STARFISH: An Open-Architecture AUV and its Applications

Chew J.L., Koay T.B., Tan Y.T., Eng Y.H., Gao R., Chitre M., Chandhavarkar N.
ARL, Tropical Marine Science Institute
National University of Singapore
18 Kent Ridge Road, Singapore 119227
Email: jeeloong@arl.nus.edu.sg

***Abstract*—Autonomous Underwater Vehicles (AUVs) have been long desired as sensing platforms for logistics and economic reasons. Most commercially available AUVs are expensive or have very limited capabilities. Many of them have limited modularity, which poses an obstacle to adaptation of these AUVs to include new functionalities. This is usually a result of proprietary vehicular interfaces and hardware/software architectures.**

In 2006, we embarked on a project to develop a small team of AUVs, called STARFISH - Small team of autonomous robotic fish. We aimed to develop a team of modular, low-cost AUVs with a design that supports extensions to add heterogeneous capabilities. An open-architecture framework that includes mechanical, electrical and software interfaces was incorporated into the design of STARFISH AUVs. This not only allows the users to easily integrate their proprietary modules with the AUV, but also permits the insertion and swapping of software subsystems within vehicle to alter any desired aspect of the vehicle.

The guidance, navigation and control (GNC) intelligence of the STARFISH AUVs was modeled as a team of interacting software agents mimicking the hierarchical command structure in a scientific or naval vessel. These software agents are modular, reconfigurable and extensible in nature with well-defined interfaces. Such flexibility allows a design methodology that makes it possible for other parties to develop new modules independently, with minimal knowledge of the detailed AUV design.

This paper describes the different configurations of the AUVs that can be realized using the interfaces. A basic summary of these open architecture interfaces is also be presented to show the simplicity of adding new functionalities into the AUV. Lastly, we describe some applications and multi-asset field experiments that we have been conducting in Singapore waters.

I. Introduction

A number of AUVs have been successfully developed over the years. These AUVs initially served as scientific research tools and only a handful enter the commercial market. These AUVs range from any order in terms of size, endurance, capabilities and cost. Cost, capabilities and restrictive proprietary subsytems have been the greatest trade-offs to the ownership and operations of any AUVs.

The “Small team of autonomous robotic fish” (STARFISH) project was started in 2006 to develop a team of modular and cost efficient open-architecture AUVs. The open-architecture framework of mechanical, electrical and software interfaces was incorporated into the design of STARFISH AUVs.

The modularity allows the users to easily reconfigure a team of AUVs as per field requirements. Users can also easily integrate their proprietary modules with the AUV. Integration is not only bounded to hardware modules but is also extended to software subsytems that can be inserted and swapped within the vehicle. This grants the users flexibility to control and reconfigure a heterogeneous team of specialist AUVs.

In Fig. 1, we have a view of 2 STARFISH AUVs called Redstar and Bluestar during one of the open water trials at Selat Pauh, Singapore.

Fig. 1. Redstar & Bluestar

In Sec. II, we introduce the mechanical and electrical interface that promotes hardware modularity. Then in Sec. III, the distributed software architecture for autonomous vehicle (DSAAV) employed by STARFISH AUVs is presented. This then leads to the guidance, navigation and control (GNC) model in Sec. IV. Sec. V-D highlights the ease of development for different sections and payloads with several examples. Then, we look at some applications with the STARFISH AUVs in Sec. VI-C. Finally, in Sec. VII, we draw conclusions and state future developments.

II. Hardware Modularity

Hardware reconfiguration and integration are primarily achieved through a mechanical and electrical interface [2] [8]. The mechanical coupling interface uses a male-female interlocking mechanism with locking teeth, as in Fig. 3.

The electrical interface consists of a pair of hybrid connectors configured to carry four high-current power lines and a number of small signal communication lines (see Table I for details).

