BACHELOR IN
INDUSTRIAL ELECTRONIC ENGINEERING

Bachelor's Thesis

"Naval orientation device for coordinate location underwater by GPS"

AUTHOR:

Luis García Gámez

SUPERVISED BY:

Prof. Andrés Roldán Aranda

DEPARTMENT:

Electronics and Computers Technologies

ACADEMIC COURSE: 2021/2022

D. Andrés María Roldán Aranda, Profesor del departamento de Electrónica y Tecnología de los Computadores de la Universidad de Granada, como director del Trabajo Fin de Grado de D. Luis García Gámez,

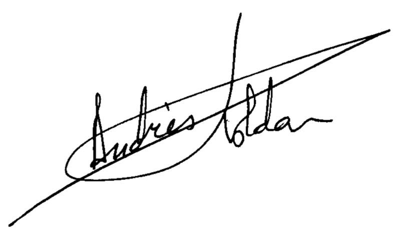
Informa:

Que el presente trabajo, titulado:

"Naval orientation device for coordinate location underwater by GPS"

ha sido realizado y redactado por el mencionado alumno bajo mi dirección, y con esta fecha autorizo a su presentación.

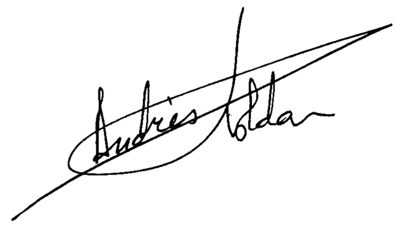
Granada, a 1 de Julio de 2022



Fdo. Prof. Andrés María Roldán Aranda

Los abajo firmantes autorizan a que la presente copia de Trabajo Fin de Grado se ubique en la Biblioteca del Centro y/o departamento para ser libremente consultada por las personas que lo deseen.

Granada, a 1 de Julio de 2022



Fdo. Luis García Gámez

Fdo. Prof. Andrés María Roldán Aranda

“Naval orientation device for coordinate location underwater by GPS”

Luis García Gámez

KEYWORDS:

Electronic, Altium Designer®, EDA, PCB Design, *Pondskater ORCA* bait boat, GPS, lead-acid battery charger, microcontroller, mobile networks BMS

ABSTRACT:

This Bachelor's Thesis is addressed to developing an orientation device to locate underwater areas of Posidonia samples or shipwrecks and cave entrances, as assistance for immersions in a diving club. These areas are often easily visible from the shore, however, once inside the water, finding them might become a challenging task since the water might be murky and the diver will be placed at a point with a worse perspective.

The system will be integrated into the *Pondskater ORCA* bait boat, which can be seen in Figure 1. Initially, this boat was designed to be guided by radio control with a remote, however, it has some errors in its design which make the boat hard to control. Moreover, the lack of functionalities was obvious, nowadays we can count on a huge quantity of different sensors and actuators as well as faster processors which could be included in the device to obtain a better performance and to make it much easier to control, this improvable design is what motivated us to start this project.

For ease of use we will completely redesign its internal electronic so that, in regular operation conditions, the ship will be programmed with coordinates GPS to locate above the underwater target acting as a buoy to allow divers to reach the desired location easily. To achieve this the device will include GPS and GPRS to communicate with the user as well as all the necessary sensors such as consumption sensors, an IMU (Inertial Measurement Unit), a display, etc. It will be based on the ESP32 that will control the propulsion motor and a servo motor for direction control as well as coordinate all the other tasks to obtain a correct operation. In addition, it will include a charging system for lead-acid batteries which will allow the correct charge of its battery and will notify the state of charge to the user (acting as a BMS).

This prototype might be useful as a starting point for similar projects as well as for possible further improvements.

Figure 1 – *Pondskater ORCA*

"Dispositivo de orientación naval para localización de coordenadas submarinas mediante GPS"

Luis García Gámez

PALABRAS CLAVE:

Electrónica, Altium Designer®, EDA, diseño de PCB, barco de cebo *Pondskater ORCA*, GPS, cargador de batería de ácido-plomo, microcontrolador, redes moviles, BMS

RESUMEN:

Este Trabajo de Fin de Grado está dirigido a desarrollar un dispositivo de orientación para localizar áreas submarinas de muestras de Posidonia, naufragios y entradas de cuevas como ayuda para las inmersiones de un club de submarinismo. Estas áreas a menudo son fácilmente visibles desde la orilla, sin embargo, una vez dentro del agua, encontrarlas puede convertirse en una ardua tarea ya que el agua usualmente se encuentra turbia y el buzo se ubicará en un punto con una peor perspectiva.

El sistema se integrará en el barco de cebo *Pondskater ORCA*, que se puede ver en la Figure 2. Inicialmente, este barco fue diseñado para ser guiado por radio control con un control remoto, sin embargo, tiene algunos errores en su diseño que hacen que el barco sea difícil de controlar. Además, la falta de funcionalidades era evidente, hoy en día podemos contar con una gran cantidad de diferentes sensores y actuadores así como procesadores más rápidos que se podrían incluir en el dispositivo para obtener un mejor rendimiento y hacerlo mucho más fácil de controlar, esto fue lo que motivo el desarrollo de una nueva versión del mismo

Para facilitar su uso rediseñaremos por completo su electrónica interna de forma que, en condiciones normales de operación, este se controlará indicando las coordenadas a las que se desea que se dirija para ubicarse sobre el objetivo submarino, actuando así como una boya para permitir que los buzos lleguen fácilmente a la ubicación deseada. Para lograr esto el dispositivo deberá ser capaz de conocer su propia ubicación mediante GPS y acceder a las redes móviles para comunicarse con el usuario, así como hacer uso de otros sensores necesarios como sensores de consumo, una IMU (Inertial Measurement Unit), una pantalla, etc. Todo ello a través de un microcontrolador que controlará el motor de propulsión y un servomotor para el control de dirección además de coordinar todas las demás tareas para obtener un correcto funcionamiento. Aparte de esto, el diseño incluirá un sistema de carga de batería de plomo que permitirá la correcta carga de esta y avisará del estado de carga al usuario (actuando como BMS).

Este prototipo podría ser útil como punto de partida para proyectos similares, así como para posibles mejoras adicionales.

Figure 2 – *Pondskater ORCA*

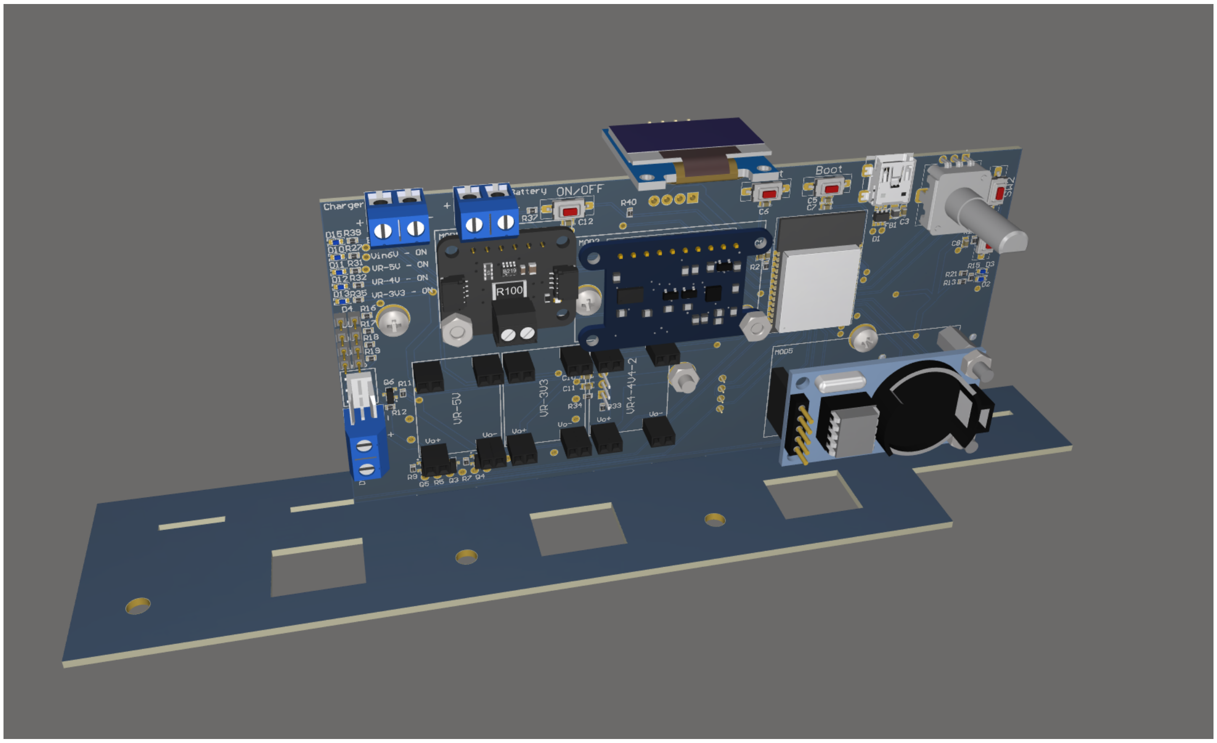


Figure 3 – *3D assembly of the project - front view*

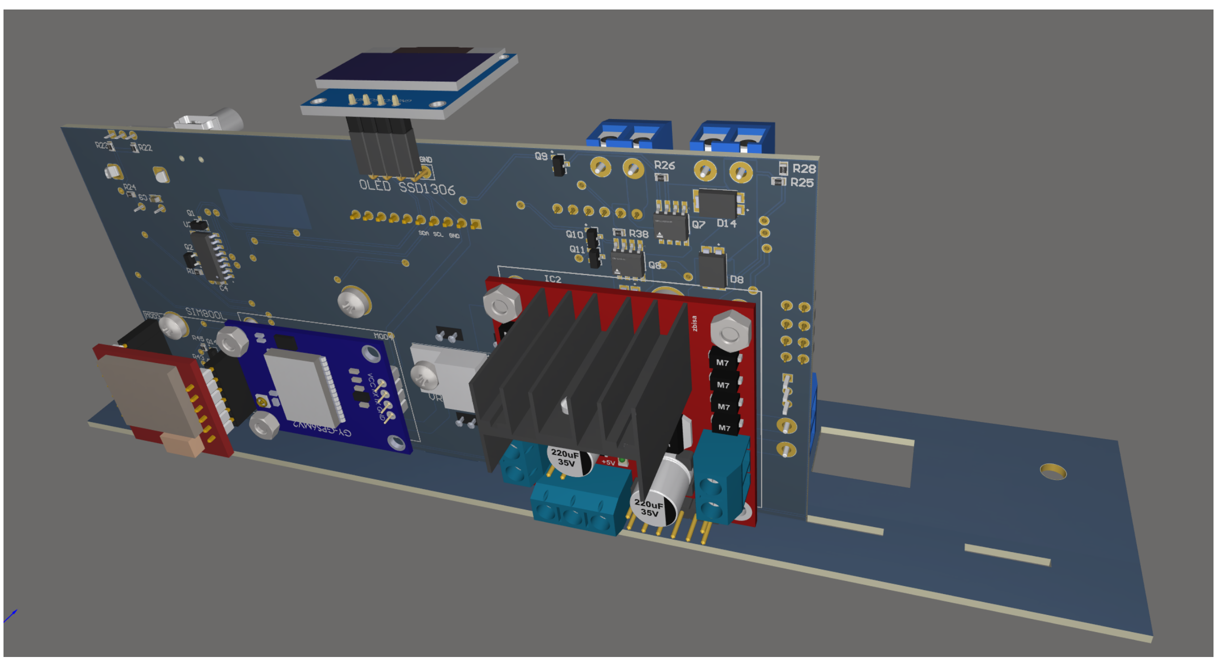


Figure 4 – *3D assembly of the project - back view*

Agradecimientos:

Quiero hacer una mención especial a todas las personas que me han permitido llegar hasta aqui.

A mi familia, por el apoyo que siempre me han mostrado, espcialmente durante el periodo de desarrollo de este proyecto. Vosotros siempre habéis sido un pilar básico que me apoya en mis mejores y peores momentos sin perder la fe en mi, gracias.

A mis compañeros de laboratorio, por las inumerables veces que han estado ahí para echarme una mano cuando parecía no existir una solución a los errores.

A todos los amigos y compañeros de clase que el transcurso de esta larga etapa me ha permitido conocer, ellos también han estado ahí para animarme en los peores momentos y acompañarme en los mejores, de vosotros me quedo con lo mejor de estos años.

