2017 17th International Conference on Control, Automation and Systems (ICCAS 2017)
Oct. 18–21, 2017, Ramada Plaza, Jeju, Korea

Mechatronic Architecture Development of UX-1

Soheil Zavari[1], Olli Usenius, Tuomas Salomaa, Jose Villa Escusol,
Arttu Heininen, Jouko Laitinen, Jussi Aaltonen, Kari T.Koskinen

[1]Department of Mechanical Engineering and Industrial Systems
Tampere University of Technology
Korkeakoulunkatu 10, 33720 Tampere, Finland
soheil.zavari@tut.fi * Corresponding author

Abstract: This paper presents novel design of underwater robot for exploring abandoned mines. The hazardous environment of such mines due to unknown hydrodynamic forces and faulty navigations, brings the need of developing a reliable system able to be controlled autonomously. This capability highly rely on the basis of low level control and mechatronic architecture of the robot which demonstrate robot potential for performing real-time operations. Following, describes rapid prototyping during development phase of the robot. Further, it investigates on mechatronic development of main controller unit, propulsion system and ballast.

Keywords: Underwater robot, Mechatronic, Control, AUV design.

1. INTRODUCTION

Due to complication in accessibility, underwater world continue to exist unexplored by humankind, however during recent decades underwater robots extended human knowledge in oceanography and geology of the sea floor rapidly. Remotely Operated Vehicles (ROV) are developed to observe deep sea levels. However, ROV replaced by AUV (Autonomous Underwater Vehicle)due to their limitation such as high cost of operation, non real time basis control and limited range of navigation due to tethered connection. To mention a few, Odyssey is an AUV that is designed in 1994 to operate in 6000 meters [1], also SAUVIM [2] is a semi autonomous underwater vehicle in 1998 which operates in depth of 6000 meters.

AUV carries navigation sensors, control units, supply power, communication units, etc. However carrying all components on board forms disproportionate structure shape which directly effect on irregular hydrodynamic forces on robot. Despite that, there are few configuration profiles which are robust and friendly in appearance. As such profiles is spherical configuration, the maneuverability and symmetrical structure of a sphere provides significant advantage for an autonomous robot.

ODIN is one of the first prototype which took the advantage of spherical design in 1991 ([3]) as a closed frame omnidirectional autonomous underwater navigator with 8 thrusters. The recent software and mechatronic development based on Windows operating system is explained in updated version of the robot known as ODIN 3 can be found in [4]. For on land applications, spherical ROSPHERE ([5]) is developed to monitor and measure the conditions of crops with two degree of freedom pendulum system.

UNEXMIN "**Un**derwater **Ex**plorer for Flooded **Min**es" project look into development of an AUV to operate in 500 meters depth of unmapped surrounding of flooded mines. The robot ([6]) benefits from spherical profile with 30 cm in radius and 8 thrusters in order to inspect narrow mine's channels. Due to confined unpredictable robot work space, the robot must be highly maneuverable. To enhance the latter, the robot is capable of heave motion not only by 4 vertical thrusters but also by a compact ballast system. Moreover, an independent attitude control through pendulum mechanism is developed to ease the robot motion.

The mechatronic architecturing of such small size AUV with multiple sensors and thrusters is quite advanced and it determines the capabilities of the robot in terms of operating time and maneuverability in unpredictable sub sea environment. In this context, in order to enhance the intelligence and reliability of the vehicle the mechatronic architecture is usually consists of multiple layers. A sophisticated low level mechatronic architecture can contribute to enhance higher controller models later in development phase. Thereby, as it will be explained in section 2,4the low level unit plays significant role.

This paper is organized in three sections . The first section is associated to the implementation of low level mechatronic architecture and rapid prototyping. Next it describes communication interfaces and it follows briefly modeling the thrusters and motion of the robot.

