TITLE:

Close range remote control with advanced capabilities

STUDENTS:

Magda Huiculescu
Marco Ferdinando Carlino
Michal Kobus
Miquel Pereira

SUPERVISORS:

Joan Vincent Castell
Mike Van der Schaar

DATE:

16th June 2016
<table>
<thead>
<tr>
<th>FAMILY NAME: Huiculescu</th>
<th>FIRST NAME: Magda</th>
</tr>
</thead>
<tbody>
<tr>
<td>HOME UNIVERSITY:</td>
<td>Universitatea Politehnica din Bucuresti</td>
</tr>
<tr>
<td>SPECIALITY:</td>
<td>Mechanical engineering – Industrial design</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>FAMILY NAME: Carlino</th>
<th>FIRST NAME: Marco Ferdinando</th>
</tr>
</thead>
<tbody>
<tr>
<td>HOME UNIVERSITY:</td>
<td>Glasgow Caledonian University</td>
</tr>
<tr>
<td>SPECIALITY:</td>
<td>Computer and electronic systems engineering – Digital systems</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>FAMILY NAME: Kobus</th>
<th>FIRST NAME: Michal</th>
</tr>
</thead>
<tbody>
<tr>
<td>HOME UNIVERSITY:</td>
<td>Lodz University</td>
</tr>
<tr>
<td>SPECIALITY:</td>
<td>Information Technology</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>FAMILY NAME: Angel Pereira</th>
<th>FIRST NAME: Miquel</th>
</tr>
</thead>
<tbody>
<tr>
<td>HOME UNIVERSITY:</td>
<td>Universitat Politècnica de Catalunya</td>
</tr>
<tr>
<td>SPECIALITY:</td>
<td>Telecommunications engineering</td>
</tr>
</tbody>
</table>
Abstract

When a marine underwater housing with electronic equipment is deployed it is usually switched off during transport to the deployment location, which can take days to weeks. The system is then switched on just before its final installation. Electronics and batteries are stored in sealed containers that can be difficult to manipulate especially while at sea; it is preferred not to open them unless absolutely necessary. Therefore, a rugged switch or connector might be installed on the exterior of the underwater housing. These external connectors come with extra cost and also add eventual path for water intrusion and might allow accidental manipulation. An alternative solution could be a radio linked remote control (RC) enough rugged and safe to be used confidently in these and other critical applications.

Key words:
Remote, wireless, underwater housing, design, Arduino, Bluetooth, Android App,
Table of Contents

Abstract .......................................................................................................................... 3
List of Figures ................................................................................................................. 5
List of Tables .................................................................................................................. 5
1 Introduction: Goals and aims of the project .............................................................. 6
2 Design Brief and initial considerations ...................................................................... 7
3 Marketing research and analysis ................................................................................ 8
4 Project development .................................................................................................... 8
  4.1 Communication ...................................................................................................... 9
    4.1.1 Wireless technology available ......................................................................... 10
    4.1.2 Bluetooth ........................................................................................................ 11
    4.1.3 Communication protocol ................................................................................ 12
    4.1.4 Communication with underwater housing electronics ..................................... 13
  4.2 Remote control ....................................................................................................... 13
    4.2.1 Hardware development .................................................................................... 13
    4.2.2 Software development ..................................................................................... 15
    4.2.3 Design .............................................................................................................. 20
    4.2.4 Testing .............................................................................................................. Error! Bookmark not defined.
    4.2.5 Challenges ....................................................................................................... Error! Bookmark not defined.
  4.3 Internal Unit .......................................................................................................... 37
    4.3.1 Hardware development .................................................................................... 37
    4.3.2 Software development ..................................................................................... 38
    4.3.3 Design .............................................................................................................. 40
    4.3.4 Testing .............................................................................................................. Error! Bookmark not defined.
    4.3.5 Challenges ....................................................................................................... Error! Bookmark not defined.
  4.4 Mobile application ................................................................................................. 40
    4.4.1 Platform choice .............................................................................................. 41
    4.4.2 Development plan .......................................................................................... 42
    4.4.3 Prototype application ..................................................................................... 42
    4.4.4 Final application ............................................................................................. 45
5 Consideration of ethical issue and design for sustainability ........................................ 47
6 Conclusions and further research .............................................................................. 48
7 Acknowledgements ..................................................................................................... 48
8 Bibliography .............................................................................................................. 49
9 Appendixes ................................................................................................................ 50
  9.1 Key terms ............................................................................................................. 50
List of Figures

Figure 1: Representation of the system .......................................................... 7
Figure 2: Protocol data structure ................................................................. 13
Figure 3: I2C protocol .................................................................................. 14
Figure 4: Use case diagram for the remote software ..................................... 16
Figure 5: Template of Arduino IDE ................................................................ 17
Figure 6: First design ideas .......................................................................... 21
Figure 7: Surrogates from the initial design .................................................. 22
Figure 8: “Mako” design ............................................................................. 22
Figure 9: Polystyrene models ....................................................................... 23
Figure 10: Hand dimensions as presented in “The Measure of Man: Human Factors in Design” by Henry Dreyfuss [6] .................................................. 24
Figure 11: 3D Models for Remote’s components .......................................... 25
Figure 12: 3D Models for the shape of the remote ...................................... 25
Figure 13: First component position approach .......................................... 25
Figure 14: Calculus on which the position of the buttons was based .......... 26
Figure 15: Preferred viewing area as it is shown in “The Measure of Man : Human Factors in Design” by Henry Dreyfuss [1] ........................................ 27
Figure 16: Position of the display and how it influences the body ............... 27
Figure 17: Materials Graph ......................................................................... 30
Figure 18: ABS accordance to the process ................................................. 30
Figure 19: Dimensions of the remote .......................................................... 31
Figure 20: Polystyrene model of the final shape of the remote ................... 31
Figure 21: 3D Model of the enclosure ......................................................... 32
Figure 22: Final remote dimensions ............................................................. 32
Figure 23: 3D Assembly ............................................................................. 33
Figure 24: Yellow hue that will be used for the remote ............................... 33
Figure 25: Safety ingress measures applied to the design ......................... 34
Figure 26: Safety accessories ..................................................................... 35
Figure 27: Graph analysis for polypropylene .............................................. 36
Figure 28: Bumper protection ...................................................................... 36
Figure 29: Water intrusion membrane ......................................................... 37
Figure 30: Different positioning of the plugs and port .................................. 40

List of Tables

Table 1: Microcontrollers comparison .......................................................... 8
Table 2: Components of the two systems developed .................................... 9
Table 3: Wireless communication comparison ............................................ 10
Table 4: Suitable wireless communication power consumption comparison ........................................ 10
Table 5: Man hand measurements .............................................................. 23
Table 6: Material’s relevant characteristics ................................................... 28
Table 7: Advantages and disadvantages of ABS and PLA .......................... 29
One of fields of research created thanks to this third avenue of human interest is acoustic marine research. As the name suggests it deals with underwater sounds. Those may come from artificial or natural sources. Monitoring noise is important for different reasons as sound pollution, created by ships or construction of oil platforms that may threaten the aquatic fauna and need to be controlled. A well placed set of underwater devices is a perfect tool for researching project on some species of animals.

One of the crucial tasks is deployment of underwater housings that gather data. This requires a lot of expertise to develop, maintain and use them and a lot of effort to deploy. Some of requirements they need to meet is perfect waterproofness and ability to operate for extended periods of time on battery power. In addition, there are logistic difficulties caused by rough sea conditions when the equipment is being delivered to its destination. Therefore, it is important to identify problems and complications and overcoming them to streamline the process.

One of entities that specialize in the previously mentioned field is Laboratory of Applied Bioacoustics (L.A.B.). L.A.B. was established in 2003 as part of the “Universitat Politècnica de Catalunya” (UPC) and operates on its campus in Vilanova i la Geltrú. It analyses data from systems of partner co-operators but also deploys its own underwater housings. L.A.B. has also history of participating in European Project Semesters (EPS), a one-semester course designed to train final-year engineering students to work in international teams. In EPS, international teams of students work on real-life projects proposed by national and international companies [1] and this project is the result of this collaboration. It is intended to be developed for the 16th June 2016 and it is realized by an international team consisting in four engineering students supervised by a UPC professor and a representative of L.A.B.

The challenge proposed is to eliminate the need of disassembling and reassembling the underwater housing to turn the electronics inside on and off on a boat just before deployment. It is inconvenient to switch it on on land as it would cost a premature power consume. As stated the sea conditions may make this difficult and time-consuming due to bad visibility, movements of the vessel or cold that forces researchers to wear gloves. The solution consists on apply a wireless technology and integrate a remote control system.

Two devices are developed: an internal unit that will be integrated in the underwater housing and a remote control. The internal unit will switch the electronics inside the underwater housing on and off. It must consume a small amount of energy to preserve battery consumption and it will be able to perform simple diagnostics on battery status and status of the other components used for the research operation as hydrophone and hard drive. The remote control will send command to the internal unit, display feedback received and it will be designed to be suitable for usage while navigating.

The communication between the two devices has to be wireless and no long range features are required. It needs to be bidirectional and safe against interferences.

In the project an app for Android phones will be included, it replicates the same functionalities of the remote, it would suit as a perfect replacement for the remote.

In order to realize the project there were organized weekly meetings with the supervisors. The work was divided using a project management tools, further information can be found in the appropriate chapter.

The timeline of actions to set goals and objectives has been the followed:

The team attended meetings with supervisors to clarify goals and known L.A.B.’s expectations.
Tasks were divided according to member’s specializations. Equipment started to be ordered and research begun. Before the midterm the team had a set of prototypes and designs ready. Those include three designs for the case of the remote, design of the boards that will compose the internal unit and as well the remote, proof of communication between the internal unit and the remote and proof of communication between an Android phone and the internal unit prototype.

Building upon this with the team by the end of the project intends to deliver: the complete internal unit ready to be integrated with the rest of the underwater housing and a remote that communicate with each other in a safe way. Both with be delivered with 3D printed case. The team will also deliver complete Android application.

2 Design Brief and initial considerations

The very first initial consideration was how to integrate a wireless technology in the pre existing underwater housing hardware, as a remote needs a system to communicate with. Options were two:
1. Implement the pre existing hardware of the underwater housing and integrate a wireless technology
2. Develop a separate unit that would communicate with the remote and also able to communicate with the pre existing hardware.

It has been agreed to create a separate unit that takes status information from the underwater housing system and features as:
- Relay, to switch On and Off the power of the underwater housing’s hardware
- Temperature and humidity sensor, to monitor eventual underwater housing overheating
- Current and voltage sensor, to monitor the power consumption of the underwater housing’s hardware

The reasons were that implementing the pre existing hardware could have caused issue on compatibility, involving the redesign of the case and PCB board. Furthermore, memory of the hardware was limited from the other functions that are already operating researching tasks.

The remote sends command but also receives feedback from the internal unit. The feedback is displayed on a screen embedded in the remote. Key features for the remote are durability and resistance against rough condition.

The connection between the internal unit and the remote has to be wireless. Technology considered were mainly Radio Frequency and Bluetooth. Further researches have been conducted to estimate eventual power consumption. Same researches have been conducted to choose microcontroller used to process data and connect sensors and component needed.

![Figure 1: Representation of the system](#)
3 Marketing research and analysis

Researches have been conducted in order to obtain the best product in terms of innovation. The product will not be on the market at the moment but eventual competitors’ analysis and marketing research showed that what distinguish this product from most of the remote on the market is the bi-directional connection. Few remote are receiving feedbacks and they can be found on Air Conditioning market or even more frequently remote for drones.

A second point is the technology used, Bluetooth is not applied for Air Conditioning and not even for drones because of the limited range (10 m). Further information on the reason why Bluetooth has been chosen can be found in Chapter 4 (4.1.2).

This product is the only one, as researches show, equipped with this features and thanks to his flexible design can be upgraded and changed in order to new future needs of the company or adapted for other use of other companies.

4 Project development

The project involves the use of wireless technology, data exchange, use of sensors, ideally all performed in small dimensions. For this reason, the use of a microcontroller is necessary. A microcontroller is a single chip microprocessor system which contains data and program memory, serial and parallel I/O, timers, external and internal interrupts, all integrated into a single chip. Microcontrollers have traditionally been programmed using assembly language of the target microcontroller. There are various forms of BASIC and C compilers available for most microcontrollers that make the programming a much simpler task [2].

A microcontroller will allow the connection between the different electronic components of the internal unit also communication between the remote control and the electronics inside the underwater housing. An Arduino Nano has been selected as microcontroller to use.

There are many microcontrollers available on the market that might be suitable for the project. Three have been taken in consideration and the one chosen was Arduino Nano.

The other two considered were a Raspberry PI board and another Arduino board, Arduino Uno. The three board comes with the following features:

### Table 1: Microcontrollers comparison

<table>
<thead>
<tr>
<th></th>
<th>Raspberry Pi Model B</th>
<th>Arduino Uno</th>
<th>Arduino Nano</th>
</tr>
</thead>
<tbody>
<tr>
<td>Digital Pin</td>
<td>16 GPIO available</td>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>Analogue Pin</td>
<td></td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>I2C supported</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Serial communication</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>OS</td>
<td>GNU based - Raspbian</td>
<td>Upload code from Integrated Development Environment</td>
<td>Upload code from Integrated Development Environment</td>
</tr>
<tr>
<td>Language supported</td>
<td>C, C++, Java, PHP, Python</td>
<td>C, C++, Java</td>
<td>C, C++, Java</td>
</tr>
<tr>
<td>Memory</td>
<td>External SD card</td>
<td>Flash Memory, SRAM and EEPROM</td>
<td>Flash Memory, SRAM and EEPROM</td>
</tr>
<tr>
<td>Power consumed</td>
<td>140mAh</td>
<td>46.5mAh</td>
<td>33.4mAh</td>
</tr>
</tbody>
</table>
**Supply voltage**  | 5.1 Volts DC | 5-9 Volts DC | 5-9 Volts DC

The power source of the internal unit would be a battery so power consumption had to be taken in consideration carefully.

Raspberry Pi Model B board has really high performance although not needed. In the project no User Interface (UI) would developed and C and C++ language would be an ideal programming language. The Arduino Nano is a small, complete, and breadboard-friendly board based on the ATmega328. It works with a Mini-B USB cable. Arduino Nano was designed and is being produced by Gravitech. Each of the 14 digital pins on the Nano can be used as an input or output, using pinMode(), digitalWrite(), and digitalRead() functions on Arduino integrated development environment (Arduino IDE). They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms [3].

Arduino Nano permit serial communication I2C communication, it was really important to check these characteristics as the serial communication is needed to communication with other board and I2C is needed to embed an LCD in the remote to display feedbacks.

Components for the system will be following:

<table>
<thead>
<tr>
<th>Table 2: Components of the two systems developed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Remote</td>
</tr>
<tr>
<td>Arduino Nano</td>
</tr>
<tr>
<td>Bluetooth module</td>
</tr>
<tr>
<td>LCD</td>
</tr>
<tr>
<td>Switch buttons</td>
</tr>
<tr>
<td>RGB led</td>
</tr>
<tr>
<td>Power source</td>
</tr>
<tr>
<td>Internal Unit</td>
</tr>
<tr>
<td>Arduino Nano</td>
</tr>
<tr>
<td>Bluetooth module</td>
</tr>
<tr>
<td>Relay</td>
</tr>
<tr>
<td>Current and voltage sensor</td>
</tr>
<tr>
<td>Temperature sensor</td>
</tr>
<tr>
<td>Serial converter</td>
</tr>
</tbody>
</table>

In the following chapter each functionality is analysed.

### 4.1 Communication

Communication had to be wireless, close-range, safe and bidirectional. The underwater water housing, that is gathering marine fauna’s sound, is left underwater for 3+ months so it has to be sure that when operate it is fully functional. The remote is receiving feedback to double check the actual completion of action and “double confirmation question” are asked before perform any function. The underwater housing is made of PVC that is wave transparent so communication would be clear and it will might have just little distortion or attenuation. Communication with the underwater housing while underwater is not required even thought in theory it would be really difficult as wireless technology use high frequency electromagnetic wave and water molecules absorbs microwaves very easily. If this features would
be required the communication technology has to change in something similar of a sonar, that is used underwater for instance in submarine.

4.1.1 Wireless technology available

Most of wireless technology available have been analysed and compared in order to choose the best in term firstly features and then power consumption.

<table>
<thead>
<tr>
<th>Table 3: Wireless communication comparison</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Near Field Communication (NFC)</strong></td>
</tr>
<tr>
<td><strong>Set-up time</strong></td>
</tr>
<tr>
<td><strong>Range</strong></td>
</tr>
<tr>
<td><strong>Usability</strong></td>
</tr>
<tr>
<td><strong>Selectivity</strong></td>
</tr>
<tr>
<td><strong>Use cases</strong></td>
</tr>
<tr>
<td><strong>Consumer experience</strong></td>
</tr>
</tbody>
</table>

As Table 1 shows, Near Field Communication (NFC) and Infrared Data association (IrDA) have a really narrow range that make those technologies not suitable. NFC is nowadays used in Debit card for contactless payments or in key card to open doors.

IrDA is a technology mostly used in TV remote. It involves the use of a transmitter (usually in the remote) and a receiver (in the TV) and they need the same line of sight plus data sent are usually bytes in hexadecimal form as $0xFFA857$ for instance, that does not make the communication enough safe. In both cases bandwidth and complexity of information are really poor.

Table 2 is showing the three technologies chosen from Table 1 that might be suitable for the project in terms of features but there are differences in terms of power consumption, a second point to take in consideration scrupulously.

<table>
<thead>
<tr>
<th>Table 4: Suitable wireless communication power consumption comparison</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>RF 315/433 MHz module</strong></td>
</tr>
<tr>
<td><strong>Power consumed</strong></td>
</tr>
</tbody>
</table>

Radio Low Frequencies are mostly used for car security system, Remote keyless entry, Garage door controller, Home security, Wireless mouse, Automation system.

These wireless transmitters work with our 434MHz receivers. They can easily fit into a breadboard and work well with microcontrollers to create a very simple wireless data link. Since these are only transmitters, they will only work communicating data one-way, you would need two pairs (of different frequencies) to act as a transmitter/receiver pair. These modules are indiscriminate and will receive a fair amount of noise. Both the
transmitter and receiver work at common frequencies. Therefore, a method of filtering this noise and pairing transmitter and receiver will be necessary.

Wi-Fi and Bluetooth are to some extent complementary in their applications and usage. Wi-Fi is usually access point-centred, with an asymmetrical client-server connection with all traffic routed through the access point, while Bluetooth is usually symmetrical, between two Bluetooth devices. Bluetooth serves well in simple applications where two devices need to connect with minimal configuration like a button press, as in headsets and remote controls, while Wi-Fi suits better in applications where some degree of client configuration is possible and high speeds are required, especially for network access through an access node. However, Bluetooth access points do exist and ad-hoc connections are possible with Wi-Fi though not as simply as with Bluetooth. Wi-Fi Direct was recently developed to add a more Bluetooth-like ad-hoc functionality to Wi-Fi even though performance in terms of power consumption are still different.

Bluetooth has been the one that appeared the best suitable technology for an innovative product. As reported on www.bluetooth.com

“What makes Bluetooth better than other technologies?

The short answer is because Bluetooth is everywhere, it operates on low power, it is easy to use and it doesn’t cost a lot to use. Let’s explore these a bit more.

Bluetooth is everywhere—you will find Bluetooth built into nearly every phone, laptop, desktop and tablet. This makes it so convenient to connect a keyboard, mouse, speakers or fitness band to your phone or computer.

Bluetooth is low power—with the advent of Bluetooth Smart (BLE or Bluetooth low energy), developers were able to create smaller sensors that run off tiny coin-cell batteries for months, and in some cases, years. This is setting the stage for Bluetooth as a key component in the Internet of Things.

Bluetooth is easy to use—for consumers, it really can’t get any easier. You go to settings, turn on your Bluetooth, hit the pairing button and wait for it start communicating. That’s it. From a development standpoint, creating a Bluetooth product starts with the core specification and then you layer profiles and services onto it. There are several tools that the SIG has to help developers.

Bluetooth is low cost—you can add Bluetooth for a minimal cost. You will need to buy a module/system on chip (SoC)/etc. and pay an administrative fee to use the brand and license the technology. The administrative fee varies on the size of the company and there are programs to help start-ups.” [4]

4.1.2 Bluetooth

The use of Bluetooth as communication technology permitted to expand the features of the project developing also a mobile application in addition of the remote.

A mobile application is easier to be changed and implemented in case of changes of company’s need. The mobile application developed will replicate remote functions, further information can be found in Chapter 4.4.

There are different versions of Bluetooth technologies, there one chooses for the project is Bluetooth 2.1 class using a module by Microchip, it is called RN42. Complete datasheet of the module can be found in Appendixes 1.

Some relevant information are:

• RN42 module is an easy to use Bluetooth SPP (Serial Port Protocol) module, designed for transparent wireless serial connection setup.
• Serial port Bluetooth module is fully qualified Bluetooth V2.1+EDR (Enhanced Data Rate) 3Mbps Modulation with complete 2.4GHz radio transceiver and baseband. It uses CSR Bluecore 04-External single chip Bluetooth system with CMOS technology and with AFH(Adaptive Frequency Hopping Feature). Footprint dimensions are 24.4mmx29.9mm.
• Typical -80dBm sensitivity
• Up to +4dBm RF transmit power
• Low Power 1.8V Operation, 1.8 to 3.6V I/O
• PIO control
• UART interface with programmable baud rate with integrated antenna with edge connector
• Bluetooth class 2 - Up to 10-meter range.

Software features:
• Default Baud rate: 38400.
• Stop bit: 1
• Parity bit data control
• Supported baud rate: 9600, 19200, 38400, 57600, 115200, 230400, 460800.
• Permit pairing device to connect as default.
• Master and slave mode
• AT command supported

RN42 has a “normal mode” and a “programming mode”, called AT command mode. Specification regarding software features can be set accessing the AT command mode but it usually done pressing a button in the module or pulling a pin on HIGH voltage plus it requires introduction of text so the module have to be set in advanced and that brings some constrains.

