Development of a system architecture for an advanced autonomous underwater vehicle, ORCA

Hyun-Taek Choi and Pan-Mook Lee

Ocean Development System Laboratory, Korea Research Institute of Ship & Ocean Engineering,
Korea Ocean Research & Development Institute, Daejeon 305-600, Korea
(Tel: +82-42-868-7563; Fax: +82-42-868-7503; Email: htchoi,pmlee@kriso.re.kr)

Abstract: Recently, great improvements have been made in developing autonomous underwater vehicles (AUVs) using state-of-the-art technologies for various kinds of sophisticated underwater missions. To meet increasing demands posed on AUVs, a powerful on-board computer system and an accurate sensor system with an well-organized control system architecture are needed. In this paper, a new control system architecture is proposed for AUV, ORCA (Oceanic Reinforced Cruising Agent) which is being currently developed by Korea Research Institute of Ships and Ocean Engineering (KRISO). The proposed architecture uses a hybrid architecture that combines a hierarchical architecture and a behavior based control architecture with an evaluator for coordinating between the architectures. This paper also proposed a sensor fusion structure based on the definition of 4 categories of sensors called grouping and 5-step data processing procedure. The development of the AUV, ORCA involving the system architecture, vehicle layout, and hardware configuration of on-board system are described.

Keywords: URV, AUV, System Architecture, Sensor Fusion System Structure

1. Introduction

Recently, great progresses have been made in AUV (Autonomous Underwater Vehicle) technology with strong performance requirements and various kinds of state-of-the-art technologies. These progresses have contributed to hydrographic survey that is the primary task of AUV [1,2]. These have also created new sophisticated tasks which need much more high technologies and their combining. Actually, a great variety of AUVs equipped with a highly accurate sensor system or new energy sources are announced. For example, Maridan A/S developed an integrated Doppler-inertial system named Marpos system. In sea tests, Marpos with M600 AUV showed a positioning accuracy of around 0.03 percent of total distance traveled, the equivalent of 1.7 meter per hour at a vehicle speed of 3 knots. C&C technologies announced Hugin AUV. It rated to 3000m, powered by a unique aluminum oxygen fuel cell for 40 hour endurance [3]. WHOI (Woods Hole Oceanographic Institution) developed REMUS (Remote Environmental Monitoring Units) in the late 1990's and transferred it to Hydroid founded Nov. 2001. Since then, Hydroid commercialized REMUS and has shipped more than 31 REMUS systems. It was deployed by the US Navy to support a range of naval operations, primarily mine countermeasure (MCM) in Operation Iraqi Freedom. [4]. GAVIA released Gavia AUV, which has a fully modular structure allowing user changeable modules for specific mission [5].

On the other hands, to let AUV work for underwater roles which are currently carried out by ROVs, the most important issue is control system architecture to increase the intelligence of a vehicle. A lot of efforts have been made for ground mobile robots [6–11]. In general, it is known that hierarchical architecture is the oldest method, but is good for global reasoning and planning. Then, behavior based architecture was proposed to achieve fast and robust response in complex and dynamic environments. Now, many researchers are working for hybrid architecture to combine advantages of two architectures. Additionally, multiple sensor fusion methodologies have been studied to combine data from various types of sensors to perform inferences that may not be possible from a specific sensor alone [12, 13]. Compared to ground mobile robots, AUV has a lot of difficulties in sensing and controlling mechanism. Some results of the recent study for ground mobile robot were not valid under underwater constraints. Nevertheless, lots of ongoing research efforts have been put in this issue. In [14], 11 AUV control system architectures were summarized and compared. More recently, behaviors based architecture with command fusion [15], intelligent task oriented control architecture [16], hybrid architecture for test-bed vehicle [17] were proposed.

Since 2001, Korea Research Institute of Ship and Ocean Engineering (KRISO) has been working for an advanced deep sea unmanned underwater vehicle project which consists of an ROV (Remotely Operated Vehicle), a launcher, and an AUV. In this project, AUV ORCA (Oceanic Reinforced Cruising Agent) is designed. This paper proposed a new system architecture including control system structure and sensor fusion system structure. The proposed architecture is a hybrid architecture to achieve global planning and reasoning capability as well as fast reactivity. A key idea of this hybrid architecture is an evaluator to make deliberative functions run at specific situations.