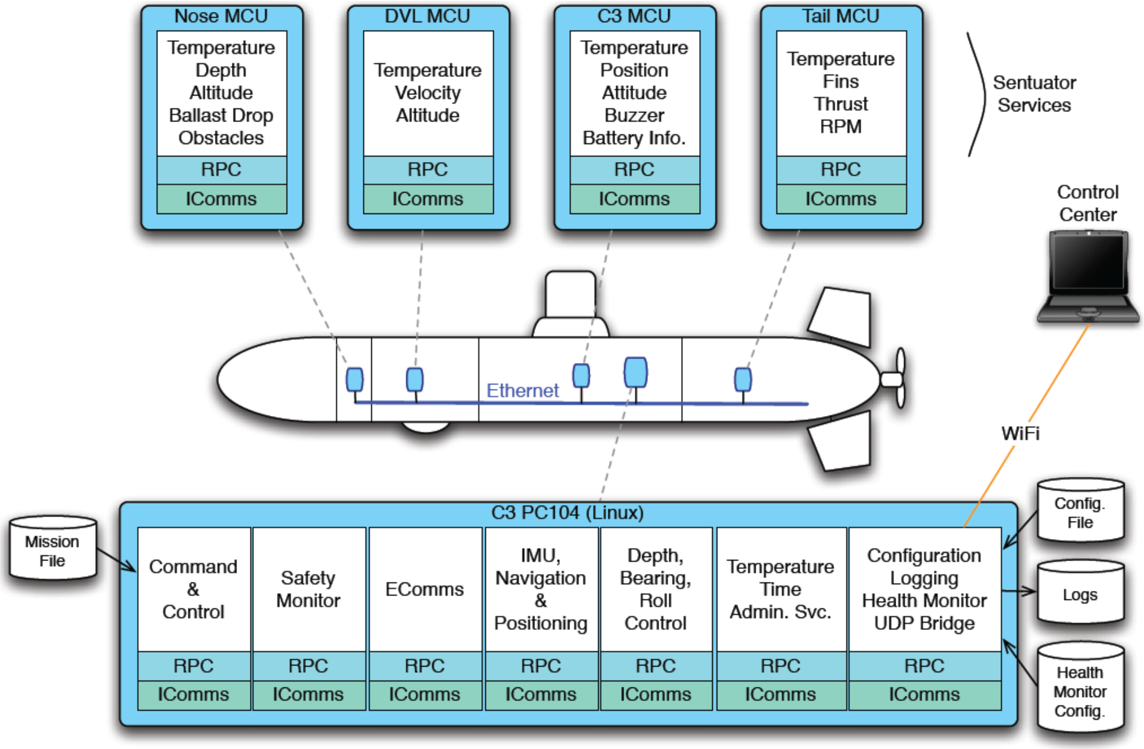
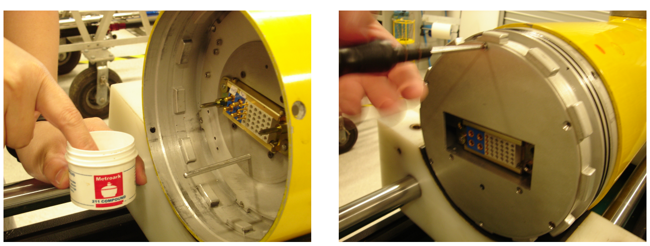


Fig. 2. Software Component Deployment



(a) Male coupling (b) Female coupling

Fig. 3. Section interface (mechanical)

TABLE I
SECTION INTERFACE (ELECTRICAL)

Connection	Description
48V battery bus	Connects to common battery pool; all the batteries in various sections connects to this bus.
48V system bus	Connects to common system power; all the electrical modules collects power from this bus
48V power return	Serves as power return bus for battery and system power bus
Ethernet bus	Serves as inter-section communication bus
Run Level	Serves as vehicle wide status indicator
Fuel gauge	Provides a means to interrogate battery status across all sections

All communication within the vehicle is achieved using Ethernet packets. Coupled together with the standardized electrical interface, electronics modularity within the section can then be achieved. As electronics within the vehicle can be distributed among various sections, this gives rise to the viability of a distributed software architecture, as to be discussed in Sec. III.

Communication to the external world is done through standard interfaces that include TCP/IP over Ethernet, WiFi, acoustic modem or GSM modem. This ensures compatibility with other systems.

III. SOFTWARE MODULARITY

The STARFISH AUVs employ the distributed software architecture for autonomous vehicle (DSAAV) [1] to provide flexible software reconfiguration. DSAAV uses remote procedure calls (RPC) construct that allows distributed deployment of software components within the AUV, as shown in Fig. 2. This makes re-deployment and even migration of software components across platforms (such as from PC104 to MCU) very easy.