RN42 can be set to auto pair with specific modules if they are within 10-meter range. One module has to act as “master” and the second as “slave”. The master module is holding the MAC addresses and auto pairing within 3-4 seconds. The slave mode can connect to whichever module as long as they have the same baud rate set and the user knows the pass key to access it. The master module is storing MAC addresses of the slave modules in a memory address register that can hold up to 8 MAC addresses, this means that one remote can support up to 8 underwater housings but underwater housings can be connected to smartphones using LAB app developed or other remote that might be developed in future. A mobile does not have constrains on number of devices to be connected with.

4.1.3 Communication protocol

RN42 support serial communication. Serial communication is a type of communication that is transmitting data one bit at a time. Error rate might depend on the baud rate used, the amount of bit sent per second. The remote and the internal unit are communication sending 9600 bps. As error detection features Parity bits is used. Parity checking uses parity bits to check that data has been transmitted accurately. The parity bit is added to every data unit (typically seven or eight bits) that are transmitted. The parity bit for each unit is set so that all bytes have either an odd number or an even number of set bits. Assume, for example, that two devices are communicating with even parity (the most common form of parity checking). As the transmitting device sends data, it counts the number of set bits in each group of seven bits. If the number of set bits is even, it sets the parity bit to 0; if the number of set bits is odd, it sets the parity bit to 1. In this way, every byte has an even number of set bits. On the receiving side, the device checks each byte to make sure that it has an even number of set bits. If it finds an odd number of set bits, the receiver knows there was an error during transmission.

It is not a reliable method as for instance if two bits are changed because of noise during transmission no error would be detected. For this reason, another communication protocol has been developed. This communication protocol is not applied on the final prototype as it is written in C language and could not be applied in the Android mobile application. It is fully working between the remote and the internal unit but makes impossible communication with the mobile app as the structure of information is different. As a future development for this project, after developing the same protocol in Java it would be possible to fully communicate in a safer way. The communication protocol is structuring data as following.
The protocol developed is using a header to validate the incoming data. It uses a fix header that the receiver is recognising, accepting then the data flow. The size of the incoming data is communicated as second and then the payload. When the payload size is equal to the size of the payload communicate the checksum starts running. Checksum is an error detection method more reliable than parity checking, it uses algorithms than can be of different complexity. As the name suggest this method check the sum of bytes of the payload performing some calculation. The one used in the protocol is MOD10 also known as Luhn algorithm. The Luhn algorithm was developed by German computer scientist Hans Peter Luhn in 1954. It calculates simple checksum formula used to validate identification numbers such as credit card numbers. The algorithm was designed to protect against accidental errors, such as a digit mistyping. It will detect any single-digit error, as well as almost all transpositions of adjacent digits. It will not, however, detect transposition of the two-digit sequence 09 to 90 (or vice versa).

The calculator below gives Luhn checksum of the given digit sequence. The sequence is considered valid if the checksum mod 10 equals to zero. It also gives the next check digit to be appended at the end of source sequence to form valid number according Luhn algorithm.

The formula is quite simple: to calculate Luhn checksum you need to sum all odd digits (calculating from right to left, so last digit is considered N1) plus sum of all even digits multiplied by 2, if the product of multiplication is greater than 9 you must subtract 9. If the last digit of the checksum is zero, the whole sequence is valid.

To produce validation digit, we can simply append "0" to source sequence and calculate Luhn checksum again. If last digit of the obtained checksum is zero then the validation digit is also zero, otherwise validation digit can be obtained by subtracting last checksum digit from 10. [5]

### 4.1.4 Communication with underwater housing electronics

The communication between the two boards, the internal unit one and the underwater housing one is serial. The underwater housing will send simple strings and because the communication is wired would be more reliable so there is no need of developing or use a different communication protocol. No actions are performed so in case of eventual error the user can ask again for the information without any eventual consequence.

### 4.2 Remote control

The remote has been fully developed from the team. In the next chapters hardware development stages, software development and design development are explained in details. The entire development has been influenced by the specific user target that the company has. Users are operators of the company that deploy the underwater housing in rough weather condition wearing gloves.

#### 4.2.1 Hardware development

The remote has been developed using firstly a breadboard to try all the component and organize them in the right way. Once results were obtained, a schematic and a PCB board has been designed using Eagle software. Schematic and PCB board for the remote can be found in Appendix 2.
PCB has been designed in conjunction with the case, team member in charge of electronics design and product design co-worked in order to obtain the best product for a comfortable and effective user experience. List of the component used can be found in Appendix 3. In the next chapter each component in further described.

### 4.2.1.1 LCD

A 20 x4 characters LCD by Midas is the element that is a key element. It makes the remote able to display feedback and creates a better user experience. The number of I/O pin in Arduino Nano is limited to 14 so I2C technology is used to interface the LCD. This protocol requires only 2 wires for data exchange, SDA and SCL. The inter-integrated circuit (I2C) serial interface protocol was developed by Philips in the late 1980x. Can be available in high-speed mode with a data rage of 3.4 Mbps. The I2C protocol has the following characteristics: 

- Synchronous in nature. A data transfer is always initiated by a master device. A clock signal (SCL) synchronizes the data transfer. The clock rate can vary without disrupting the data. The data rate will simply change along with the changes in the clock rate.

The LCD comes with I2C communication features and clear understanding and explanation of the process can be found in the related datasheet.

![I2C Protocol](image)

**Figure 3: I2C protocol**

Full datasheet can be found in Appendix 4.

### 4.2.1.2 Bluetooth module

Bluetooth module in remote has been set with 9600bps baud rate. It is in master mode holding just one MAC address at the moment. It is auto paring within 3-4 seconds with the internal unit Bluetooth module. Because it is in master mode it cannot be discovered by other Bluetooth devices. The code to access AT command mode can be found in Appendix 5. Arduino Nano has been used to access AT mode and operate. The same passkey has been set for the two Bluetooth module to avoid text input.

### 4.2.1.3 RGB led

An RGB led is used as visual indicator for users to instantly know the status of the battery in the buoy. RGB led is reading the status of the current sensor and switch colour in order to the percentage of the battery left. The status of battery is counted upon the amount of current that flew through the current sensor. If the user needs more information there is a function in the menu where more detail and live information are displayed.
Full datasheet can be found in Appendix 6.

4.2.2 Software development

The software is structured following the flow chart included in Appendix 7. Between the two systems the “intelligence” is all performed in the internal unit so the remote acts only as a feedback reader. The principle functions used in the source code are

```cpp
Serial.available();
lcd.print(Serial.read);
Serial.write(" ");
```

Serial.available is a function that is checking any incoming data from the serial port. In this case the serial incoming data are the one from the Bluetooth module. It is possible to have more than one module that is receiving serial data and then define a function of each of them. An example of that is used in the internal unit software and further information can be found in the Software development chapter of the internal unit.

The second function is taking what is read from the serial module and print it in the screen. Further information about the screen can be found in chapter “Screen handling”.

Serial.write() is sending strings that the internal unit would recognise and match in a switch case. If the two strings matches than the function is performed. Full source code of the remote can be found in Appendix 8.

4.2.2.1 Menu requirements

As stated in the project title and previous chapter - the remote has advanced capabilities. It is something more than regular TV remote that has number of buttons and sends messages assigned to them. It has to display feedback and be suitable for use in rough conditions. To do so it has number of buttons limited to three and incorporates a screen. A challenge that arises from such a setup is the fact there that the remote can perform more functions than there are buttons. Such property results in specific set of requirements.

Functional requirements:
- Software has to implement layered menus.
- Software has to suitable for 20x4 character screen that has no animations.
- Menus have to be easy to navigate with three buttons - “SCROLL”, “OK” and “BACK”.
- Software runs on an Arduino Nano board with ATmega328 microcontroller.

Non-functional requirements:
- Software should allow easy modification of menu.
- Software should display status of the remote - for example battery icon.

The user’s target of the remote will be employees of L.A.B. - computer literate people. Using the remote should not be a problem for them given the software is functional and clear. It software will be interacted with in ways shown in Figure 4.
4.2.2.2 Menu implementation

After specifying the requirements, it was time to choose programming language and programming environment. Fortunately taking this decision was straightforward. Arduino is not only a name of a board. It is a company and community that provide both hardware and software. The IDE provided for the boards is free, efficient and easy to use. What is more choosing Arduino meant that the team will program it using mixture of C and C++ as only this language is recognized by the boards. Both C and C++ are popular and proven; another advantage is the experience that the team already had with those.

Choosing Arduino had another consequence - specific structure of the programs. Usually a C or C++ program has a main function that is the starting point of its execution. In Arduino programs do not have it, instead they have two basic functions - setup and loop. The code in first one runs once at the beginning of execution. Code in loop is running continuously. Standard Arduino program template can be seen in Figure 5.
The software comprises of one file and follows structured programming paradigm. It is most convenient taking into account that programming for microcontrollers encourages small size and simplicity of the programs and that menu it not too complicated to be easily maintained when written that way. The team experimented also with more object oriented approach. It was rejected and is described in later chapter.

To provide clear description of the program, it will be grouped into parts. Each part will focus on specific task or set of tasks performed by the program and describe functions responsible for them. It is assumed that the reader has experience with programming. Full code is featured in the appendix. Explanations of key terms for those, who are not familiar with programming but wish to understand chapters below are also provided there.

4.2.2.3 Screen handling

The first step was to utilize the screen. Its specifications together with I2C protocol were described previously. However, the team encountered problems as the screen did not work with known library. And as described previously, the team had to customize a library to be able to use it. First it needs to be included in the program:

```c
#include <LCD.h>
#include <LCD_C0220BiZ.h>
#include <ST7036.h>
```

A variable representing the screen needs to be created and given data.

```c
ST7036 lcd = ST7036 (2, 20, 0x78);
```

With that done, the screen variable needs to initialized:

```c
lcd.init();
```

Displaying text on the screen is very straightforward and is done with following functions:

```c
lcd.setCursor(1,5);
lcd.print(text);
```
The first one sets the cursor - place where the first printed character will appear. The second prints the characters. It takes as an argument string of characters or a single character. Before printing new information the `lcd.clear()`; function can be used to erase previous contents of the screen.

### 4.2.2.4 Displaying information

Displaying information is handled by function named “draw”. It takes a pointer to an array of strings, number of strings in that array and a true/false argument to differentiate if the displayed text is a menu or not.

```plaintext
void draw(String *text, int number, bool isMenu)
```

If the text is a menu, it will not be written from the beginnings of the lines. One character will be left out as a place for the cursor. In this case it is represented with “>” sign and marks the currently highlighted option. In that case battery power and potentially other information will may be displayed in corner of the screen. The team planned to differentiate highlighted options by actually highlighting the text, but the used screen does not allow that. If the text is not a menu, it will be displayed from the beginning of each line. It is user’s responsibility to provide appropriate number of strings as size of the array is not checked.

The function also restricts a number of lines that can be displayed to four because of screen limitations. Important notice: cursor mentioned in this chapter is not the same as the cursor mentioned in previous one. Second one is associated with printing text on the screen by the software. First one was implemented by the team as visual aid to the users and is controlled by them. In the coming chapters word “cursor” will refer to the first cursor if not stated otherwise.

### 4.2.2.5 Menu logic

The logic that controls the menu is the part of the software that proved the hardest to develop. It had three iterations and only the last one was deemed satisfying. Final menu is controlled through set of functions. Each of them represents one layer of menu (or one submenu) with up to 4 items user can choose. Functions have the same signatures:

```plaintext
void submenu112 (bool OK)
```

They return nothing and take one argument that can be false or true. This argument specifies if the “OK” or “BACK” button was pushed. The position of cursor is at all times known by the program and kept in variable that is accessed by each submenu function. According to this variables appropriate option in current submenu is chosen or previous submenu is displayed. The only exception is the first layer of the submenu from which the user cannot back off. Each of the items may start both another layer of the menu or another function that for example sends data from the remote. The trickiest part was how to control the flow of the program. At first glance it seems easy - a function waits until one of the buttons is pressed and then acts according to input. But structure of Arduino programs made this solution impossible. To listen for input, the control needs to be passed to loop function. When input comes an appropriate function needs to be executed quickly and control should once again return to loop or else the remote would stop responding to button presses. A working and easy to maintain solution to this problem is pointer to a function. In C it forces all submenu functions to have the same return type and arguments list (in that case void and with one bool argument). Pointers to the text of current submenu text and a number of items in it are also stored. Whenever function responsible for current menu layer is needed, the program calls whichever one is pointed to by the function pointer.

To assist in moving through submenus and make code more readable helper function was written:

```plaintext
void goToMenu(void (*function)(bool), String *text, int number)
```
It changes variable holding number of items in current submenu and pointers to submenu function and text. It also resets position of cursor to first item. It is called by submenu functions each time the current submenu changes.

4.2.2.6 Handling buttons

As mentioned previously - buttons are handled in continuously running loop function. Each time it runs it checks and saves the status of the buttons. Then it compares the current state to the state from a previous run. If the state is different and the button is being pushed action is taken. In case of “OK” and “BACK” function pointed by the pointer is started with appropriate argument. If the scroll button was pressed, position of the cursor is changed and current menu is redrawn to show this change. Lastly the current states are copied to variables that store previous state and the function ends.

4.2.2.7 Functions

4.2.2.7.1 Bluetooth

Bluetooth module is attached to the default serial port Arduino Nano. Pin 0 for RX and Pin 1 for TX. There is not need to define anything else, the function used are Serial.write() to send strings that are recognised in the switch case of the internal unit.

4.2.2.7.2 Reading feedback

Serial.available() and Serial.read() are the two functions used to read incoming data from the internal unit and print what are received on the LCD.

4.2.2.7.3 RGB led

This function takes three arguments, one for the brightness of the red, green and blue LEDs. In each case the number will be in the range 0 to 255, where 0 means off and 255 means maximum brightness. The function then calls 'analogWrite' to set the brightness of each LED. Full code can be found in Appendix 10. The reason that you can mix any color you like by varying the quantities of red, green and blue light is that eye has three types of light receptor in it (red, green and blue). Eyes and brain process the amounts of red, green and blue and convert it into a color of the spectrum.

4.2.2.8 Past iterations

4.2.2.8.1 First iteration

This iteration was the simplest approach. The menu used one big switch statement. The statement used an integer number to determine which menu to display. It proved flawed for two reasons. Firstly, the reader can try to imagine an algorithm that conveys position in multilayer menu with a simple number. Its creation and understanding proved to be hard. Secondly consider a menu that has four items each of them leading to another menu that also has four items. This gives us sixteen cases in the switch statement (one for each item) and will grow fast as the menu grows. The switch statement also basically requires the programmer to model a structure that is easiest to imagine as two-dimensional construction as one-dimensional. Overall this approach used most basic techniques but was a nightmare to maintain and develop.

4.2.2.8.2 Object-oriented approach

After failure of the first approach, the team turned to more advanced techniques. C language was not developed with object-oriented approach in mind, but C++ was. It was hoped that crucial parts of C++ will be available. A submenu object was defined. It held its text, pointers to previous and next submenus or functions and had another set of functions that handled button presses and other situations. However, each object has to be initialized - it needs to be given its text, pointers to functions and other submenu pointers. Plus, definition of this object was placed in the same file as the rest of the code. This lead to many lines of code...
that were hard to read and maintain. Overall this approach turned out to be unclear, complicated and not reliable.

4.2.2.8.3 Custom characters

Originally the remote software was written for different screen that was used in the first prototype. This screen cooperated with NewLiquidCrystal library. It also holds up to eight custom defined characters. Displaying such characters in a corner of the screen was an efficient way of conveying information such as battery level. They are defined in following way:

```cpp
lcd.createChar(2, fullBattery);
```

Where the number is an index assigned to the character and fullBattery is a one-dimensional array of integers. Basically this array represents the character itself. On the screen it will be displayed in rectangle eight pixels high and five pixels wide. And bits that constitute each of the integers in the array control which pixel will be lit and which not. So a character resembling a empty battery will be represented in a following way:

```cpp
uint8_t lowBattery[] = {
    B01110,
    B11111,
    B10001,
    B10001,
    B10001,
    B10001,
    B10001,
    B11111
};
```

Print function and assigned number are used to display the custom character.
The team did not manage to implement this function on the final screen although it should be technically possible.

Customs characters were shown to be a useful feature on simple displays allowing to display icons that convey additional information.

4.2.3 Design

The design process implies following a succession of steps in order to achieve a final product. In this part of the report it will be presented how the design of the remote was developed, from the idea to the prototype.

4.2.3.1 Design concepts

Before starting the creation process, the most important part is to understand what is the problem that needs to be solved. The problem can be identified by answering the following question: why is the remote needed?
As it was presented previously, at the beginning of the report, the need of the remote, apart from the power consumption aspect, is influenced by the environment where it will be used.
The design of the remote should respond to a certain usage, meaning that it should be created in such a way that it can be suitable for sea usage, even in rough conditions.

4.2.3.2 Brainstorming stage

The first step in creation is brainstorming. By using this method there will not be any boundaries traced from the very beginning that might cause a creative obstruction. The brainstorming stage for this project was essential in order to create a unique and functional design.
In order to understand better the environment in which the remote will be used, the team asked the company supervisor which are the challenges that mainly appear on a boat, especially in the case of rough conditions. They are presented in the following list:

• Continuous movement of the boat caused by waves
• Worsening of instability directly proportional with the speed the boat is moving with
• Possibility of high waves that might splash the device
• Heavy rain that might be a possible water intrusion source as well
• Powerful sun light
• Strong wind

Before developing the first concepts the shape was thought starting from the functionality of the product. A list of the main issues to be solved has been made. When it comes to design, the remote should first respond to the three following aspects:

• It has to be easy to handle (even with gloves)
• It has to offer good protection to the electronics inside
• It should have aesthetic qualities

A short list of initial constraints resulted as an outcome of the first phase of the brainstorming:

• The remote has to be designed in such a way it will be used with only one hand
• It has to be splash proof (the objective is to reach an ingress protection up to IP65)
• The colour of the product will be strongly influenced by the environment in which it is used

4.2.3.3 Sketching stage

Knowing the initial design requirements and constraints, the next step was to create the first shape sketches, in order to present them to the supervisors. The main source of inspiration was, on the first hand the industrial field and secondly the marine life.

Six initial designs were made, each of them different in any shape aspects. The team, which includes the supervisors, has decided in three out of six designs.

![First design ideas](image-url)
Together, with the supervisors, the team chose the first three designs illustrated in Fig. 1. The next objective was to develop new ideas from them in order to find the best design for the remote. After developing new surrogates from the previous designs, new shapes were developed:

![Figure 7: Surrogates from the initial design](image)

But the shape that draw the attention was the following one:

![Figure 8: “Mako” design](image)

The shape for this design, as the name is suggesting, was inspired by the shape of the Mako shark tooth. The principal advantage of this shape is the easy way it can be handled, offering to the user a good grip and lowering the chances of dropping it and damage the enclosure. But the advantage could not only be seen from the sketch, so the next step of the project was to taste the efficiency of the design by creating polystyrene models.

4.2.3.4 Polystyrene models

This step of the project is essential in order to decide the shape of the remote. The step consists in carving the pieces of polystyrene in the shapes that were drawn. The objective is to analyse the dimensions and also to realise if something needs to be changed in the shape. The main purpose is to test if the grip of the remote is ergonomically and if there are difficulties in holding it or in working with it. The challenge is to find a shape that suits best the user. The process requires a polystyrene board, a cutter, two types of emery paper and epoxy foam or any adhesive. After deciding together with the supervisors which shape should be analysed, the total of polystyrene models was three.
The final stage of this step is to give them to the team members and the supervisors to in order to test them. The team had the ability to choose the suitable design by handling and interacting with it. This results of the step were essential in understanding what are the pros and the cons of each shape and to decide which one will be used for the remote. The next step will consist in a research on how to improve the ergonomics by analysing further the average man hand.

4.2.3.5 Dimensions and ergonomics

In order to apply to the design, the best measurements, the team asked the company supervisor who will be the average user of the remote. The user will be mostly man. This piece of information leads the research into a more specific field: the man hand dimensions. [1]

Table 5: Man hand measurements

<table>
<thead>
<tr>
<th>MAN HAND DATA</th>
<th>Small [cm]</th>
<th>Medium [cm]</th>
<th>Large [cm]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hand length</td>
<td>17.28</td>
<td>19.05</td>
<td>20.82</td>
</tr>
<tr>
<td>Hand breadth</td>
<td>8.13</td>
<td>8.89</td>
<td>9.65</td>
</tr>
<tr>
<td>3rd finger length</td>
<td>10.16</td>
<td>11.43</td>
<td>12.7</td>
</tr>
<tr>
<td>Thumb length</td>
<td>6.1</td>
<td>6.85</td>
<td>7.6</td>
</tr>
</tbody>
</table>

In designing the remote, the team focused on the “Medium” column of the table. But the dimensions in the table will be slightly modified when the user is wearing gloves. Doing a more elaborate research the team has discovered how the initial dimensions will change. The following figure is showing how exactly the dimensions are influenced in case of wearing “heavy winter gloves”.

Figure 9: Polystyrene models
Figure 10: Hand dimensions as presented in "The Measure of Man: Human Factors in Design" by Henry Dreyfuss [6]

The information provided by this research will be implemented in the shape of the enclosure mostly regarding the grasp. These measurements will help in designing a remote that can be easily handled and they will reduce the possibility of an eventual drop.

4.2.3.5.1 Electronics dimensions

This step has started after the team members involved in system electronics design have finished deciding which electronic components will be used for the remote. In this stage the dimensions of the electronics are known. The purpose of this stage is to realise the 3D models for the components and enclosures. The challenge of the stage is to find the most suitable position of the electronics by keeping the design unchanged as much as possible.
After modelling the components, other 3D Models were made for the shape of the remote. The models followed the initial design but were adapted to the size of the components. This 3D Models for the enclosures do not have in this stage any details. The position of the display and the buttons will be analysed in the “Ergonomics” chapter of this report.

This task was very useful in finding new shapes and mostly to have an idea about the way the Printed Circuit Board’s shape will be modelled. After this stage was complete a decision was made on which design will be kept and improved until the final prototype. The following image shows which is the design and also a first look at the position of the components inside the enclosure.
4.2.3.5.2 Ergonomics

In this stage of the report will be analysed only two aspects that involve ergonomics:
The position of the buttons
The position of the buttons is a very important aspect in obtaining the best way to handle the remote. In order to discover which button position is most suitable for the design, on which the team has decided on above, first there were made mathematical calculus. The calculus was based on the finger dimensions of an average user and the remote grasp. The following figure illustrates how the position of the buttons was calculated.