2. MECHATRONIC ARCHITECTURE

One of the challenging phase during development phase is choosing right mechatronic components which are compatible with system in terms of communication protocol, baudrate, interface protocols. Therefore, testing and development process might continue simultaneously and it usually encounters with a number of trial and error. Nevertheless, as it will be explained in next section, recent off-the-shelf products such as Matlab and dSpace provide an easy approach for rapid protoryping in a real time environment. To increase the pace of development

Fig. 1 3D view of design

phase, UX1 (**U**nderwater e**X**plorer) utilized Speedgoat embedded PC from Matlab which offer hard real-time capabilities and it will be explained by detail in section 2.2 .

2.1 Rapid Prototyping

This phase of development might be tedious to examine whether the mechatronic characteristic of the component are in accordance with general specification requirement of the robot. In another word, real time capability of the robot would define the minimum characteristic of every mechatronic module in the system. In this context Speedgoat as target computer provides a real time test bench in order to examine main components of the system such as thrusters, speed controller, etc in short amount of time without the need of driver programming and testing the component (plant) in real time environment. In this case we were able to test the actual hardware and our controller model independent of the interface protocol of the hardware module. Practically, matlab real time library developed ready interface block which allow us to integrated any interface virtually into the model.

As it is demonstrated in Fig. 2, when the model is created in development computer, a standalone real time application can be download over target computer in order to control and operate independently with hardware modules. Once the hardware is connected to the target computer, the generated code from the model is executed. The parameters and signals of each real time task can be tuned via real-time explorer while the task is running and interacting with hardware elements. This process significantly reduce the time during development phase. Fig. 3 is illustrated the CAN bus, where 8 speed controllers and IMU sensor is connected to the target computer. At this point it is feasible to deterministically control the data acquisition sampling rate real-time from the model, moreover the test could verify the functionality of the speed controllers and sensors with the model.

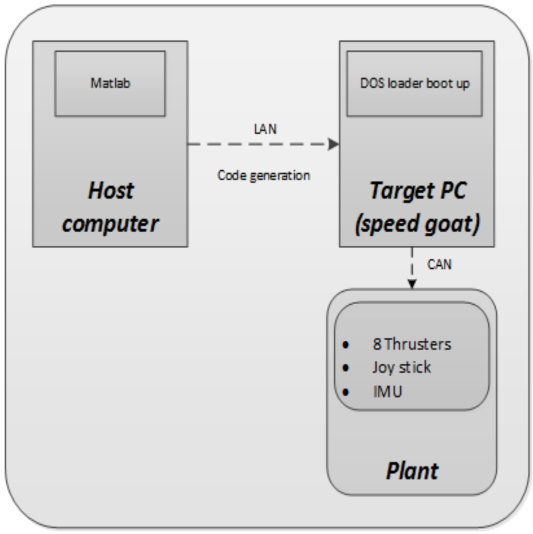

Fig. 2 Rapid Prototyping

2.2 Hardware Components

The main mechatronic architecture of the UX1 in low level control is consist of Speedgoat computer , 8 VESC speed controller, Inertia Measurement Unit (IMU) and joystick. There are 4 laser scanners, 3 HD cameras and Doppler Velocity Log (DVL), sonar and multibeam spectral unit as navigation sensors during development of high level control system, which is out of the scope of this paper. The main controller unit Speedgoat is core i7 2.5 Ghz and SSD 256GB, with 4 serial interfaces RS232 which supports up to 115Kbps (In our case, serial driver (RS232) simulink block is configured at 115, 16 bit and it will be executed every 10 ms) and 4 Ethernet ports used for the real-time UDP connection and USB 3 and USB 2 ports as well.