Y por supuesto, a mi tutor, Andrés Roldán Aranada, por guiarme y enseñarme las herramientas necesarias para que este proyecto haya sido posible.

Cada uno de los mencionados son partícipes de este Trabajo. A todos ellos, y a los que no pudieron verlo, muchas gracias.

Contents

List of Figures

List of Tables

Glossary

Altium Designer® EDA software utilizado para diseñar PCBs a partir de esquemas. Permite Diseño 3D, así como simulación electrónica.

Altium Designer® EDA software used to design PCBs from schematics. It allows 3D Design, as well as electronics simulation.

Buck-boost converter is a type of DC-to-DC converter that has an output voltage magnitude that is either greater than or less than the input voltage magnitude.

Bypass capacitor Capacitor that shorts AC signals to ground so that any AC noise that may be present on a DC signal is removed, producing a much cleaner and pure DC signal.

C General-purpose computer programming language.

C (current unit) is a current unit used to indicate charge and discharge currents of batteries and that depends on its capacity. 1C is defined as the current needed to fully discharge a battery in one hour, this means that for a 7Ah battery 1C would be equal to 7A while 0.5C would be 3.5A and 2C would be 14A .

C++ General-purpose programming language created as an extension of the C programming language and which includes object-oriented programming as well as several other new functionalities.

Common drain is one of the three basic MOSFET amplifier topologies. In this circuit the gate of the transistor serves as the input, the source is the output and the drain is connected to the power supply .

Crystal oscillator electronic oscillator circuit that operates on the principle of inverse piezoelectric effect in which an alternating voltage applied across the crystal surfaces causes it to vibrate at its natural frequency. It is these vibrations which eventually get converted into oscillations. The oscillator frequency is often used to keep track of time, as in quartz wristwatches, to provide a stable clock signal for digital integrated circuits, and to stabilize frequencies for radio transmitters and receivers [7].

Differential pair A differential signal is one that travels through two conductors, called (+) and (-), instead of doing it through only one, in such a way that the voltages and currents in the conductors are symmetrical. The useful value of the signal is obtained by subtracting the signals of both conductors: V(+) - V(-). Differential signals are used because they are more robust against interference, since the interference adds equally to the signals on both conductors, so when calculating V(+) - V(-), the interference cancels .

EDA (Electronic design automation) Category of software tools focused on the design and production of electronic systems, ranging from the design of integrated circuits to the development of PCBs.

EDA (Electronic design automation) (es) Categoría de herramientas de software para diseñar y producir sistemas electrónicos como circuitos integrados y PCBs.

EEPROM (electrically erasable programmable read-only memory) is a type of non-volatile memory used in computers, integrated in microcontrollers and other electronic devices to store relatively small amounts of data by allowing individual bytes to be erased and reprogrammed [34].

ESD (Electrostatic discharge) Electrostatic phenomenon that causes an electric current to flow suddenly and momentarily between two objects of different electric potential. They are a serious hazard to solid-state electronics as they can render electronic devices useless .

GPIO (general-purpose input/output) Digital signal pin on an integrated circuit or electronic circuit board which may be used as an input or output, or both, and is controllable by software. GPIOs have no predefined purpose and are unused by default [35].

GPRS (General Packet Radio Services) Best-effort packet-switching protocol for wireless and cellular network communication services [36].

GranaSAT GranaSAT is an academic project from the University of Granada originally consisting in designing and developing a picosatellite. Coordinated by the Professor Andrés María Roldán Aranda, GranaSAT is a multidisciplinary project with students from different degrees, where they can acquire and enlarge the knowledge necessary to face an actual aerospace project.

GSM (Global System for Mobile Communications) Standard developed by the European Telecommunications Standards Institute (ETSI) to describe the protocols for second-generation (2G) digital cellular networks used by mobile devices such as mobile phones and tablets [37].

I2C (Inter-Integrated Circuit) Communication interface based in half duplex mode using a master-slave architecture usually with a single master. It is widely used for attaching low-speed peripheral ICs to microcontrollers in short-distance, intra-board communication..

I2S (Inter-IC Sound) Electrical serial bus interface standard used for connecting digital audio devices together.

IMU (Inertial Measurement Unit) Device that can measure and report specific force, angular rate, and the orientation of the object to which it is attached [39]. An IMU typically consists of:

- Gyroscopes: providing a measurement of the angular rate.
- Accelerometers: providing a measurement of the specific force/acceleration.
- Magnetometers (optional): providing a measurement of the magnetic field surrounding the system.

.

Perforated bakelite Material that allows electronic prototyping. During the electronic design process, it is necessary to test certain electronic circuits using this kind of board. This stage is prior to the design of a PCB.

Posidonia Genus of flowering plants, found in the seas of the Mediterranean and around the south coast of Australia.

PWM (Pulse Width modulation) Method of reducing the average power delivered by an electrical signal, by effectively chopping it up into discrete parts. The average value of voltage (and current) fed to the load is controlled by turning the switch between supply and load on and off at a fast rate. The longer the switch is on compared to the off periods, the higher the total power supplied to the load [40]. (see subsubsection 5.2.3.2).

Reverse engineering Process through which it is attempted to understand through deductive reasoning how an already designed device, process or system works or was designed..

SolidWorks is a software specially created for 2D and 3D mechanical modeling.

SPI(Serial Peripheral Interface) Synchronous serial communication interface specification used for short-distance communication, primarily in embedded systems. SPI devices communicate in full duplex mode using a master-slave architecture usually with a single master [41].

Step-down converter (also called buck converter) is a type of DC-to-DC converter which obtains a lower voltage at its output than at its input.

Step-up converter (also called boost converter) is a type of DC-to-DC converter which obtains a higher voltage at its output than at its input .

Switch bouncing The push buttons have a bounce effect when pressed. That is, when it is pressed or released, there is a fluctuation in the signal that passes through its contacts and could cause it to go from a HIGH to a LOW state or vice versa. This can produce an unwanted effect on a microcontroller, like activating an element when what we really wanted is to turn it off, etc. That's due to that the sensor interprets the bounces as if the button had been pressed more than once .

UART (universal asynchronous receiver-transmitter) An individual (or part of an) IC, usually used for serial communications over a computer or peripheral device serial port, in which the data format and transmission speeds are configurable. It sends data bits one by one, from the least significant to the most significant, framed by start and stop bits so that precise timing is handled by the communication channel [42].

Acronyms

AC Alternating current.

ADC Analog to digital converter.

BJT Bipolar junction transistor.

BMS Battery management system.

BOM Bill of materials.

DAC Digital to analog converter.

DC Direct current.

DTR Data terminal ready.

ETSI European Telecommunications Standards Institute.

GPS Global positioning system.

IC Integrated circuit.

LED Light-Emitting Diode.

MCU Microcontroller unit.

MOSFET metal-oxide-semiconductor field-effect transistor.

PCB Printed circuit board.

RTC Real time clock.

RTS Request to send.

SCL Serial clock.

SDA Serial data.

SMD Surface mount device.

UGR University of Granada.

UPS Uninterruptible Power Supply.

USB Universal Serial Bus.

Chapter 1

Introduction

The goal of this thesis is to design an orientation device to locate underwater areas of Posidonia samples or shipwrecks and cave entrances, the project idea is developed at the request of a diving club from Granada's coast to GranaSAT.

GranaSAT is an aerospace development group from the University of Granada (UGR), which is made up entirely of students and under the supervision of the professor Dr. Andrés María Roldán Aranda.

Figure 1.1 – *Granasat logo*

The Granasat project, which logo is shown in Figure 1.1, began as a student initiative. In 2013, several students who were interested in aerospace engineering, and wanted to focus their technical studies in the aerospace scope, decided to participate in the BEXUS/REXUS program, defined as [23]:

Once finished the participation and after the successful work developed for the BEXUS program [13], the Granasat group increased its popularity and became well known among the educational community and for many students who wanted to repeat the deed achieved.

This made it possible to face new engineering challenges, more complex and based on a long-term project.

1.1 Objectives and challenges

Creating a device able to navigate through the sea is a challenging task that requires to be divided into different key goals to be accomplished. Each of them can be developed at different moments since they do not strictly depend on each other, however, it is always necessary to take all of them into consideration whenever a development is made and to consider the project as a unit that will have to be built in a single device.

This device will have to the following **technical objectives**:

- To be able to withstand the different conditions to which it could be exposed in a maritime environment.
- To be able to establish wireless communication through relatively long distances (from the sea to the shore).
- To be able to power up in an environment without easy access to electricity, for what it will be necessary a battery, which entails a challenge in its management, proper charge and safe discharge.
- To be able to navigate, for what it will have to:
  - be able to obtain its own location.
  - be able to know its direction
  - be able to move and change its direction over the sea.
- To have an interface to be controlled by the user.

Moreover, it is also necessary to emphasize that not only are the technical objectives a priority but also there are **goals to achieve as a student**:

- To know the process followed to developmen a product within the framework of engineering.
- To apply all the knowledge learned throughout the 4 years of the degree to design a functional product.
- To learn to search for complementary information to expand that knowledge.
- To learn to create the proper documentation of a project, not only for the panelist but also for those who might continue improving my work in the future.
- To be able to make decisions taking into account all the technical aspects of the different elements.
- To manage the time and schedule in a long-term challenge.
- To add and make good use of the heritage work done before by the GranaSAT team.
- To validate, under time constraints, the best possible model.

1.2 Chapter Description

Following the objectives listed above , the project was developed after planning the next structure:

- **Chapter 2: Previous work.**
  This chapter will describe all the previous knowledge and designs received from my tutor from previous work to understand the work described in the following chapters. The Pondskater OSTRA will be shown and described (2.1) as well as the design previously done (2.2) including a prototype made in a perforated bakelite board.
- **Chapter 3: Verification and testing of the previous work.**
  In this chapter all the test needed to ensure the proper functioning of the device obtained from a previous work will be described, in particular we verified the information and connections shown in the schematics files and tested the prototype (3.2) as well as the different components included in the boat(3.3).
- **Chapter 4: System requirements and constraints.**
  This chapter will set the requirements (Functional requirements, Design requirements, Firmware requirements) and constraints (4.4) the system shall fulfill to achieve the desired purpose.

- **Chapter 5: System description and design.**
  In this chapter the design of the project will be described thoroughly, the final result obtained as well as all the previous unsatisfactory design that the development may bring within.

  A overview of all the schematics(5.2) that define our device will be shown, including mainly the components considered to be included in the design and comparisons between their respective homologous as well as, of course, their interconnections.

  After that, the design of the PCB made to live up the previously mentioned schematics will be described(5.3) as well as how might this design be manufactured in our lab or by a professional manufacturer.

- **Chapter 6: Conclusions, future work and lessons learned.**
  In the last chapter, the conclusions raised after the work (6.1), the lessons learned, future work identified during the development (6.2) and other possible applications (6.3) of this bachelor's thesis is described.

Chapter 2

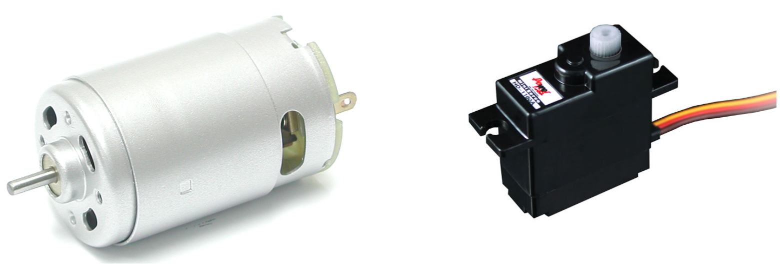
Previous work

2.1 The boat "Pondskater OSTRA"

Our design will depart from the previously designed ***Pondskater ORCA***, which can be seen in Figure 2.4. Initially, this boat was designed to be guided by radio control with a remote, however, it has some errors in its design which make the boat hard to control. Moreover, the lack of functionalities was obvious. Nowadays we count on a huge quantity of different sensors and actuators as well as faster processors which could be included in the device to obtain a better performance and to make it much easier to control.

This boat that we will use as a starting point for our design includes:

1. The chassis.
2. A 6V DC motor to propel the boat.
3. Two servomotors:
    (a) A servomotor for steering control through the rudder.
    (b) A servomotor to control the door of a locker (in which initially bait would be placed).



(a) *DC motor* **(b)** *Servomotor*

Figure 2.1 – *Motors*

4. A 6V 7Ah lead-acid battery, which can be seen in Figure 2.2.

Figure 2.2 – *6V 7Ah lead-acid battery*

5. Seven 5mm position LEDs as those shown in Figure 2.3:

   (a) In the Bow (front part):

      i. A blue LED.

      ii. A red LED.

      iii. A green LED.