![Figure 14: Calculus on which the position of the buttons was based](image)

The next stage of this step was to apply the calculated position on a polystyrene model that was tested by the team and the supervisors. The position of the buttons can be easily seen in the previous Figure 8.
The position of the display
The position of the display was strongly influenced by the idea of implementing the following ergonomic concept:
This research was referring to the eyes position while staying in front of a computer. The team decided to apply this idea to the position of the remote's display. In this way, it will offer a strength position of the body while using it and also a relax position of the eyes as the research is suggesting. In the following drawing it is shown how this concept was thought by the team:

The team has decided to bend the display with 20° in order to obtain the desired results.

4.2.3.6 Materials and manufacture

The material not only influences the functionality and the characteristics of the product, but also the shape and the ergonomic. Choosing the right material is an essential and a critical step in creating a design.
For this objective of the project a material analysis has been made. This analysis consisted in two main steps: doing a research on material characteristics and using a program to generate a graph.

4.2.3.6.1 Research on material characteristics

The research is based on two important aspects: the remote will be used in extreme working conditions and it has to be durable enough to any shock that might cause damage. In order to satisfy these main requirements a list of material characteristics has been made:

- High impact resistance
- Rigidity
- Solar/salt corrosion resistance
- Adherence
- Low temperature conductivity
- Recyclability
- Colour

The research has advanced then in a direction restricting to only one category of materials. A decision was made that the characteristics mentioned above will be fulfilled best by a plastic polymer. After the limited area was bordered, in deciding which list of materials it will get down to, another quick research was made to analyse the manufacturing methods available. The manufacturing methods of enclosures, just like the ones used for remotes, are made by using injection. This process consists in introducing material into an injection moulding machine thru a hopper. The injection moulding machine contains a heated barrel equipped with a screw, which leads the molten polymer into a temperature controlled split mould through a channel system. The screw melts the polymer, and also acts as a ram during the injection phase. The polymer is injected into a mould tool that defines the shape of the moulded part. This manufacturing method fits best for a mass production. [2]

The quantity of products restricted to this project is one prototype. Therefore, another opportunity has occurred. The manufacturing method does not need to be injection moulded. The prototype will be realized by using a 3D Printer.

The material that will be chosen for the prototype has to meet the requirements for two types of manufacturing methods: one that will be used for the prototype of the project which is the 3D Printing and one that might be used in the future for a greater amount in quantity production, which is Injection Moulding. This decision was taken so the material will not be changed in case a large production is desired.

There is a great amount of plastic materials that can be injected, but the list of plastic materials that can be 3D Printed is far lower. By researching and intersecting the list of materials that can be both injected and 3D printed the following list was obtained: Polylactic Acid (PLA), Nylon, Acrylonitrile butadiene styrene (ABS), Polypropylene (PP), High impact polystyrene (HIPS), Polyethylene Terephthalate (PET). [3][4]

A list of each material’s, suitable for the project, characteristics was made to analyse and conclude which one will be used for the remote. [5]

<table>
<thead>
<tr>
<th>Table 6: Material’s relevant characteristics</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABS</td>
</tr>
<tr>
<td>---</td>
</tr>
<tr>
<td>• Excellent impact resistance</td>
</tr>
<tr>
<td>• Excellent aesthetic qualities</td>
</tr>
<tr>
<td>1. Does not absorb water</td>
</tr>
</tbody>
</table>

28
After analysing the characteristics that best fit the project requirements the list of plastic materials that can actually be used was reduced at two: ABS and PP. Therefore, a very important aspect was not neglected: 3D Printer’s technical features. The available 3D Printers for this project are Up Box 3d Printer and Colido 2.0 Plus. Both 3D Printers are suitable for either ABS or PLA, but not for PP. [6][7] This constraint was useful in deciding which material will be used for the prototype. The materials that this type of 3D Printers are working with are ABS and PLA. So, the actual decision had to be done between ABS and PLA. Below there is a comparison of how ABS and PLA when they are 3DPrinted [8]:

### Table 7: Advantages and disadvantages of ABS and PLA

<table>
<thead>
<tr>
<th>ABS</th>
<th>PLA</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Advantages:</strong></td>
<td><strong>Advantages:</strong></td>
</tr>
<tr>
<td>• Strength</td>
<td>• Higher printing speeds</td>
</tr>
<tr>
<td>• Flexibility</td>
<td>• Sharper printed corners</td>
</tr>
<tr>
<td>• Machinability</td>
<td>• Glossy surfaces</td>
</tr>
<tr>
<td>• Higher temperature resistance</td>
<td></td>
</tr>
<tr>
<td><strong>Disadvantages:</strong></td>
<td><strong>Disadvantages:</strong></td>
</tr>
<tr>
<td>• Made out of oil, so more damaging to the environment</td>
<td>• Can deform because of heat (like a cassette in a car)</td>
</tr>
<tr>
<td>• Hot plastic fumes when printing</td>
<td>• Less sturdy (than ABS)</td>
</tr>
<tr>
<td>• More difficult to print</td>
<td></td>
</tr>
</tbody>
</table>

The interest, when choosing the material, is focused less on the aesthetic characteristics and more on the capability of the material to offer good protection, and reduce as much as possible the risk of damaging the remote. Therefore, ABS remains the suitable material, even though its surface finish characteristics when it is 3D Printed are less glossy than PLAs. The next step of this chapter was very important to decide if the properties of ABS best fit also in the case of a mass production therefore an analysis has been made.

4.2.3.7  **Graph analysis**
To analyse the materials that will be used for realising the remote it was used CES EduPack 2015. CES EduPack 2015 is a programme that combines the science of materials, engineering and design and responds to the user needs with a wide variety of tools. In this stage, this programme was used to generate a graph of materials. The two axis of the graph that were chosen were: X Axis- “Processing properties - Polymer injection moulding” and Y Axis- “Impact and fracture properties - Fracture Toughness”. (Fig. 1)

![Figure 17: Materials Graph](image)

In this section are the materials with the highest related properties. The material that presents interest in this step is ABS. Therefor the challenge is to discover if ABS has “excellent” properties according to this graph and if it can be used in an injecting mass production. By enlarging the “Excellent” area of the graph and arranging the list of materials in “Alphabet order” the ABS will be shown as a proper material to be used. (Fig. 2)

![Figure 18: ABS accordance to the process](image)

By finishing the two steps of this objective, it can be concluded that the material that meets the requirements and follows the constraints is ABS. This material fits both the prototype and the quantity production.

4.2.3.8 Final design
The design that the team has decided on above is large enough for the components but, even though it was thought accordingly the hand measurements, the dimension of the remote can be rescaled. Not the width of the grasp is the main problem of the dimensions therefore it will be just slightly modified so it will not influence the hand measurements, but the length.

![Figure 19: Dimensions of the remote](image19)

The polystyrene model at this dimensions was made. The polystyrene model was then cut to the minimum dimensions the remote can have. The resulting shape and dimension is presented in Figure

![Figure 20: Polystyrene model of the final shape of the remote](image20)
The new dimensions of the remote are illustrated in the Figure above:

Accordingly, with the new shape the position of the buttons was as well modified. After testing new positions on the given model the new button position, as well as the new components position can be seen in the 3D Model below:
4.2.3.8.1 Colour

Choosing the right colour is a very important step into developing a new product not only to offer it good aesthetic properties but also to integrate it in the environment it will function. In the industrial field the chromatics are not various because the utility of a colour is not to offer the product a good aspect, but to offer it a good visibility. Therefore the colour used mostly is yellow. According to the Occupational Safety and Health Administration federal agency the “OSHA safety yellow” code is 13591 and the hex is #EAC234. This is the hue that will be used for the remote. [9]

This colour has the ability to stand out in any environment and to focus the attention where it is needed. This colour also helps at creating an identity of the company. Yellow is used because the original underwater housing shares the same colour and, in this way, a correlation between the two related objects is made. [10] The colour is directly related to the shape. According to studies the shape that corresponds to yellow is the triangle. The triangle assimilates all shapes of diagonal character, such as rhombus, trapezoid and their derivatives, which is a very important aspect, given the shape of the remote. [11] Yellow shows its brightness and aggressive shine on black, detail that can be really useful for increasing the visibility if needed.

4.2.3.8.2 Ingress protection reflected in the design
At the beginning of the report, one of the requirements was to obtain an ingress protection up to IP65. The first number that follows the letters, 6, means no ingress of dust; complete protection against contact (dust tight). The second number, 5, means the protection for water projected by a nozzle (6.3 mm) against enclosure from any direction shall have no harmful effects. [12]
In order to obtain this results in the design it was included a special gap for a silicone gasket. The silicone gasket has followed the contour of the enclosure as it can be seen in the picture below:

![Figure 25: Safety ingress measures applied to the design](image)

4.2.3.8.3 Safety handle
Given the rough conditions and the environment the remote is used into, apart from the shape of the enclosure that is comfortable enough to be easily handled, the team thought of a method that it can be secured to the hand or the belt while the remote is used.
The team had two options in deciding what kind of method to apply:
1. An accessory that can be placed around the wrist
2. An accessory to hang the remote either on the belt or even on the wall
The team decided to model both accessories and in this way to offer more possibilities to the user, who can decide by himself what is the safety measure he wants to apply.
Both accessories are presented in the Figure below:
Both accessories are provided with block-release mechanism that permit the cords size to be modified. In the hook accessory the cord should be tightened inside while the remote is hanging on a wall and unwounded at the users’ preferences and accessibility while the hook is secured to the belt but the remote is in the users hand. For the wrist safety accessory, the block-release mechanism is used to modify the size of the cord accordingly with the users’ wrist diameter.

4.2.3.9 Future perspectives

The following next two subchapters of the report are showing how can the concept be improved in the future.

4.2.3.9.1 Materials

A future perspective is covering the materials from which the remote should be done. As it is known ABS has some important to be considered disadvantages such as it is not suitable for enclosures that are exposed to the UV radiation and also they are not very recommended in rough conditions. This kind of disadvantages might affect the enclosure, especially when it is used in the environment that this remote was designed for. The team has decided to offer, in case of a mass production the alternative of another material. In the “Materials and manufacturing” chapter it was discovered that polypropylene can be a suitable replacement. The graph analysis was applied as well for this material. The results are shown in the following figure:
As it is shown in the graph, polypropylene would be a suitable material for in case of a mass production.

4.2.3.9.2 Ingress protection

As an addition to the Ingress protection design presented in the previous subchapter the team has developed the following future perspectives.

The addition of an elastic grip to the side of the remote

The elastic grip can be seen as a bumper that is protecting the side and the corners of the remote. It prevents the user to drop it as it is adherent and it offers a better grip of the device. It also covers the corners and in this way it reduces the drop impact.

But the most important advantage of the “bumper” is that it prevents water to enter into the enclosure by covering the side of the remote, as it can be seen in the following figure:

4.2.3.9.3 Buttons and switch protection

The team has made a research in order to improve the water protection for the buttons and switches. The research has shown that there is a membrane that can be applied on the buttons and switches that can stop any water intrusion. [13]
This is a future perspective that can be considered in case the user wants to make the remote completely waterproof. A similar membrane can be applied on the display.

4.3 Internal Unit

To establish communication between the designed remote control, or the mobile application, and the underwater housing is necessary to implement a system in the underwater housing, it will be in addition to the pre existing one.

The pre existing hardware accomplish all the data gathering, managing the hydrophone that records the data and the hard disk that store all this data. The separate unit developed will manage the current that the pre existing hardware is using. Even if the new unit is always “awake” is still consuming less energy than the other one and in this way it possible to achieve the objective of the project to reduce battery use and switch on and off the underwater housing on the boat without manual action.

4.3.1 Hardware development

The power source of the internal unit will be the battery that is already running the system, the internal unit will drain just a little bit of current not decreasing performances of the underwater housing. It will be used to process its function and act as a switch to give power to the underwater housings system. The internal unit board is using 5V voltage so a step down dc-dc converter by Traco power is used to reduce the power from 12V to 5V. The internal unit is able to hold a current up to 36V as reported in the datasheet of the DC-DC converter.

4.3.1.1 Relay

Switching on and off a current require a relay. It is a switch controlled by an electrical circuit that, across an electromagnet and a coil, defines the closing or the opening of other circuits. For the implementation of the internal unit it will be used a relay bi-stable component (ON, OFF) with two coils inside. The datasheet of the relay used can be found in the Appendix.

The relay used is 5V dc current operate, it would switch a current of 12V coming from the battery going to the underwater housing electronics. An addition switch has been embedded in the internal unit PCB board. It performs the same function as the remote. It has been made in case of manual operation.

4.3.1.2 ACS712

It is important to controlling the state of the current but it is also useful to track how much current is consumed and estimate so the remaining power. To achieve that a current sensor is used, a ACS712 chip by Allegro.

It is a current sensor using the Hall Effect to work providing an economic and precise solution to measure AC or DC current. The current sensor send data using an analogue output.

The Arduino measures the input at the analog pin, converts it to millivolts, subtracts the offset and then finally divides it by the scale factor of the current sensor.
Full datasheet can be found in 12.

4.3.1.3  DHT11

To detect and send information related to the condition of the underwater housing, such as temperature or humidity information, it is necessary to connect to the microcontroller a digital temperature and humidity sensor. The DHT11 is a basic, ultra low-cost digital temperature and humidity sensor. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air, and spits out a digital signal on the data pin (no analogue input pins needed). It's fairly simple to use, but requires careful timing to grab data. The only real downside of this sensor is you can only get new data from it once every 2 seconds, so when using the library, sensor readings can be up to 2 seconds old, that anyway does not influence the internal unit performance.

- Low cost
- 3 to 5V power and I/O
- 2.5mA max current use during conversion (while requesting data)
- Good for 20-80% humidity readings with 5% accuracy
- Good for 0-50°C temperature readings ±2°C accuracy
- No more than 1 Hz sampling rate (once every second)
- Body size 15.5mm x 12mm x 5.5mm
- 4 pins with 0.1” spacing

Full datasheet can be found in Appendix 13.

4.3.1.4  MAX232

To communicate with the electronics located inside of the underwater housing and communicate serial a level converter is needed has Arduino pin are working with 5V I/O and RS232 use 12V voltage. It was necessary to include the use of the integrated circuit MAX 232. The MAX232 is a dual driver/receiver and typically converts the RX, TX, CTS and RTS signals. The drivers provide TIA-232 voltage level outputs (about ±7.5 volts) from a single 5-volt supply by on-chip charge pumps and external capacitors.

Full datasheet can be found in Appendix 14.

4.3.1.5  Bluetooth module

The Bluetooth module is set a “slave”. The mac address is stored in the remote memory address register. If the remote would be within a 10-meter range from the internal unit, the two devices auto pair within 4 seconds. The Bluetooth module accepts connection from whichever devices but a pass key is asked.

Full datasheet can be found in Appendix 1.

4.3.2  Software development

Source code of the internal unit is performing all the action. It is organised in a switch case that read strings sent from the remote or the mobile application and run a functions. Several libraries are used. DHT11.h is the library that contains functions for the digital and humidity sensor, SoftwareSerial.h is a library that permit to turn whichever I/O pin in a serial pin, by default there is only one Pin use as transmitter and only one used as receiver. The use of this library is necessary as the Bluetooth communication is serial and also the one via RS232 cable.

4.3.2.1  Relay function
Relay is working on two states. There are two wires that manage the switch of the current keeping the circuit OPEN or CLOSED. Pulling on HIGH voltage (5V) one of the two wires the related action is performed. As can be seen in schematic and code, pin 12 close the circuit and let current flow.

```cpp
void turnLEDOnOff(){
    if(blinkFlag) digitalWrite(12, HIGH);
    else digitalWrite(11, HIGH);
}
```

When the strings is sent the blinkflag will be true pulling pin 12 on HIGH voltage in pin 11 on LOW voltage, in this way the current will flow.

### 4.3.2.2 RS232 function

The internal unit is designed to communicate with the underwater housing to receive feedback on current status of the Hard Disk and the correct functionality of the equipment. The board in the underwater housing is running Linux OS. There have been two tests made and concept proved. The first use a TTL converter to communication using an USB port and the second one uses MAX232 level converter to interface with RS232. Most microcontrollers these days have built in UARTs (universally asynchronous receiver/transmitter) that can be used to receive and transmit data serially. UARTs transmit one bit at a time at a specified data rate (i.e. 9600bps, 115200bps, etc.). This method of serial communication is sometimes referred to as TTL serial (transistor-transistor logic). Serial communication at a TTL level will always remain between the limits of 0V and Vcc, which is often 5V or 3.3V. A logic high (‘1’) is represented by Vcc, while a logic low (‘0’) is 0V.

RS-232 signals are similar to microcontroller’s serial signals in that they transmit one bit at a time, at a specific baud rate, with or without parity and/or stop bits. The two differ solely at a hardware level. By the RS-232 standard a logic high (‘1’) is represented by a negative voltage – anywhere from -3 to -25V – while a logic low (‘0’) transmits a positive voltage that can be anywhere from +3 to +25V. On most PCs these signals swing from -13 to +13V.

The more extreme voltages of an RS-232 signal help to make it less susceptible to noise, interference, and degradation. This means that an RS-232 signal can generally travel longer physical distances than their TTL counterparts, while still providing a reliable data transmission.

The software has been developed using Python and Bash. The software can run whichever bash command than splits information so to make easier which data to transfer and display in the remote as there is limited space and scrolling information is not a functionality included.

The test has been conducted using Putty in the case of MAX232 and using the LCD in case of the connection between Raspberry Pi and Arduino Nano using TTL USB converter.

In both cases the communication is fully working and the internal unit board has a MAX232 embedded.

```python
import sh

hdall = sh.df('-H')
print hdall

hd = sh.df('-H', '/')
print hd

hd_data = hd.split()
print hd_data[9]
```
print hd_data[10]
print hd_data[11]

SH is a module for Python that can run bash command, the code can use in the underwater housing to grab information from the hd disk and sent it to the internal unit with the code below. The following is just sending a letter but the two codes can be merged easily.

```python
import serial

ser = serial.Serial('/dev/ttyUSB0',9600)
while True:
    ser.write('A'+r'
')
```

Just use the data can has to be sent like hd_data as an argument is `ser.write()`.

### 4.3.3 Design

The internal unit has to be positioned inside the underwater housing, screwed and secured to avoid movement. It has been decided to use a rectangular shape as it would be easy to attached on whichever inside wall of the internal unit. Furthermore, plugs for incoming and out coming power and RS232 cable can be easily accessible. The box used has to be wave transparent but robust. The one chosen is a project box available on RS made by Hammond manufacturing. Material is ABS and has on one side a longer top lid that permit to screw it on a wall. Dimensions are 191 x 110 x 57mm. A three pin plug is used for current flowing. Different combinations were tried to obtain the most effective way of place plug and the final result is the one showed in the figure below.

![Different positioning of the plugs and port](image)

### 4.4 Mobile application

The core of the project consists of internal unit and remote described in the previous chapters. During development the team faced series of decisions to fulfil requirements. This resulted in following features:
- Limiting number of buttons to three
- Using Bluetooth unit that automatically connect to up to eight remembered devices
Those decisions have two major consequences. First one means that the remote will have menus, from which user can pick items. Inputting text would be very tedious and is not planned to be implemented. Secondly the remote will connect to eight particular units which addresses are remembered. Changing those addresses would require opening the remote and connecting additional wires to the Bluetooth unit. This solution was chosen, because it presented best balance between offered capabilities, cost and ease of implementation. Significant changes to the requirements may require changing both software and hardware. It may be complicated and time consuming.

The application provide solution to potential problems mentioned above. Firstly, it was created for a mobile device such as smartphone. It offers rich input possibilities with textboxes and on-screen keyboard. A smartphone has also wide possibilities when it comes to Bluetooth connectivity. It can pair with multiple devices, and already come with functionalities such as searching and managing found ones. It is only a matter of accessing those functionalities from the application. Another advantage of the mobile application is the fact that probably all functionalities of the remote can be reproduced on the smartphone. What is more - adding new will not require changes in the hardware, but only in software. Therefore, the app may work as a replacement of the remote in case of:
Loss of malfunction of the remote;
Appearance of new requirements, that exceed capabilities of current design and urgently need to be implemented;
Accommodating need to control more that eight underwater housings.

However, the application is not without its drawbacks. Most mobile devices are not waterproof nor can be operated in gloves. They make it a worthwhile alternative, but prevent it from replacing the remote altogether.

4.4.1 Platform choice

After deciding to develop the app, the team had to decide on which operating system it should be developed. Application created for one OS cannot be run on another without significant changes to its code so this decision was important. According to IDC portal three most popular operating systems on smartphone market in Q2 of 2015 are Android by Google, IOS by Apple and Windows Phone by Microsoft with following shares [10]:

As it can be seen most popular OS is Android. It means that future users are more probable to have access to devices that run Android OS.
Another advantages of developing for this system come from its popularity. Both Google and independent sources provide huge amount of documentation and tutorials that greatly simplify creating applications for the system [11]. There is also a dedicated integrated development environment called Android Studio. It comes with such features as possibility to emulate Android device so the app can be tested even if no smartphone is available [12].

Lastly apart from popularity, powerful tools and rich knowledge base, the team had most experience with programming for Android and this was the final argument in favour of choosing this platform.

4.4.2 Development plan

The application was developed in two stages. First a prototype was created. It served as a demonstration of technology and provided team with possibility to learn more about connecting Android devices to Arduino boards via Bluetooth and establishing bidirectional flow of data. Next stage was development of the final application. It started together with development of layered menus for the remote. Both remote and the app have similar capabilities. The only big difference is the fact, that the final application is able to seek devices and let the user choose to which they wish to connect. In the next chapters the prototype and the final application will be presented.

4.4.3 Prototype application

Purpose of the prototype is to connect to an Arduino board via Bluetooth, send and receive data and disconnect. The board has a slave Bluetooth module and has a program that can listen to and understand commands sent by the smartphone. After successful connection the prototype allows the user to turn blinking of Arduinos built in LED diode, change frequency of this blinking or check current blink frequency and display it to the user.

In this case users of the application are members of the team that present the prototype and potentially supervisors. They are computer literate and will have no problem in understanding applications feedback, principles of operation or how to pair devices.