2.3 Communication of main CPU and low level control

There is always risk of losing the robot during operation in hazardous underwater environments due to either failing in mechanical, naviagation or unknown hydrodynamic forces. Yet, the first and most common means of malfunctions are due to unpredictable breakdown in mechanical system. With that being said, the second layer of low level control is design to enhance the reliability of mechanical modules of the robot during operations. As it is indicated in the Fig. 3, this layer interacts with 2 microcontrollers LQFP64 NXP which are responsible for communication between main CPU and first layer of low level control system. The microcontrollers receive the force vector from main CPU and assign each force value to proper VESC ID through the CAN bus. Simultaneously the microcontroller monitor the power consumed by each thruster in order to detect any possible significant perturbation in current as a result of mechanical malfunction. In the case of failure in any of the thrusters, the microcontroller cease the supplied power to non-functional thruster and transfer the load to the other three horizontal or vertical thrusters.

2.4 Communication Interfaces

Due to critical operating environment of UX1 robot, reliability of the communication protocol in low level plays a significant role to minimize the errors in the system. On the other hand, the process of data communication must assure to maximize the real-time capability for an autonomous control of the robot. The latter indicates the soft and hard real-time capabilities of every sub module in the system. Among all industrial communication protocols (Ethernet, I2C, CAN), CAN protocol is the most reliable and affordable protocol among high-tech sensors or actuators in the market. As it is demonstrated in Fig. 3, CAN bus line includes 8 speed controllers and IMU sensor. IMU is located at the end of the line before the 120 ohm resistor, the other end of the CAN bus is connected to CAN-serial converter. The CAN bus is further extended from converter to a joystick, which directly set the command to thrusters in the pool test.

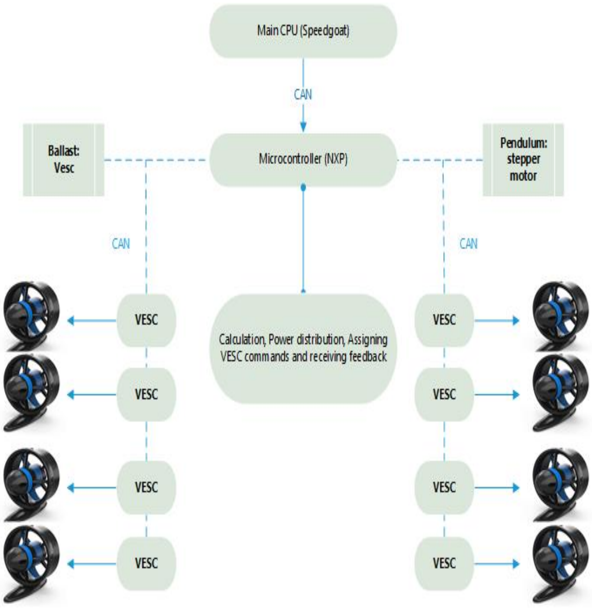

Fig. 3 Communication layer

All thrusters are controlled by VESC, which is open source speed controller with STM32F5 microcontroller and it is equipped with CAN and USB as communication interface. Hence prior developing control model, each speed controller configured via BLDC tool at 500 Kbaud/s and assigned with an ID between 1 to 8. VESC controller can operate the thruster either by duty cycle or electrical RPM. Consequently, every speed controller unit receive 4 bytes command as electrical RPM (refer to 3).in CAN data frame. Consequently, speed controller publish 4 bytes of electrical RPM (EPRM), 2 bytes of current and 2 bytes of duty cycle as feedback in a CAN frame data structure. Sending commands and receiving feedback data occur in every 1 millisecond.

In order to communicate with hardware modules, the driver interface must be added to the core model, the Speedgoat real-time library includes most common driver interfaces for variety of communication protocols. The structure of data follow ASCII format to communicate with other devices. Hence all control commands for sending to VESC drivers are encoded to ASCII format simply via serial signal write block. Note that the command data must be compatible to CAN Id and CAN data structure. For this purpose, data converted to string of ASCII format via Uint16 and Uint32 in Matlab simulink. Similarly, all data which is received by the computer must be decoded to CAN data frame.