Section 2 summarizes main idea, advantages, and disadvantages of hierarchical architecture, behavior based architecture, and hybrid architecture. Then, a architecture with sensor fusion system structure is proposed in section 3. And, section 4 describes system configuration of the vehicle with on-board system, then finally conclusion is drawn.

This work was supported by the Ministry of Marine Affairs and Fisheries (MOMAF) of Korea for the development of a deep-sea unmanned underwater vehicle.

2. Control System Architecture

2.1. Hierarchical Architecture

Hierarchical architecture uses a top-down approach to organize the system with layers. The higher layers handle an overall mission, and provide sub-tasks for the lower layers, then these sub-tasks to achieve a mission is solved in the lower layers. Since direct communication between two adjacent layers is only available, it is a serial and tightly coupled structure. Therefore, communication and control flow occurs in predictable and predetermined manner, and this makes it easy to verify a performance such as controllability and stability [6, 14]. But, this architecture lacks flexibility in order to add some new functions, as a result, any modifications require significant work on the whole system. Since there is no direct communication path from higher layer to lower layer, response time from senor measurement to system action is long and sensor fusion is difficult [14]. Another feature of this architecture is the sequence of three primitives, Sense, Plan, and Act, where Plan is carried out based on a model. All sensor observations are fused into on a global data structure, which is generally referred to as a *world model* [8]. This architecture often requires strong assumptions about the world model such as consistency, reliability, and certainty. If the information used in Plan is inaccurate or has changed since obtained, the performance may be degraded seriously [6]. In a highly dynamic or complex environment, a response might be delayed [16], but in structured and highly predictable environments, hierarchical architecture is seemingly well suited [6].

2.2. Behavior-based Architecture

Behavior-based Architecture was proposed to overcome drawbacks associated with hierarchical architecture including the perceived lack of responsiveness in unstructured and uncertain environments. Sense and Act are tightly coupled into behaviors without representational symbolic knowledge and all activities emerge as the result of these behaviors operating either sequencially or concurrently. Each behavior (situation-action pair) becomes contextually meaningful unit through a decomposition method [6, 8]. Decomposition is based on the desired behaviors for the vehicle and missions are normally described as a sequence of phases with a set of active behaviors [16]. Then, a coordination mechanism is responsible for determining the relative strength of each behavior in a particular moment. There are two types of coordination method, competitive or cooperative method. In competitive method such as subsumption method, only one behavior is selected for an action at the moment, and in cooperative method like vector summation for behavioral fusion, different behaviors could contribute to make an action with weighted summation [10]. Behaviors are inherently modular and easy to test in isolation from the system. Also, behaviors support incremental expansion of the capabilities of a vehicle [8]. This architecture could reduce communication overhead since modules can access each other directly, but due to the lack of supervision, the communication among the modules can be very intensive and its controllability becomes a problem. As the number of behaviors is increasing, it is hard to synchronize a timing between behaviors. Therefore, the system performance and stability is very difficult to verify [14].

2.3. Hybrid Architecture

One issue of behavior based architecture is a lack of global planning and reasoning capability. A trend in architecture design has moved to hybrid architecture since 1990's [8]. It is a combination of the hierarchical Architecture and behavior based Architecture, where it can be divided into two levels, higher level and lower level. The higher level uses a hierarchical architecture to implement strategic, global mission planning functionality, and the lower level uses a behavior based architecture to achieve fast response in uncertain and complex environment. Higher level commands are translated and distributed to corresponding behaviors. In this scheme, while preserving advantages of hierarchical architecture, flexibility at the lower level can be achieved [14]. In this paper, new hybrid architecture is proposed for autonomous underwater vehicle. An evaluator actively controls global planning and reasoning modules by estimation and comparison. Next section describes a proposed architecture in details.