   (b) In the stern (back part):

      i. Two white LEDs.

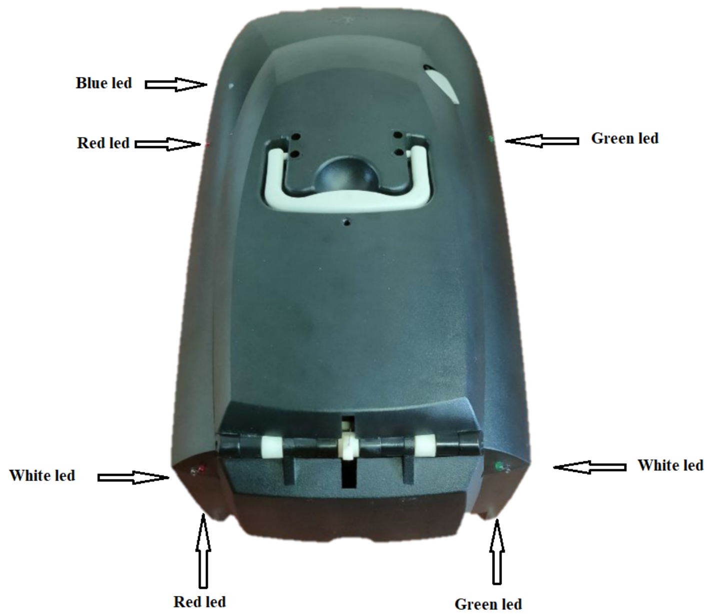
      ii. A red LED.

      iii. A green LED.

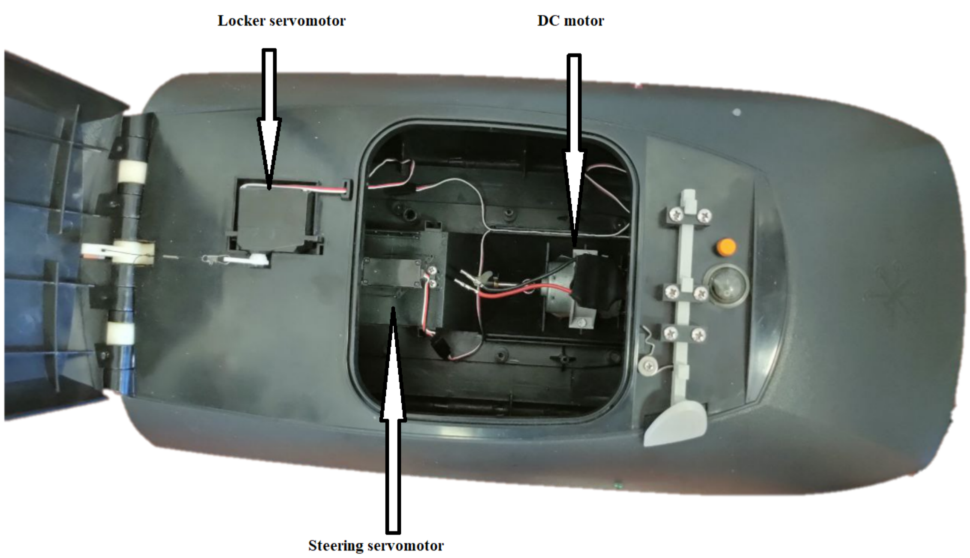
Figure 2.3 – *5mm LEDs*

The layout of the elements can be seen in Figure 2.4. This design delimits our following design considering that we would have to make it according to a 6V power source. We must take into account that using a different battery is always possible, however, it would entail changing the DC motor as well since it works at the same voltage than the battery or using a voltage regulator to adapt it.

2



(a) *Leds layout*



(b) *Motors layout*

Figure 2.4 – *Boat description*

2.2 Previous work at GranaSAT

2.2.1 Prototype for testing

Our first step is to apply reverse engineering methods to find out which elements compound the prototype. As can be observed in Figure 2.6, the prototype made in a perforated bakelite board includes:

- A **ESP32** (datasheet in [11]), a low-cost, low-power general-purpose microcontroller with integrated Wi-Fi and dual-mode Bluetooth as well as quite an extensive set of peripherals.

- A **L7805CV** (datasheet in [25]), a linear voltage regulator (see subsubsection 5.2.8.5 for more information regarding voltage regulators) which will act as a 5 V power supply for the ESP32. This regulator accepts input voltages from the output voltage plus 1 V up to 35 V. It is worth mentioning that the microcontroller works at 3.3V, nevertheless, this module includes a voltage regulator that allows higher voltage supplies to be powered with voltages from 5V to 12V.

- A **LM317A** (datasheet in [31]), an adjustable linear voltage regulator which will supply the SIM800L module, which works at 4 V. This regulator accepts input voltages from 4.25 V up to 40 V over a 1.25 V to 37 V output range. Its output is modified by a tension divider as the one shown in Figure 2.5, obtained from the datasheet.

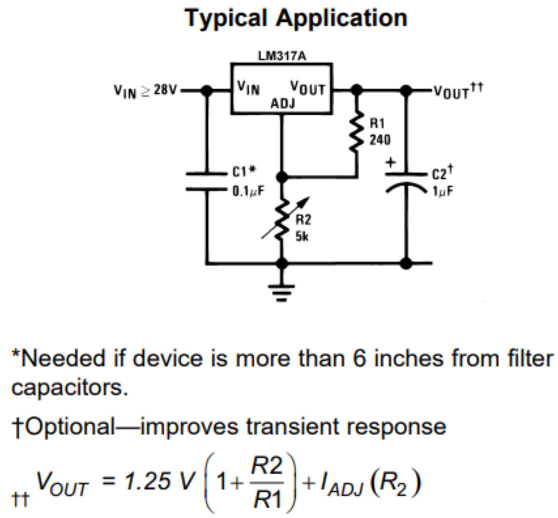


Figure 2.5 – *LM317A typical application*

- A **tension divider** formed by two resistors whose values are 2.2KΩ and 1KΩ and through which the output of the adjustable regulator is set to 4 V.

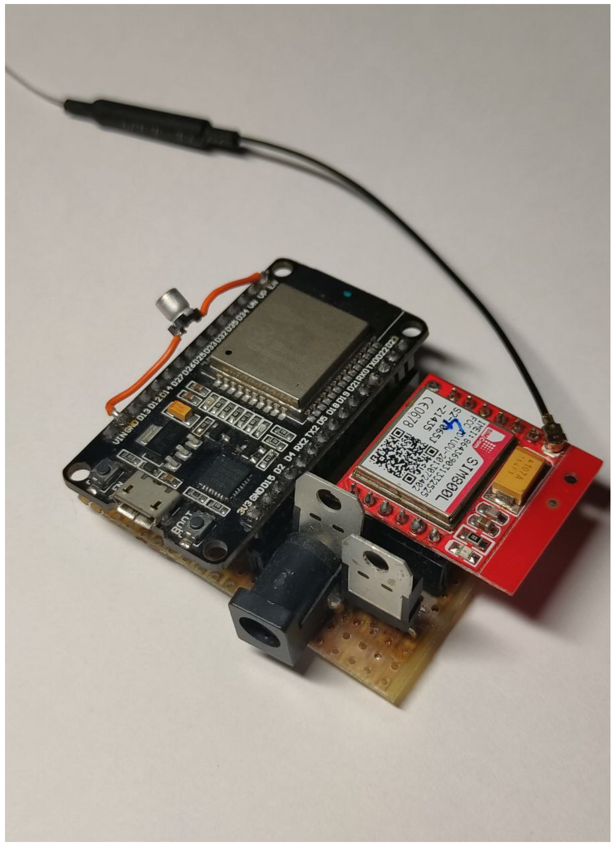
- A **SIM800L** (datasheet in [24]), a module able to establish mobile communications through GSM (Global System for Mobile Communications) and GPRS (General Packet Radio Services) with a SIM slot, which includes an antenna.

- A **1 μF capacitor** connected to the ENABLE and GND pins of the ESP32, which theoretically avoids errors while the program is being loaded.

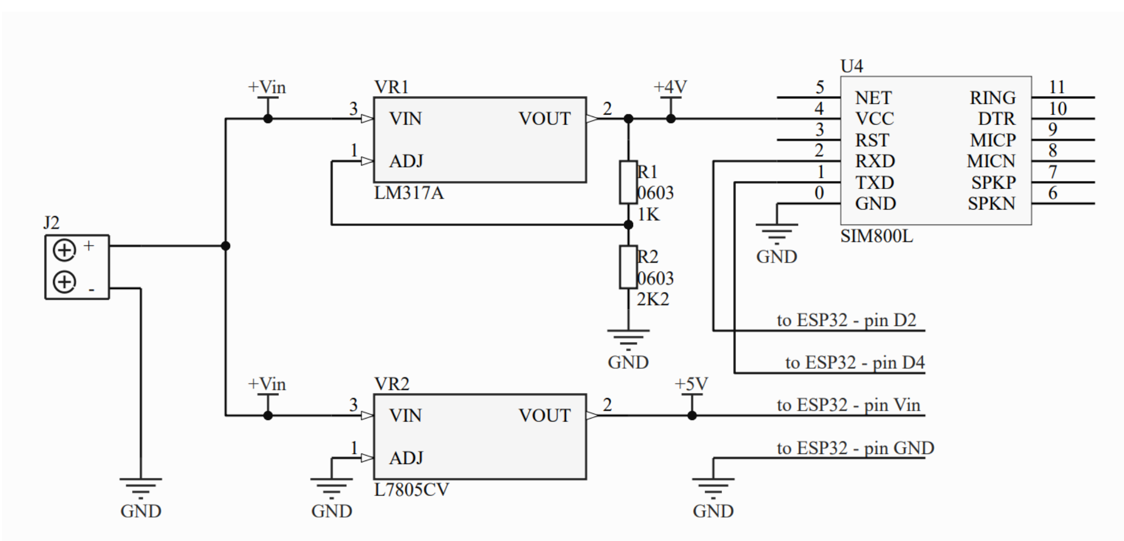
- And finally, a **power connector**.

2

This device, whose **schematic** is shown in Figure 2.6 (b), allows us to start with the first tests and code sketches for our future project.



(a) *Physical appearance*



(b) *Schematic*

Figure 2.6 – *Prototype description*

2.2.2 Sketch and first schematic design

The schematics previously designed for this projects in Altium Designer® are shown in Appendix D.

They are 7 schematic sheets with some of the necessary components for the required functionalities but, as can be observed, they are halfway to be finished. Those components will be described later in the component selection for the design since all of them have been considered and so analyzed for the new design.

Since it was not finished, there was no adequate documentation neither, so we had to use only these schematic designs for the compression of the project and the functionality of the components, which hindered this task.

Chapter 3

Verification and testing of the previous work

3.1 Schematic verification

The fact of being halfway to being finished is notorious in several different aspects (Appendix D). It still lacks the necessary 5V and 3.3V voltage sources as well as some essential features such as the servomotor connections, a GPS, a power switch or a voltage and current sensors. In addition, it has several design errors as for example the RTC, which must be powered at 5V instead of at 3.3V, or the value of the resistors of the voltage divider that configure the output of the LM317A at 4V, which are not correctly chosen. Moreover, there are several ports connected to the ESP32 which are not linked to any component or sheet.

Motivated by these errors, we decided to **review the entire project**, checking such critical factors as whether the supply voltage of each component was correct or whether the connection made with the ESP32 was adequate, among others.

Although the utility of these missing components may not yet be clear, this aspect will be discussed later when delving into the design carried out.

3.2 Prototype testing

After analyzing the previous work to know by what components and connections is this prototype compounded, we can start the first tests.

As we already know that this device can be powered up with voltages up to 40 V, we proceeded to turn it on with the help of an external power supply at 7 V. In this stage we learned our first lesson and it is that the circular power connectors like the one that this prototype includes don't have a standard that define where the positive and the negative pin are connected.

Due to that, we, innocently, used a wire available in the lab to connect the external source to a circular power connector, connecting the red pin with the red wire and the black pin with the black wire without checking first if the internal connection of the prototype corresponded with the internal connection of the connector attached to the wires. This led to a current of several amperes measured by the power source and a slight burning smell. We have just burned one of the voltage regulators.

After this, we were forced to desolder the damaged component to solder a new one, after which the lesson was clear enough and it did not happen again.

