

The Autonomous Acoustic Buoy (June 2013)

R. Reitsma, F. Pellicer, J. Agüera and A. Marinas

I. INTRODUCTION

Together with the *Universitat Politècnica de Catalunya (UPC)*, the *Laboratori D'Aplicacions Bioacústiques (LAB)* has provided another Autonomous Acoustic Buoy (AAB) project within the *European Project Semester 2013 (EPS)*. The project is being supervised by the *Laboratori D'Aplicacions Bioacústiques (Figure 1)* is a group of multi-disciplinary scientists that investigates sound productions of marine life. Their main goal is to decrease the deterioration of the seas and its inhabitants. It was created as a response to the acoustic degradation of marine habitat in the Mediterranean Sea and the North Atlantic with support from the Spanish government and the city of Vilanova i la Geltrú [1].



Figure 1 - Laboratori D'Aplicacions Bioacústiques logo

II. AUTONOMOUS ACOUSTIC BUOY SYSTEM

The purpose of an Autonomous Acoustic Buoy (AAB) system is that it can monitor acoustics underwater. This gives it that it listens to different sounds that are present in the underwater environment. Whether it is produced by wildlife, mankind or man-made machines, the AAB can recognize and classify it. The system is especially useful in areas on sea where there is a lot of human

activity with noise producing machinery, such as oil exploration or construction. In those areas it is important that the presence of wildlife is recognized before starting operations, also it can be used to measure if the noise itself remains below legislative limits.

An AAB system consists of several components. It uses a so called main buoy, equipped with a hydrophone and a watchdog system. The hydrophone is a microphone capable of recording sound underwater. The watchdog system provides the system with the capability of recognizing the different sound types. Another component is a receiver that is capable of showing the collected acoustic data to an end-user, whether it is on a ship or on-shore. To cover all possible distances in between the main buoy and the receiver, repeater buoys can be used. These buoys have the sole purpose of repeating the system over and over again. Multiple repeater buoys can be added to cover longer distances.

In previous years of the project, the buoy was designed, software and hardware have been developed for data acquisition, and the watchdog system has been built to ensure a safe and autonomous unattended operation of the electronics. This year's project focuses on developing the communication between the buoy(s) and the end user, including the development of a user-interface.

Because it is difficult to send acquired data directly from a buoy that is far away on sea to an end-user. A mesh network of buoys is needed to amplify and repeat the signal over and over until it reaches its

destination. To monitor a large area, multiple buoys can be deployed. Therefore the communication system must have the capability of acting both as a transmitter as well as a repeater in order to reach the end-user.

III. A COMMUNICATION SYSTEM

The last remaining part for the Autonomous Acoustic Buoy system is the communication system, which has been developed this year. To achieve the common goal of developing a communication system to transfer data from a collecting buoy to the end user, several steps needed to be taken:

- Development and programming of electronic components.
- Design and programming of a user-interface.
- Construction of the housing and integration of components.

The project started with the selection of different electronic components that were suitable for the type of communication that was needed. This led to complicated programming and the development of an extra component. Meanwhile the interface as well as the economic feasibility and the physical construction of components started. The team's completed product in the end has been named *Corangino*.

IV. ELECTRONICS

At the start of the project the team was supplied with several *XBee Pro 868 RF* (Figure 2) communication modules by the LAB. XBee is the name of a communication protocol.

The team was given the choice to choose other or to keep these modules. In the end the XBee has been chosen to be the

communication protocol that should be used. The reason these modules are chosen is that they have a high transmission power and sensitivity.

Therefore making them ideal for use with long distances. In addition, its small size of 2.443 by 3.322 cm and low power consumption make it very suitable for use in an isolated system such as an autonomous buoy. [2]

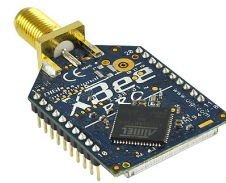


Figure 2 – XBee Pro 868 RF module

At this point the brains of the communication system needed to be chosen in form of a microcontroller, there were two options for this. Either buying a complete electronic board with a microcontroller inside or buying only a microcontroller and build complete board around it. The decision was made to buy a complete electronic board, the type that has been chosen is the *Arduino Mega 2560 module* (Figure 3).

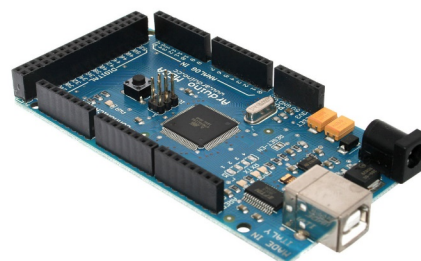


Figure 3 – Arduino Mega 2560

One of the main reasons for choosing this board is that the microcontroller had to have at least three connections for use in this project to connect the components such as the GPS, the watchdog system and hydrophone. Next to that, constructing an own board suitable for a microcontroller

would have been very time consuming so choosing an electronic board that already had four ports was the quick and easy way. Next to that the *Arduino Mega 2560* is compatible with the *XBee* modules and is based on an open-source platform making a lot of different uses possible through programming.[3] The *Arduino Mega 2560* controls the communications, processes the information that it receives, controls the GPS and its data and display the data on the LCD display. As a connection adapter for the *XBee Pro 686 RF* on the *Arduino Mega 2560*, a so called *XBee Shield* is used. This shield is the only solution on the market to make this connection, therefore making it ideal [4].