Data "plumbing" or data flow between subsystems can be easily configured during deployment. For an example, in the current STARFISH AUVs, an altimeter provides altitude data to all subsystems that require it. When an optional DVL section is added, the higher quality altitude measurements from the DVL can instead be used by all subsystems without any change to their source code. The desired data "plumbing" can be achieved easily by changing the configuration file during deployment.

IV. GNC MODEL

The guidance, navigation and control (GNC) model for STARFISH AUVs employs a hybrid hierarchical model [3] [7]

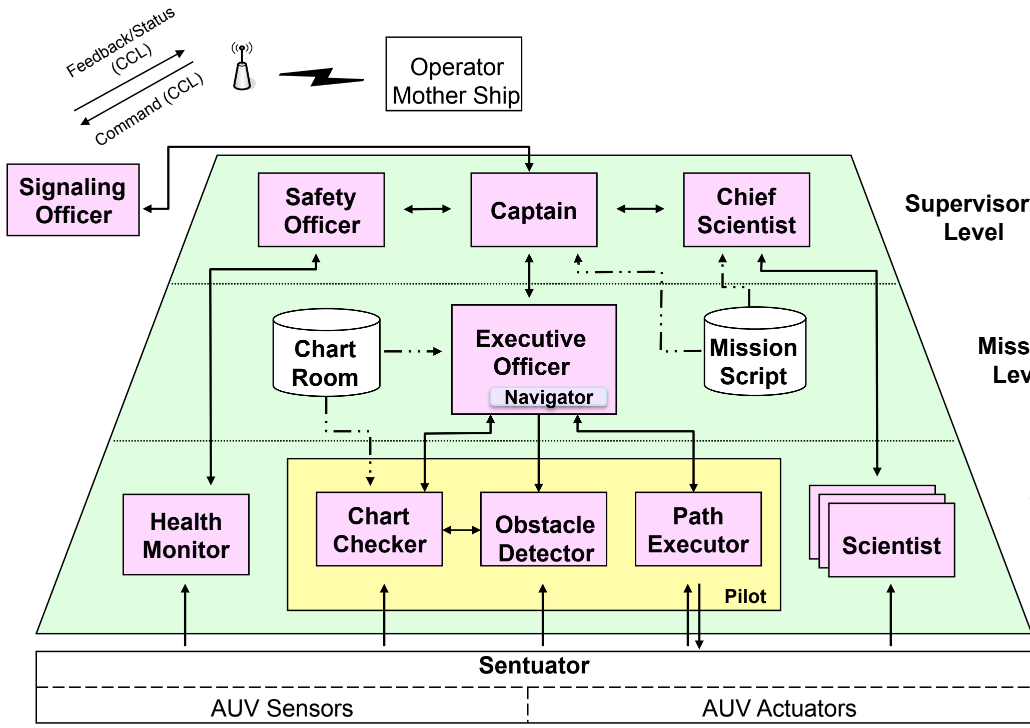


Fig. 4. Hybrid Control Architecture

for its software agents, as shown in Fig. 4. This hierarchical model mimics the command structure in a scientific or naval vessel. The hybrid architecture allows for deliberative high level mission control while decoupling the low level reactive vehicle control. The segregation of the task into individual agent components also present an explicit view of a clearly defined control responsibilities at different level of control hierarchy.

V. System Development

Modularity in hardware, software and operating model offers flexibility in system development of the STARFISH AUVs. Sections or payloads such as proprietary sensing modules can be independently developed by users or external parties for integration with the STARFISH AUVs. The only requirement is to ensure that the mechanical and electrical interface adheres to the standardized interface specifications. New and/or updated software subsystems can also be easily introduced into the STARFISH AUVs. This modularity and open-architecture offers users the ability to control and configure any desired aspect of the vehicle. The following sections illustrate these flexibilities with some examples that we have.

A. Nose section

Our current STARFISH AUVs are equipped with a sector scanning Forward Looking Sonar (FLS) in the nose section, among other sensors. If the need arises for the introduction of a multibeam or imaging sonar, a new nose section can be developed with the new sonar module while having the change transparent to other software agents. Software agents that require the multibeam sonar sensory information can subscribe to it through an update in the configuration file.