Anecdotes aside, we proceeded to **turn it on** again, after changing the polarity of one of the connector, and after observing the indicator LED on we went ahead to the design of a firmware sketch.

Our **first sketch** had the function of **making a LED blink** just to verify that the ESP32 worked properly and could be programmed. This simple sketch is shown in section C.1. Later, after checking the correct operation of the previous sketch, we went ahead to create a more complex firmware, making use of the SIM800L. In this one, we programmed the **ESP32 to make phone calls and to send and receive SMS.** This sketch is shown in section C.2

Although SMS could be used to control the boat through commands, this is a too old-fashioned technology so the most appropriate operation mode for our boat will be to implement a firmware to connect the module to the internet and to transmit and receive data from a mobile application installed in our smartphone. This means that we will have to keep on working on the firmware.

3.3 Boat's components testing

The battery included in the boat is a lead-acid battery as previously commented. Before working with lead-acid batteries we must know that they are especially sensitive to over-discharge, moreover, they are also one of the fastest self-discharging batteries, between 2% and 20% of capacity typically declined per month. Taking both factors into consideration, it seems obvious that this kind of battery needs proper maintenance to avoid degradation and, since we don't know how long the battery has been unused, we concluded that **it was convenient to check the battery condition.**

This battery is 6V 7AH so, in regular cycles, its voltage should move between 5V and 7.5V depending on the discharge current (C) as shown in Figure 3.1.

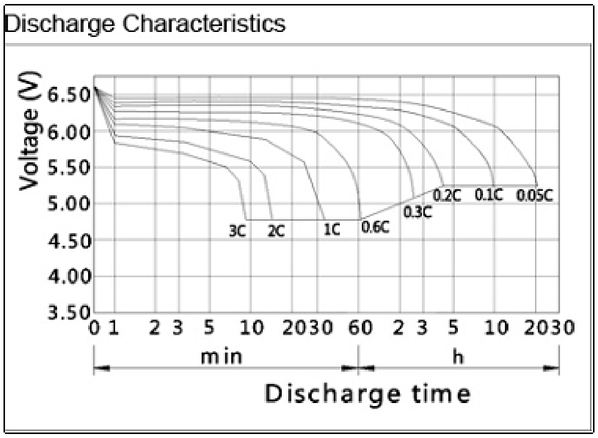


Figure 3.1 – *Discharge characteristic curve of a 6V sealed lead-acid battery*

Knowing that, we can now test the battery voltage with our multimeter, from where we obtained the voltage that can be seen in Figure 3.2, 1.58 V.

As we can see, the voltage is far below a safe value and it might mean that the battery is already broken. Our intention after that was to charge and to discharge it with a VOLTCRAFT ALC 85000 expert, a professional charger for different kinds of batteries, however, the software would not detect the battery if the voltage has such an atypical value. Therefore, we tried to increase the battery voltage "manually" with an external source to at least 4.5 V. To achieve that, the battery was connected to an external power source

Figure 3.2 – *Measured battery voltage*

set at 7V with a wire with a 1 ohm power resistance to avoid shortcircuits.

Once connected, the current measured by the source's ammeter was 0 A, this meant that the battery was not charging. However, as the source ammeter is not really accurate we decided to connect our multimeter to get a more exact measure, obtaining the value shown in Figure 3.3. A current of 40.5 μA is too low to charge a battery, in addition, we measured almost 0 V in the resistance due to the low current and the same voltage in the battery as in the source. All this means that our battery is acting as an open circuit and we are not going to be able to charge it so **it is definitely broken.**

Figure 3.3 – *Measured battery charge current*

Due to that, we didn't have any option left other than buying a new battery.

Last but not least, we tested the position LEDs and the motors which all of them worked properly. Therefore, we could continue with the design of the project.

Chapter 4

System requirements and constraints

As previously mentioned in section 1.1, our final product will have to fulfil a combination of requirements. In this chapter, we will go deeper into this topic to define in more detail how our device should be in order to accomplish the desired purpose in the best way possible.

4.1 Functional requirements

In this section the specific functions that our boat must include are going to be described.

Firstly, a **movement system** is indispensable, which is already included in its chassis. This system is composed of the DC Motor which will be the propeller and the servomotor which will control the steering.

Moreover, to move through the water towards the desired location it is also essential to know where the device is located to be able to calculate a route, this means that a **GPS** module must be included. Nevertheless, this module can online acquire data about its position but not its orientation, that is why an **IMU** (specifically a magnetometer) is necessary as well. In addition, this module can be useful to detect whether the boat is or not moving, for example, in the case of running aground.

On the other hand, the device must include a **wireless communication system** in order to be controlled by the user from relatively long distances as our usage environment entails distances that could be from a few meters to even several kilometres as the ship could move away from the coast.

Besides, the fact of using a battery make it necessary to use certain components for its proper use, control and maintenance as:

- A **battery charger**, which must be specially designed for ours in order to extend its life and duration as the charging process varies among the different types of batteries. In our case, for a lead-acid battery, a 3-phases-based charging process must be followed which consists of a constant current phase, followed by a constant voltage phase, and a float charge phase to end. This process will be explained more deeply in subsubsection 5.2.8.4.
- A **power switch** to turn it on and off when desired both manually or by the microprocessor if any problem is detected or if the state of charge of the battery is too low.
- A **battery voltage and current sensor** to prevent it from dropping to an excessively low value as it might cause irreparable damage to the battery. In addition, a **state of charge estimation** would help to avoid that, which could be obtained from the measurements of the current sensor.

Finally, any boat has to meet some **visibility requirements** imposed by the navigation regulations in Spain and this won't be an exception. The regulation rules that for ships propelled by a motor and smaller than 12 meters it is mandatory to count on a white light on the top of the craft and two Band lights (on the sides): a red one and a green one. The red light indicates that the port (left) side is in view or the green light indicates that the starboard (right) side is in view. [21]

4.2 Design requirements

The design of a product usually includes components that might not be known or accessible to the user but that without them the device would not work properly or the developing phase will become harder. In this section, these complementary functions that must be included to create an operational design will be described.

First of all, it is mandatory to include a **microcontroller** that will act as a "brain" to coordinate all the tasks, acquire the sensors' data, transmit the information to the user and control the motor and other outputs.

In addition, the device needs firmware which means that we must include an **interface** to be able to establish communication between it and a computer or any other device **to program it**.

Moreover, including some **other interfaces** for the user such as indicator LEDs to show if there is an energy supply, a screen to show information or buttons to interact with the device might be useful, especially during the development stage.

Taking all the facts mentioned above into consideration, probably **voltage regulator devices** will have to be included too as the different components might work at different voltages. In addition, batteries don't provide a stable voltage value over time, this value depends on the state of charge and on the current it is delivering, so thanks to those regulators we will be able to have a stable voltage source for our components.

4.3 Firmware requirements

Through the firmware we will provide the microcontroller with the capability of carrying out all the tasks, without it our device won't be different from a paperweight. Before its creation, we must take into account which one is the microcontroller we are working with as it will determine its implementation. Most commercial microprocessors are designed to be programmed in C or C++, however, most of them include small differences in their programming directives in order to use their resources more efficiently or to include special features.

4.4 Constraints

Not only the requirements must be taken into account before starting with the development stage but also the constraints we might find.

On the one hand, **we must be able to manufacture our design**. Initially, the product is thought to be manufactured in our lab so it is necessary to take into consideration the tools that will be at our disposal and the possibilities that they offer us. Apart from that, sending our designs to be manufactured by an external supplier is always viable and it might reduce or manufacturing constraints since they are specialised in it and they have more adequate tools and more precise machines to perform the task, however, it will take longer to be available as they are usually sent from china and its delivery might take up to a month or two. Not to mention the experience and knowledge about the manufacturing process that this product fabrication will provide. The possibility of both manufacturing methods makes it necessary to carry out a design that must be able to be performed in any of both ways.

On the other hand, like any good engineer worth his salt, **we should always seek to minimize costs**. This might not be one of the most restrictive constraints but it is always a key factor when choosing between different components. We should always take into account the available components in our labs as primary options and we must always justify our purchase if any other component is required.

In addition, the fact of designing a device based on a previously designed product creates several constraints as **we must adapt our design to the shape and size of the boat** as well as to the LEDs and motors it includes.

Finally, the last factor we must consider is the time, we must carry out our design taking into consideration that **the time is limited**, the same way as we will have a deadline to finish our tasks once outside in the professional world.

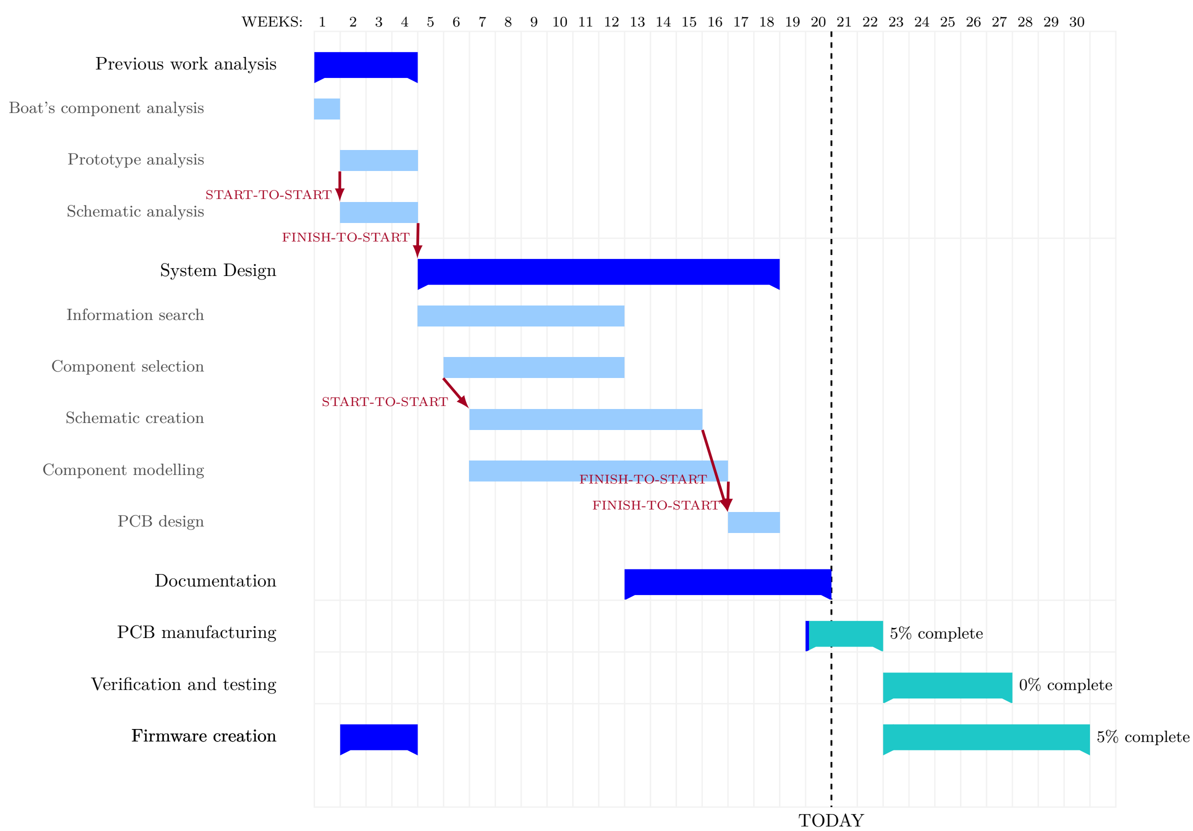
Chapter 5

System description and design

5.1 Project planning

Once that the technical base of the project are known and before starting it, it's necessary to identify our available time, the tasks that must be carried out and to set how much time we will employ in each task. For this purpose we defined the following **Gantt diagram**:

Gantt-Chart



As can be observed there are still a few remaining tasks addressed to create a functional product which could not be finished in the available time, nevertheless, this project was born already bearing in mind that probably it would not be possible to finish it completely. In fact, these tasks could even suppose another whole final project for prospective students. The proposed future work will be described in the conclusions, section 6.2.

5.2 Schematic design

In this section all the components that form the hardware of our project will be described as well as the path followed to choose each one of them among any others that could be found in the market and the interconnections between them.

5.2.1 The microcontroller

According to [8] "a Microcontroller (sometimes called Microcontroller unit (MCU)) is a single IC typically used for a specific application and designed to implement certain tasks. Products and devices that must be automatically controlled in certain situations, like appliances, power tools, automobile engine control systems, and computers are great examples, but microcontrollers reach much further than just these applications.