The task of the app is to demonstrate technology and provide basis for further development.

The requirements are as follows: the device has to allow for installation of applications from unknown sources, have Bluetooth and run on Android Jelly Bean 4.1.2 version or higher. This system version was chosen as it will provide compatibility with most Android devices currently on the market. The app is very basic and virtually have no other requirements. There are no security features implemented at this stage. Only exceptional situation is when the user introduces invalid frequency. No further development in planned.

The author is Michał Kobus. No costs were associated with the development.

Application was implemented for Android OS in Android Studio IDE and is written in Java. Rather uninspired “Arduino Bluetooth” was chosen for a name of the prototype.

Below relevant UML diagrams concerning the prototype are presented. Screenshots and manual are provided as appendix (Appendix 4).
The use case diagram shows activities performed by the app. There are two actors - User that interacts with the app and receives feedback and Arduino board with Bluetooth module that is given commands and requested by the app to provide data.

The more straightforward variables and functions are described here. However, connectivity calls for more in-depth explanation that is provided in the next chapter. There is only one class, which extends the class Activity. Activity class comes with the Android API and is one of most important classes. It takes care of setting up a window in which the UI will be placed and which the user will be interacting.
EditText is where the frequency will be inputted. The application needs a reference to it to access inputted variable. Reference to buttons is needed to show or hide them. Lastly two Booleans are flags - first used to determine whether continue listening to incoming data and second keeps information whether the blink on Arduino is on or off.

Method `onCreate` mainly finds references to buttons and text file and assigns them to variables so they can be then accessed from the code. Functions with names beginning with `onClick` handle button presses.

Method `onClickConnect` establishes connection and makes other buttons visible and `onClickDisconnect` terminates it and hides them. Method `onClickAllowChange` makes the textbox and button responsible for sending the new frequency to Arduino visible. Method `onClickChangeFrequency` checks if frequency introduced by the user is correct. Then it converts it and sends as a command understandable for software on Arduino board. Two last methods also send appropriate commands to Arduino.

Method `onClickCheckFreq` then checks if any data arrived and displays it to the user.

The state transition diagram shows the states in which the prototype application can be. In both states in upper half of the diagram only “Connect” button is visible. In other states this button is replaced by “Disconnect”, “Check frequency”, “Change frequency”, and “Turn the blink on/off”.

### 4.4.3.1 Connectivity

To connect the application makes use of many classes and features provided by the Android API. Firstly, `PORT_UUID` is needed to establish communication with Arduino. Variable device is instance of class `BluetoothDevice` and combined with `PORT_UUID` creates a representation of Arduino board the app wants to connect to. Then the device is passed to socket that is an instance of class `BluetoothSocket`. This socket is used to send and listen to data. To do it the app needs to access both streams that are parts of the socket. They are held in `inputStream` and `outputStream` variables. Streams are very useful as they take care of most tasks other that reading and writing data from and into them. Lastly buffer is used to accumulate incoming data.

Method `BTInit` checks if the device supports Bluetooth. If it does, applications request the Android system to enable Bluetooth. This results in a window that asks for users permission to do so. Then Bluetooth information is accessed and list of paired devices is checked. If it is empty user is asked to pair a device.

Method `BTConnect` makes the connection based on chosen device.

Method `onClickConnect` utilizes both previously mentioned functions. First is calls `BTInit`. In case of its success a list of paired devices is saved. When the “Connect” button is pressed again the list is used to inflate a drop-down menu that appears on top of the button. When user clicks one of items on this list application saves the chosen device and `BTConnect` is called to attempt creating connection to the device.

If the connection is established `beginListenForData` runs. It makes the smartphone listen for incoming signals from Arduino. However, it presented a development challenge as Android applications are by default run as processes with one thread of execution. It means that instructions are executed one after another and one
at a time. If the listening was to be done in the main, default thread it would mean that the app would be frozen until it was done. This is obviously undesirable. Therefore, another thread of execution is launched exclusively to do the listening. Sadly, only the main thread can interact with the user interface of the application. To do it from another thread a handler is needed. This special class needs to be associated with thread where it should post information or commands. Then they can be posted there from another thread. Handler is used in this case to show information received from the Arduino board. Method onClickDisconnect terminates the connection and hides all buttons while showing the Connect button. To terminate the listening thread, it sets continueThread back to false.

4.4.4 Final application

The core purpose of the final application is to enable an Android device to serve as a replacement for the remote. The app also has some functions that the remote does not have, as stated in the introduction. The intended users are employees of LAB. Therefore, the team expects a high level of computer literacy, understanding how the app works, what functions it provides and what feedback it displays. The tasks of the app are to connect to an Arduino Nano board equipped with a Bluetooth module. The board in this case is the heart of internal unit and the unit will be placed inside of the underwater housing. It also allows to turn the underwater housing on or off, asks for data about the status of its systems and provides feedback.

The requirements are as follows: the device has to allow installation of applications from unknown sources, have Bluetooth and run on Android Jelly Bean 4.1.2 version or higher. The application is simple and has low requirements in respect to memory or computational power. The security features Comprise of password and some modifications to app behaviour. Further development is taken into account, but probably will not be carried out by current team. Its direction is also unknown.

The author is Michal Kobus. No costs were associated with the development. Application was implemented for Android OS in Android Studio IDE and is written in Java. “LAB Remote” was chosen for a name of the app to make clear connection to the company that commissioned the project.

The application was designed to perform following tasks:
As it can be seen on Diagram, the Application allows user to check available devices, connect to chosen one, turn the underwater housing on and off, check its status and is protected by a password. The basic states of the application are described on state transition diagram:

They correspond to activities that the application implements. And activity on Android can be described as single screen devoted to particular task with which the user can interact.

4.4.4.1 Password and other improvements
The core of the final application is similar as in case of the prototype. The connectivity is programmed almost in the same way. There are however two main differences.

Firstly, the app implements multiple activities. Incoming feedback triggers one of them there it is displayed. There are also different screens for login and password change.

However, the most important novelty is the introduction of password. When the application is launched for the first time, user is presented with a possibility of inputting a password. If the password is correct main menu is shown and previous activity is finished. That way it cannot be navigated back to by pressing the mobiles back button. The user may also choose to log out. In that case all other activities are closed and user is presented with login screen. That way other activities become unavailable until right password is inputted. This is not standard behaviour. In Android one application is usually a one task. This task operates as a stack. If it starts with activity This activity is the only item on the stack. If an activity A launches activity B, the activity B is on top. If the back button is pressed, activity on top (in this case B) is closed and A comes into focus. If user switches to another tack, stack of previous task is preserved. This is why logout operation was programmed. The team does not want unauthorised user to access activities that were left open by the user.

To store the password and change it shared preferences are used. They store pairs of values - a name that identify each of them and a value itself. They are preserved between the application launches and therefore allow the changed password to persist. They are also accessible only to the application created them (if not stated otherwise during their creation). Usage of shared preferences together with modifying the behaviour of activities greatly increased security of the app. As long as the authorised user remembers to logout, the app cannot be easily accessed.

5 Consideration of ethical issue and design for sustainability

According to the studies made using the eco-audit tool, a remote control made of ABS material and using Alkaline batteries generates 528 MJ of energy and 37.6 kg of CO2 footprint. According to this data, this remote control supplied by Alkaline batteries generates 542 MJ of total energy and 38.5 Kg of total CO2 footprint adding the data related to the manufacture, transport use and disposal. This information is shown in the next graph related to the Relative Contribution of Life Phase (%) of an ABS material using Alkaline batteries.

A remote control made of PLA needs less energy (MJ) than one remote made of ABS material therefore it has a better performance but it has a bigger CO2 Footprint than a one remote made of ABS material. A remote control supplied by Alkaline batteries is the best option for the Eco design of a remote control due to the fact that it is the type of batteries with the lowest CO2 Footprint compared other batteries like Ni-Cd or Li-Ion batteries. In the next figure is shown the CO₂ footprint (kg) of a remote made of PLA using Ni-Cd batteries and another remote control made of ABS and using Alkaline batteries.

Many metals required for production are rare and their mining and processing them have significant impact on environment. The same goes for the batteries. The option chosen by the team is optimal, however there yet is no such a thing as a green accumulator suitable for our system. Luckily there are more and more advances in recycling of such elements as the resources become more precious and legislation in Europe is putting bigger emphasis on recycling. Nevertheless, electronics will almost always have high carbon footprints.

The material for casings is also not very environment-friendly. It is hard to recycle and material obtained in that way will be of lower quality that the casings themselves. Overall the remote itself after being decommissioned have negative impact on the planet. However, it's important to note, that the project will produce only one prototype of internal unit and one prototype for the remote control, consequently their impact will be rather small.

It is also important to note that the system will be used in a field that positively impacts environment. Marine research expands humankind’s understanding of marine ecosystems and how to care for them. Underwater housings such as the one for which the system is developed can also be used to monitor underwater sound
pollution generated by humans. Therefore, system can be indirectly beneficial for the natural habitats of our planet and for their preservation for future generations. A full report generated with Eco Audit tool can be found in Appendix 15.

6 Conclusions and further research

At the end of the semester the team developed a functioning prototype of the proposed system - internal unit and remote that communicate with each other and a mobile application. It was also proved that serial communication between the internal unit and buoy’s hardware is possible. Bluetooth’s popularity was justified as it proved to be ideal for close range communication. It is relatively low power consumption coupled with sufficient bandwidth, high accessibility of hardware and software and rich knowledge base are strong advantages. The team had no major problems with finding libraries and modules and writing software. Other components of the project were Arduino boards. They were proved to be reliable, versatile and very easy to program thanks to the dedicated IDE and powerful community. They are recommended for both teams that have little experience with microcontrollers and those more advanced. The team believe that this choice could not have been better. Android also was not a letdown. All desired features were implemented in the final application. Therefore, the team strongly recommends using mentioned technologies and devices in similar projects. It is also hoped that numerous, smaller comments made previously will be useful.

Probably the most problematic component was the screen. Utilizing it required modification of existing library. A lack of proper documentation from the manufacturer was also an obstacle. After this experience, the team learned that when choosing a component, it is equally important for it to fulfill technical requirements as to make sure it will be easy to integrate with other parts of the project. Time is an important resource and may be worth considering if it can be spend to compensate for example for a lack of proper support.

Relevant to the project is the concept of Internet of Things (IoT). It revolves around devices becoming “smart” and able to communicate and interface with other devices. The software and hardware required is becoming cheaper, more accessible and standardized. Adding connectivity is becoming possible or easy in more and more cases. Joining existing technologies, it the right way results in increased flexibility and potential. It brings us closer to realization of science fiction concepts such as widespread smart houses. It also brings more direct advantages. The internal unit and remote proposed by the team not only work as a switch, but also supply data about the buoy and allow for bidirectional flow of information. The project is not only solving a problem but improving and implementing the way how the device works. It can also be used in other aquatic buoys or similar hardware. Overall in case of connectivity the team suggests to evaluate if adding option to connect wirelessly will enhance the project. Factors such as power consumption, usefulness of data that can be transferred and security may be taken into account. If advantages overbalanced disadvantages, the team recommends usage of technologies that are compatible with those with strong presence on the market and to take into account future development together with idea of IoT.

7 Acknowledgements

The team is thankful to supervisors Joan Castell and Mike Van der Schaar for their guidance. Through the development process they provided team with numerous forms of support. The LAB laboratory in main building of EPSEVG was a place of meetings and work for the team almost since the beginning of the project. Tools and parts there helped greatly in assembling prototypes. The team was also provided with financial support to buy components for the project. Overall the project would not be realized if not for this help.
8 Bibliography


[26] Opara, Best practices for graphic designers: An essential guide to understanding and applying colour design principles.

Appendixes

9.1 Key terms

**Function** - in programming function is a set of instructions identified by unique name. Usually instructions are grouped together in such way that each function perform a specific task. For example, there may be a function that controls the screen. It may take some text as an argument when it is started and display it.

**Variable** - a piece of data that has a specific name that identifies it. It may be one or zero, character, set of characters called strings or many other types of data.

**Structured programming paradigm** - key feature needed to understand code developed for this project is that that structured programming code is based on functions. One function can start another function. So the screen function can be started by function that took user’s input and therefore determined text that should be displayed. The program is composed of a set of functions that depend on and use each other and of variables.

**Object-oriented paradigm** - this paradigm focuses on objects - sets of functions and variables that represent a greater whole. For example, object-oriented software that control an engine will not be comprised of functions that check temperature or oil level. It will rather create opportunity to create an object that represents whole engine. Than this object will have functions that allow to ask it for temperature and oil level. However, everyone outside the object will have no insight into how those functions work, but will only be presented with results.

**Library** - is a set of functions that perform a certain task and are provided separately to be used in programs where they are needed to spare the effort of writing them from scratch. In that case they control a piece of hardware. It can be imagined libraries are like hammers. Hammers are available in shops so when somebody needs to hammer a nail they do not need to make their own hammer.

**Pin** - a connector that is a wire or pad coming out of device. It usually transmits data and power device from power source.

**Pointer** - each variable name identifies its variable that resides in a certain place in the computer's memory. Modifying the variable modifies whatever is in that place in the memory. Pointer on the other hand holds an address that points to a place in the memory. This address can be used to access the variable it is pointing to. However, the value of a pointer can be easily changed to point to other variable as long as it is of the same type.

**IDE** - Integrated Programming Environment is an application or set of application that are used to create, modify and test software.

**API** - Application Programming Interface is a set of functions, protocols and tools to build software.

```c
/*
Function for measuring current using ACS712
*/
```
const int analogIn = A0;
int mVperAmp = 185; // use 100 for 20A Module and 66 for 30A Module
int RawValue= 0;
int ACSoffset = 2500;
double Voltage = 0;
double Amps = 0;

void CurrentSensor(){
    RawValue = analogRead(analogIn);
    Voltage = (RawValue / 1024.0) * 5000; // Gets you mV
    Amps = ((Voltage - ACSoffset) / mVperAmp);

    // print results
    Serial.print("Raw Value = "); // shows pre-scaled value
    Serial.print(RawValue);
    Serial.print(" mV = "); // shows the voltage measured
    Serial.print(Voltage,3); // the '3' after voltage allows you to display
    Serial.print("\t Amps = "); // shows the voltage measured
    Serial.println(Amps,3); // the '3' after voltage allows you to display 3
    delay (2500);
}
/* Function for measuring temperature and humidity using DHT11 */

#include <DHT.h>

#define DHT11Pin 13
DHT dht(DHT11Pin, DHT11);

void TempHum()
{
    // wait 1 seconds between readings
    delay (1000);
    // get humidity
    float humidity = dht.readHumidity();
    // get temperature as C
    float celsius = dht.readTemperature();
    // get temperature as F
    float fahrenheit = dht.readTemperature(true);

    // print results
    Serial.print("Humidity: "); Serial.println(humidity);
    Serial.print(" Celsius: "); Serial.println(celsius);
    Serial.print(" Fahrenheit: "); Serial.println(fahrenheit);
}

/*
Function to read serial incoming data from RS232
*/

#include <SoftwareSerial.h> // import the serial library

SoftwareSerial RS(6, 5); // RX, TX

void RS232()
{
    Serial.begin(9600);
    RS.begin(9600);
    Serial.println("***Welcome - RS232 communication***");
}

void loop()
{
    // Keep reading from RS232 and send to Arduino
    if (RS.available())
        Serial.write(RS.read());

    // Keep reading from Arduino Serial Monitor and send to Arduino
    if (Serial.available())
        RS.write(Serial.read());
}
/**
 * Main code including functions
 */

int RelayPinON = 12;
int RelayPinOFF = 11;
int delayTime = 1000;
bool blinkFlag = false;
bool serialReadingFinished = false;
String serialInput;

void setup() {
    // put your setup code here, to run once:
    pinMode(RelayPinON, OUTPUT);
    pinMode(RelayPinOFF, OUTPUT);
    Serial.begin(9600);
    serialInput.reserve(50);
}

void loop() {
    // put your main code here, to run repeatedly:

    if (serialReadingFinished){
        if (serialInput.startsWith("TEMP")) {
            TempHum();
        }
        if (serialInput.startsWith("CURRENT")) {
            Current();
        }
        if (serialInput.startsWith("RS232")) {
            RS232();
        }
        if (serialInput.startsWith("BLINK ")) {
            //takes the string from the 6th element onwards
            serialInput = serialInput.substring(6);
            if (serialInput.startsWith("ON")) blinkFlag = true;
            else if (serialInput.startsWith("OFF")) blinkFlag = false;
            RelayOff();
        }
    }

    serialReadingFinished = false;
    serialInput = "";
}

void turnLEDOnOff() {
    if(blinkFlag) {
        digitalWrite(12, LOW);
    }
}
digitalWrite(12, HIGH);
}
else {
digitalWrite(12, LOW);
digitalWrite(11, HIGH);
}

// runs every time there is data incoming
void serialEvent() {
    while (Serial.available()) {
        // get the new byte:
        char inChar = (char)Serial.read();
        // if the incoming character is a newline, set a flag
        // so the main loop can do something about it:
        if (inChar == '\n') serialReadingFinished = true;
        // add it to the serialInput:
        serialInput += inChar;
    }
}


#include <LCD.h>
#include <LCD_C0220BiZ.h>
#include <ST7036.h>

// Set the pins on the I2C chip used for LCD connections:
//          addr, en,rw,rs,d4,d5,d6,d7,bl,blpol
ST7036 lcd = ST7036 (2, 20, 0x78);  // Set the LCD I2C address

/*uint8_t fullBattery[] = {
B01110,
B11111,
B11111,
B11111,
B11111,
B11111,
B11111,
B11111};

uint8_t mediumBattery[] = {
B01110,
B11111,
B10001,
B10001,
B10001,
B11111,
B11111,
B11111};

uint8_t lowBattery[] = {
B01110,
B11111,
B10001,
B10001,
B10001,
B10001,
B10001,
B11111};

will be useful when we have custom characters once again */

// digital pins 2, 3 and 4 have a pushbuttons attached to it.
int pushButtonUP = 10;
int pushButtonOK = 11;
int pushButtonBACK = 12;

// variables for reading the pushbutton status
int buttonStateOK = 0;
int buttonStateUP = 0;
int buttonStateBACK = 0;

int prevButtonStateOK = 0;
int prevButtonStateUP = 0;
int prevButtonStateBACK = 0;

bool buttonRead = false; // flag used to read only one button press at a time
int batteryLevel = 2; // used to display battery level; 0 - empty, 1 - half, 2 - full

//----------variables used to identify current menu----------
int numberOfItems = 4, positionInMenu = 0; // number of elements in current menu and position of cursor in that menu
void (*currentFunction)(bool OK) = NULL;
String *currentText = NULL; // pointer to the text for current menu

//----------text for the menus----------
String mainMenuText[] = {"option1", "option2", "option3", "Battery"};
String submenu11Text[] = {"option1-1", "option1-2", "option1-3"};
String submenu12Text[] = {"option2-1", "option2-2"};
String submenu13Text[] = {"option3-1"};
String submenu14Text[] = {"Confirm change?"};
String submenu111Text[] = {"option1-1-1", "option1-1-2"};
String submenu112Text[] = {"option1-2-1"};
String submenu113Text[] = {"option1-3-1"};
void mainMenu (bool OK);

void setup()
{
    pinMode(pushButtonOK, INPUT); // make the pushbutton's pin an input
    pinMode(pushButtonUP, INPUT);  // make the pushbutton's pin an input:
    pinMode(pushButtonBACK, INPUT); // make the pushbutton's pin an input:

    pinMode(13, OUTPUT);
    digitalWrite(13, LOW);  // keep RESET LCD pin on HIGH voltage level
    digitalWrite(13, HIGH); // keep RESET LCD pin on HIGH voltage level
    delay(5);
    lcd.init();            // initialise LCD
    lcd.setCursor(0, 3);
    lcd.print("L.A.B. remote");
    lcd.setCursor(2, 6);
    lcd.print("WELCOME");
    delay(30000);
lcd.clear();

//---------create icons of full and empty battery---------
/*lcd.createChar(2, fullBattery);
lcd.createChar(1, mediumBattery);
lcd.createChar(0, lowBattery);*/

currentFunction = &mainMenu;
currentText = mainMenuText;
draw(currentText, numberOfItems, true);
}

void loop()
{
  buttonStateOK = digitalRead(pushButtonOK);
  buttonStateUP = digitalRead(pushButtonUP);
  buttonStateBACK = digitalRead(pushButtonBACK);

  if(prevButtonStateUP != buttonStateUP && buttonStateUP == HIGH)
    up();

  if(prevButtonStateOK != buttonStateOK && buttonStateOK == HIGH)
    currentFunction(true);

  if(prevButtonStateBACK != buttonStateBACK && buttonStateBACK == HIGH)
    currentFunction(false);

  prevButtonStateOK = buttonStateOK;
  prevButtonStateUP = buttonStateUP;
  prevButtonStateBACK = buttonStateBACK;
}

//---------functions that control drawing the menus---------

void draw(String *text, int number, bool isMenu)
{
  lcd.clear();
  if(isMenu){
    for (int i = 0; i < number && i < 4; i++){
      if (i == positionInMenu){
        lcd.setCursor(i,0);
        lcd.print(">" );
      }
      lcd.setCursor(i,1);
      lcd.print(text[i]);
    }

    lcd.setCursor(0, 15);
    if(batteryLevel == 0) lcd.print("(  =)");
    if(batteryLevel == 1) lcd.print("( ==) ");
  }
}
if(batteryLevel == 2) lcd.print("(==)");

// code using custom characters
//lcd.setCursor(0,19);
//lcd.print((char)batteryLevel);
}
else{
  for (int i = 0; i < number && i < 4; i++){
    lcd.setCursor(i,0);
    lcd.print(text[i]);
  }
}
}

void up(){
  if(positionInMenu < (numberOfItems - 1)) positionInMenu++;
  else positionInMenu = 0;
  draw(currentText, numberOfItems, true);
}