In Redstar's nose dry section, we have also included another set of battery pack. This additional power source isn't exclusive to the nose section but it becomes a part of the common power source in the AUV. It is then be shared across all sections.

B. C3 section

The modular architecture also allows flexibility in implementing different flavours of the vehicle. Different combinations of electronics modules have been implemented across the development cycle of the vehicle without disturbing the software and vehicle integrity.

Examples taken from STARFISH AUVs include the replacement of different versions of the acoustic modem for Bluestar and Redstar. An Inertial Measurement Unit (IMU) was implemented for Bluestar while not in Redstar. In addition, Redstar is equipped with an actively controlled strobe light and an additional GPS module.

C. Tail section

Another illustration of the architectural flexibility is the implementation of an internal roll ballast control system in Redstar, as depicted in Fig. 6. This internal roll ballast control system works hand-in-hand with the rudder and fins of the AUV to achieve better maneuverability, control and stability

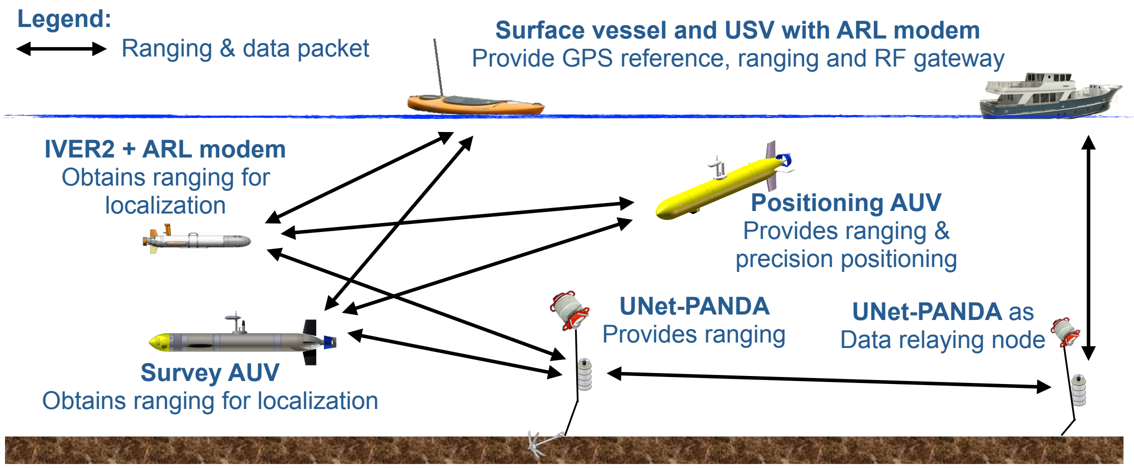


Fig. 5. A team of heterogeneous assets to sense marine environment collaboratively

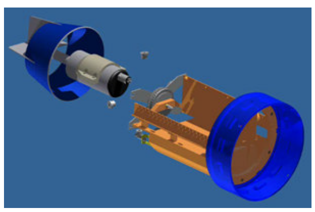
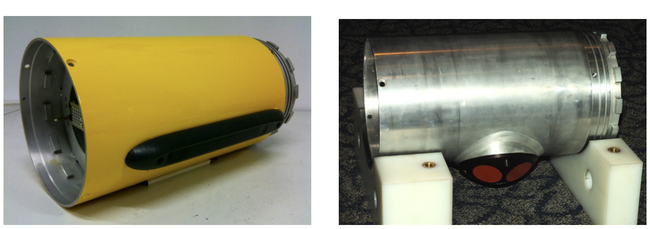


Fig. 6. Construction of Internal Roll Control

[5]. The new intelligence is encoded in the software agent in-charge of the low level vehicular control, while it is transparent to the high level GNC agents.

D. Payloads

Additional payloads integrated with STARFISH AUVs can immediately be utilized to execute its implementation objectives. Examples as with the sidescan that can be used to acquire bathymetry data and objects detection. DVL can be added in addition to an altimeter of the STARFISH AUV to yield higher positioning accuracy. In Fig. 7, the sidescan and Doppler Velocity Log (DVL) payload sections developed at ARL with collaboration with Center for Environmental Sensing and Modeling (CENSAM) at Singapore-MIT Alliance for Research & Technology (SMART) can be seen.