Note that the size of data received by computer at each sample time is 104 bytes $(8*8+2*8+3*8byte)$ which indicates the software and hardware FIFO size should be in maximum size in order to minimize the interrupt time and consequently allocate more time to process the input data in the model. As a result, the serial port hardware FIFO size is configured to 64 bytes (16C750 UART) before the interrupt occurs, which result in only 0,2 milisecond to process the data. On the other hand, the software FIFO size also set as 1024 bytes which is about ten times larger than each set of input data.

3. MODELING THRUSTERS

The directional control system of UX1 is consist of 3 individual mechanisms. The propulsion unit which mainly is responsible for surge, heave and stirring motion of the robot. The pendulum mechanism enables the robot to pitch in certain angle and maintain the orientation accordingly. Moreover, the robot operate on long distance heave motion via ballast system. It is demonstrated in Fig. 4, the blue marked cylindrical container in the center of sphere is oil tank and the semicircle rack around the oil tank is holder for 3 batteries.

In the early phase of design, a repetitious process took place in order to come up with optimal solution in terms of number, pose of thrusters, while the ideal solution must minimize the occupied space, and yet provide the required Degree Of Freedom (DOF). The robot has 8 thrusters and the thruster pose can determine the DOF. In our case, there are 4 thrusters located horizontally face to face symmetrically in each hemisphere. For forward and backward motion only two thrusters are operational which push the water out, in this case each thruster use 25 to 75 percent of their maximum power.

As it is illustrated in Fig. 1, the vertical thrusters are located on vertical plane, which cross the center of sphere. Note that spherical design and the symmetrical pose of thrusters significantly contributes to simplifying

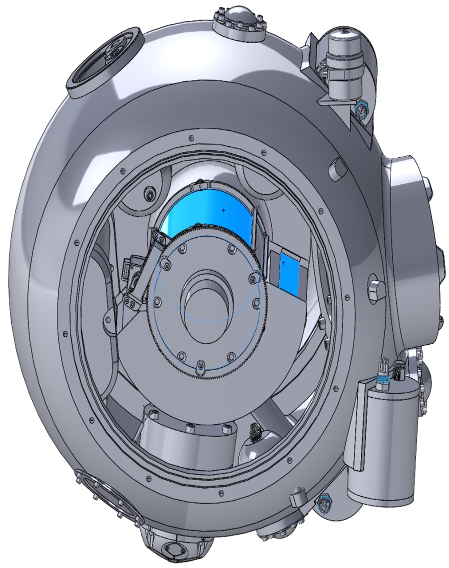

Fig. 4 Cross section view : pendulum system and oil tank

the dynamic equation of motion.

Thrust force (F_d)is modeled according to bilinear as following equation (refer to [7]), which can be written in terms of advance speed of water in propeller (V_a) and rotational velocity of the propeller (n):

$$F_d = \rho \cdot D^4 \cdot \alpha_1 \cdot n^2 + \rho \cdot D^3 \cdot \alpha_2 \cdot n \cdot V_a \quad (1)$$

Where α_1, α_1 represents constant values and D is the diameter of the propeller equal to 76 mm. According to specification requirement the maximum veloctiy of the robot in mine would be about 0.5 m/s where measured data in such mines didn't identify any trace of water flow. Hence, the assumption of stationary water reduced the equation 1 only to its first term, which can be simplified as follow:

$$F_d = \alpha \cdot n^2 \quad (2)$$

The α can be obtained from thrust force versus rotational velocity graph provided by the manufacturer which lead to $\alpha = 0.035$.

Note that the 8 thrusters are considered identical and modeled correspondingly. The input control signal is assign as thrust froce (newton) which will be converted to its equivalent electrical rotational speed as input to VESC speed controllers.