3. ORCA System Architecture

3.1. Motivation

As mentioned in Introduction, many studies present that hybrid architectures are the best general architectural solution, where a hot issue is how to combine advantages of these two architectures. Let us think about human behavior. When we go to somewhere, we plan a global path using a given goal, current environmental information, previous learning data, and then start to go. In front of unexpected situations, we usually react to avoid the situation and think about the reaction and the goal. Actually thinking is a kind of evaluating. We evaluate the current situation including our reaction to make decisions such as keep going, planning again, or give up. One can understand that a evaluation plays very important role between global planning and reacting. In this paper, new hybrid architecture is proposed for AUV, where an evaluator can trigger deliberative modules based on the performance evaluation of reactive modules. In the rest of this section, details of the proposed architecture are described.

3.2. ORCA Control System Architecture

As shown in Fig.5 the architecture of AUV ORCA has a hybrid form of combining a hierarchical architecture and a behavior based architecture. Basically, this architecture forms hierarchical layers with physical layer, driver layer, reactive layer, deliberative layer, and strategic layer. Among them, reactive layer includes a behavior based architecture. Right parts of each layer are in charge of input such as sensor data, feedback of internal decision, and user commands, while left parts of each layer are related to output flow. Role definitions of five layers are below.

⋄ *Strategic layer:* This layer has authority to make decisions whether a given mission can be completed or not, and to manage a learning mechanism. Strategic Coordinator (SC)

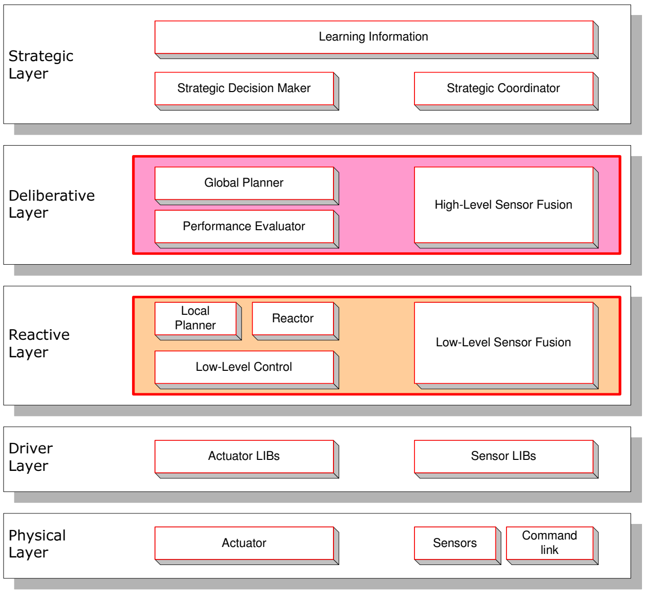

Fig. 1. New Hybrid System Architecture

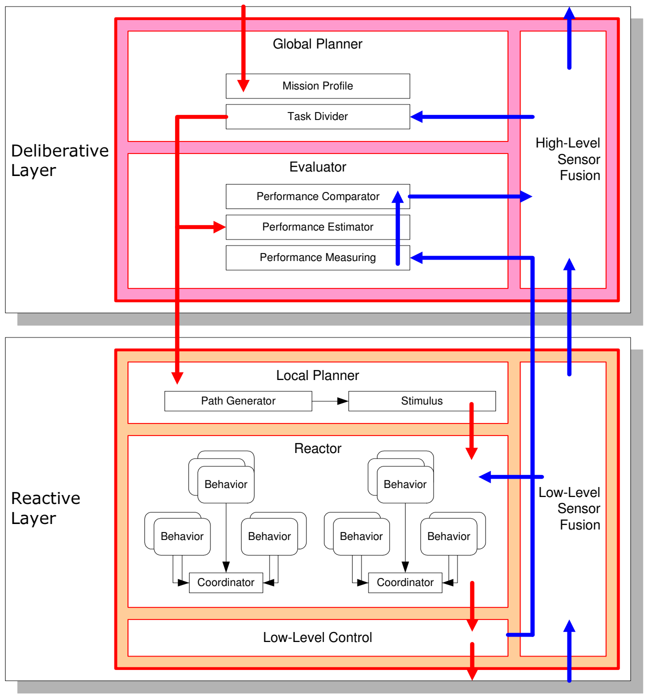

Fig. 2. Deliberative layer and Reactive Layer

analyses a mission script from an operator, then if the mission is a simple managing command, for example, transferring acquired data, testing a certain part of vehicle, moving to a specific location for recovery, the SC sends commands directly to corresponding modules. In case of the operational mission, it is sent to Strategic Decision Maker (SDM), which makes a decision based on a given mission, internal and external vehicle status, previous information from learning module and predefined operational guideline. After the end of mission, the SDM determines to update learning information. Learning Information (LI) in this layer is a kind of information storage, not mechanism itself. The LI saves parameters for intelligent controls and a sensor fusion system.