(a) Sidescan (b) DVL

Fig. 7. Payload sections - Developed by ARL

In Fig. 8, a LED Induced Fluorescence (LEDIF) optical sensor payload that is under development by Prof. Hemond's group at CENSAM is shown. The LEDIF sensor payload is part of a suite of sensors under development that can be used for in-water chemical sensing.

Fig. 8. LEDIF sensor payload - Development by Prof. Hemond's group at CENSAM

A myriad of payloads can be integrated with the STARFISH AUVs. Several of the payloads under development at ARL and CENSAM include a module with environmental and chemical sensors, a vector thrusting module and a tag release module for target tagging.

The list of applications possible with various payloads integration are simply endless. With the flexible software and hardware architecture in place, users can independently develop proprietary modules for their desired requirements; such as surveillance, weapons and counter-measures modules. Another advantage of such architecture is the user can easily adapt the vehicle according to the needs without having to release proprietary or sensitive information of the implementation.

VI. Heterogeneous Multi-assets Deployment

STARFISH AUVs are designed with the vision of operating within the setup of heterogeneous autonomous assets. With the modularity and flexibility of reconfiguration supported by the open architecture, it allows easy adaptations of the vehicle to interoperate with other assets. On top of that, other assets can also take advantage of its flexible architecture.

As a result, STARFISH AUV can be seamlessly deployed alongside other systems to form a set of collaborative assets, a possible configuration is depicted in Fig. 5. In this configuration, assets with heterogeneous capabilities provide different functionalities within the team. For example, surface vessel, USV and positioning AUV will provide good position fixes, bottom mounted systems such as UNet-PANDA for ranging and data relaying, along with survey AUVs for data collections. Many of these assets such as STARFISH AUVs, USV and surface vessel have employed variations of hardware and software configurations from the architecture.

An example of such adaptation is the common communication channel among the assets. In this case, an acoustic modem with integrated ranging functionality has been installed in the AUV with negligible hardware modifications. From software perspective, only the communication driver has been replaced.

The architecture will then make the communication and the new ranging functionality available to the other software agents within the vehicle. Appropriate agents such as Safety Officer and Chart-checker can then utilize the new ranging information in their algorithms if they choose to. This demonstrates the flexibility of the architecture in adapting the STARFISH AUV for the operation scenario.

The following sections describe some of the work in progress that will contribute to form the scenario in Fig. 5.

A. Cooperative Positioning between two STARFISH AUVs

In a scenario where there are limited payloads for field deployment, a heterogeneous team of assets that can cooperate with one another would be advantageous. We are currently working on a team of STARFISH AUVs equipped with complementary payloads to accomplish a desired mission objective cooperatively.

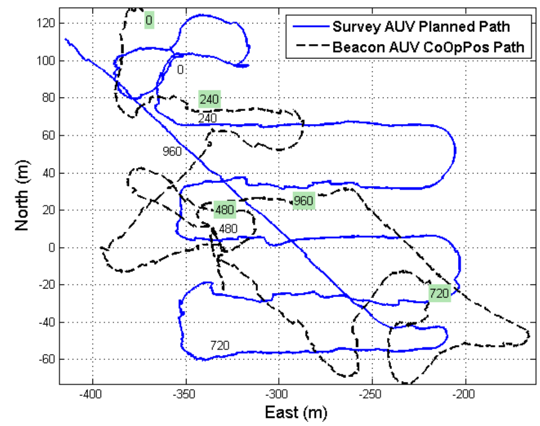
In an experiment setup to demonstrate cooperative positioning [4] [6], two STARFISH AUVs, Bluestar and Redstar was respectively configured as *beacon* (positioning AUV) and *survey* AUV. Bluestar was equipped with a DVL payload to execute the role as a positioning beacon. On the other hand, Redstar was equipped with a sidescan payload for the purpose of bottom imaging.

Our AUV architecture has allowed easy adaptation of STARFISH AUVs into these complimentary configurations. Specific payloads with appropriate software services are introduced and only adaptation in the position estimate software modules are needed while hte rest of the agents are untouched.