A *Liquid Crystal Display (LCD)* has been selected to show necessary information about the status, mode and transmitting data of the buoy. A *Fordata FDCC1602N LCD* has been chosen (Figure 4).

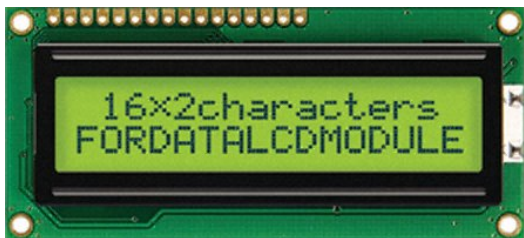


Figure 4 – Fordata Liquid Crystal Display

This LCD has sixteen columns and two rows (16x2) to have enough space for showing text. It has a *Light-Emitting Diode (LED)* backlight for clear visibility in low-light or dark environments. The reason for choosing this particular LCD is that it has a *SPLC780D* controller [5]. This controller is compatible with the drivers in the *Liquid Cristal Library* of the *Arduino Mega 2560*.

After all these components were chosen it left the team with a big gap, the question was how to connect everything together with different components needing different voltages and connections. The

answer is a self-developed board called named *Boarduino*. Its main function is to adapt signals and convert voltages to appropriate levels. It has one input for a 12V voltage and three other voltage outputs. The *Arduino Mega 2560* needs 5V so an *R-78B5.0-1.0* switching converter is used to convert 12V to 5V [6]. Both the GPS and the LCD display need 6.5V so another switching converter (*R-78AA5.0-1 SMD*) is used that allows to control turn the output on and off via digital signals making it very useful to save energy increasing the AAB's autonomy [7]. The decision to choose the switching converters is because this conversion method is more power efficient (often 75% to 98%) than linear voltage regulation and the reduced cost.

The *Boarduino* (Figure 5) has been designed with a software design suite called *Proteus* [8].

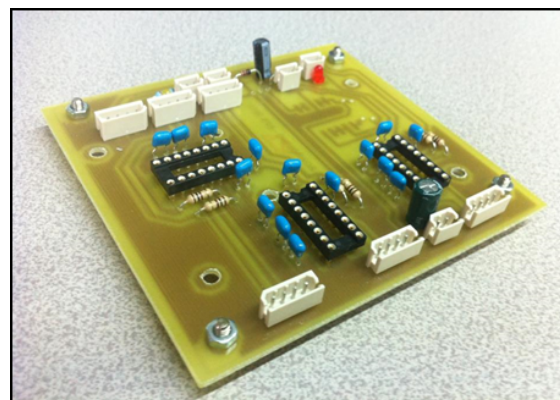


Figure 5 – Boarduino

V. PROGRAMMING OF ELECTRONICS

The electronic components on themselves are not able to do anything. This is where programming comes in. For our system, smart programming techniques are used, resulting in a device that can automatically detect which operation mode it should be in. These three modes are to operate as the transmitter in the main buoy, acting as

repeater buoy to amplify the system or a receiver to collect the data.

[1] When the device is connected to the existing embedded board in the main buoy and the Global Positioning System (GPS). It will turn into the 'Main' operation mode. Meaning it collects and transmits the GPS position and embedded board data (*alarms*). The embedded board is the board that controls all the systems like the hydrophone and watchdog system.

[2] When only a GPS is connected the device switches to 'Repeater' mode, meaning it will listen to incoming signals, when it receives a signal it transmits it again. Either to another repeater or to the Receiver.

[3] When device is connected to a personal computer it automatically switched to 'Receiver' mode. Its then collects data and transfers it to the connected personal computer.

VI. USER-INTERFACE

A graphical user-interface was needed to ensure easy access for the end-user. The user-interface is designed with the end-user in mind, to make it as accessible as possible the decision was made to develop a webpage that can be accessed from a personal computer or smartphone. The user can see alarm data, time and GPS location anywhere around the world with *Google Maps* [11] in real time.

The Software chosen to create the graphical interface is software called *Adobe Dreamweaver CS6* [12], which is web design software.