Essentially, a microcontroller gathers input, processes this information, and outputs a certain action based on the information gathered. Microcontrollers usually operate at relatively low speeds, around the 1MHz to 200 MHz range, and need to be designed to consume less power because they are embedded inside other devices that can have greater power consumption in other areas".

To sum up, **this device will act as the "brain" of our device** coordinating all the tasks, gathering the sensors' information and activating the actuators in consequence so all these components will be connected to it.

Among the wide variety of them found in the market, three have been chosen as the most appropriate elections for our device. All of them are considerably known for being general-purpose microcrontroller powerful enough for most electronic projects, especially among the teaching community. They are the **ATmega328p** (datasheet in [4]), usually found in the Arduino UNO board, the **ESP8266EX** (datasheet in [10]), usually found as a module but which can be also found as a single IC, and the **ESP32-WROOM-32**(datasheet in [9]), the successor of the ESP8266EX, which can be found integrated in the ESP32 module, the one used in the prototype, Figure 2.6.

For choosing which one is more suitable for our application, we will make the table trade-off that can be seen in Table 5.1.

	ATmega328p	ESP8266EX	ESP32-WROOM-32
Bits number	8	32	32
Cores number	1	1	2
Operating voltage	1.8V ∼ 5.5V	2.5V ∼ 3.6V	2.5V ∼ 3.6V
Current consumption	45mA ∼ 80mA	15µA ∼ 500mA	20mA ∼ 500mA
Current consumption deep sleep	35mA	0.5µA	5µA
Digital GPIO	14	13	36
Digital GPIO with PWM	6	13	36
ADC	6(10 bits)	1(10 bits)	2(12 bits)
DAC	0	0	2(8 bits)
SPI/I2C/I2S/UART	1/1/1/1	2/1/2/2	4/2/2/4
Flash Memory	32KB	4MB	4MB
SRAM	2KB	64KB	520KB
EEPROM	1024 bytes	512 bytes	0
Clock Speed	16MHz	52MHz	80MHz/160MHz
WIFI	No	Yes	Yes
Bluetooth	No	No	Yes
Price	1.76€ URL	1.29€ URL	2.21€ URL
Our choice	×	×	✓

Table 5.1 – *Microcrontroller comparison*

5

Note that the **ESP32-WROOM-32 overcomes the other two options in almost any feature**. Although its price is higher, it counts with a higher number of GPIO and a higher number of communication interfaces that might be especially useful since it will be necessary to establish communication between the microcontroller and the programmer device, the wireless communication device and several other sensors. However, its more relevant feature is that it possesses **two cores** and, above all, a **higher clock speed** that will allow it to process the information faster.

This microcontroller doesn't include a EEPROM, which might be especially useful for saving configurations as it is a non-volatile memory storage system, nevertheless, this feature is not necessary for our project and could be included as an external chip if it was necessary.

Once the decision is made, there is one more factor we must take into consideration: the seller offers two different versions of the ESP32-WROOM-32:

- The ESP32-WROOM-32D, which includes an onboard antenna.
- The ESP32-WROOM-32U, which includes an U.FL connector that needs to be connected to an external IPEX antenna). (see Figure 5.1)

In principle, the communication will be establish through GSM/GPRS, this means that the WiFi won't be necessary and so neither will the antenna be, therefore, our first option would be the ESP32-WROOM-32U version. Nevertheless, considering that this design might be useful for other future designs, we have decided to use the ESP32-WROOM-32D to facilitate the use of all its features.

In addition, this component is available in our workshop since other students have chosen it as the best option for other projects as well. All these factors make the **ESP32-WROOM-32D the most appropriate option** for our project.

(a) *ESP32-WROOM-32D* **(b)** *ESP32-WROOM-32U*

Figure 5.1 – *ESP32-WROOM-32 versions*

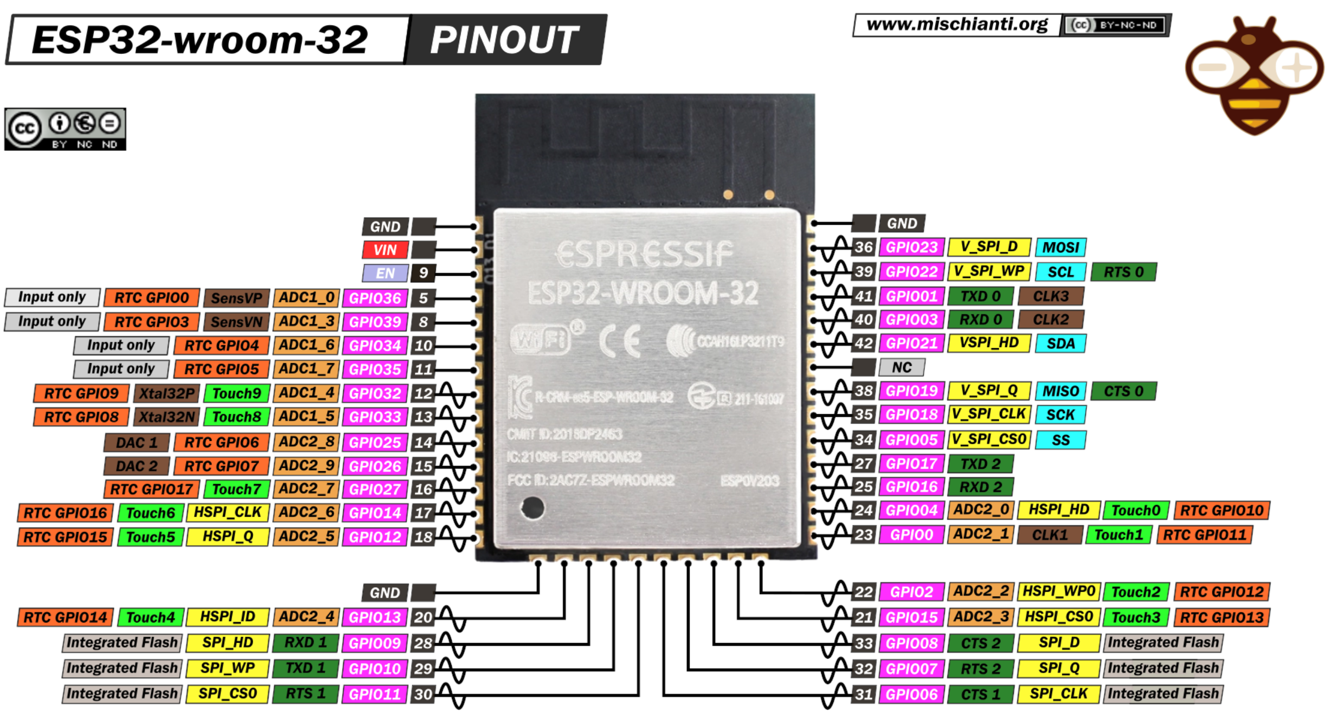
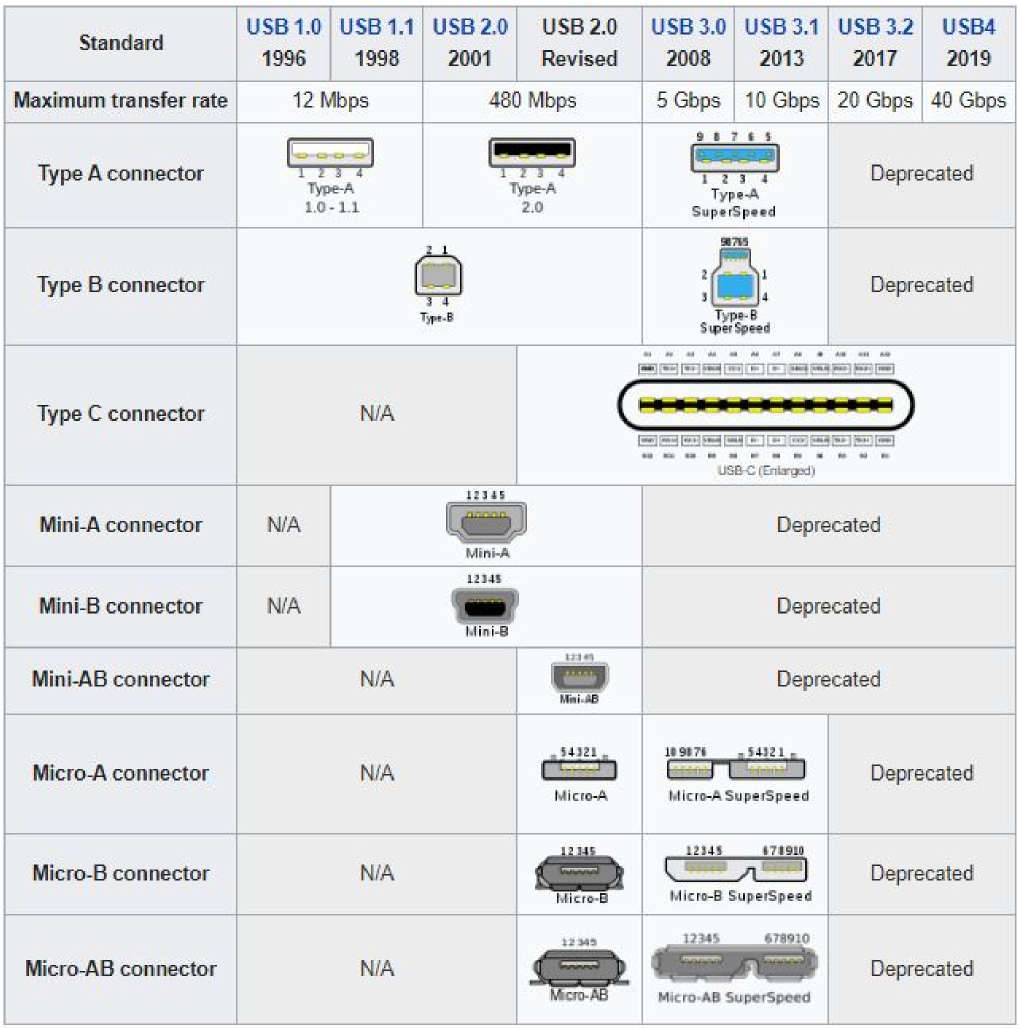


Figure 5.2 – *ESP32-WROOM-32 pinout (www.mischianti.org)*

5.2.2 The programming interface

5.2.2.1 The port

In order to program our device an interface to connect the programmer device, typically a computer, is necessary. This interface will be a USB port since it is an industry standard and the great majority of devices include at least one, however, it exists several different versions of USB connectors since the standard has been updated over the years as can be seen in Figure 5.3.



Standard	USB 1.0 1996	USB 1.1 1998	USB 2.0 2001	USB 2.0 Revised	USB 3.0 2008	USB 3.1 2013	USB 3.2 2017	USB4 2019
Maximum transfer rate	12 Mbps		480 Mbps		5 Gbps	10 Gbps	20 Gbps	40 Gbps
Type A connector	Type-A 1.0 - 1.1		Type-A 2.0		Type-A SuperSpeed		Deprecated	
Type B connector	Type-B				Type-B SuperSpeed		Deprecated	
Type C connector	N/A			USB-C (Enlarged)				
Mini-A connector	N/A	Mini-A			Deprecated			
Mini-B connector	N/A	Mini-B			Deprecated			
Mini-AB connector	N/A			Mini-AB	Deprecated			
Micro-A connector	N/A			Micro-A	Micro-A SuperSpeed		Deprecated	
Micro-B connector	N/A			Micro-B	Micro-B SuperSpeed		Deprecated	
Micro-AB connector	N/A			Micro-AB	Micro-AB SuperSpeed		Deprecated	

Figure 5.3 – *Available connectors by USB standard. [43]*

A high data transfer won't be necessary as the upload time is not a critical aspect, this leads us to take the dimension and the price as the main factors.

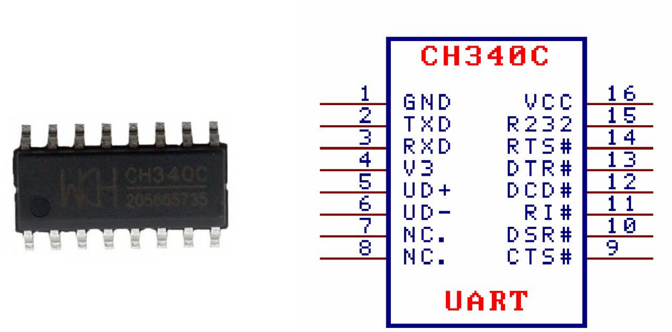
As a consequence of this, the type C connector has been dismissed due to that it entails a higher difficulty of implementation and price. After that, our first option was the micro-B type as it is the smallest and is commonly used, however, after starting our design, we decide to change our election after receiving advice from our mentor. Apparently, the mentioned connector has a weaker solder due to a smaller contact surface so its union is easily weakened after several uses and the component can be easily ripped off. Therefore, the election was changed to a **mini B type connector**, which can be implemented with a stronger soldering and approximately the same price as can be seen in Table 5.2.