//----------function that control going to next menu----------
void goToMenu(void (*function)(bool), String *text, int number){
  currentFunction = function;
  currentText = text;
  numberOfItems = number;
  positionInMenu = 0;
}
//----------functions that control going menu flow----------
void submenu111 (bool OK){
  if (OK){
    switch (positionInMenu){
      case 0:
      {
        String txt[] = "function 1-1-1";
        draw(txt, 1, false);
        delay(2000);
        draw(currentText, numberOfItems, true);
        break;
      }
      case 1:
      {
        String txt[] = "function 1-1-2";
        draw(txt, 1, false);
        delay(2000);
        draw(currentText, numberOfItems, true);
        break;
      }
    }
  }
}
break;
}
}
else{
    goToMenu(submenu11, submenu11Text, 3);
}
draw(currentText, numberOfItems, true);

void submenu12 (bool OK){
    if (OK){
        String txt[] = "function 1-2-1";
        draw(txt, 1, false);
        delay(2000);
        draw(currentText, numberOfItems, true);
    }
    else{
        goToMenu(submenu11, submenu11Text, 3);
    }
    draw(currentText, numberOfItems, true);
}

void submenu13 (bool OK){
    if (OK){
        String txt[] = "function 1-3-1";
        draw(txt, 1, false);
        delay(2000);
        draw(currentText, numberOfItems, true);
    }
    else{
        goToMenu(submenu11, submenu11Text, 3);
    }
    draw(currentText, numberOfItems, true);
}

void submenu1 (bool OK){
    if (OK){
        switch (positionInMenu){
            case 0:
                goToMenu(submenu11, submenu11Text, 2);
                break;
            case 1:
                goToMenu(submenu112, submenu112Text, 1);
                break;
            case 2:
                goToMenu(submenu113, submenu113Text, 1);
                break;
                break;
else{
    goToMenu(mainMenu, mainMenuText, 4);
}
draw(currentText, numberOfItems, true);
}

void submenu12 (bool OK){
    if (OK){
        switch (positionInMenu){
            case 0:
            {
                String txt[] = "function 2-1";
                draw(txt, 1, false);
                delay(2000);
                draw(currentText, numberOfItems, true);
            }
            case 1:
            {
                String txt[] = "function 2-2";
                draw(txt, 1, false);
                delay(2000);
                draw(currentText, numberOfItems, true);
            }
            }
            else{
    goToMenu(mainMenu, mainMenuText, 4);
}
    draw(currentText, numberOfItems, true);
}

void submenu13 (bool OK){
    if (OK){
        String txt[] = "function 3";
        draw(txt, 1, false);
        delay(2000);
        draw(currentText, numberOfItems, true);
    }
    else{
    goToMenu(mainMenu, mainMenuText, 4);
    }
    draw(currentText, numberOfItems, true);
}

void submenu14 (bool OK){
    if (OK){

if (batteryLevel < 2) batteryLevel++; 
else batteryLevel = 0;
String txt[] = "State changed."
; 
draw(txt, 1, false);
delay(2000);
draw(currentText, numberOfItems, true);
}
else{
    goToMenu(mainMenu, mainMenuText, 4);
}
draw(currentText, numberOfItems, true);
}

void mainMenu (bool OK){
if (OK){
    switch (positionInMenu){
        case 0:
            goToMenu(submenu11, submenu11Text, 3);
            break;
        case 1:
            goToMenu(submenu12, submenu12Text, 2);
            break;
        case 2:
            goToMenu(submenu13, submenu13Text, 1);
            break;
        case 3:
            goToMenu(submenu14, submenu14Text, 1);
            break;
    }
    draw(currentText, numberOfItems, true);
}
}
/*
RGB LED functions
*/

int redPin = 4;
int greenPin = 5;
int bluePin = 6;

//uncomment this line if using a Common Anode LED
//#define COMMON_ANODE

void setup()
{
    pinMode(redPin, OUTPUT);
    pinMode(greenPin, OUTPUT);
    pinMode(bluePin, OUTPUT);
}

void red()
{
    setColor(255, 0, 0);  // red
}

void green()
{
    setColor(0, 255, 0);  // green
}

void blue()
{
    setColor(0, 0, 255);  // blue
}

void setColor(int red, int green, int blue)
{
    #ifdef COMMON_ANODE
    red = 255 - red;
    green = 255 - green;
    blue = 255 - blue;
    #endif
    analogWrite(redPin, red);
    analogWrite(greenPin, green);
    analogWrite(bluePin, blue);
}
Manual for prototype application

Following steps explain the process of the application, layout and actions.

Step 1
After opening the app for the first time the only option available is to connect a device (Screenshot 1). Note on the top of the screen missing Bluetooth icon - it is currently off. If the Bluetooth is already on skip to step 3. It is also important to note that Arduino’s Bluetooth module, to which we want to connect must be paired with our Android device before using the app.

Step 2
If Bluetooth is off and “Connect” button is pushed, the app will ask for permission to turn the Bluetooth on as seen on Screenshot 2. Push “yes” to proceed.

Attachment: [Screenshot 1]

Attachment: [Screenshot 2]
Step 3
On the first glance nothing will change. “Connect” will be the only button available. But pushing it again will display list of paired devices as seen on Screenshot 3. If there are no paired devices suitable notification will be displayed. Once again: The Bluetooth module needs to paired before the app is used.

Step 4
After establishing connection proper information will appear set of visible buttons will be changed to this in Screenshot 4. We believe all the buttons are self-explanatory. If the connection was not established, try again or restart the app.

Security feature. The app has an hard coded default password that can be changed once the user access the first time.
European Project Semester 2016
LAB team
Internal Unit PCB
Marco Ferdinando Carlino
European Project Semester 2016
LAB team
Remote schematic
Marco Ferdinando Carlino
European Project Semester 2016
LAB team
Pushbuttons PCB
Marco Ferdinando Carlino
<table>
<thead>
<tr>
<th>Descripción</th>
<th>Cantidad</th>
<th>Precio unitario</th>
<th>Coste</th>
</tr>
</thead>
<tbody>
<tr>
<td>Regulador de conmutación, entrada 6.5 36V dc, Salida 5V, 1A, SW 6664379</td>
<td>1</td>
<td>4,31 € Unidad</td>
<td>4,31 €</td>
</tr>
<tr>
<td>TRACOPOWER TSR 1-24/50 Estado RoHS - Conforme</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Interruptor táctil tipo Botón, Negro, contactos SPST-NA 4791435 TE Connectivty FSM8JH Estado RoHS - Conforme</td>
<td>20</td>
<td>0,107 € Ud.</td>
<td>2,14 €</td>
</tr>
<tr>
<td>Condensador electrolítico de aluminio Panasonic ECEA1HK5010, 1μF, ±20%, 50 V dc, Orificio Pasante, Serie KS 0116997 Panasonic ECEA1HK5010 Estado RoHS - Conforme</td>
<td>15</td>
<td>0,166 € Ud.</td>
<td>2,49 €</td>
</tr>
<tr>
<td>LED Kingbright, 4 pines, Montaje en orificio pasante Tri-color, Azul, Verde, Rojo, 50 °, 5 mm (T-1 3/4) 8614290 Kingbright L-15445SUREQBFZGEC Estado RoHS - Conforme</td>
<td>5</td>
<td>1,632 € Ud.</td>
<td>8,16 €</td>
</tr>
<tr>
<td>Transceptor de línea, MAX232N, 2-TX 2-RX 2-TRX PDIP, EIA/TIA-232-F, RS-232, V.28, 5 V, 16 pines 1951195 Texas Instruments MAX232N Estado RoHS - Conforme</td>
<td>1</td>
<td>1,04 € Unidad</td>
<td>1,04 €</td>
</tr>
<tr>
<td>Relé de enclavamiento, SPDT, 5V dc, Montaje en PCB, Potencia 6803767 TE Connectivty RT314F05 Estado RoHS - Conforme</td>
<td>1</td>
<td>6,99 € Unidad</td>
<td>6,99 €</td>
</tr>
<tr>
<td>Sensor de efecto Hall Allegro Microsystems ACS712ELCTR-20A-T SOIC, 8 pines, alimentación 4,5 5,5 V 6807135 Allegro Microsystems ACS712ELCTR-20A-T Estado RoHS - Conforme</td>
<td>1</td>
<td>4,52 € Unidad</td>
<td>4,52 €</td>
</tr>
<tr>
<td>Código</td>
<td>Descripción</td>
<td>Cantidad</td>
<td>Precio por unidad</td>
</tr>
<tr>
<td>--------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>----------</td>
<td>-------------------</td>
</tr>
<tr>
<td>4152868</td>
<td>Caja de plástico Hammond 1591ESFLBK, 1591, IP54, 191 x 110 x 57mm Hammond 1591ESFLBK Estado RoHS - Conforme</td>
<td>1</td>
<td>0,04 € / Ud.</td>
</tr>
<tr>
<td>0132731</td>
<td>Resistencia de película de carbono, TE Connectivity, 10kΩ, ±5%, 0,5W, Axial, Serie CFR50 0132731 TE Connectivity CFR5010K Estado RoHS - Conforme</td>
<td>10</td>
<td>0,04 € / Ud.</td>
</tr>
<tr>
<td>0148972</td>
<td>Resistencia de película de metal, TE Connectivity, 100kΩ, ±1%, 0,6W, Axial, Serie LR1F 0148972 TE Connectivity LR1F100K Estado RoHS - Conforme</td>
<td>10</td>
<td>0,045 € / Ud.</td>
</tr>
<tr>
<td>6835423</td>
<td>Resistencia de película de metal, Vishay, 180kΩ, ±5%, 1W, Axial, Serie PR01 6835423 Vishay PR0100101803JA100 Estado RoHS - Conforme</td>
<td>10</td>
<td>0,138 € / Ud.</td>
</tr>
<tr>
<td>7077726</td>
<td>Resistencia de película de carbono, RS Pro, 4,7kΩ, ±5%, 0,25W, Axial, Serie RS 7077726 RS Pro RS-Carbon-4k7-5%-0,25W Estado RoHS - Conforme</td>
<td>10</td>
<td>0,021 € / Ud.</td>
</tr>
<tr>
<td>7017639</td>
<td>Resistencia de película de metal, TE Connectivity, 5,76kΩ, ±0.1%, 0,25W, Axial, Serie HOLCO 7017639 TE Connectivity HB5K76BYA Estado RoHS - Conforme</td>
<td>5</td>
<td>0,94 € / Ud.</td>
</tr>
<tr>
<td>5373549</td>
<td>Condensador cerámico multicapa, AVX, 100pF, ±5%, 100 V dc, Orificio Pasante, C0G dieléctrico 5373549 AVX SR151A101JAR Estado RoHS - Conforme</td>
<td>25</td>
<td>0,20 € / Ud.</td>
</tr>
<tr>
<td>5373757</td>
<td>Condensador cerámico multicapa, AVX, 1nF, ±5%, 100 V dc, Orificio Pasante, C0G dieléctrico 5373757 AVX SR211A102JAR Estado RoHS - Conforme</td>
<td>25</td>
<td>0,236 € / Ud.</td>
</tr>
<tr>
<td>7065445</td>
<td>Convertidor dc - dc aislado, Salida 5V dc, 200mA, ±10%, ±1.2% 7065445 TRACOPOWER TME 030SS Estado RoHS - Conforme</td>
<td>1</td>
<td>5,58 € / Unidad</td>
</tr>
<tr>
<td>8195112</td>
<td>Display Monocromo LCD Midas, Alfanumérico, retroiluminación LED, Transmisivo, 4 filas x 20 caracteres, fondo Azul 8195112 Midas MCCCGG42005A6W-BNMLWI Estado RoHS - Conforme</td>
<td>1</td>
<td>29,82 € / Unidad</td>
</tr>
</tbody>
</table>
### Nano 3.0 Atmel Atmega 328 MCU board

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Arduino</strong></td>
<td><strong>A000005</strong></td>
<td><strong>Estado RoHS - Conforme</strong></td>
<td><strong>2</strong></td>
</tr>
<tr>
<td><strong>2 Disponible para entrega en 24/48 horas</strong></td>
<td><strong>18,25 € Unidad</strong></td>
<td><strong>36,50 €</strong></td>
<td></td>
</tr>
</tbody>
</table>

### Módulo Bluetooth, Microchip, Bluetooth 2.1, +4dBm

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Microchip</strong></td>
<td><strong>RN42XVP-I/RM</strong></td>
<td><strong>Estado RoHS - Conforme</strong></td>
<td><strong>2</strong></td>
</tr>
<tr>
<td><strong>2 Temporalmente fuera de stock. Disponible a partir del 29/07/2016, con entrega en 2 día(s) laborable(s).</strong></td>
<td><strong>21,94 € Unidad</strong></td>
<td><strong>43,88 €</strong></td>
<td></td>
</tr>
</tbody>
</table>

### Forma de envío

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Entrega a domicilio (Cuenta crédito o Tarjeta crédito/débito). Ver condiciones - Preguntas Frecuentes (parte inferior izquierda)</strong></td>
<td><strong>Total productos 173,23 €</strong></td>
</tr>
<tr>
<td><strong>Entrega Gratuita</strong></td>
<td><strong>Total del pedido 209,61 €</strong></td>
</tr>
<tr>
<td><strong>IVA 36,38 €</strong></td>
<td></td>
</tr>
</tbody>
</table>

---

902 100 711 rsonline.es 902 100 611

Página 3 de 3
Start

Initialize Serial

15 seconds

Send string

Received string back?

Yes

30 seconds timer. Switch OFF in case of no answer

Connection available. Do you want to communicate?

Check battery status

Warning

Yes

Battery low?

No

Welcome

Menu

Buoy power menu

Check battery

Check status

Is buoy ON?

Yes

Buoy is currently ON and working. Do you want to switch power OFF?

No

Turn buoy power to OFF

Is buoy ON?

Yes

Buoy has to be ON to check status

Temperature HDD

No

Battery status and voltage

Is buoy ON?

No

Turn buoy power to OFF

Buoy is currently OFF and not working. Do you want to switch power ON?

Appendix 3
The input voltage to the board when it is running from external power. Not USB bus power.

Absolute MAX per pin 40mA recommended 20mA

Absolute MAX 200mA for entire package

Analog exclusively Pins

The power sum for each pin’s group should not exceed 100mA
RN41XV & RN42XV Bluetooth Module

Features:
- Fully qualified Bluetooth® version 2.1 Class 1 (RN41XV) and Class 2 (RN42XV) data module, supports version 2.1 + Enhanced Data Rate (EDR)
- Backwards-compatible with Bluetooth version 2.0, 1.2, and 1.1
- Pin compatible with widely used 2 x 10 2-mm socket typically used by 802.15.4 applications
- RN42XV: 26 μA sleep, 3 mA connected, 30 mA transmit
- RN41XV: 30 mA connected, < 10 mA sniff mode
- UART data connection interface
- Supports secure simple pairing (SPP)
- Sustained data rates: 240 Kbps (slave), 300 Kbps (master)
- Embedded Bluetooth stack profiles: SPP and HID profile support as well as GAP, SDP, RFCOMM, and L2CAP protocols
- Bluetooth SIG certified
- Certifications: FCC, IC, CE
- Environmentally friendly, RoHS compliant

Applications:
- Bluetooth replacement for 802.15.4 modules
- Cable replacement
- Barcode scanners/readers
- Measurement and monitoring systems
- Industrial sensors and controls
- Medical devices
- Computer accessories
- Asset tracking

1.0 DESCRIPTION

The RN41XV and RN42XV are small form factor, low-power Bluetooth radio modules offering plug-in compatibility for the widely used 2 x 10 (2-mm) socket typically used by 802.15.4 radio modules.

Based on the popular 2 x 10 (2-mm) socket footprint often found in embedded applications, the Roving Networks’ RN41XV and RN42XV modules provide Bluetooth connectivity in legacy and existing designs that may have been based upon the 802.15.4 standard.

The RN41XV Class 1 Bluetooth module is based on the RN41, and the RN42XV Class 2 Bluetooth module is based on the RN42. These modules are simple to design in and are fully certified, making them a complete embedded Bluetooth solution. The RN42 is functionally compatible with the RN41.

The Class 1 RN41 module has a range up to 100 meters. The Class 2 RN42 module has a range up to 20 meters.
2.0 OVERVIEW

- Pin compatible with 2 x 10 (2-mm) socket
- Baud rate speeds: 1,200 bps up to 921 Kbps, non-standard baud rates can be programmed
- RN41XV: Class 1 radio, 330 feet (100 m) range, +16 dBm output transmitter, -80 dBm typical receive sensitivity
- RN42XV: Class 2 radio, 60 feet (20 m) distance, +4 dBm output transmitter, -80 dBm typical receive sensitivity
- Frequency 2,402 ~ 2,480 MHz
- FHSS/GFSK modulation, 79 channels at 1-MHz intervals
- Secure communications, 128-bit encryption
- Error correction for guaranteed packet delivery
- Configuration via the local UART and over-the-air RF
- Auto-connect master, I/O pin (DTR), and character-based trigger modes

The module’s moisture sensitivity level (MSL) is 1. Table 2-1 shows the module’s size and weight.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>RN41XVC</th>
<th>RN42XVP</th>
<th>RN41XVU</th>
<th>RN42XVU</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Size</td>
<td>24.4 x 29.9 in.</td>
<td>24.4 x 29.0 in.</td>
<td>24.4 x 29.9 in.</td>
<td>24.4 x 29.0 in.</td>
<td>in.</td>
</tr>
<tr>
<td>Weight</td>
<td>5.5 g</td>
<td>5.5 g</td>
<td>5.5 g</td>
<td>5.5 g</td>
<td>g</td>
</tr>
</tbody>
</table>

Table 2-2, Table 2-3, Table 2-4, Table 2-5, Table 2-6, and Table 2-7 provide detailed specifications for the module.

The module’s moisture sensitivity level (MSL) is 1. Table 2-1 shows the module’s size and weight.

### TABLE 2-1: MODULE SIZE & WEIGHT

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Temperature Range (Operating)</td>
<td>-40°C ~ 85°C</td>
</tr>
<tr>
<td>Temperature Range (Storage)</td>
<td>-40°C ~ 85°C</td>
</tr>
<tr>
<td>Relative Humidity (Operating)</td>
<td>≤ 90%</td>
</tr>
<tr>
<td>Relative Humidity (Storage)</td>
<td>≤ 90%</td>
</tr>
</tbody>
</table>

### TABLE 2-2: ENVIRONMENTAL CONDITIONS

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Temperature Range (Operating)</td>
<td>-40°C ~ 85°C</td>
</tr>
<tr>
<td>Temperature Range (Storage)</td>
<td>-40°C ~ 85°C</td>
</tr>
<tr>
<td>Relative Humidity (Operating)</td>
<td>≤ 90%</td>
</tr>
<tr>
<td>Relative Humidity (Storage)</td>
<td>≤ 90%</td>
</tr>
</tbody>
</table>

### TABLE 2-3: RN41XV & RN42XV DIGITAL I/O CHARACTERISTICS

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Min.</th>
<th>Typ.</th>
<th>Max.</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input Logic Level Low</td>
<td>-0.4</td>
<td>-</td>
<td>+0.8</td>
<td>V</td>
</tr>
<tr>
<td>Input Logic Level High</td>
<td>0.7 VDD</td>
<td>-</td>
<td>VDD + 0.4</td>
<td>V</td>
</tr>
<tr>
<td>Output Logic Level Low</td>
<td>-</td>
<td>-</td>
<td>0.2</td>
<td>V</td>
</tr>
<tr>
<td>Output Logic Level High</td>
<td>VDD - 0.2</td>
<td>-</td>
<td>-</td>
<td>V</td>
</tr>
<tr>
<td>All I/O pins (Except Reset) Default to Weak Pull Down</td>
<td>+0.2</td>
<td>+1.0</td>
<td>+5.0</td>
<td>μA</td>
</tr>
</tbody>
</table>

### TABLE 2-4: RN41XV ELECTRICAL CHARACTERISTICS

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Min.</th>
<th>Typ.</th>
<th>Max.</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supply Voltage (DC)</td>
<td>3.0</td>
<td>3.3</td>
<td>3.6</td>
<td>V</td>
</tr>
<tr>
<td>RX Supply Current</td>
<td>35</td>
<td>60</td>
<td>mA</td>
<td></td>
</tr>
<tr>
<td>TX Supply Current</td>
<td>65</td>
<td>100</td>
<td>mA</td>
<td></td>
</tr>
<tr>
<td>Average Power Consumption</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Standby/Idle (Default Settings)</td>
<td>25</td>
<td></td>
<td>mA</td>
<td></td>
</tr>
<tr>
<td>Connected (Normal Mode)</td>
<td>30</td>
<td></td>
<td>mA</td>
<td></td>
</tr>
<tr>
<td>Connected (Low-Power Sniff)</td>
<td>8</td>
<td></td>
<td>mA</td>
<td></td>
</tr>
<tr>
<td>Standby/Idle (Deep Sleep Enabled)</td>
<td>250</td>
<td>2.5</td>
<td>mA</td>
<td></td>
</tr>
</tbody>
</table>

TABLE 2-5, Table 2-6, and Table 2-7 provide detailed specifications for the module.
TABLE 2-5: RN42XV ELECTRICAL CHARACTERISTICS

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Min.</th>
<th>Typ.</th>
<th>Max.</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supply Voltage (DC)</td>
<td>3.0</td>
<td>3.3</td>
<td>3.6</td>
<td>V</td>
</tr>
<tr>
<td><strong>Average Power Consumption</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Radio On (Discovery or Inquiry Window Time), Note 1</td>
<td>40</td>
<td></td>
<td></td>
<td>mA</td>
</tr>
<tr>
<td>Connected Idle (No Sniff)</td>
<td>25</td>
<td></td>
<td></td>
<td>mA</td>
</tr>
<tr>
<td>Connected Idle (Sniff 100 ms)</td>
<td>12</td>
<td></td>
<td></td>
<td>mA</td>
</tr>
<tr>
<td>Connected with Data Transfer</td>
<td>40</td>
<td>45</td>
<td>50</td>
<td>mA</td>
</tr>
<tr>
<td>Deep Sleep Idle Mode</td>
<td>26</td>
<td></td>
<td></td>
<td>μA</td>
</tr>
</tbody>
</table>

**Note 1:** In slave mode, there are bursts of radio on time that vary with the inquiry windows. The average current depends on how you set the inquiry window.