3.1 Heave and Pitch

According to requirement specification UX1 must be able to traverse 500 meters vertically due to the depth of mines. In this case, continuous operation of 4 vertical thrusters for vertical motion requires significant amount of power supply and consequently additional number of batteries, while the available space for batteries is quite limited. Therefore, a ballast system is design which consists of 2 cylindrical barrels with pistons. A small size pump (takako TFH-040, max 140bar, 65 l×30 H ×30 W mm) transfer the oil from the oil tank in the center of robot to the barrels. VESC speed controller receive commands from the main CPU in order to set the amount of oil need to be transferred to the barrels and be replaced by water. In total 2.8 liters is the maximum adjustable volume through the piston cylinder mechanism. The properties of the ballast system is demonstrated in the table 1.

Table 1 Ballast specification

Specification	
Cylinder size	318 l ×80 d mm
Variable volume	2.8 L
Pump	takako TFH-040
oil tank size	4 L

3.2 Pendulum

The compact size of the UX1 and concentration of multiple navigation sensors on outer layer in certain positions around the sphere, lay the foundation of the necessity to control pitch angle actively. particularly DVL angle view must always be projected to the wall in order to estimate accurate velocity value. As a result, a pendulum mechanism, which complies with requirements such as least possible volume in size and providing $\pm 90°$ degree pitch angle for the robot, would be the ideal solution .The properties of the pendulum is demonstrated in the table 2, the pendulum carries the weight of three batteries (each 2.4 kg) under the robot center of gravity and the batteries are distributed aligned in circular line. Since the acceleration of pitch motion is not the priority, a small size (35 mm) stepper motor with 24 DCV and 0.186 N.m torque is the suitable actuator for the pendulum system. The properties of this system is demonstrated in table 2.

Table 2 Pendulum Properties

Specification	
Radius	76 mm
Weight	7.2 kg
Maximum Angel	$\pm 90°$
Operating time	10 s

4. CONCLUSION

This work tried to investigates on mechatronic layout of the robot. All elements of the system developed cautiously to minimize the delay and enhance the deterministic behavior of the robot as real-time system , in this context physical communication layer and software interfaces ease the process of rapid prototyping. The fast process of developing control models for actual hardware through target PC is addressed.

Second layer of low level control system will be developed further to contribute into a sophisticated error handling unit, which communicates with most of instrumentation, navigation sensors and motors in the UX1.

The functionality of the propulsion system is examined during pool test while the robot is control by a workstation and joystick outside of the pool. Surge, heave and heading motion are tested and the acceleration is recorded by IMU along short distances in the pool.

In next phase, the Hardware In Loop (HIL) will be applied in harsher environment, where smooth flow of water can be induced to the system, to tune the control model parameters.

ACKNOWLEDGMENT

This project has received funding from the European Unions Horizon 2020 research and innovation programme under grant agreement No 690008.

REFERENCES

[1] Bellingham, J. G., et al. *A second generation survey AUV.* Autonomous Underwater Vehicle Technology, 1994. AUV'94., Proceedings of the 1994 Symposium on. IEEE, 1994.

[2] Yuh, J., et al. *Design of a semi-autonomous underwater vehicle for intervention missions (SAUVIM).* Underwater Technology, 1998. Proceedings of the 1998 International Symposium on. IEEE, 1998.

[3] Choi, S. K., J. Yuh, and N. Keevil. *Design of omni-directional underwater robotic vehicle.* OCEANS'93. Engineering in Harmony with Ocean. Proceedings. IEEE, 1993.

[4] Yuh, Junku, Jing Nie, and CS George Lee. *Experimental study on adaptive control of underwater robots.* Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on. Vol. 1. IEEE, 1999.

[5] Hernndez, Juan D., et al. *Moisture measurement in crops using spherical robots.* Industrial Robot: An International Journal 40.1 (2013): 59-66.

[6] Zavari, Soheil, et al. *Early stage design of a spherical underwater robotic vehicle.* System Theory, Control and Computing (ICSTCC), 2016 20th International Conference on. IEEE, 2016.

[7] Fossen, Thor I. *Guidance and control of ocean vehicles.* John Wiley Sons Inc, 1994.