Type	Micro-B	Mini-B	C
Maximum transfer rate	480 Mbps	480 Mbps	40 Gbps
Size	6.85 x 1.8 mm	7 x 3 mm	8.3 x 3.1 mm
Price	0.28€ URL	0.32€ URL	1.05€ URL
Our choice	×	✓	×

Table 5.2 – *USB port comparison*

5.2.2.2 The communication protocol

Once the port is selected, we can move on to the next aspect: the communication protocol. As seen previously when describing the microcontroller, subsection 5.2.1, it can communicate through several protocols: SPI,I2C,I2S and UART, but anyone of them is the USB protocol so we must translate the signal. For this purpose the **CH340C** (datasheet in [33]) will be used, which is able to convert data from USB to UART, RS232 and RS485. This component has been selected due to that it is the most commonly used in this type of application, for example, it can be found in the Arduino UNO and ESP32 board, in addition, it is available in the laboratory stock.

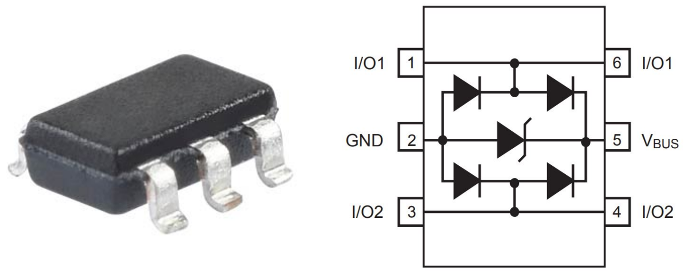


(a) *Packaging (SOP-16)* **(b)** *Pinout (top view)*

Figure 5.4 – *CH340C*

This chip is a quite sensitive module, especially in terms of power supply. Due to that, the seller recommends adding a bypass capacitor of 0.1 µF at the power supply pin. In addition, we have decided to include an **ESD protection** to prevent possible damage to the device. In particular, the **USBLC6-2** (datasheet in [26]), which can be seen in Figure 5.5, owing to that it is the one available in the laboratory stock. The power supply provided by the USB has been chosen to feed this device as the manufacturer's datasheet recommends. Moreover, we have decided to include an extra ferrite bead chip and a bypass capacitor to obtain a cleaner voltage supply.

Moreover, the DATA+ and DATA- signals that come from the USB port form a differential pair that must be taken into consideration once in the design phase as the connections' length must be equal to maintain symmetry in the signals, otherwise, they might put out of step and so provoke errors in the data transmission.



(a) *Packaging (SOT-23-6L)* **(b)** *Functional diagram (top view)*

Figure 5.5 – *USBLC6-2 ESD protection*

All this concepts lead us to obtain the schematic that can be seen in Figure 5.6, where *USB_TX* and *USB_RX* are connected to the default UART ports of the ESP32 (to GPIO01 (RXD_0) and GPIO03 (TXD_0) respectively as can be seen in Figure 5.2).

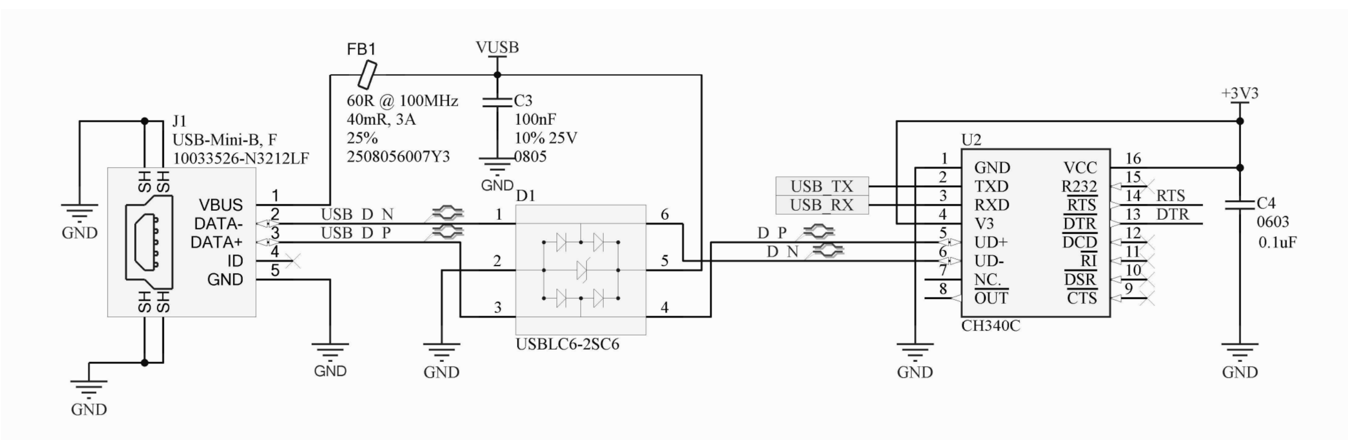
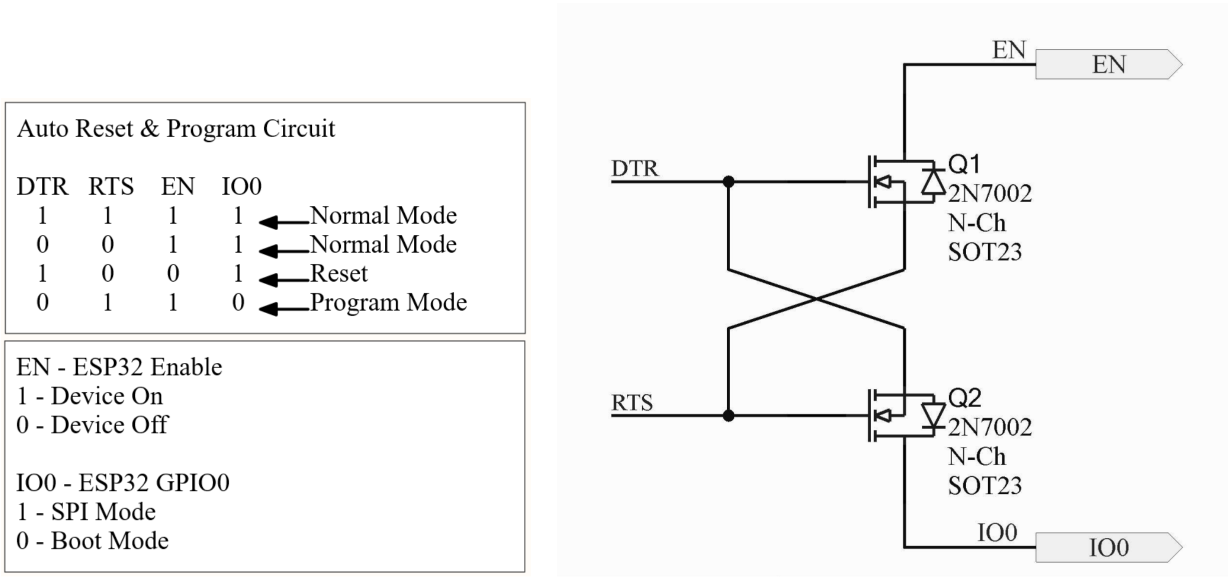


Figure 5.6 – *USB to UART schematic*

5.2.2.3 Setting the ESP32 in boot mode

Apart from this connections, **the microcontroller must be set in programming mode (boot mode)** while the data transfer is being executed. This is achieved through the EN (ENABLE) pin and the GPIO0 pin, whose operation mode is described in Figure 5.7 (a). To control those pins from the Request to send (RTS) and Data terminal ready (DTR) signal the circuit shown in Figure 5.7 (b) is used.

In the same way, this circuit could have been implemented with BJTs, however, these would need a gate resistor to prevent possible damages caused by a too high Vbe (Voltage between the base and the emitter). Moreover, MOSFETs usually have a higher switching speed than most BJTs. A suitable option to implement this design is the **2N7002** (datasheet in [6]) due to its size and price. Besides, it can be found available in



(a) *Operation modes* **(b)** *EN and GPIO0 circuit*

Figure 5.7 – *Operation modes of the ESP32*

the laboratory stock. For this reason, this device will be included in the following designs whenever a low power transistor is needed.

Its main characteristics are listed in Table 5.3.

Name	**2N7002**
Type	MOSFET
Channel	N-channel
Vds max	60 V
Id max	115 mA
Rds max	7.5 Ω @Vgs = 5V
Packaging	SOT23
Price	0.36 € URL

Table 5.3 – *2N7002 MOSFET*

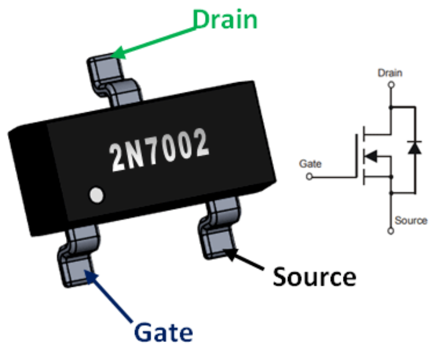


Figure 5.8 – *2N7002 packaging and pinout*

An extra feature added to our design is the possibility of **modifying these operation modes manually through buttons**. To add this functionality a typical button with a pull-up resistor will be implemented in our design, to which a capacitor has been added to avoid the switch bouncing.

The last correction we will do to this design is adding a 1 µF capacitor in the ENABLE pin due to the following explanation. No combination of RTS and DTR will cause EN = 0 and IO0 = 0 which is the precondition to reset the microcontroller. In theory, this event should never occur, however, depending on propagation delays and the speed of the device it might take place provoking the ESP32 to get stuck in a state where it is not programmable. The previously mentioned capacitor will prevent this due to it will slow down the speed of the signal and consequently, the propagation delays won't be significant. This consideration produces the circuit that can be seen in Figure 5.9.

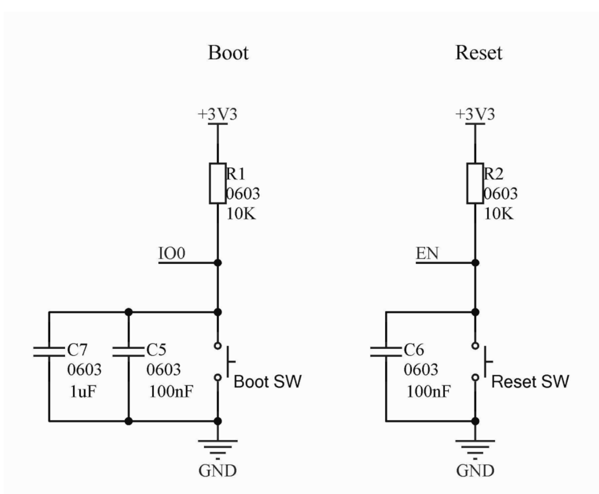


Figure 5.9 – *Reset and boot buttons*

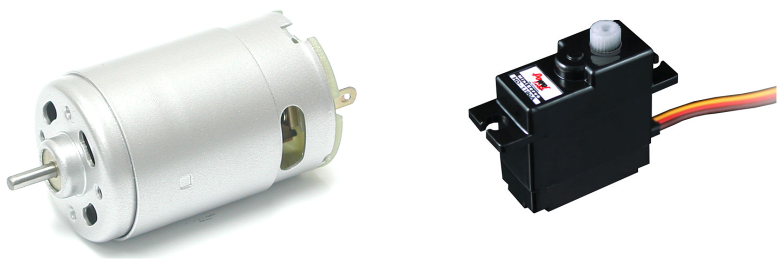
All these aspects regarding the USB communication are implemented in the schematic that can be seen in the altium files, Appendix E, 7th sheet.

5.2.3 The motors

As previously mentioned, the boat includes a 6V DC motor and two servomotors. For this project the servomotor that controls the locker door will be unused since this functionality is not necessary. The other two motors will conform to our movement system. Their technical characteristics can be seen in Table 5.4

Name	DC motor	Servomotor
Power supply (main)	0 ∼ 6 V	5V
Power supply (logic)	-	5V
Power consumption	5 A (at maximum load)	500 mA
Price	3.77 €	2.9 €

Table 5.4 – *Motors technical characteristics*



(a) *DC motor* **(b)** *Servomotor*

Figure 5.10 – *Motors appearance*

5.2.3.1 The Servomotor

A servomotor is a device that **has the ability to be located in any position within its operating range and remain stable in said position**. It is basically a motor that incorporates a regulation system to control both the speed and position. Thanks to this system they are controlled by sending a variable width electrical pulse, or PWM (Pulse Width modulation), through a single control wire.

