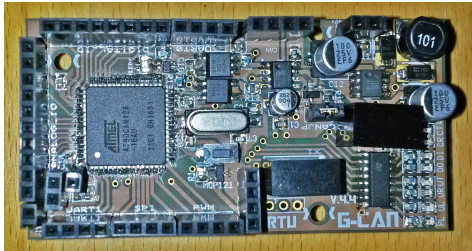
To monitor the system properly there are temperature sensors and current sensors on the PSU which make it possible to detect problems inside the hull and make the system act accordingly to add an extra dimension of safety and robustness to the system.

### B.2 CAN-card

A generic CAN-card has been created to form the base of the Controller Area Network (CAN) bus. The Microcontroller Unit (MCU) on these are the AT90CAN128. In order for the system to be modular the CAN-cards are placed in stacks using a pin header on the card and a socket header on the stacks which makes it possible to slide the cards into the socket header. In a similar fashion, there are extension

---

boards to change the functions of the generic CAN-cards. For example there is an extension board for the motor signals, the sensor handling, and for the actuators. The CAN-card can be seen in fig. 9.



**Figure 9:** *The generic CAN-card.*

### B.3 Extension boards

There are a number of extension boards in the electrical system of Naiad.

- Motor extension - the thrusters are controlled using a Phoenix Ice2 HV 60 motor controllers used with the Castle Link V3.52.10 software. The signal to the motor controllers come from the motor extension board which provides the controllers with 5V and GND from the CAN-cards as well as a PWM-signal to each motor. One motor extension board can control all six motors on Naiad.
- Sensor extension - the sensor extension board is used to transfer the sensor data from the sensors to the CAN-bus. At the moment the only sensor that uses this board is the pressure sensor.

- Actuator extension - the actuator extension board is used to control the solenoid that releases the markers. It is also prepared for releasing torpedoes in a pneumatic system, which is yet to be implemented.
- INS extension - the INS board contains the electronics to get the measured data from the Inertial Measurement Unit (IMU) through a UART interface. It is also able to transfer the data from the Fiber Optical Gyroscope (FOG) to the CAN-card using a SPI-interface.

## 3. SOFTWARE

The software section has been divided into three parts, one for the main control which handles motion and internal communication in Naiad, another section for the vision system and a third section regarding the simulator.

### A. Main control

The main computer on Naiad is a Beagle Bone Black (BBB) where, both on this card and on the CAN-cards, all code is written in Ada taking Ravenscar into consideration. All of the node to node communication is based on either CAN or Transmission Control Protocol (TCP) where the former is communication between CAN-cards. The latter is between nodes that are either on a BBB, a GIMME-2 card or other nodes that are connected via Ethernet.

A node on the CAN-bus is still able to communicate with a node on a BBB by passing through a translation node.

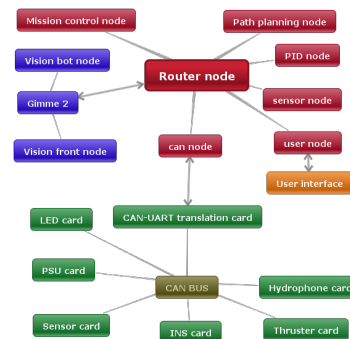
A node can choose what type of values to subscribe to by sending a message containing this information to the routing node. When a node sends data to the routing node, the data will be distributed to all nodes that needs this information. With this approach, the system and nodes do not need to be changed when implementing a new node/task. This leads to starting a new node for the first time can be implemented when the code for that node is completed, reducing development time for new functions.

With all communication going through a central node, and no direct communication, there will be an overhead, but with small messages the benefit of this modular system outweighs this drawback. An other advantage of this type of system is safety, in a safety-critical system such as a robot, a node dying can cause the whole system to fail, with a central node keeping track of all nodes still being alive this problem can be found and solved. A flowchart of the node communication in the system can be seen in fig. 10.

To control the motion of Naiad, one Proportional Integration Derivative (PID) and five Proportional Derivative (PD) controllers were used. The PD/PID controller bases its control values on the difference (error) between the desired and the current state of the robot.

To determine the position of the AUV, the accelerometer data from the

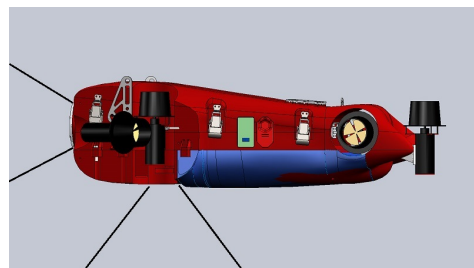
IMU are integrated twice. However, there is a lot of noise on this signal and when the signal is integrated, the noise increases. To reduce the noise a Kalman filter is used.