◇ *Deliberative layer:* In this layer, a mission is decomposed into sub-tasks. For this, a high-level sensor fusion system synthesizes all information coming up from the lower layer. A key function of this layer is a global planning and replanning triggered by an evaluator. Global Planner (GP) organizes sub-tasks based on mission scripts from an operator and an environmental model from a sensor fusion system. A mission has at least one sub-task, usually several sub-tasks carried out in sequence and/or concurrently. Performance Evaluator (PE) estimates the vehicle status at specific time and compares them with current status. The status includes vehicle position and orientation data, on-board system condition, battery usage, sensor and actuator failure and so on. If differences between planned situation and current situation are too big to recover, the PE generates triggering signal to plan again. And if there are repeated trigger signals more than predefined value, high-level sensor fusion system sends signal to the SDM on the strategic layer to make sure if the mission can be completed under current situation. High-level sensor fusion system is explained in next section.

◇ *Reactive layer:* The main job of this layer is reactive work for a given sub-tasks in real-time calculation. Behavior based architecture is governing fast and robust actions against uncertain and dynamic environments. Local Planner (LP) generates initial path for a task and activates corresponding behaviors. Reactor module is made up of behaviors and coordinators. In this module, activated behaviors react using information from low-level sensor fusion system, and reactive action of behaviors is merged by a competitive or cooperative method in a coordinator. Outcome of coordinators, low-level commands are sent to low-level control module, where physical commands are generated. Low-level sensor fusion system is explained in next section.

◇ *Driver layer:* This layer consists of the interface module for the sensors, AD/DA board, image processing board and communication devices. Each module is implemented in libraries (LIBs) for reconfiguration of a specific sensor and for resetting of the interface.

◇ *Physical layer:* This layer represents physical devices such as sensor, thruster, actuators, and communication link.

Fig.2 shows internal structure and data flow of deliberative layer and reactive layer.

3.3. Sensor fusion structure

In AUV, a lot of sensors are used for internal monitoring, navigation, and specific missions. First, to build multi-sensor fusion system, these sensors on the vehicle are categorized into four groups by meaning of measured data. This grouping concept is effective to reorganize sensor system when a sensor is added or is removed.

◇ Group 1: (*Identical Sensor Group*) Measuring same physical data with same sensors on different locations. These sensors are installed for fault tolerance. For example, two pressure sensors attached at different positions on the vehicle. But, counter example is two temperature sensors, with one on the on-board system and the other on the vehicle body, which case is in Group 4.

◇ Group 2: (*Redundant Sensor Group*) Measuring same physical data with different types of sensor. Example can be the case of magnetic compass and a heading gyro. This configuration could increase reliability and quality of fused data.

◇ Group 3: (*Associated Sensor Group*) Measuring same logical data after some data manipulations. Linear acceleration data of IMU could be associated with velocity data of DVL (doppler Velocity log) after post-processing.

◇ Group 4: (*Independent Sensor Group*) Measuring different data. For example, angle from IMU, distance from sonar, image from front camera, and image of rear camera are all different.

The sensor fusion structure consists of 5 steps as shown in Fig. 3. Preliminary Filtering and Data Transformation are in the reactive layer as a low-level structure and Quality Improvement, Data Identification, and Situation Mapping are in the deliberative layer as a high-level structure. Each step has a role as follows:

◇ *Preliminary filtering:* From a raw sensor signal, it determines sensor faults using physical limits or protocols given by the sensor manufacture. Then, filter out noise from each sensor signal if it is necessary. Raw signals include internal sensor data, navigation sensor data, operator commands, and feedback of the low-level control module.