TABLE 2-6: RN41XV RADIO CHARACTERISTICS

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Sensitivity at 0.1% BER</td>
<td>2.402</td>
<td>-</td>
<td>-80</td>
<td>-86</td>
<td>≤ -70</td>
<td>dBm</td>
</tr>
<tr>
<td></td>
<td>2.441</td>
<td>-</td>
<td>-80</td>
<td>-86</td>
<td>dBm</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.480</td>
<td>-</td>
<td>-80</td>
<td>-86</td>
<td>dBm</td>
<td></td>
</tr>
<tr>
<td>RF Transmit Power</td>
<td>2.402</td>
<td>15.0</td>
<td>16.0</td>
<td>16.0</td>
<td>≤ 20</td>
<td>dBm</td>
</tr>
<tr>
<td></td>
<td>2.441</td>
<td>15.0</td>
<td>16.0</td>
<td>16.0</td>
<td>dBm</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.480</td>
<td>15.0</td>
<td>16.0</td>
<td>16.0</td>
<td>dBm</td>
<td></td>
</tr>
<tr>
<td>Initial Carrier Frequency Tolerance</td>
<td>2.402</td>
<td>-</td>
<td>5</td>
<td>75</td>
<td>75</td>
<td>kHz</td>
</tr>
<tr>
<td></td>
<td>2.441</td>
<td>-</td>
<td>5</td>
<td>75</td>
<td>kHz</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.480</td>
<td>-</td>
<td>5</td>
<td>75</td>
<td>kHz</td>
<td></td>
</tr>
<tr>
<td>20 dB Bandwidth for Modulated Carrier</td>
<td>-</td>
<td>900</td>
<td>1,000</td>
<td>1,000</td>
<td>≤ 1,000</td>
<td>kHz</td>
</tr>
<tr>
<td>Drift (Five Slots Packet)</td>
<td>-</td>
<td>15</td>
<td>-</td>
<td>-</td>
<td>40</td>
<td>kHz</td>
</tr>
<tr>
<td>Drift Rate</td>
<td>-</td>
<td>13</td>
<td>-</td>
<td>-</td>
<td>20</td>
<td>kHz</td>
</tr>
<tr>
<td>Δf1avg Maximum Modulation</td>
<td>2.402</td>
<td>140</td>
<td>165</td>
<td>175</td>
<td>&gt; 140</td>
<td>kHz</td>
</tr>
<tr>
<td></td>
<td>2.441</td>
<td>140</td>
<td>165</td>
<td>175</td>
<td>kHz</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.480</td>
<td>140</td>
<td>165</td>
<td>175</td>
<td>kHz</td>
<td></td>
</tr>
<tr>
<td>Δf2avg Minimum Modulation</td>
<td>2.402</td>
<td>140</td>
<td>190</td>
<td>-</td>
<td>115</td>
<td>kHz</td>
</tr>
<tr>
<td></td>
<td>2.441</td>
<td>140</td>
<td>190</td>
<td>-</td>
<td>kHz</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.480</td>
<td>140</td>
<td>190</td>
<td>-</td>
<td>kHz</td>
<td></td>
</tr>
</tbody>
</table>
## TABLE 2-7: RN42XV RADIO CHARACTERISTICS

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Sensitivity at 0.1% BER</td>
<td>2.402</td>
<td>-70</td>
<td>-86</td>
<td>-86</td>
<td>≤ -70</td>
<td>dBm</td>
</tr>
<tr>
<td></td>
<td>2.441</td>
<td>-70</td>
<td>-86</td>
<td>-86</td>
<td>dBm</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.480</td>
<td>-70</td>
<td>-86</td>
<td>-86</td>
<td>dBm</td>
<td></td>
</tr>
<tr>
<td>RF Transmit Power</td>
<td>2.402</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>≤ 4</td>
<td>dBm</td>
</tr>
<tr>
<td></td>
<td>2.441</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>dBm</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.480</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>dBm</td>
<td></td>
</tr>
<tr>
<td>Initial Carrier Frequency Tolerance</td>
<td>2.402</td>
<td>-5</td>
<td>75</td>
<td>75</td>
<td>75</td>
<td>kHz</td>
</tr>
<tr>
<td></td>
<td>2.441</td>
<td>-5</td>
<td>75</td>
<td>75</td>
<td>kHz</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.480</td>
<td>-5</td>
<td>75</td>
<td>75</td>
<td>kHz</td>
<td></td>
</tr>
<tr>
<td>20 dB Bandwidth for Modulated Carrier</td>
<td>-</td>
<td>900</td>
<td>1,000</td>
<td>1,000</td>
<td>≤ 1,000</td>
<td>kHz</td>
</tr>
<tr>
<td>Drift (Five Slots Packet)</td>
<td>-</td>
<td>15</td>
<td>-</td>
<td>40</td>
<td>kHz</td>
<td></td>
</tr>
<tr>
<td>Drift Rate</td>
<td>-</td>
<td>13</td>
<td>-</td>
<td>20</td>
<td>kHz</td>
<td></td>
</tr>
<tr>
<td>Δf1_{avg} Maximum Modulation</td>
<td>2.402</td>
<td>140</td>
<td>165</td>
<td>175</td>
<td>&gt; 140</td>
<td>kHz</td>
</tr>
<tr>
<td></td>
<td>2.441</td>
<td>140</td>
<td>165</td>
<td>175</td>
<td>kHz</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.480</td>
<td>140</td>
<td>165</td>
<td>175</td>
<td>kHz</td>
<td></td>
</tr>
<tr>
<td>Δf2_{avg} Minimum Modulation</td>
<td>2.402</td>
<td>140</td>
<td>190</td>
<td>-</td>
<td>115</td>
<td>kHz</td>
</tr>
<tr>
<td></td>
<td>2.441</td>
<td>140</td>
<td>190</td>
<td>-</td>
<td>kHz</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.480</td>
<td>140</td>
<td>190</td>
<td>-</td>
<td>kHz</td>
<td></td>
</tr>
</tbody>
</table>

*Figure 2-1 shows the module’s dimensions and Table 2-8 describes the pins.*
TABLE 2-8: PIN DESCRIPTION (PART 1 OF 2)

<table>
<thead>
<tr>
<th>Pin Number</th>
<th>Signal Name</th>
<th>Description</th>
<th>Optional Function</th>
<th>Direction</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>VDD_3V3</td>
<td>3.3 V regulated power input to the module.</td>
<td>Power</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>TXD</td>
<td>UART TX, 8 mA drive, 3.3-V tolerant.</td>
<td>From module</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>RXD</td>
<td>UART RX, 3.3 V tolerant.</td>
<td>To module</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>GPIO7</td>
<td>GPIO, 24 mA drive, 3.3 V tolerant/ADC input.</td>
<td>I/O</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>RESET_N</td>
<td>Optional module reset signal (active low), 100 k pull up, apply pulse of at least 160 μs, 3.3 V tolerant.</td>
<td>Input</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>GPIO6</td>
<td>GPIO, 24 mA drive, 3.3-V tolerant/ADC input.</td>
<td>Data TX/RX</td>
<td>From module</td>
</tr>
<tr>
<td>7</td>
<td>GPIO9</td>
<td>GPIO, 24 mA drive, 3.3-V tolerant/ADC input.</td>
<td>I/O</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>GPIO4</td>
<td>GPIO, 24 mA drive, 3.3 V tolerant/ADC input.</td>
<td>I/O</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>GPIO11</td>
<td>GPIO, 8 mA drive, 3.3 V tolerant.</td>
<td>I/O</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>GND</td>
<td>Ground.</td>
<td>Ground</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>GPIO8</td>
<td>GPIO, 8 mA drive, 3.3-V tolerant. The RN41XV and RN42XV drive GPIO8 high on powerup, which overrides software configured powerup values, on GPIO8.</td>
<td>I/O</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>RTS</td>
<td>UART RTS flow control, 8 mA drive, 3.3 V tolerant.</td>
<td>From module</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>GPIO2</td>
<td>GPIO, 24 mA drive, 3.3 V tolerant/ADC input.</td>
<td>I/O</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>Not Used</td>
<td>No connect.</td>
<td>No Connect</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>GPIO5</td>
<td>GPIO, 24 mA drive, 3.3 V tolerant/ADC input.</td>
<td>I/O</td>
<td></td>
</tr>
</tbody>
</table>
3.0 TYPICAL APPLICATION SCHEMATIC

Figure 3-12 shows a typical application schematic. Because the RN41XV and RN42XV are functionally compatible, this diagram applies to both modules. The RN-XV-EK evaluation kit is shown as a reference. The RN-XV-EK evaluation board provides a 2 x 10 (2-mm) socket for both the RN41XV and RN41XV modules. The design offers USB/serial connection to the on-board UART, a reset switch (pin 5), and switch to control GPIO4 (pin 8).

**FIGURE 3-1: APPLICATION SCHEMATIC, Note 1**

---

**TABLE 2-8: PIN DESCRIPTION (PART 2 OF 2)**

<table>
<thead>
<tr>
<th>Pin Number</th>
<th>Signal Name</th>
<th>Description</th>
<th>Optional Function</th>
<th>Direction</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>CTS</td>
<td>UART CTS flow control, 3.3 V tolerant.</td>
<td>To module</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>GPIO3</td>
<td>GPIO, 24 mA drive, 3.3 V tolerant/ADC input.</td>
<td>I/O</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>GPIO7</td>
<td>GPIO, 24 mA drive, 3.3 V tolerant/ADC input.</td>
<td>I/O</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>AIO0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>AIO1</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note 1: The RN-XV-EK evaluation kit is shown as a reference.
4.0 DESIGN CONCERNS

The following sections provide information on designing with the RN41XV and RN42XV module, including radio interference, factory reset, connection status, etc.

4.1 Powering the Module

Apply ONLY 3.3 V ±10% regulated power to pin 1 (VDD) and pin 10 (ground). The module does not have an on-board voltage regulator and MUST be powered from a regulated 3.3 V power supply.

4.2 Reset Circuit

The RN41XV and RN42XV modules contain a 1kΩ pull-up to VCC, and the reset polarity is active low. The module’s reset pin has an optional power-on-reset circuit with a delay, which should only be required if the input power supply has a very slow ramp or tends to bounce or have instability on power up. Often a microcontroller or embedded CPU I/O is available to generate the reset once power is stable. If not, designers can use one of the many low-cost power supervisor chips currently available, such as the MCP809, MCP102/121, and Torex XC61F.

4.3 Factory Reset Using GPIO4

Roving Networks recommends that designers connect GPIO4 (pin 8) to a switch, jumper, or resistor so it can be accessed. This pin can be used to reset the module to its factory default settings, which is critical in situations where the module has been misconfigured. To reset the module to the factory defaults, GPIO4 should be high on power-up and then toggle low, high, low, high with a 1 second wait between the transitions.

4.4 Connection Status

The RN41XV and RN42XV modules have an on-board green LED to indicate the connection status. The connection status LED is located in the lower right corner of the module.

<table>
<thead>
<tr>
<th>LED Status</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blink at 1 Hz</td>
<td>The module is discoverable and waiting for a connection.</td>
</tr>
<tr>
<td>Blink at 10 Hz</td>
<td>The module is in command mode.</td>
</tr>
<tr>
<td>Solid</td>
<td>The module is connected to another device over Bluetooth.</td>
</tr>
</tbody>
</table>
### 5.0 COMPLIANCE INFORMATION

Table 5-1 and Table 5-2 describe the RN41XV and RN42XV module’s compliance information, respectively.

#### TABLE 5-1: RN41XV COMPLIANCE INFORMATION

<table>
<thead>
<tr>
<th>Category</th>
<th>Country</th>
<th>Standard</th>
</tr>
</thead>
<tbody>
<tr>
<td>Radio</td>
<td>USA</td>
<td>FCC CFR47 Part 15 C, para 15.247</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FCC ID: T9J-R41-1</td>
</tr>
<tr>
<td></td>
<td>Europe</td>
<td>EN 300 328-1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EN 300 328-2 2.4GHz</td>
</tr>
<tr>
<td></td>
<td>Canada</td>
<td>IC Canada ID: IC RSS-210 low power comm. device</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6514A-RN411</td>
</tr>
<tr>
<td>EMC</td>
<td>USA</td>
<td>FCC CFR47 Part 15 subclass B</td>
</tr>
<tr>
<td></td>
<td>Europe</td>
<td>EN 55022 Class B radiated</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EN61000-4-2 ESD immunity</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EN61000-4-3 radiated field</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EN61000-4-6 RF immunity</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EN61000-4-8 power magnetic immunity</td>
</tr>
<tr>
<td>Bluetooth</td>
<td>LISTED</td>
<td>B013180</td>
</tr>
<tr>
<td>Environmental</td>
<td>RoHS</td>
<td>RoHS compliant</td>
</tr>
</tbody>
</table>

#### TABLE 5-2: RN42XV COMPLIANCE INFORMATION

<table>
<thead>
<tr>
<th>Category</th>
<th>Country</th>
<th>Standard</th>
</tr>
</thead>
<tbody>
<tr>
<td>Radio</td>
<td>USA</td>
<td>FCC Part 15 Subpart B: 2008 Class B</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FCC CRF Title 47 Part 15 Subpart C</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FCC ID: T9J-RN42</td>
</tr>
<tr>
<td></td>
<td>Europe</td>
<td>ETSI EN 301 489-1 V1.8.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ETSI EN 301 489-17 V2.1.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>ETSI EN 300 328 V1.7.1</td>
</tr>
<tr>
<td></td>
<td>Canada</td>
<td>IC RSS-210 low power comm. device</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Certification Number: 6514A-RN42</td>
</tr>
<tr>
<td>EMC</td>
<td>USA</td>
<td>FCC CFR47 Part 15 subclass B</td>
</tr>
<tr>
<td></td>
<td>Europe</td>
<td>EN 55022 Class B radiated</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EN61000-4-2 ESD immunity</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EN61000-4-3 radiated field</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EN61000-4-6 RF immunity</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EN61000-4-8 power magnetic immunity</td>
</tr>
<tr>
<td>Bluetooth</td>
<td>BQB LISTED</td>
<td>B014867- SPP and DUN profiles</td>
</tr>
<tr>
<td>Environmental</td>
<td>RoHS</td>
<td>RoHS compliant</td>
</tr>
</tbody>
</table>
6.0 ORDERING INFORMATION

Table 6-1 provides ordering information.

<table>
<thead>
<tr>
<th>Part Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RN41XVC-I/RM</td>
<td>RN41 XV footprint, chip antenna.</td>
</tr>
<tr>
<td>RN41XVU-I/RM</td>
<td>RN41 XV footprint, U.FL. antenna.</td>
</tr>
<tr>
<td>RN42XVP-I/RM</td>
<td>RN42 XV footprint, PCB antenna.</td>
</tr>
<tr>
<td>RN42XVU-I/RM</td>
<td>RN42 XV footprint, U.FL. antenna.</td>
</tr>
</tbody>
</table>

For other configurations, contact Roving Networks directly.

Go to [http://www.rovingnetworks.com](http://www.rovingnetworks.com) for current pricing and a list of distributors carrying Roving Networks products.

Table 6-2 provides information on related products that work with the RN41XV and RN42XV.

<table>
<thead>
<tr>
<th>Part Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RN-XV-EK</td>
<td>Evaluation platform for the RN41XV or RN42XV module. Used for learning and demonstrating WiFi commands and Roving Networks hardware capabilities.</td>
</tr>
<tr>
<td>RN-XV-RD2</td>
<td>Evaluation platform for the RN41XV or RN42XV module. Contains 2 10-amp, 250 V relays.</td>
</tr>
</tbody>
</table>
Temperature and humidity module

DHT11 Product Manual
1. **Product Overview**

DHT11 digital temperature and humidity sensor is a composite sensor containing a calibrated digital signal output of the temperature and humidity. Application of dedicated digital modules collection technology and temperature and humidity sensing technology, to ensure that the product has high reliability and excellent long-term stability. The sensor includes a resistive sense of wet components and an NTC temperature measurement device, and connected with a high-performance 8-bit microcontroller.

2. **Applications**

HVAC, dehumidifier, testing and inspection equipment, consumer goods, automotive, automatic control, data loggers, weather stations, home appliances, humidity regulator, medical and other humidity measurement and control.

3. **Features**

Low cost, long-term stability, relative humidity and temperature measurement, excellent quality, fast response, strong anti-interference ability, long-distance signal transmission, digital signal output, and precise calibration.

4. **Dimensions** (unit: mm)
5. **Product parameters**

Relative humidity  
Resolution: 16Bit  
Repeatability: ±1% RH  
Accuracy: At 25℃ ±5% RH  
Interchangeability: fully interchangeable

Response time: 1 / e (63%) of 25℃ 6s  
1m / s air 6s  
Hysteresis: <± 0.3% RH  
Long-term stability: <± 0.5% RH / yr in

Temperature  
Resolution: 16Bit  
Repeatability: ±0.2℃  
Range: At 25℃ ±2℃  
Response time: 1 / e (63%) 10S

**Electrical Characteristics**

Power supply: DC 3.5 ~ 5.5V  
Supply Current: measurement 0.3mA standby 60μA  
Sampling period: more than 2 seconds

**Pin Description**

1, the VDD power supply 3.5 ~ 5.5V DC  
2, DATA serial data, a single bus  
3, NC, empty pin  
4, GND ground, the negative power
6、Typical circuit

Microprocessor and DHT11 of connection typical application circuit as shown above, DATA pull the microprocessor I / O ports are connected.

1. Typical application circuit recommended in the short cable length of 20 meters on the 5.1K pull-up resistor, the resistance of greater than 20 meters under the pull-up resistor on the lower of the actual situation.

2. When using a 3.5V voltage supply cable length shall not be greater than 20cm. Otherwise, the line voltage drop will cause the sensor power supply shortage, caused by measurement error.

3. Each read out the temperature and humidity values are the results of the last measurement For real-time data, sequential read twice, but is not recommended to repeatedly read the sensors, each read sensor interval is greater than 5 seconds can be obtained accurate data.

7、Serial communication instructions (single–wire bi–directional)

◎ Single bus Description

DHT11 uses a simplified single–bus communication. Single bus that only one data line, the system of data exchange, control by a single bus to complete. Device (master or slave) through an open–drain or tri–state port connected to the data line to allow the device does not send data to release the bus, while other devices use the bus; single bus usually require an external one about 5.1kΩ pull–up resistor, so that when the bus is idle, its status is high. Because they are the master–slave structure, and only when the host calls the slave, the slave can answer, the host access devices must strictly follow the single–bus sequence, if the chaotic sequence, the device will not respond to the host.

◎ Single bus to transfer data defined

DATA For communication and synchronization between the microprocessor and DHT11, single–bus data format, a transmission of 40 data, the high first–out.
Data format:
The 8bit humidity integer data + 8bit the Humidity decimal data + 8bit temperature integer data + 8bit fractional temperature data + 8bit parity bit.

Parity bit data definition
“8bit humidity integer data + 8bit humidity decimal data + 8bit temperature integer data + 8bit temperature fractional data” 8bit checksum is equal to the results of the last eight.

Example 1: 40 data is received:

```
0011 0101
0000 0000
0001 1000
0000 0000
0100 1101
```

High humidity 8 Low humidity 8 High temp. 8 Low temp. 8 Parity bit

Calculate:
0011 0101+0000 0000+0001 1000+0000 0000= 0100 1101

Received data is correct;
Humidity: 0011 0101=35H=53%RH
Temperature: 0001 1000=18H=24℃

Example 2: 40 data is received:

```
0011 0101
0000 0000
0001 1000
0000 0000
0100 1001
```

High humidity 8 Low humidity 8 High temp. 8 Low temp. 8 Parity bit

Calculate:
0011 0101+0000 0000+0001 1000+0000 0000= 0100 1101

01001101≠0100 1001
The received data is not correct, give up, to re–receive data.

Data Timing Diagram
User host (MCU) to send a signal, DHT11 converted from low–power mode to high–speed mode, until the host began to signal the end of the DHT11 send a response signal to send 40bit data, and trigger a letter collection. The signal is sent as shown.

Note: The host reads the temperature and humidity data from DHT11 always the last measured value, such as twice the measured interval of time is very long, continuous read twice to the second value of real–time temperature and humidity values.
Peripherals read steps

Communication between the master and slave can be done through the following steps (peripherals (such as microprocessors) read DHT11 the data of steps).

Step 1:

After power on DHT11 (DHT11 on after power to wait 1S across the unstable state during this period can not send any instruction), the test environment temperature and humidity data, and record the data, while DHT11 the DATA data lines pulled by pull–up resistor has been to maintainhigh; the DHT11 the DATA pin is in input state, the moment of detection of external signals.

Step 2:

Microprocessor I / O set to output at the same time output low, and low hold time can not be less than 18ms, then the microprocessor I / O is set to input state, due to the pull–up resistor, a microprocessor/ O DHT11 the dATA data lines also will be high, waiting DHT11 to answer signal, send the signal as shown:

Step 3:

DATA pin is detected to an external signal of DHT11 low, waiting for external signal low end the delay DHT11 DATA pin in the output state, the output low of 80 microseconds as the response signal, followed by the output of 80 micro–seconds of high notification peripheral is ready to receive data, the microprocessor I / O at this time in the input state is detected the I / O low (DHT11 response signal), wait 80 microseconds highdata receiving and sending signals as shown:

Step 4:

Output by DHT11 the DATA pin 40, the microprocessor receives 40 data bits of data '0' format: the low level of 50 microseconds and 26–28 microseconds according to the changes in the I / O levellevel, bit data "1" format: the high level of low plus, 50 microseconds to 70 microseconds. Bit data '0', '1' signal format as shown:
End signal:
Continue to output the low 50 microseconds after DHT11 the DATA pin output 40 data, and changed the input state, along with pull–up resistor goes high. But DHT11 internal re–test environmental temperature and humidity data, and record the data, waiting for the arrival of the external signal.

8. Application of information
1. Work and storage conditions
Outside the sensor the proposed scope of work may lead to temporary drift of the signal up to 300%RH. Return to normal working conditions, sensor calibration status will slowly toward recovery. To speed up the recovery process may refer to "resume processing". Prolonged use of non–normal operating conditions, will accelerate the aging of the product.