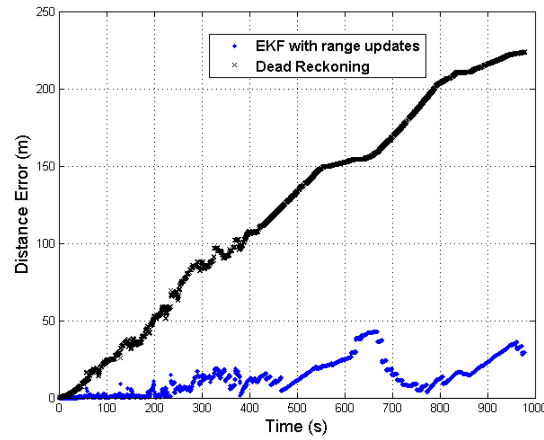
Several open-water trials were carried out at Selat Pauh where Redstar performed surveys in lawn-mover missions using only dead-reckoning for position estimates and ranging informations from BlueStar to improve the positioning accuracy.

In Fig. 9, the mission path executed by Bluestar and Redstar can be observed along with the bounded positioning error of Redstar. The results clearly exhibited the efficiency of cooperative positioning where positioning errors of a simply equipped AUV without high accuracy positioning capabilities could still yield good bounded estimates.

Cost savings in terms of payloads deployment is also achieved. Effectively, the required quantity of the DVL and sidescan payloads have been reduced by 1 each as compared to equipping every AUVs with both DVL and sidescan. This also increases operational and logistics efficiencies.



(a) Mission



(b) Results

Fig. 9. Cooperative Positioning

B. Cooperative mission between STARFISH AUV and USV

This section shows the capability of STARFISH's DSAAV software architecture to work with the USV developed by MIT. We aim to extend the cooperative positioning capability between two STARFISH AUV to between a STARFISH AUV and an USV from SMART. In this case, we need to run the same cooperative algorithms coded in DSAAV framework in the USV.

During a joint sea trial between ARL and CENSAM from SMART in June 2010, a STARFISH AUV and a USV (in the form of autonomous kayak) was deployed to conduct initial acoustic communication and ranging test between the 2 vehicles and STARFISH's C2 has been integrated with the USV. The USV operates on MOOS [9] while STARFISH AUVs operate on DSAAV [1]. In effort to allow interoperability, a software interface was coded to facilitate communication between MOOS and DSAAV, as in Fig. 10.

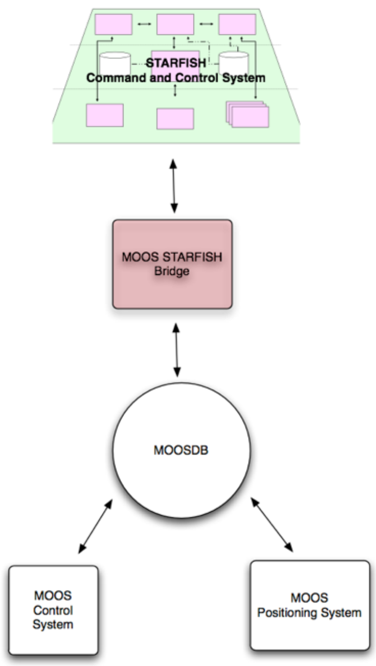


Fig. 10. Software Interface between STARFISH & MOOS

With the simple software bridge, DSAAV's C2 managed to utilize the positioning system, sensors along with the control system of the USV without any modification to the USV's C2 implementation in MOOS.

C. Underwater Networked Pop-Up Ambient Noise Data Acquisition (UNet-PANDA) system and STARFISH

A set of bottom mounted networked nodes, UNet-PANDA [11] [10], are currently being built in ARL as part of the heterogeneous assets. Each UNet-PANDA carries a compatible acoustic modem that provides both ranging and data communication to STARFISH AUVs.

The UNet-PANDA acquires GPS position before being deployed, which will provide relatively good estimates of the UNet-PANDA's underwater position in shallow water such as in Singapore waters. This will provide reasonable ranging information in improving the position estimates of AUVs that do not carry precision positioning system but uses similar modem. With the flexibility of the STARFISH software architecture, the C2 and other software agents will make use of the best positioning estimates available to them when the STARFISH AUV is within the communication range of the UNet-PANDAs.