◇ *Data Transformation:* After calibration of the signals, check the data reliability using Group 1 and Group 2 properties to produce meaningful data. Then, remove redundant data. In this step, possible information called collateral information could be added. These data use as a input to behaviors and are processed in the real-time.

◇ *Quality Improvement:* Improve quality of the data using fusion techniques, estimation algorithms, and interpolation algorithms. In this step, some of sensor physical limits such as different latency time are overcame.

◇ *Data Identification:* All fused data are combined to figure out the status of a vehicle, and to build a world model with pre-defined data structure including diagnostic information.

◇ *Situation Mapping:* This step is highly dependent on a mission. Results of above step could reorganize for task-specific format and possibly combined with given geometric

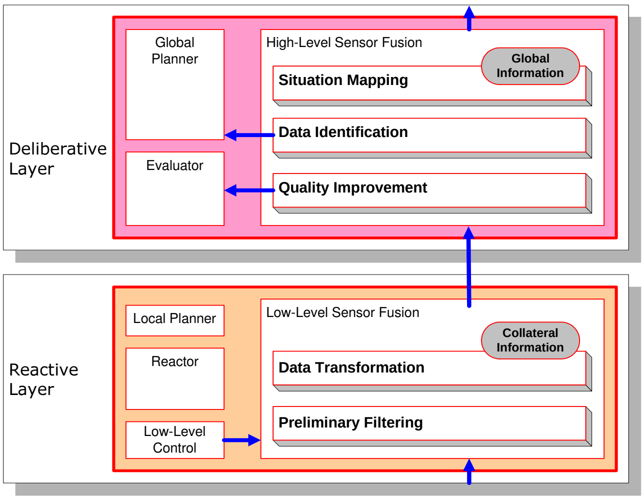

Fig. 3. Sensor Fusion System Structure

Fig. 4. Oceanic Reinforced Cruising Agent, ORCA

environmental information. This polished information is either used for the mission in the vehicle system or sent to the ground station.

4. ORCA System Configuration

4.1. Mechanical design

ORCA is made of all aluminum, pressure housing to equip an on-board system and sensors. It is designed by minimizing drag and maximizing volume as shown in Fig.4. The vehicle is 1.7 m long with the body diameter of 0.25m, and weight 70 kg in air. It has one main thruster and 4 fin actuators. ORCA PT (prototype) was designed to operate up to 100m depth, and finally ORCA will be designed for 6000m depth. It has pressure sensor (Copal Electronics, PA-500-102G-10), altitude and heading reference system (Microinfinity, Marion MI-A3370X), doppler velocity log (RD Instrument, Workhorse Navigator 1200kHz), two underwater cameras (RF Concepts, VB21-CSHR-R43), range sonar (made by KRISO) as navigation sensors, and 7 temperature sensors (National Semiconductor, LM35), 2 leakage sensors (SY high-tech, SY-HS-2), 2 battery monitoring sensors (made by KRISO) as internal monitoring sensors. For a propulsion system, BLDC motor (maxon EC32) with amplifier (maxon, DEC 50/5) are mounted. Later, a mission sensor package including a side-scan sonar will be considered.

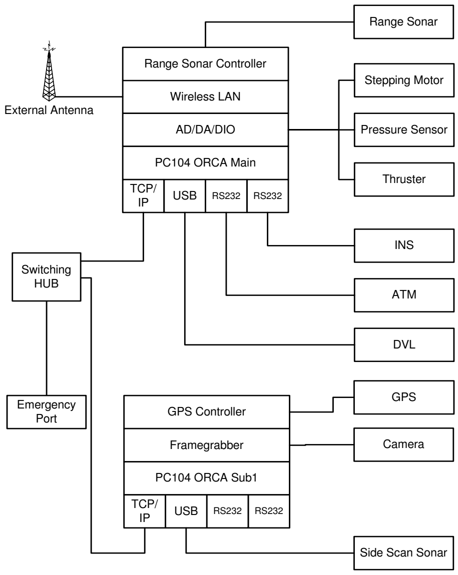

Fig. 5. On-board System Layout

4.2. On-board System

To meet demands posed on AUVs, basically, a powerful on-board computer system in terms of strong computing power, reliability, flexible interface to various sensor systems and communication systems, an easy developing and debugging environment, and expandability becomes a core component. There are additional requirements such as low-power consumption, low-heat generation, small size, light weight, and low cost, which should be compromised with functional requirements. ORCA has PC104+ based system. Two PC104+ CPU Modules and several peripheral boards form basic assembly. Later it can be expanded with another PC104+ using a high speed LAN depending on specific missions. Table 1 shows the details of on-board assembly.