Avoid placing the components on the long–term condensation and dry environment, as well as the following environment.
A, salt spray
B, acidic or oxidizing gases such as sulfur dioxide, hydrochloric acid

Recommended storage environment
Temperature: 10 ~ 40 ℃ Humidity: 60% RH or less

2. The impact of exposure to chemicals
The capacitive humidity sensor has a layer by chemical vapor interference, the proliferation of chemicals in the sensing layer may lead to drift and decreased sensitivity of the measured values. In a pure environment, contaminants will slowly be released. Resume processing as described below will accelerate this process. The high concentration of chemical pollution (such as ethanol) will lead to the complete damage of the sensitive layer of the sensor.

3. The temperature influence
Relative humidity of the gas to a large extent dependent on temperature. Therefore, in the measurement of humidity, should be to ensure that the work of the humidity sensor at the same temperature. With the release of heat of electronic components share a printed circuit board, the installation should be as far as possible the sensor away from the electronic components and mounted below the heat source, while maintaining good ventilation of the enclosure. To reduce the thermal conductivity sensor and printed circuit board copper plating should be the smallest possible, and leaving a gap between the two.

4. Light impact
Prolonged exposure to sunlight or strong ultraviolet radiation, and degrade performance.
5. Resume processing
Placed under extreme working conditions or chemical vapor sensor, which allows it to return to the status of calibration by the following handler. Maintain two hours in the humidity conditions of 45°C and <10% RH (dry); followed by 20–30°C and >70% RH humidity conditions to maintain more than five hours.

6. Wiring precautions
The quality of the signal wire will affect the quality of the voltage output, it is recommended to use high quality shielded cable.

7. Welding information
Manual welding, in the maximum temperature of 300°C under the conditions of contact time shall be less than 3 seconds.

8. Product upgrades
Details, please the consultation Aosong electronics department.

9. The license agreement
Without the prior written permission of the copyright holder, shall not in any form or by any means, electronic or mechanical (including photocopying), copy any part of this manual, nor shall its contents be communicated to a third party. The contents are subject to change without notice.

The Company and third parties have ownership of the software, the user may use only signed a contract or software license.

10. Warnings and personal injury
This product is not applied to the safety or emergency stop devices, as well as the failure of the product may result in injury to any other application, unless a particular purpose or use authorized. Installation, handling, use or maintenance of the product refer to product data sheets and application notes. Failure to comply with this recommendation may result in death and serious personal injury. The Company will bear all damages resulting personal injury or death, and waive any claims that the resulting subsidiary company managers and employees and agents, distributors, etc. that may arise, including: a variety of costs, compensation costs, attorneys' fees, and so on.

11. Quality Assurance
The company and its direct purchaser of the product quality guarantee period of three months (from the date of delivery). Publishes the technical specifications of the product data sheet shall prevail. Within the warranty period, the product was confirmed that the quality is really defective, the company will provide free repair or replacement. The user must satisfy the following conditions:

① The product is found defective within 14 days written notice to the Company;
② The product shall be paid by mail back to the company;
③ The product should be within the warranty period.

The Company is only responsible for those used in the occasion of the technical condition of the product defective product. Without any guarantee, warranty or written statement of its products used in special applications. Company for its products applied to the reliability of the product or circuit does not make any commitment.
Features and Benefits

- Low-noise analog signal path
- Device bandwidth is set via the new FILTER pin
- 5 µs output rise time in response to step input current
- 80 kHz bandwidth
- Total output error 1.5% at $T_A = 25^\circ C$
- Small footprint, low-profile SOIC8 package
- 1.2 mΩ internal conductor resistance
- 2.1 kVRMS minimum isolation voltage from pins 1-4 to pins 5-8
- 5.0 V, single supply operation
- 66 to 185 mV/A output sensitivity
- Output voltage proportional to AC or DC currents
- Factory-trimmed for accuracy
- Extremely stable output offset voltage
- Nearly zero magnetic hysteresis
- Ratiometric output from supply voltage

Description

The Allegro™ ACS712 provides economical and precise solutions for AC or DC current sensing in industrial, commercial, and communications systems. The device package allows for easy implementation by the customer. Typical applications include motor control, load detection and management, switch-mode power supplies, and overcurrent fault protection. The device is not intended for automotive applications.

The device consists of a precise, low-offset, linear Hall circuit with a copper conduction path located near the surface of the die. Applied current flowing through this copper conduction path generates a magnetic field which the Hall IC converts into a proportional voltage. Device accuracy is optimized through the close proximity of the magnetic signal to the Hall transducer. A precise, proportional voltage is provided by the low-offset, chopper-stabilized BiCMOS Hall IC, which is programmed for accuracy after packaging.

The output of the device has a positive slope ($>V_{OUT(min)}$) when an increasing current flows through the primary copper conduction path (from pins 1 and 2, to pins 3 and 4), which is the path used for current sampling. The internal resistance of this conductive path is 1.2 mΩ typical, providing low power loss. The thickness of the copper conductor allows survival of

Typical Application

Application 1. The ACS712 outputs an analog signal, $V_{OUT}$, that varies linearly with the uni- or bi-directional AC or DC primary sampled current, $I_P$, within the range specified. $C_F$ is recommended for noise management, with values that depend on the application.
**ACS712**

**Fully Integrated, Hall Effect-Based Linear Current Sensor IC with 2.1 kV RMS Isolation and a Low-Resistance Current Conductor**

**Description (continued)**

The ACS712 is provided in a small, surface mount SOIC8 package. The leadframe is plated with 100% matte tin, which is compatible with standard lead (Pb) free printed circuit board assembly processes. Internally, the device is Pb-free, except for flip-chip high-temperature Pb-based solder balls, currently exempt from RoHS. The device is fully calibrated prior to shipment from the factory.

**Selection Guide**

<table>
<thead>
<tr>
<th>Part Number</th>
<th>Packing*</th>
<th>$T_A$ (°C)</th>
<th>Optimized Range, $I_p$ (A)</th>
<th>Sensitivity, Sens (Typ) (mV/A)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACS712ELCTR-05B-T</td>
<td>Tape and reel, 3000 pieces/reel</td>
<td>-40 to 85</td>
<td>±5</td>
<td>185</td>
</tr>
<tr>
<td>ACS712ELCTR-20A-T</td>
<td>Tape and reel, 3000 pieces/reel</td>
<td>-40 to 85</td>
<td>±20</td>
<td>100</td>
</tr>
<tr>
<td>ACS712ELCTR-30A-T</td>
<td>Tape and reel, 3000 pieces/reel</td>
<td>-40 to 85</td>
<td>±30</td>
<td>66</td>
</tr>
</tbody>
</table>

*Contact Allegro for additional packing options.

**Absolute Maximum Ratings**

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Symbol</th>
<th>Notes</th>
<th>Rating</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supply Voltage</td>
<td>$V_{CC}$</td>
<td></td>
<td>8</td>
<td>V</td>
</tr>
<tr>
<td>Reverse Supply Voltage</td>
<td>$V_{RCC}$</td>
<td></td>
<td>-0.1</td>
<td>V</td>
</tr>
<tr>
<td>Output Voltage</td>
<td>$V_{OUT}$</td>
<td></td>
<td>8</td>
<td>V</td>
</tr>
<tr>
<td>Reverse Output Voltage</td>
<td>$V_{ROUT}$</td>
<td></td>
<td>-0.1</td>
<td>V</td>
</tr>
<tr>
<td>Output Current Source</td>
<td>$I_{OUT(Source)}$</td>
<td></td>
<td>3</td>
<td>mA</td>
</tr>
<tr>
<td>Output Current Sink</td>
<td>$I_{OUT(Sink)}$</td>
<td></td>
<td>10</td>
<td>mA</td>
</tr>
<tr>
<td>Overcurrent Transient Tolerance</td>
<td>$I_p$</td>
<td>1 pulse, 100 ms</td>
<td>100</td>
<td>A</td>
</tr>
<tr>
<td>Nominal Operating Ambient Temperature</td>
<td>$T_A$</td>
<td>Range E</td>
<td>-40 to 85</td>
<td>°C</td>
</tr>
<tr>
<td>Maximum Junction Temperature</td>
<td>$T_J(max)$</td>
<td></td>
<td>165</td>
<td>°C</td>
</tr>
<tr>
<td>Storage Temperature</td>
<td>$T_{stg}$</td>
<td></td>
<td>-65 to 170</td>
<td>°C</td>
</tr>
</tbody>
</table>

**Isolation Characteristics**

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Symbol</th>
<th>Notes</th>
<th>Rating</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dielectric Strength Test Voltage*</td>
<td>$V_{ISO}$</td>
<td>Agency type-tested for 60 seconds per UL standard 60950-1, 1st Edition</td>
<td>2100</td>
<td>VAC</td>
</tr>
<tr>
<td>Working Voltage for Basic Isolation</td>
<td>$V_{WFSI}$</td>
<td>For basic (single) isolation per UL standard 60950-1, 1st Edition</td>
<td>354</td>
<td>VDC or $V_{pk}$</td>
</tr>
<tr>
<td>Working Voltage for Reinforced Isolation</td>
<td>$V_{WFFI}$</td>
<td>For reinforced (double) isolation per UL standard 60950-1, 1st Edition</td>
<td>184</td>
<td>VDC or $V_{pk}$</td>
</tr>
</tbody>
</table>

* Allegro does not conduct 60-second testing. It is done only during the UL certification process.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Specification</th>
</tr>
</thead>
</table>
| Fire and Electric Shock | CAN/CSA-C22.2 No. 60950-1-03  
UL 60950-1:2003  
EN 60950-1:2001 |
Functional Block Diagram

Pin-out Diagram

Terminal List Table

<table>
<thead>
<tr>
<th>Number</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 and 2</td>
<td>IP+</td>
<td>Terminals for current being sampled; fused internally</td>
</tr>
<tr>
<td>3 and 4</td>
<td>IP–</td>
<td>Terminals for current being sampled; fused internally</td>
</tr>
<tr>
<td>5</td>
<td>GND</td>
<td>Signal ground terminal</td>
</tr>
<tr>
<td>6</td>
<td>FILTER</td>
<td>Terminal for external capacitor that sets bandwidth</td>
</tr>
<tr>
<td>7</td>
<td>VIOUT</td>
<td>Analog output signal</td>
</tr>
<tr>
<td>8</td>
<td>VCC</td>
<td>Device power supply terminal</td>
</tr>
</tbody>
</table>
## COMMON OPERATING CHARACTERISTICS

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Symbol</th>
<th>Test Conditions</th>
<th>Min.</th>
<th>Typ.</th>
<th>Max.</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eiectrical Characteristics</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Supply Voltage</td>
<td>$V_{CC}$</td>
<td>$V_{CC} = 5.0$ V, output open</td>
<td>4.5</td>
<td>5.0</td>
<td>5.5</td>
<td>V</td>
</tr>
<tr>
<td>Supply Current</td>
<td>$I_{CC}$</td>
<td>$V_{CC} = 5.0$ V, output open</td>
<td>–</td>
<td>10</td>
<td>13</td>
<td>mA</td>
</tr>
<tr>
<td>Output Capacitance Load</td>
<td>$C_{LOAD}$</td>
<td>VIOUT to GND</td>
<td>–</td>
<td>10</td>
<td>10</td>
<td>nF</td>
</tr>
<tr>
<td>Output Resistive Load</td>
<td>$R_{LOAD}$</td>
<td>VIOUT to GND</td>
<td>4.7</td>
<td>–</td>
<td>–</td>
<td>kΩ</td>
</tr>
<tr>
<td>Primary Conductor Resistance</td>
<td>$R_{PRIMARY}$</td>
<td>$T_A = 25^\circ$ C</td>
<td>–</td>
<td>1.2</td>
<td>–</td>
<td>mΩ</td>
</tr>
<tr>
<td>Rise Time</td>
<td>$t_r$</td>
<td>$I_P = I_P(max)$, $T_A = 25^\circ$ C, $C_{OUT} = open$</td>
<td>–</td>
<td>3.5</td>
<td>–</td>
<td>μs</td>
</tr>
<tr>
<td>Frequency Bandwidth</td>
<td>$f$</td>
<td>$-3$ dB, $T_A = 25^\circ$ C, $I_P$ is 10 A peak-to-peak</td>
<td>–</td>
<td>80</td>
<td>–</td>
<td>kHz</td>
</tr>
<tr>
<td>Nonlinearity</td>
<td>$E_{LIN}$</td>
<td>Over full range of $I_P$</td>
<td>–</td>
<td>1.5</td>
<td>–</td>
<td>%</td>
</tr>
<tr>
<td>Symmetry</td>
<td>$E_{SYM}$</td>
<td>Over full range of $I_P$</td>
<td>98</td>
<td>100</td>
<td>102</td>
<td>%</td>
</tr>
<tr>
<td>Zero Current Output Voltage</td>
<td>$V_{OUT(Q)}$</td>
<td>Bidirectional; $I_P = 0$ A, $T_A = 25^\circ$ C</td>
<td>–</td>
<td>$V_{CC} \times 0.5$</td>
<td>–</td>
<td>V</td>
</tr>
<tr>
<td>Power-On Time</td>
<td>$t_{PO}$</td>
<td>Output reaches 90% of steady-state level, $T_J = 25^\circ$ C, 20 A present on leadframe</td>
<td>–</td>
<td>35</td>
<td>–</td>
<td>μs</td>
</tr>
<tr>
<td>Magnetic Coupling</td>
<td></td>
<td></td>
<td>–</td>
<td>12</td>
<td>–</td>
<td>G/A</td>
</tr>
<tr>
<td>Internal Filter Resistance</td>
<td>$R_{F(INT)}$</td>
<td></td>
<td>1.7</td>
<td>–</td>
<td>1.7</td>
<td>kΩ</td>
</tr>
</tbody>
</table>

1 Device may be operated at higher primary current levels, $I_P$, and ambient, $T_A$, and internal leadframe temperatures, $T_A$, provided that the Maximum Junction Temperature, $T_J(max)$, is not exceeded.

2 $1G = 0.1$ mT.

3 $R_{F(INT)}$ forms an RC circuit via the FILTER pin.

## COMMON THERMAL CHARACTERISTICS

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Symbol</th>
<th>Test Conditions</th>
<th>Min.</th>
<th>Typ.</th>
<th>Max.</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating Internal Leadframe Temp</td>
<td>$T_A$</td>
<td>$E$ range</td>
<td>–40</td>
<td>–</td>
<td>85</td>
<td>°C</td>
</tr>
<tr>
<td>Junction-to-Lead Thermal Resistance</td>
<td>$R_{JL}$</td>
<td>Mounted on the Allegro ASEK 712 evaluation board</td>
<td>5</td>
<td>°C/W</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Junction-to-Ambient Thermal Resistance</td>
<td>$R_{JA}$</td>
<td>Mounted on the Allegro 85-0322 evaluation board, includes the power consumed by the board</td>
<td>23</td>
<td>°C/W</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1 Additional thermal information is available on the Allegro website.

2 The Allegro evaluation board has 1500 mm$^2$ of 2 oz. copper on each side, connected to pins 1 and 2, and to pins 3 and 4, with thermal vias connecting the layers. Performance values include the power consumed by the PCB. Further details on the board are available from the Frequently Asked Questions document on our website. Further information about board design and thermal performance also can be found in the Applications Information section of this datasheet.
Fully Integrated, Hall Effect-Based Linear Current Sensor IC
with 2.1 kVRMS Isolation and a Low-Resistance Current Conductor

ACS712

x05B PERFORMANCE CHARACTERISTICS

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Symbol</th>
<th>Test Conditions</th>
<th>Min.</th>
<th>Typ.</th>
<th>Max.</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optimized Accuracy Range</td>
<td>IP</td>
<td>Over full range of IP, TA = 25°C</td>
<td>–</td>
<td>5</td>
<td></td>
<td>A</td>
</tr>
<tr>
<td>Sensitivity</td>
<td>Sens</td>
<td>Peak-to-peak, TA = 25°C, 185 mV/A programmed Sensitivity, C_F = 47 nF, C_OUT = open, 2 kHz bandwidth</td>
<td>180</td>
<td>185</td>
<td>190</td>
<td>mV/A</td>
</tr>
<tr>
<td>Noise</td>
<td>V_NOISE(PP)</td>
<td>–</td>
<td>21</td>
<td></td>
<td></td>
<td>mV</td>
</tr>
<tr>
<td>Zero Current Output Slope</td>
<td>ΔV_OUT(Q)</td>
<td>TA = –40°C to 25°C</td>
<td>–</td>
<td>0.26</td>
<td></td>
<td>mV/C</td>
</tr>
<tr>
<td>Sensitivity Slope</td>
<td>ΔSens</td>
<td>TA = –40°C to 25°C</td>
<td>–</td>
<td>0.054</td>
<td></td>
<td>mV/A</td>
</tr>
<tr>
<td>Total Output Error^2</td>
<td>E_TOT</td>
<td>IP = ±5 A, TA = 25°C</td>
<td>±1.5</td>
<td></td>
<td></td>
<td>%</td>
</tr>
</tbody>
</table>

^1 Device may be operated at higher primary current levels, IP, and ambient temperatures, TA, provided that the Maximum Junction Temperature, T_J(max), is not exceeded.

^2 Percentage of IP, with IP = 5 A. Output filtered.

x20A PERFORMANCE CHARACTERISTICS

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Symbol</th>
<th>Test Conditions</th>
<th>Min.</th>
<th>Typ.</th>
<th>Max.</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optimized Accuracy Range</td>
<td>IP</td>
<td>Over full range of IP, TA = 25°C</td>
<td>–</td>
<td>20</td>
<td></td>
<td>A</td>
</tr>
<tr>
<td>Sensitivity</td>
<td>Sens</td>
<td>Peak-to-peak, TA = 25°C, 100 mV/A programmed Sensitivity, C_F = 47 nF, C_OUT = open, 2 kHz bandwidth</td>
<td>96</td>
<td>100</td>
<td>104</td>
<td>mV/A</td>
</tr>
<tr>
<td>Noise</td>
<td>V_NOISE(PP)</td>
<td>–</td>
<td>11</td>
<td></td>
<td></td>
<td>mV</td>
</tr>
<tr>
<td>Zero Current Output Slope</td>
<td>ΔV_OUT(Q)</td>
<td>TA = –40°C to 25°C</td>
<td>–</td>
<td>0.34</td>
<td></td>
<td>mV/C</td>
</tr>
<tr>
<td>Sensitivity Slope</td>
<td>ΔSens</td>
<td>TA = 25°C to 150°C</td>
<td>–</td>
<td>0.07</td>
<td></td>
<td>mV/A</td>
</tr>
<tr>
<td>Total Output Error^2</td>
<td>E_TOT</td>
<td>IP = ±20 A, TA = 25°C</td>
<td>±1.5</td>
<td></td>
<td></td>
<td>%</td>
</tr>
</tbody>
</table>

^1 Device may be operated at higher primary current levels, IP, and ambient temperatures, TA, provided that the Maximum Junction Temperature, T_J(max), is not exceeded.

^2 Percentage of IP, with IP = 20 A. Output filtered.

x30A PERFORMANCE CHARACTERISTICS

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Symbol</th>
<th>Test Conditions</th>
<th>Min.</th>
<th>Typ.</th>
<th>Max.</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optimized Accuracy Range</td>
<td>IP</td>
<td>Over full range of IP, TA = 25°C</td>
<td>–</td>
<td>30</td>
<td></td>
<td>A</td>
</tr>
<tr>
<td>Sensitivity</td>
<td>Sens</td>
<td>Peak-to-peak, TA = 25°C, 66 mV/A programmed Sensitivity, C_F = 47 nF, C_OUT = open, 2 kHz bandwidth</td>
<td>63</td>
<td>66</td>
<td>69</td>
<td>mV/A</td>
</tr>
<tr>
<td>Noise</td>
<td>V_NOISE(PP)</td>
<td>–</td>
<td>7</td>
<td></td>
<td></td>
<td>mV</td>
</tr>
<tr>
<td>Zero Current Output Slope</td>
<td>ΔV_OUT(Q)</td>
<td>TA = –40°C to 25°C</td>
<td>–</td>
<td>0.35</td>
<td></td>
<td>mV/C</td>
</tr>
<tr>
<td>Sensitivity Slope</td>
<td>ΔSens</td>
<td>TA = 25°C to 150°C</td>
<td>–</td>
<td>0.08</td>
<td></td>
<td>mV/A</td>
</tr>
<tr>
<td>Total Output Error^2</td>
<td>E_TOT</td>
<td>IP = ±30 A, TA = 25°C</td>
<td>±1.5</td>
<td></td>
<td></td>
<td>%</td>
</tr>
</tbody>
</table>

^1 Device may be operated at higher primary current levels, IP, and ambient temperatures, TA, provided that the Maximum Junction Temperature, T_J(max), is not exceeded.

^2 Percentage of IP, with IP = 30 A. Output filtered.
Fully Integrated, Hall Effect-Based Linear Current Sensor IC
with 2.1 kVRMS Isolation and a Low-Resistance Current Conductor

Characteristic Performance
$I_P = 5 \, \text{A}, \text{unless otherwise specified}$

- **Mean Supply Current versus Ambient Temperature**
- **Supply Current versus Supply Voltage**
- **Magnetic Offset versus Ambient Temperature**
- **Nonlinearity versus Ambient Temperature**
- **Mean Total Output Error versus Ambient Temperature**
- **Sensitivity versus Ambient Temperature**
- **Output Voltage versus Sensed Current**
- **Sensitivity versus Sensed Current**
- **0 A Output Voltage versus Ambient Temperature**
- **0 A Output Voltage Current versus Ambient Temperature**
ACS712

Fully Integrated, Hall Effect-Based Linear Current Sensor IC with 2.1 kVRMS Isolation and a Low-Resistance Current Conductor

Characteristic Performance

$\text{IP} = 20 \text{ A, unless otherwise specified}$

**Mean Supply Current versus Ambient Temperature**

**Supply Current versus Supply Voltage**

**Magnetic Offset versus Ambient Temperature**

**Nonlinearity versus Ambient Temperature**

**Mean Total Output Error versus Ambient Temperature**

**Sensitivity versus Ambient Temperature**

**Output Voltage versus Sensed Current**

**Sensitivity versus Sensed Current**

**0 A Output Voltage versus Ambient Temperature**

**0 A Output Voltage Current versus Ambient Temperature**
Fully Integrated, Hall Effect-Based Linear Current Sensor IC with 2.1 kVRMS Isolation and a Low-Resistance Current Conductor

Characteristic Performance
$I_p = 30$ A, unless otherwise specified

Mean Supply Current versus Ambient Temperature

Supply Current versus Supply Voltage

Magnetic Offset versus Ambient Temperature

Nonlinearity versus Ambient Temperature

Mean Total Output Error versus Ambient Temperature

Sensitivity versus Ambient Temperature

Output Voltage versus Sensed Current

Sensitivity versus Sensed Current

0 A Output Voltage versus Ambient Temperature

0 A Output Voltage Current versus Ambient Temperature
Definitions of Accuracy Characteristics

**Sensitivity (\text{Sens})**. The change in device output in response to a 1 A change through the primary conductor. The sensitivity is the product of the magnetic circuit sensitivity (G/A) and the linear IC amplifier gain (mV/G). The linear IC amplifier gain is programmed at the factory to optimize the sensitivity (mV/A) for the full-scale current of the device.