The user-interface consists of the following four pages (Figure 6):

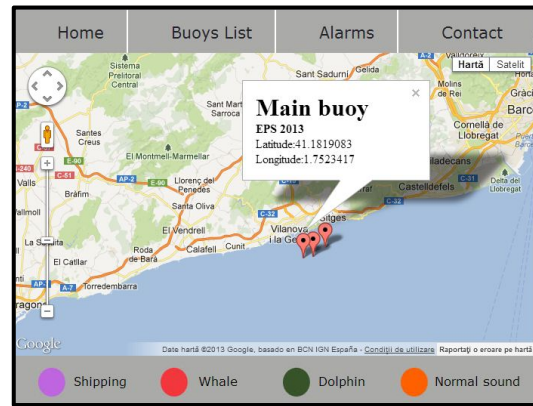


Figure 6 – User-interface page

1. A *home page*, for giving an overview of the interface page and directly showing the map with real time information about possible alarms.
2. A *list of buoys* to see information about all the type of buoys, not only when an alarm appears.
3. An *alarm history page* for displaying not only recent alarms but also past alarms, while also registering all the alarms for future reference.
4. And a *contact page* in case anything is wrong with the product or any other questions.

Dividing the interface in more than one page is important for the page to have structure. A more structured page allows for easier access on handheld devices.

VII. CONSTRUCTION OF THE HOUSING

By putting all the components into a housing and connecting all the components together, our system was created, given the name '*Corangino*'.

Since the buoys and the communication systems within will be on the water, a waterproof housing is a minimum requirement. A standard size box from the market has been chosen to speed up the process. It has a size of 190 x 110 x 60 mm (L x W x H) and is made of lightweight and easy to modify Acrylonitrile Butadiene Styrene (ABS) plastic [9].

Before actual construction, drawings were made (Figure 7) that were used as a template for cutting, drilling, grinding the box for fitment of the electronic components.

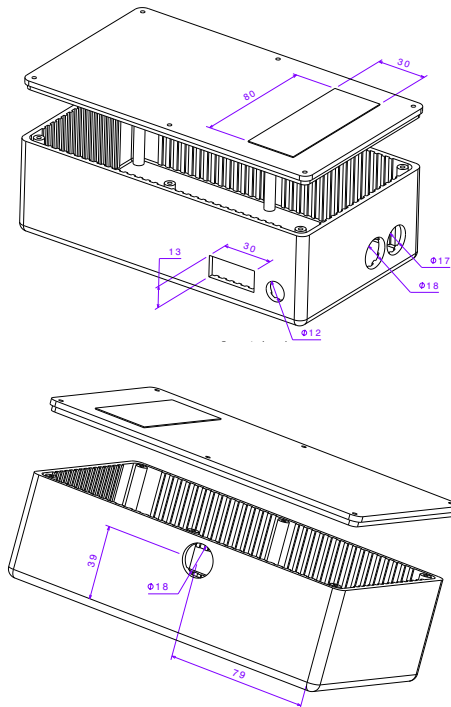


Figure 7 – Corangino Drawings

Using both power tools and hand tools, the housing was modified according to the template ensuring a perfect fitment of all the components. Power tools such as a pillar drill, a soldering iron and a dremel were used to construct most of the box. Sanding paper and different files were used to put on the finishing touches. The electronic components are secured in the box with Velcro [10] for easy installation and take out. Connectors are installed with screws and or threads. All the components put together into the housing resulted in the following result (

Figure 8):



VIII. CONCLUSION

The main goal of this project was to develop a communication system to transfer data from a collecting buoy to the end user. Exactly this has been developed.

Looking back at the electronics part, it can safely be said that an efficient device has been designed, tested and manufactured. Resulting in the Corangino prototype, consisting of a two main parts. An Arduino Mega 2560 board with a smart functions programmed into its memory and a self-constructed board named the Boarduino. The electronic part together with communication part has established a stable point to point connection using the XBee modules. However, the next step of testing will be to establish a communication in a mesh pattern with several XBee modules supporting and repeating one another.

For the user-interface, two programs have been developed. The first, a Java based program that collects data from the receiver coming in through the serial port. And second, an html page where the end-user can find data about an alarm's origin,

the time it occurred and the location in real time with Google Maps (Figure 9).



Figure 9 – Corangino & Interface

REFERENCES

- [1] <http://www.lab.upc.edu/>
- [2] <http://www.digi.com/products/wireless-wired-embedded-solutions/zigbee-rf-modules/point-multipoint-rfmodules/xbee-pro-868>
- [3] <http://arduino.cc/en/Main/ArduinoBoardMega2560>
- [4] <http://docs.europe.electrocomponents.com/webdocs/0db9/0900766b80db99cc.pdf>
- [5] <http://www.fordata.cn/pdf/short/fdcc1602n.pdf>
- [6] http://www.recom-international.com/pdf/Innoline/R-78Bxx-1.0_L.pdf
- [7] http://www.recom-power.com/pdf/Innoline/R-78Axx-1.0_SMD.pdf
- [8] <http://www.labcenter.com/index.cfm>
- [9] <http://nl.rs-online.com/web/p/general-purpose-enclosures/2816857/>
- [10] <http://en.wikipedia.org/wiki/Velcro>
- [11] <https://developers.google.com/maps/?hl=nl>
- [12] <http://www.adobe.com/nl/products/dreamweaver.html>