As shown in Fig.5, Two PC104+ modules in ORCA on-board assembly are connected by 100 Mbps LAN with a 5 Ports switching hub. The ORCA-Main is in charge of supervising overall system and controlling the vehicle with running the proposed system architecture. For this, the ORCA-Main receives navigation sensor data and internal sensor data, and handles communication devices. The ORCA-Sub1 manages additional sensor such as image processing board with CCD cameras and GPS (Global Position System). After raw data processing, data are sent to the ORCA-Main or the ground station.

Operating system plays an important role in an embedded system as a backbone. Since AUV system typically needs various communication interfaces, selecting appropriate OS could lessen a burden caused by interfacing software, and provides flexibility of a system configuration. ORCA uses Microsoft Windows XP Embedded with Real Time eXtension (RTX) supplied by VentureCom [18]. Windows XP Embedded allows relatively small OS image through customizing components as well as familiar developing and debugging environments with a graphical user interface. Optimized OS image including RTX shows effective performance as a soft real-time embedded OS [17].

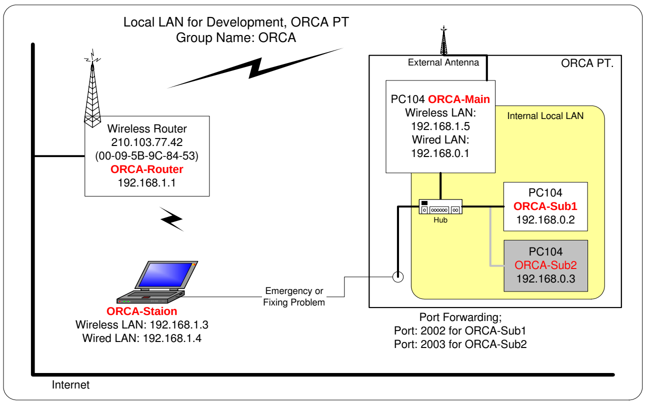

Fig. 6. Overall System Configuration

4.3. Operating and Developing Environments

Overall system consists of the vehicle and the ground station, where these two systems communicate with wireless LAN when the vehicle on the surface and with an acoustic modem in underwater operation. Fig.6 shows that ORCA system has a nested local LAN. An wireless router unites the vehicle and the ground station using DHCP (Dynamic Host Configuration Protocol). In ORCA, the ORCA-Main acts like DHCP server for the ORCA-Sub1 and additional PC104+ module. To send commands to the ORCA-Sub1 from the ground station directly, the ORCA-Main has a port forwarding configuration. This environment including network configuration uses both in development and in operation without any changes. ORCA has a direct LAN port for fixing a problem.

4.4. Implementation Issues

So far, main focuses of recent works were on concepts and their verifications. However, wet environment and a vehicle housing give us very hard limitations to develop and debug algorithms. Goal of this implementation is to create easy environments for operation, development, and debugging under the proposed architecture. For this, we divided this implementation into a system part and an algorithm part. Both parts have classes and sub-classes defined using object-oriented programming concepts, and each module is programmed as a DLL (Dynamic Linking Library) or LIB. In this structure, we designed that algorithm modules can make virtual connection between the vehicle and the ground station to adjust parameters as well as the modules can be downloaded from the ground station on site.

5. Conclusion

In this paper, a new system architecture was proposed for AUV, ORCA. In combining of global planning capability of hierarchical architecture and reactivity of behavior based architecture, we proposed an evaluation mechanism between initial plan and current reaction, which can trigger the global planner considering the differences. As a part of the system architecture, the sensor fusion system structure with the grouping concept was described. Additionally, details of ORCA including mechanical design, on-board system, network configuration were presented.