**Figure 10:** Flowchart of the complete software system

## B. Vision

There are two stereo vision camera systems on Naiad, one that is facing forward and one downwards, which can be seen in fig. 11.



**Figure 11:** Illustration of the placement of both stereo vision systems.

The vision-system in Naiad are two GIMME-2 stereo-vision boards. The

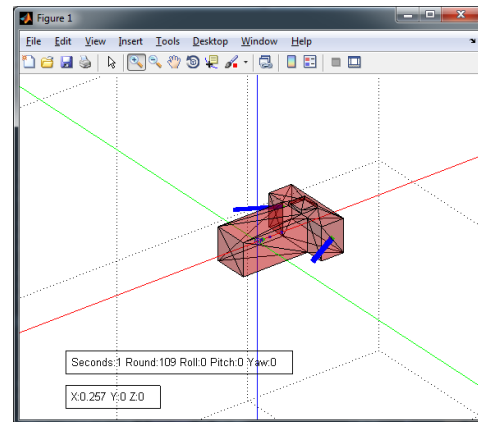
processor on these boards is a zynq processor which has an ARM Cortex A9 dual-core processor combined with an FPGA in the same chip, which enabled some of the vision to be done directly on an FPGA, which makes the processing fast. There are two 10 megapixel image sensors on each GIMME-2 board. Since the image processing is done on the GIMME-2 board, the system for front and downward vision are completely separated. The OpenCV library is used for image processing, which significantly reduces software development time. In order to make the use of OpenCV easier, the code for the vision system is written in C/C++.

The images are analysed directly on the boards and the relevant information is sent to the router node in the system. Relevant information is for example information about distance and angle towards the object.

### C. Simulation

The simulator for Naiad has two modes, either simulation or IMU mode. In simulation mode a virtual simulated representation of the Naiad robot is run with simulated sensors and actuators. The simulation allows you to be connected directly to the PID controller through TCP sockets, sending equal data as the real sensor node would have done if the robot was running a real task and receiving the real motor values that the PID controller would send to the real robot. These motor values are turned into estimated forces

that is shown as dynamic lines coming from the different motor positions to give a good visualization of the forces that is acting on the robot. An illustration of the simulation can be found in fig. 12.



**Figure 12:** Concept image from the simulation environment.

In IMU mode the simulator can be set into a passive mode that shows a virtual visual representation of the estimated position and orientation that the system in real time has estimated. No calculations is needed in this mode more than rotating the virtual robot in the right order.

## 4. STRATEGY AND CURRENT STATUS

With about one month left for the competition the main issue is to get the object detection to work properly. The robot is able to move properly on command but without vision it is hard to know where to move. Therefore this is the main issue at this point.

---

The markers and the manipulator are yet not attached to the hull, which is a concern for the team. The aim for the mechanics group is, at this point, to try getting these finished in the time left before the competition. The first aim is to get the missions that does not need these features.

Since the university does not have a pool on its own the public pool has been used, which made testing harder since it becomes an all day task for almost the entire team to move everything there, do the testing and then moving everything back. A smaller garden pool which is 5 m wide has also been available on campus. This pool has two major limitations; since it is placed outside it is not possible to use it during the winter and its size makes it impossible to use it for testing full missions.

## 5. ACKNOWLEDGEMENTS

We would like to thank everyone, both faculty and others, who has helped the team with the development of Naiad; Emil Bergström, Nils Brynedal Ignell, Cezar Chiru, Gerard Duff, Fredrik Ekstrand, Martin Ekström, Simon Elgland, Christoffer Holmstedt, Omar Jamal, Konstantinos Konstantopoulos, Joel Larsson, Daniel Lindqvist, Johan Mattsson, Michael Mayer, Per-Erik Måhl, Nahro Nadir, Debajyoti Nag, Niclas Rasmusson, Aseem Rastogi, Patrik Pärlefjord, Rizwin Shooja, Wenkai Wang. And our sponsors;

**Level 1:** Würth Elektronik

**Level 2:** Robotdalen, FoU Rådet, Deep-

vision

**Level 3:** Kvaser, Packsize