The STARFISH C2 provide a framework through a software agent, Signaling Officer, to take advantage of heterogeneous physical layers from different communications. Users can communicate with the vehicle via electro-magnetic communications such as WiFi and GSM on surface, while through acoustic communications when in water. This allows the flexibility of being able to communicate with the AUV both for mission critical commands and data transfer. The UNet-PANDA provides a possibility of extending the communication range through networking facility.

VII. Conclusion

A modular and open-architecture framework of the STARFISH AUVs was presented. The architecture was designed to allow a fleet of heterogeneous, collaborative, low-cost AUVs to be easily used in real applications.

The section modularity in the hardware design and distributed software architecture allow easy migration of functionalities and control algorithms across different platforms. The clearly defined section architecture, both in terms of mechanical and electrical interfaces have provided an easy way to implement new sections with novel functionalities to extend the basic AUV.

Demonstrations of the operation of STARFISH AUVs along with the interoperability of STARFISH AUV with other systems have been presented. These highlight the effectiveness and advantages of the open-architecture design of the STARFISH AUV.

Currently, we are in development to build another 3 STARFISH AUVs along with prototype for the vector thrusting and tag release payloads. Trials are also in progress with the experimentations of STARFISH AUVs, SMART's AUVs and USVs along with UNet-PANDA.

Acknowledgment

The authors would like to acknowledge DSTA for their funding support for the STARFISH project. We would also like to acknowledge SMART for their funding of the collaborative CENSAM project that provided numerous opportunities for testing the STARFISH AUVs at sea.

The authors would like to thank Prof. Hemond and Dr Kelvin Ng from SMART for the information pertaining to LEDIF.

References

[1] Chitre M., "DSAAV - A distributed software architecture for autonomous vehicles," in Proceedings of IEEE OCEANS 2008, September 2008.

[2] Sangekar M., Chitre M., and Koay T. B., "Hardware architecture for a modular autonomous underwater vehicle STARFISH," in Proceedings of IEEE OCEANS 2008, September 2008.

[3] Tan Y.T., Chitre M. A., Vadakkepat P., and Shahabudeen S., "Design and Development of Command and Control System for Autonomous Underwater Vehicles," in DTA 2009, March 2009.

[4] Gao R., and Chitre M., "Cooperative positioning using range-only measurements between two AUVs," in IEEE OCEANS'10 Sydney, (Australia), May 2010.

[5] Y. H. Eng, G. S. Hong, and Chitre M., "Depth control of an autonomous underwater vehicle, STARFISH," in Proceedings of IEEE OCEANS'10 Sydney, May 2010.

[6] Chitre M., "Path planning for cooperative underwater range-only navigation using a single beacon," in International Conference on Autonomous and Intelligent Systems (AIS) 2010, (Povoa de Varzim, Portugal), June 2010.

[7] Tan Y.T., Chitre M., and Vadakkepat, P., "Hierarchical agent-based command and control system for autonomous underwater vehicles," in International Conference on Autonomous and Intelligent Systems (AIS) 2010, (Povoa de Varzim, Portugal), pp.1-6, 21-23 June 2010.

[8] Koay T. B., Tan T. Y., Eng Y. H., Rui G., Chitre M., Chew J. L., Chandhavarkar N., Khan R.R., Taher T. and Koh J., "STARFISH - A small team of autonomous robotics fish," in 3rd International Conference on Underwater System Technology: Theory and Applications 2010, Cyberjaya, Malaysia, Nov 2010.

[9] Newman P.M., “MOOS - Mission Orientated Operating Suite,” available at: http://www.robots.ox.ac.uk/ pnewman/TheMOOS/index.html. Accessed 16 July 2008.

[10] T. B. Koay, J. R. Potter, V. Pallayil, and E. T. Tan, “PANDA: A Rapidly Deployable, Self-Recovering Shallow Water Acquisition Platform,” MTS Sea Technology Magazine, 2002.

[11] T. B. Koay, J. Potter, T. Johansson, and P. Venugopalan, “PANDA; a self-recovering shallow water acoustic logger,” in OCEANS, 2001. MTS/IEEE Conference and Exhibition, vol. 3, pp. 1879-1885 vol.3, 2001.