Table 1. Specifications of On-board System

Module	Manufacturer	Model	Features
PC104 CPU Module	Lippert	CoolRoadRunner III	600MHz Pentium III CPU, 512 MB RAM, 2 Serial Port, 2 USB, CompactFlash Socket, 100 Mbps LAN
Storage	Fujitsu	MHT2040AT	40 GB for development
	Sandisk	CompactFlash	1 GB for operation
IO	Diamond Systems	Diamond-MM-32-AT	16 bits AD, 32 Chs., 12 bits DA, 4 Chs., DIO 24 Chs., Temperature Auto-Calibration
Wireless LAN	Kontron	PC104 PCMCIA	Type I and II
	Buffalo	WLI-PCM-L11GP	Port for External Antenna
Main Power	Diamond Systems	Jupiter-MM-SIO	50 W, ±12V, ±5V, 2 Serial Port
Sub Power	Tri-M Engineering	V104	25 W, 12V, 5V
Framegrabber	Arvoo	Picasoo 104-Duo2	2 Video Inputs

Now, we are implementing the architecture on the on-board system. Since main objective of ORCA is surveying around a specific area and transmitting the data to the ground station using an underwater docking system, accurate navigation and docking algorithm are going to be developed and verified in the vehicle after complete of the development,

References

[1] J. Yuh, Ed., *Underwater Robotic Vehicles: Design and Control*, New Mexico: TSI Press., 1995.

[2] Gwyn Griffiths, Ed., *Technology and Applications of Autonomous Underwater Vehicles*, New York, Taylor & Trance Inc., 2003

[3] John Romeo, Graham Lester, "Navigation is key to AUV Missions," Sea Technology, January 2002

[4] Kevin McCarthy, "REMUS - a role technology transfer," International Ocean Systems, November/December 2003 pp 22-23

[5] http://www.gavia.is/products/index.html

[6] Ronald C. Arkin, *Behavior-Based Robotics*, The MIT Press, 1998.

[7] Rodney A. Brooks, *Cambrian Intelligence: The Early History of the New AI*, The MIT Press 1999

[8] Robin R. Murphy, *Introduction to AI Robotics*, The MIT Press, 2000

[9] Marco Dorigo, Marco Colombetti, *Robot Shaping An Experiment in Behavior Engineering*, The MIT Press 1998

[10] Stefano Nolfi and Dario Floreano, *Evolutionary Robotics: The Biology, Intelligence, and Technology of Self-Organizing Machines*, The MIT Press, 2000

[11] Roland Siegwart, Illah R. Nourbakhsh, *Introduction to Autonomous Mobile Robots*, The MIT Press 2004

[12] David L. Hall, *Mathematical Techniques in Multisensor Data Fusion*, Artech House Inc., 1992

[13] R. C. Luo, K. L. Su, "A Review of High-level Multisensor Fusion: Approaches and applications," Proceedings of the IEEE International Conference on Multisensor Fusion nad Integration for Intelligent Systems, Taipei, Taiwan, 1999

[14] Kimon P. Valavanis, Denis Gracanin, Maja Matijasevic, Ramesh Kolluru, and Georgios A. Demetriou, "Control Architectures for Autonomous Underwater Vehicles," IEEE Control Systems Magazine, Vol. 17, No. 6, pp 48-64, December 1997

[15] Julio K. Rosenblatt, Stefan Williams, and Hugh Durrant-Whyte, "A Behavior-Based Architecture for Autonomous Underwater Exploration," International Journal of Information Sciences, Vol. 145, No. 1-2, pp 69-87, September 2002

[16] P. Radio, J. Yuh, J. Batlle, and K. Sugihara, "On AUV Control Architecture," Proceedings of the 2000 IEEE/RSJ International Conference on Intelligent Robots and Systems, Takamatsu, Japan, Oct. 2000

[17] Hyun-Taek Choi, Aaron Hanai, Song Choi, and Junku Yuh, "Development of an Underwater Robot, ODIN-III," 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems, Las Vegas, USA, Oct. 2003

[18] RTX 5.0 Documentation, VenturCom, www.vci.com