Typically, a servo motor can only turn 90° in any direction for a full 180° of motion which for our functional requirements is enough. The PWM sent to the motor determines the position of the axis and, based on the duration of the pulse sent through the control cable, the rotor will rotate to the desired position.

As shown in Table 5.4, this logic works at 5 V and, since our microcontroller works at 3.3V, **we will need a voltage conversion** to adapt the control signal, otherwise, it is most likely that our assembly will not work, and even we might damage the device.

There are many ways to adapt voltage levels, from simple solutions, such as using a voltage divider, to specific ICs that allow high switching speeds in both directions. One of the most common is by using a simple transistor, which is a simple, high-speed, bidirectional solution; this is the so called **level shifter**, whose diagram can be sen in Figure 5.11.

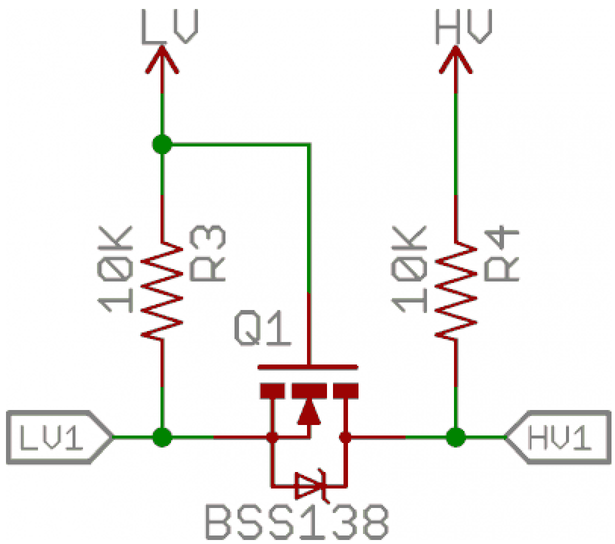


Figure 5.11 – *Level shifter (luisllamas.es)*

We will implement the same circuit where LV (low level) is 3.3V and HV (high level) is 5V. The only difference is that we will change the transistor for the **2N7002** previously mentioned in subsubsection 5.2.2.3, which allows high commutation frequency as well.

5.2.3.2 The DC motor

5

Numerous methods of different complexity and cost can be used to control the speed and torque of a Dc motor. Among them, one has been chosen thanks to its simplicity and low cost, it is the so called **regulation by trimmers or choppers**.

This regulation system allows the voltage level to be reduced thanks to devices called trimmers or choppers. These elements have several static switches which allow controlling at what instants of time they conduct or not the current through the circuit. Varying, therefore, the connection and disconnection times, a square-shaped output signal will be generated (see Figure 5.12), with a greater or lesser pulse amplitude whose average value corresponds to the output voltage. This output voltage is given by Equation 5.2.1.

$$V_{average} = V_{in} * D \tag{5.2.1}$$

$$D(\text{duty cicle}) = \frac{t_{on}(\text{pulse width})}{T(\text{Period})} \tag{5.2.2}$$

This PWM signal can be generated from our microcontroller as previously mentioned, nevertheless, the power output that the pin can deliver is not enough for such a powerful motor. Therefore, a motor driver will be used which will be controlled by the same signal but that will allow a much higher current, the H-bridge.

As is defined by [38]: "An **H-Bridge** is an electronic circuit that is generally used to allow a DC electric motor to rotate in both directions, forward and backwards. They are widely used in robotics and as power converters, as in our case. H-bridges are available as integrated circuits, but can also be built from discrete components.

The term *H-bridge* comes from the typical graphical representation of the circuit. An H-bridge is built with 4 switches (mechanical or using transistors). When switches S1 and S4 (see Figure 5.14) are closed

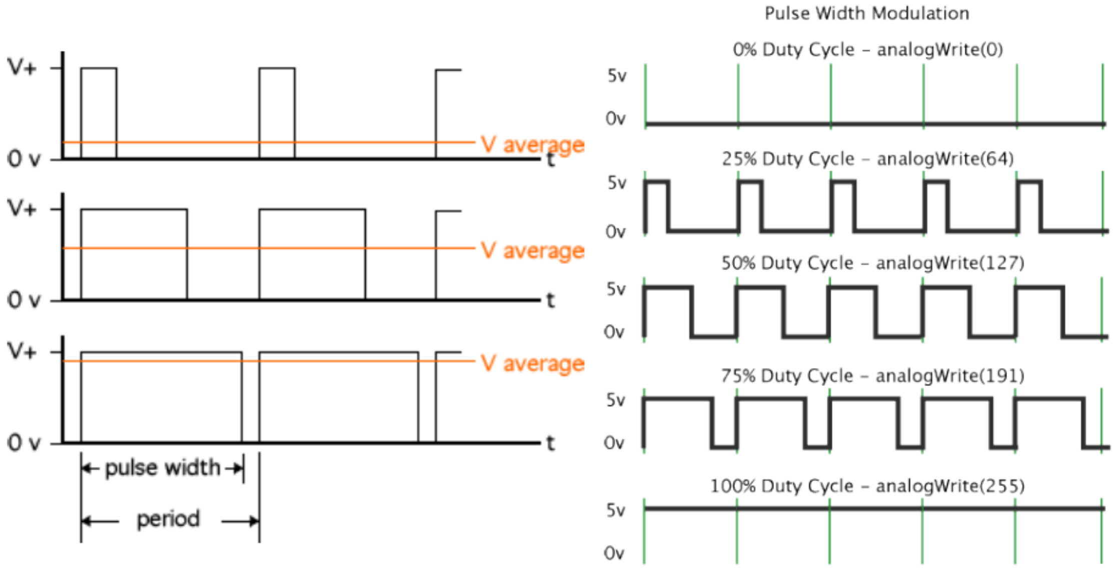


Figure 5.12 – *PWM (Pulse Width modulation)* - *desingbuildcode.weebly.com*

(and S2 and S3 open), a positive voltage is applied to the motor, making it rotate in one direction. By opening switches S1 and S4 (and closing S2 and S3), the voltage is reversed, allowing the motor to rotate in the reverse direction.

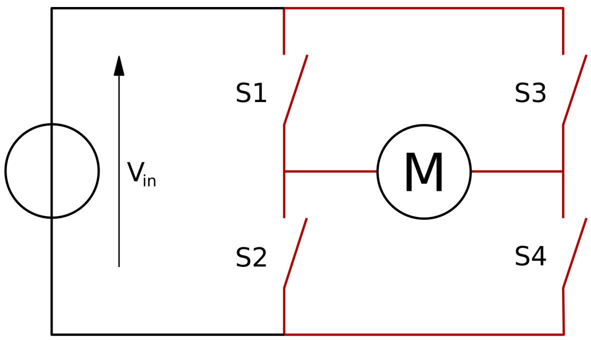


Figure 5.13 – *Structure of an H-bridge*

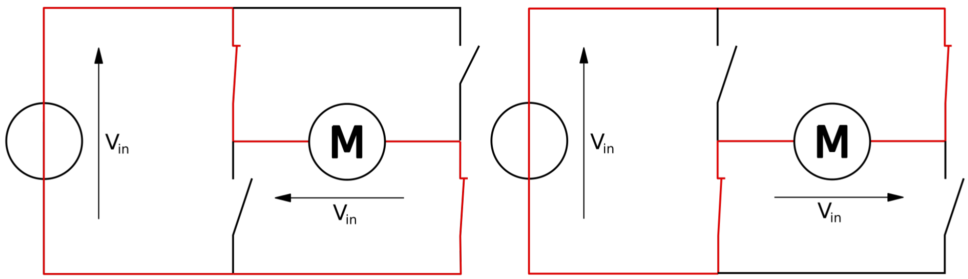


Figure 5.14 – *The two basic states of an H-bridge*

With the nomenclature we are using, switches S1 and S2 can never be closed at the same time, because this would short-circuit the voltage source. The same happens with S3 and S4 .

The H-bridge arrangement is generally used to reverse the polarity/direction of the motor, but can also be used to 'brake' the motor, where the motor comes to a sudden stop, as the motor's terminals are shorted. In shorted case, the kinetic energy of a rotating motor is consumed rapidly in form of electrical current in the shorted circuit. The other case, to let the motor 'free run' to a stop, as the motor is effectively disconnected from the circuit. The following Figure 5.15 summarizes operation, with S1-S4 corresponding to the diagram above. In the table below, "1" is used to represent "on" state of the switch, "0" to represent the "off" state [38]. "

S1	S2	S3	S4	Result
1	0	0	**1**	Motor moves right
0	**1**	**1**	0	Motor moves left
0	0	0	0	Motor coasts
1	0	0	0	
0	**1**	0	0	
0	0	**1**	0	
0	0	0	**1**	
0	**1**	0	**1**	Motor brakes
1	0	**1**	0	
x	x	**1**	**1**	Short circuit
1	**1**	x	x	

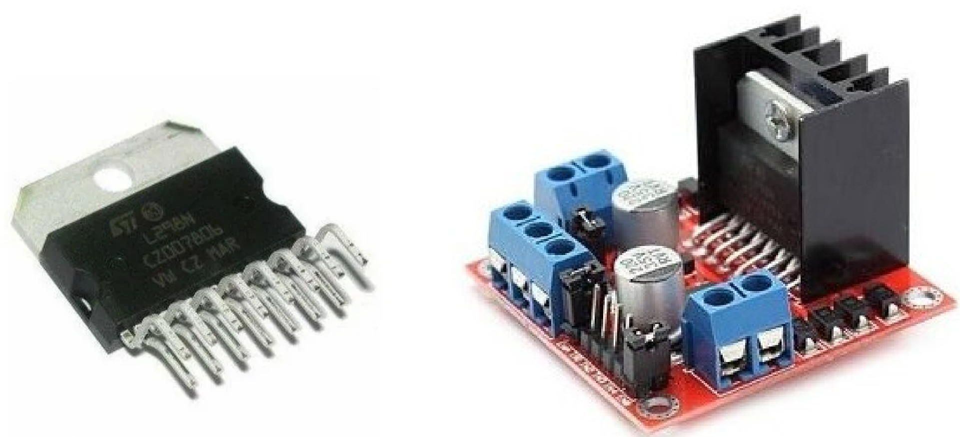
Figure 5.15 – *H-bridge operation*

Once its operating principle has been explained, we can move on to the next aspect, the choice of the specific component. One of the most used H-bridge drivers for low-power applications like this one is the **L298** (link to datasheet). Likewise, we could have used a discrete circuit or any other model, however, the popularity of this device makes it a cheap and simple option. Its technical features can be seen in Table 5.5.

Name	L298
Type	H-bridge driver
Operating supply voltage	2.3V ∼ 46V
Logic supply voltage	4.5 ∼ 7V
Peak output current (per channel)	3A (Non repetitive) 2.5A (Repetitive (80% on - 20%off) 2A (DC operation)
Power dissipation	25W
Number of channels	2
Price	1.47 € URL

Table 5.5 – *L298 H-bridge*

As you might have already noticed, it seems to have a design error since the maximum current of the motor is 5A and the maximum of the driver is 2A ∼ 3A. Nevertheless, **this limitation has been solved** as follows: as the driver possesses two channels (and the peak output current is defined as per channel) it is possible to connect both outputs (and inputs respectively) to obtain the double maximum current. In my personal opinion, the best option should have been using a discrete circuit or any other driver that supports a higher current value, nevertheless, the L298N was available in our lab stock and since it could be adapted for our project it became the cheapest and simplest option.



(a) *L298N IC* **(b)** *L298N module*

Figure 5.16 – *L298 versions*

The pinout of the module can be seen in Figure 5.17, where IN1 to IN4 would be equivalent to the previously mentioned S1 to S4 in Figure 5.15, the ENA and ENB pins enable the pins IN1 to IN2 and IN3 to IN4 respectively. In our case, IN1 and IN3, IN2 and IN4 and ENA and ENB will be connected as previously mentioned to **use both outputs**.