**Noise (V_{NOISE})**. The product of the linear IC amplifier gain (mV/G) and the noise floor for the Allegro Hall effect linear IC (≤1 G). The noise floor is derived from the thermal and shot noise observed in Hall elements. Dividing the noise (mV) by the sensitivity (mV/A) provides the smallest current that the device is able to resolve.

**Linearity (E_{LIN})**. The degree to which the voltage output from the IC varies in proportion to the primary current through its full-scale amplitude. Nonlinearity in the output can be attributed to the saturation of the flux concentrator approximating the full-scale current. The following equation is used to derive the linearity:

\[
100 \left(1 - \frac{\Delta \text{gain} \times \% \text{sat} \left( V_{\text{OUT}_{\text{full-scale amperes}}} - V_{\text{OUT}(Q)} \right)}{2 \left( V_{\text{OUT}_{\text{half-scale amperes}}} - V_{\text{OUT}(Q)} \right)} \right)
\]

where \( V_{\text{OUT}_{\text{full-scale amperes}}} \) = the output voltage (V) when the sampled current approximates full-scale ±I_p.

**Symmetry (E_{SYM})**. The degree to which the absolute voltage output from the IC varies in direct proportion to the primary current when the primary current is zero. For a unipolar supply voltage, \( V_{\text{OUT}(Q)} \) (nominally equal to \( V_{\text{CC}} / 2 \)) can be attributed to the resolution of the Allegro linear IC quiescent voltage trim and thermal drift.

**Quiescent output voltage (\text{V}_{\text{OUT}(Q)})**. The output of the device when the primary current is zero. For a unipolar supply voltage, it nominally remains at \( V_{\text{CC}} / 2 \). Thus, \( V_{\text{CC}} = 5 \text{ V} \) translates into \( V_{\text{OUT}(Q)} = 2.5 \text{ V} \). Variation in \( V_{\text{OUT}(Q)} \) can be attributed to the resolution of the Allegro linear IC quiescent voltage trim and thermal drift.

**Electrical offset voltage (\text{V}_{OE})**. The deviation of the device output from its ideal quiescent value of \( V_{\text{CC}} / 2 \) due to nonmagnetic causes. To convert this voltage to amperes, divide by the device sensitivity, Sens.

**Accuracy (E_{TOT})**. The accuracy represents the maximum deviation of the actual output from its ideal value. This is also known as the total output error. The accuracy is illustrated graphically in the output voltage versus current chart at right.

Accuracy is divided into four areas:
- **0 A at 25°C**. Accuracy at the zero current flow at 25°C, without the effects of temperature.
- **0 A over Δ temperature**. Accuracy at the zero current flow including temperature effects.
- **Full-scale current at 25°C**. Accuracy at the the full-scale current at 25°C, without the effects of temperature.
- **Full-scale current over Δ temperature**. Accuracy at the full-scale current flow including temperature effects.

**Ratiometry**. The ratiometric feature means that its 0 A output, \( V_{\text{OUT}(0)} \), is proportional to its supply voltage, \( V_{\text{CC}} \). The following formula is used to derive the ratiometric change in 0 A output voltage, \( \Delta V_{\text{OUT}(0)} / V_{\text{CC}} \), as:

\[
100 \left( \frac{V_{\text{OUT}(0)} / V_{\text{CC}}}{V_{\text{OUT}(0)} / V_{\text{CC}}} \right)
\]

The ratiometric change in sensitivity, \( \Delta \text{Sens}_{\text{RAT}} / V_{\text{CC}} \), is defined as:

\[
100 \left( \frac{\text{Sens}_{\text{RAT}} / V_{\text{CC}}}{\text{Sens}_{\text{pp}} / V_{\text{CC}}} \right)
\]

**Output Voltage versus Sampled Current**
Accuracy at 0 A and at Full-Scale Current

---

Allegro MicroSystems, LLC
115 Northeast Cutoff
Worcester, Massachusetts 01615-0036 U.S.A.
1.508.853.5000; www.allegromicro.com
Definitions of Dynamic Response Characteristics

Power-On Time ($t_{PO}$). When the supply is ramped to its operating voltage, the device requires a finite time to power its internal components before responding to an input magnetic field. Power-On Time, $t_{PO}$, is defined as the time it takes for the output voltage to settle within ±10% of its steady state value under an applied magnetic field, after the power supply has reached its minimum specified operating voltage, $V_{CC(min)}$, as shown in the chart at right.

Rise time ($t_r$). The time interval between a) when the device reaches 10% of its full scale value, and b) when it reaches 90% of its full scale value. The rise time to a step response is used to derive the bandwidth of the device, in which $f_{-3\,\text{dB}} = 0.35/t_r$. Both $t_r$ and $t_{RESPONSE}$ are detrimentally affected by eddy current losses observed in the conductive IC ground plane.
Chopper Stabilization is an innovative circuit technique that is used to minimize the offset voltage of a Hall element and an associated on-chip amplifier. Allegro patented a Chopper Stabilization technique that nearly eliminates Hall IC output drift induced by temperature or package stress effects. This offset reduction technique is based on a signal modulation-demodulation process. Modulation is used to separate the undesired DC offset signal from the magnetically induced signal in the frequency domain. Then, using a low-pass filter, the modulated DC offset is suppressed while the magnetically induced signal passes through the filter. As a result of this chopper stabilization approach, the output voltage from the Hall IC is desensitized to the effects of temperature and mechanical stress. This technique produces devices that have an extremely stable Electrical Offset Voltage, are immune to thermal stress, and have precise recoverability after temperature cycling.

This technique is made possible through the use of a BiCMOS process that allows the use of low-offset and low-noise amplifiers in combination with high-density logic integration and sample and hold circuits.
Fully Integrated, Hall Effect-Based Linear Current Sensor IC with 2.1 kVRMS Isolation and a Low-Resistance Current Conductor

Typical Applications

Application 2. Peak Detecting Circuit

Application 3. This configuration increases gain to 610 mV/A (tested using the ACS712ELC-05A).

Application 4. Rectified Output. 3.3 V scaling and rectification application for A-to-D converters. Replaces current transformer solutions with simpler ACS circuit. C1 is a function of the load resistance and filtering desired. R1 can be omitted if the full range is desired.

Application 5. 10 A Overcurrent Fault Latch. Fault threshold set by R1 and R2. This circuit latches an overcurrent fault and holds it until the 5 V rail is powered down.
Improving Sensing System Accuracy Using the FILTER Pin

In low-frequency sensing applications, it is often advantageous to add a simple RC filter to the output of the device. Such a low-pass filter improves the signal-to-noise ratio, and therefore the resolution, of the device output signal. However, the addition of an RC filter to the output of a sensor IC can result in undesirable device output attenuation — even for DC signals.

Signal attenuation, \( \Delta V_{ATT} \), is a result of the resistive divider effect between the resistance of the external filter, \( R_F \) (see Application 6), and the input impedance and resistance of the customer interface circuit, \( R_{INTFC} \). The transfer function of this resistive divider is given by:

\[
\Delta V_{ATT} = V_{OUT} \left( \frac{R_{INTFC}}{R_F + R_{INTFC}} \right)
\]

Even if \( R_F \) and \( R_{INTFC} \) are designed to match, the two individual resistance values will most likely drift by different amounts over temperature. Therefore, signal attenuation will vary as a function of temperature. Note that, in many cases, the input impedance, \( R_{INTFC} \), of a typical analog-to-digital converter (ADC) can be as low as 10 kΩ.

The ACS712 contains an internal resistor, a FILTER pin connection to the printed circuit board, and an internal buffer amplifier. With this circuit architecture, users can implement a simple RC filter via the addition of a capacitor, \( C_F \) (see Application 7) from the FILTER pin to ground. The buffer amplifier inside of the ACS712 (located after the internal resistor and FILTER pin connection) eliminates the attenuation caused by the resistive divider effect described in the equation for \( \Delta V_{ATT} \). Therefore, the ACS712 device is ideal for use in high-accuracy applications that cannot afford the signal attenuation associated with the use of an external RC low-pass filter.
Fully Integrated, Hall Effect-Based Linear Current Sensor IC with 2.1 kVRMS Isolation and a Low-Resistance Current Conductor

Package LC, 8-pin SOIC

For Reference Only; not for tooling use (reference MS-012AA)

Dimensions in millimeters

Dimensions exclusive of mold flash, gate burns, and dambar protrusions

Exact case and lead configuration at supplier discretion within limits shown

Terminal #1 mark area

Branding scale and appearance at supplier discretion

Reference land pattern layout (reference IPC7351

SOIC127P600X175-8M); all pads a minimum of 0.20 mm from all adjacent pads; adjust as necessary to meet application process requirements and PCB layout tolerances
Fully Integrated, Hall Effect-Based Linear Current Sensor IC
with 2.1 kVRMS Isolation and a Low-Resistance Current Conductor

Revision History

<table>
<thead>
<tr>
<th>Revision</th>
<th>Revision Date</th>
<th>Description of Revision</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rev. 15</td>
<td>November 16, 2012</td>
<td>Update rise time and isolation, $I_{OUT}$ reference data, patents</td>
</tr>
</tbody>
</table>

For the latest version of this document, visit our website:

www.allegromicro.com

Copyright ©2006-2013, Allegro MicroSystems, LLC

The products described herein are protected by U.S. patents: 5,621,319; 7,598,601; and 7,709,754.

Allegro MicroSystems, LLC reserves the right to make, from time to time, such departures from the detail specifications as may be required to permit improvements in the performance, reliability, or manufacturability of its products. Before placing an order, the user is cautioned to verify that the information being relied upon is current.

Allegro’s products are not to be used in life support devices or systems, if a failure of an Allegro product can reasonably be expected to cause the failure of that life support device or system, or to affect the safety or effectiveness of that device or system.

The information included herein is believed to be accurate and reliable. However, Allegro MicroSystems, LLC assumes no responsibility for its use; nor for any infringement of patents or other rights of third parties which may result from its use.
Power PCB Relay RT1 bistable

- 1 pole 16A, 1 form C (CO) or 1 form A (NO) contact
- Polarized bistable version with 1 or 2 coils
- 5kV/10mm coil-contact
- Reinforced insulation

Typical applications
Battery powered equipment or applications with “memory function”

Approvals
VDE Cert. No. 40007571, UL E214025, cCSAus 1142018

Contact Data

<table>
<thead>
<tr>
<th>Type</th>
<th>Contact</th>
<th>Load</th>
<th>Cycles</th>
</tr>
</thead>
<tbody>
<tr>
<td>IEC 61810</td>
<td>A (NO)</td>
<td>16A, 250VAC resistive, 85°C</td>
<td>30x10³</td>
</tr>
<tr>
<td>RT314</td>
<td>A (NO)</td>
<td>16A, 250VAC resistive, 85°C</td>
<td>10x10³</td>
</tr>
<tr>
<td>UL 508</td>
<td>A/B (NO/NC)</td>
<td>20A, 250VAC, general purpose, 85°C</td>
<td>6x10³</td>
</tr>
<tr>
<td>RT334</td>
<td>A (NO)</td>
<td>16A, 250VAC, general purpose, 85°C</td>
<td>5x10³</td>
</tr>
<tr>
<td>RT314</td>
<td>A (NO)</td>
<td>1hp, 240VAC, 40°C</td>
<td>1x10³</td>
</tr>
</tbody>
</table>

- Reinforced insulation
- Bistable version with 1 or 2 coils
- 5kV/10mm coil-contact

Typical applications
Battery powered equipment or applications with “memory function”

Coil Data, bistable coils

<table>
<thead>
<tr>
<th>Coil code</th>
<th>Rated voltage</th>
<th>Set voltage</th>
<th>Reset voltage</th>
<th>Coil resistance</th>
<th>Rated coil power</th>
</tr>
</thead>
<tbody>
<tr>
<td>bistable 1 coil</td>
<td>A03</td>
<td>3</td>
<td>2.1</td>
<td>1.7</td>
<td>21</td>
</tr>
<tr>
<td>A05</td>
<td>5</td>
<td>3.5</td>
<td>2.8</td>
<td>62</td>
<td>403</td>
</tr>
<tr>
<td>A06</td>
<td>6</td>
<td>4.2</td>
<td>3.3</td>
<td>90</td>
<td>400</td>
</tr>
<tr>
<td>A12</td>
<td>12</td>
<td>8.4</td>
<td>6.6</td>
<td>360</td>
<td>400</td>
</tr>
<tr>
<td>A24</td>
<td>24</td>
<td>16.8</td>
<td>13.2</td>
<td>1440</td>
<td>400</td>
</tr>
</tbody>
</table>

- All figures are given for coil without pre-energization, at ambient temperature +23°C.
- Other coil voltages on request.

Bistable coils - operation

<table>
<thead>
<tr>
<th>Version</th>
<th>1 coil</th>
<th>2 coils</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>A2</td>
<td>A1</td>
</tr>
<tr>
<td>+</td>
<td>-</td>
<td>+</td>
</tr>
<tr>
<td>Contact position not defined at delivery</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Power PCB Relay RT1 bistable (Continued)

<table>
<thead>
<tr>
<th><strong>Insulation Data</strong></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial dielectric strength</td>
<td></td>
</tr>
<tr>
<td>between open contacts</td>
<td>1000V(_{\text{rms}})</td>
</tr>
<tr>
<td>between contact and coil</td>
<td>5000V(_{\text{rms}})</td>
</tr>
<tr>
<td>Clearance/creepage</td>
<td>≥ 10/10mm</td>
</tr>
<tr>
<td>Material group of insulation parts</td>
<td>IIIa</td>
</tr>
<tr>
<td>Tracking index of relay base</td>
<td>PTI 250V</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Other Data (continued)</strong></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Terminal type</td>
<td>PCB-THT, plug-in(^1)</td>
</tr>
<tr>
<td>Weight</td>
<td>14g</td>
</tr>
<tr>
<td>Resistance to soldering heat</td>
<td>THT, IEC 60068-2-20</td>
</tr>
<tr>
<td>RTII - flux proof</td>
<td>270°C/10s</td>
</tr>
<tr>
<td>RTIII - wash tight</td>
<td>290°C/5s</td>
</tr>
<tr>
<td>Packaging/unit</td>
<td>tube/20 pcs., box/500 pcs.</td>
</tr>
</tbody>
</table>

\(^1\) socket available for 1 coil version only, see Accessories.

<table>
<thead>
<tr>
<th><strong>Other Data</strong></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Material compliance: EU RoHS/ELV, China RoHS, REACH, Halogen content</td>
<td>refer to the Product Compliance Support Center at <a href="http://www.te.com/customersupport/rohssupportcenter">www.te.com/customersupport/rohssupportcenter</a></td>
</tr>
<tr>
<td>Ambient temperature</td>
<td>-40 to 85°C</td>
</tr>
<tr>
<td>Category of environmental protection</td>
<td>IEC 61810</td>
</tr>
<tr>
<td>RTII - flux proof, RTIII - wash tight</td>
<td></td>
</tr>
<tr>
<td>Vibration/shock resistance (functional), opening B contact</td>
<td>3/5g</td>
</tr>
<tr>
<td>opening closed A contact</td>
<td>6/15g</td>
</tr>
<tr>
<td>Shock resistance (destructive)</td>
<td>100g</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>Dimensions</strong></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>a) Indicated contact position during or after coil energization with reset voltage.</td>
<td></td>
</tr>
<tr>
<td>b) for 2 coil version only</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>PCB layout / terminal assignment</strong></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>16A, pinning 5mm, 1 form A (NO) contact</td>
<td></td>
</tr>
<tr>
<td>12A, pinning 3.5mm, 1 form A (NO) contact</td>
<td></td>
</tr>
<tr>
<td>16A, pinning 5mm, 1 form C (CO) contact</td>
<td></td>
</tr>
</tbody>
</table>

\(^1\) With the recommended PCB hole sizes a grid pattern from 2.5mm to 2.54mm can be used.
### Power PCB Relay RT1 bistable (Continued)

<table>
<thead>
<tr>
<th>Type</th>
<th>RT Power PCB Relay RT1 bistable</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version</td>
<td>1 12A, pinning 3.5mm, flux proof</td>
</tr>
<tr>
<td>Contact configuration</td>
<td>1 1 form C (CO) contact</td>
</tr>
<tr>
<td>Contact material</td>
<td>4 AgNi 90/10</td>
</tr>
</tbody>
</table>

**Product code structure**

<table>
<thead>
<tr>
<th>Product code</th>
<th>Version</th>
<th>Contacts</th>
<th>Contact material</th>
<th>Coil version</th>
<th>Coil</th>
<th>Part number</th>
</tr>
</thead>
<tbody>
<tr>
<td>RT314A03</td>
<td>16A; pinning 3.5mm, flux proof</td>
<td>1 form C (CO) contact</td>
<td>AgNi 90/10</td>
<td>Bistable 1 coil</td>
<td>3VDC</td>
<td>7-1393239-7</td>
</tr>
<tr>
<td>RT314A06</td>
<td>16A; pinning 5mm, flux proof</td>
<td>1 form C (CO) contact</td>
<td>AgNi 90/10</td>
<td>Bistable 1 coil</td>
<td>5VDC</td>
<td>7-1393239-8</td>
</tr>
<tr>
<td>RT314A12</td>
<td>16A; pinning 5mm, flux proof</td>
<td>1 form C (CO) contact</td>
<td>AgNi 90/10</td>
<td>Bistable 1 coil</td>
<td>6VDC</td>
<td>7-1393239-9</td>
</tr>
<tr>
<td>RT314F03</td>
<td>16A; pinning 5mm, flux proof</td>
<td>1 form C (CO) contact</td>
<td>AgNi 90/10</td>
<td>Bistable 2 coils</td>
<td>12VDC</td>
<td>8-1393239-0</td>
</tr>
<tr>
<td>RT314F05</td>
<td>16A; pinning 5mm, flux proof</td>
<td>1 form C (CO) contact</td>
<td>AgNi 90/10</td>
<td>Bistable 2 coils</td>
<td>3VDC</td>
<td>8-1393239-4</td>
</tr>
<tr>
<td>RT314F06</td>
<td>16A; pinning 5mm, flux proof</td>
<td>1 form C (CO) contact</td>
<td>AgNi 90/10</td>
<td>Bistable 2 coils</td>
<td>5VDC</td>
<td>8-1393239-5</td>
</tr>
<tr>
<td>RT314F12</td>
<td>16A; pinning 5mm, flux proof</td>
<td>1 form C (CO) contact</td>
<td>AgNi 90/10</td>
<td>Bistable 2 coils</td>
<td>6VDC</td>
<td>8-1393239-6</td>
</tr>
<tr>
<td>RT314F24</td>
<td>16A; pinning 5mm, flux proof</td>
<td>1 form C (CO) contact</td>
<td>AgNi 90/10</td>
<td>Bistable 2 coils</td>
<td>24VDC</td>
<td>8-1393239-8</td>
</tr>
<tr>
<td>RT134F12</td>
<td>12A; pinning 3.5mm, flux proof</td>
<td>1 form A (NO) contact</td>
<td>AgNi 90/10</td>
<td>Bistable 2 coils</td>
<td>12VDC</td>
<td>4-1415382-1</td>
</tr>
</tbody>
</table>

Other types on request.

This list represents the most common types and does not show all variants covered by this datasheet.
HAND MEASUREMENTS OF MEN, WOMEN AND CHILDREN

RIGHT HAND AV. MAN

profile of heavy winter gloves A.A.F.

fingers vary

fist circums.
10 4 - 5. M.
11 6 - Av. 12 7 - L. M.

HAND DATA

<table>
<thead>
<tr>
<th>HAND DATA</th>
<th>MEN 2.5 % tile</th>
<th>50. % tile</th>
<th>97.5 % tile</th>
<th>WOMEN 2.5 % tile</th>
<th>50. % tile</th>
<th>97.5 % tile</th>
<th>CHILDREN 6 yr.</th>
<th>8 yr.</th>
<th>11 yr.</th>
<th>14 yr.</th>
</tr>
</thead>
<tbody>
<tr>
<td>hand length</td>
<td>6.6</td>
<td>7.5</td>
<td>8.2</td>
<td>6.2</td>
<td>6.9</td>
<td>7.5</td>
<td>5.1</td>
<td>5.6</td>
<td>6.3</td>
<td>7.0</td>
</tr>
<tr>
<td>hand breadth</td>
<td>3.2</td>
<td>3.5</td>
<td>3.8</td>
<td>2.6</td>
<td>2.9</td>
<td>3.1</td>
<td>2.3</td>
<td>2.5</td>
<td>2.6</td>
<td>—</td>
</tr>
<tr>
<td>3d. finger lg.</td>
<td>4.0</td>
<td>4.5</td>
<td>5.0</td>
<td>3.6</td>
<td>4.0</td>
<td>4.4</td>
<td>2.9</td>
<td>3.2</td>
<td>3.5</td>
<td>4.0</td>
</tr>
<tr>
<td>dorsum lg.</td>
<td>2.8</td>
<td>3.0</td>
<td>3.2</td>
<td>2.6</td>
<td>2.9</td>
<td>3.1</td>
<td>2.2</td>
<td>2.4</td>
<td>2.8</td>
<td>3.0</td>
</tr>
<tr>
<td>thumb length</td>
<td>2.4</td>
<td>2.7</td>
<td>3.0</td>
<td>2.2</td>
<td>2.4</td>
<td>2.6</td>
<td>1.8</td>
<td>2.0</td>
<td>2.2</td>
<td>2.4</td>
</tr>
</tbody>
</table>