In addition, the module includes a 5V voltage regulator to supply its logic circuit. The output of the regulator is also connected to the +5V pin so it is possible to use it as a power source for other components if needed. Moreover, the regulator can be disabled by connecting the jumper, in which situation the +5V

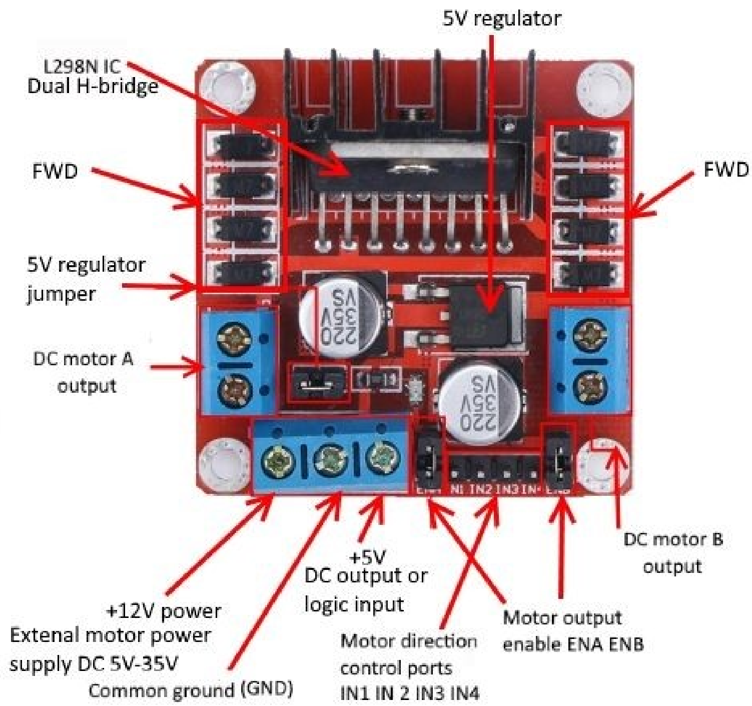


Figure 5.17 – *L298N pinout (etechnophiles.com)*

5

pin would act as an input to feeds the logic circuit by an external source. For our project, **the jumper will be activated** to not need to feed this logic circuit externally.

Finally, in the same way as in the previous section, the signal to control the logic circuit of this IC has to be powered up at a different voltage than the ESP32, 5V as the recommended value, a cause of that we will be using **three level shifters**, one for each one of the three signals that control the motor.

All these considerations are transformed into the schematic that can be seen in the altium files, Appendix E, 13th sheet.

5.2.4 The wireless communication

Moving on to the next feature, as the user is the one who will set the destination point, the device must include a wireless communication system. An enormous variety of wireless communication systems can be used, nevertheless, taking into consideration the constraints that the work environment of the device will imply, our number of options is reduced significantly. Due to that, any kind of short-range communication is dismissed.

After many thoughts, the mobile networks (GSM/GPRS) were chosen as the best option thanks to their simplicity and availability. Consequently, a **SIM800L** (datasheet in [24]) module was selected. This is widely spread for use in projects and can be usually founds included in the most used microcontroller boards whenever a connection to the mobile networks is needed. Thanks to that there is available a huge quantity of online content to facilitate its usage such as forums and tutorials from the community. Besides, this component is available in our laboratory, therefore, justifying the purchase of a new component with the same function would have been complicated.

The component is the one that can be seen in Figure 5.18 and whose technical specifications are those shown in Table 5.6.

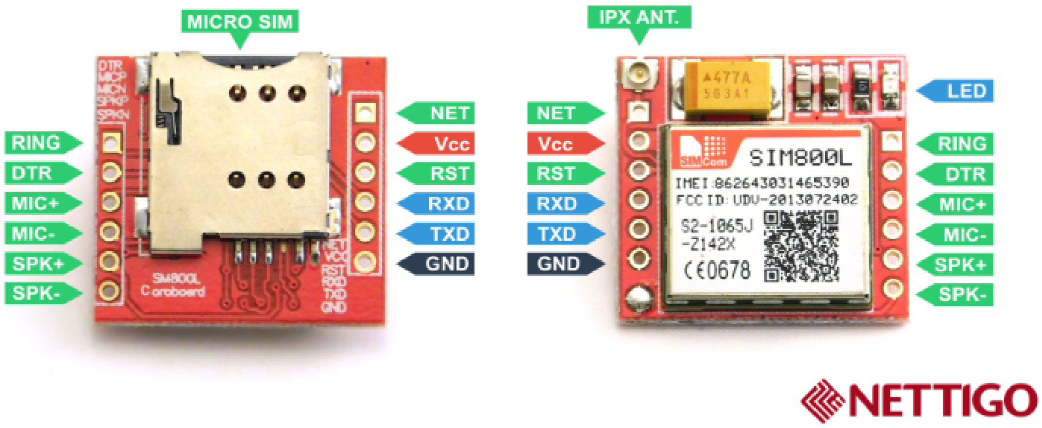


Figure 5.18 – *SIM800L (nettigo.eu)*

Name	SIM800L
Type	GSM & GPRS module
Operating voltage	3.8 V ~ 4.2 V
Current required	1A ~ 2Amax
Baud rate	9600
PCB size	23 x 25 mm
Price	1.6 € URL

Table 5.6 – *SIM800L technical characteristics*

Finally regarding this component, as can be observed in Table 5.6, this device is powered at 4 V which is not a standard value. This implies that it will be necessary to convert the voltage of the signals that connect it with the microcontroller. This is not essential since the voltage difference between the module (4 V) and the microcontroller(3.3 V) is not too big, however, the manufacturer recommends adapting the voltage since the module is quite sensitive. Therefore, we have concluded that the most appropriate option would be to adapt it. To achieve that a **level shifter** structure has been used as the previously mentioned in the servomotor description, subsubsection 5.2.3.1. In particular, three of them will be necessary to convert each one of the three signals that are used to establish the communication (RX, TX and RST), where the RX and TX signals will be connected to the ports reserved to UART communication.

5.2.5 The GPS

In order to be able to move to a desired location, an autonomous device must know, at least, the starting point and the destination point of the trip ahead to calculate a route. The destination will be selected by the user, however, the starting point must be acquired through a sensor onboard.

5

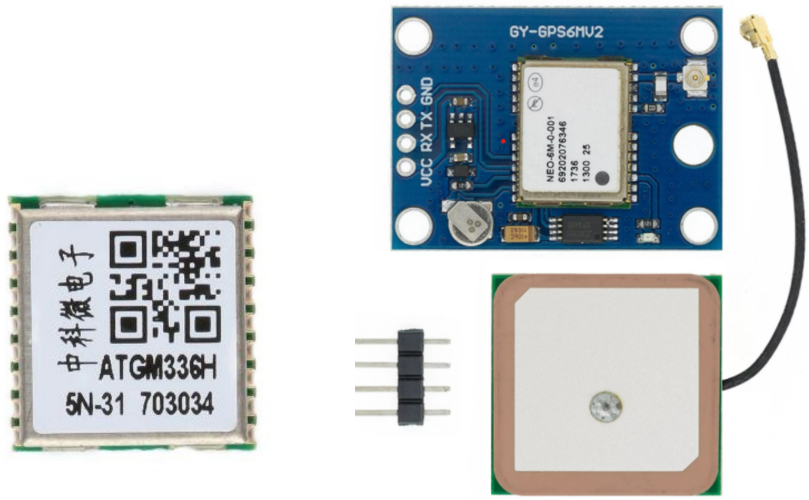
The GPS selected for this purpose in the design previously made by another student was the **ATGM336H-5N** (datasheet in [14]), a IC which would need the following components:

- An external **antenna** with its respective connector.
- A **backup supply** to keep the GPS on while the microcontroller is turned off. It is important to keep the component running due to that the first connection between a GPS chip and the satellites can take up to 20 minutes. This means that if this power supply is not included it will become necessary to wait 20 minutes each time the device is turned on before using it.
- An EEPROM to store all the data obtained while the microcontroller is off.

The integration of all these components would increase the cost and complexity of the device and, owing to that the time for this project is limited we decided to look for a module which already includes all the necessary components to facilitate the design. As a result of that, the **GY-NEO6MV2 module** (datasheet in [27]) was chosen, which, in fact, is cheaper than the previously selected and keeps almost the same technical specifications as can be seen in Table 5.7.

Name	**ATGM336H-5N**	**GY-NEO6MV2 Module**
Type	IC	module
IC	ATGM336H-5N	GY-NEO6M
Operating voltage	2.7V ∼ 3.6V	3V ∼ 5V
Power consumption	25 mA @3.3V (IC) 100 mA @3.3V (IC + antenna)	34 mA @3.3V (IC) 115 mA @3.3V (module)
Serial interfaces	SPI/I2C//UART	SPI/I2C/UART/USB V2.0 (in IC) UART (in module)
EEPROM	No	Yes
Backup battery	No	Yes
Antenna	No	Yes
Sensitivity	-162 dBm	-161 dBm
Price	3.77 € URL	2.9 € URL
Our choice	×	✓

Table 5.7 – *GPS comparison*



(a) *ATGM336H-5N* **(b)** *GY-NEO6MV2 Module*

Figure 5.19 – *GPS comparison*

This device will be **connected with the ESP32 through the two pins of the UART** since it is the only communication protocol that the GPS module permits.

5.2.6 The direction sensor

Another important sensor is the direction sensor, obtained thanks to a digital compass, which is essential to provide our device with the capability of moving through the sea.

A digital compass is a sensor that measures the value of the magnetic field in three axes, as for example the **HMC5883L** (datasheet in [15]). With this measurement, it is possible to estimate the orientation of the device with respect to the earth's magnetic field.

This type of sensor is usually found combined with a IMU (Inertial Measurement Unit), an electronic device that measures and reports a body's specific force, angular rate, and sometimes the orientation of the body, using a combination of accelerometers, gyroscopes, and sometimes magnetometers as the **adafruit 10-DOF module** (more information in [1]) which can be found in our lab stock.

This device combines 3 of the sensors available on the market to give us 11 axes of data: 3 axes of accelerometer data, 3 axes of gyroscopic, 3 axes of magnetic (compass), barometric pressure/altitude and temperature:

- **LSM303DLHC**: a 3-axis accelerometer (up to +/-16g) and a 3-axis magnetometer (up to +/-8.1 gauss) on a single die.
- **L3GD20**: a 3-axis gyroscope (up to +/-2000 dps).
- **BMP180**: a barometric pressure sensor (300..1100 hPa) that can be used to calculate altitude, with an additional on-board temperature sensor.

"The L3DG20H gyroscope, LSM303DLHC accelerometer compass and BMP180 barometric and temperature sensors are all on one breakout here, to save space and money. Since all of them use I2C, you can communicate with all of them using only two wires. Most will be pretty happy with just the plain I2C interfacing, but it also includes the 'data ready' and 'interrupt' pins, so advanced users can interface with if they choose. A 3V regulator with reverse-polarity protection means you don't have to worry about frying the boards by accident. There's level shifting circuitry so the IMU can be used with 3 or 5V logic boards. And it includes mounting holes so you can securely attach this board to your rocket, robot or any project." [1]

It is true that all these sensors are not essential for our project and we might simply use a magnetometer, however, their availability can be useful to include extra features such as using it as a grounding detector in case the boat run ashore in which situation the accelerometers would not detect any movement. For all this, **we end up choosing the adafruit 10-DOf module** for our project, whose technical specifications can be seen in Table 5.8.

Name	Adafruit 10-Dof
Type	IMU + compass + barometer + temperature sensor
Operating voltage	3 V ∼ 5 V
Maximum current consumption	30 mA
Price	8.35 € URL

Table 5.8 – *Adafruit 10-DOF*

Figure 5.20 – *Adafruit 10-DOF*

5.2.6.1 The voltage and current sensor

As we will be using a battery to power up our device, a difficulty emerges associated with it. Tracking the device consumption to **estimate the state of charge** of the battery as well as **measuring the battery voltage** is indispensable to **prevent the battery from overdischarging**, otherwise, most probably it will suffer irreparable damage even it could burn, depending on its technology, in the worst case or having its lifespan quickly reduced in a not that catastrophic situation. This functionality is denominated as **BMS** and for this reason it is also implemented in any device found in the market that uses a battery.

Nowadays, it exists numerous methods to estimate the state of charge of a battery from the simplest one up to methods involving enormous computational complexity. Despite that, as this vast field could involve a complete final degree project, we have decided to use the simplest method which is based on It is based on the measurements of the battery voltage and the output current over time. This will be described in depth in the power supply chapter, subsubsection 5.2.8.2.

To measure those two variables two sensors are needed as well as a **shunt resistor**. A shunt is a resistive load through which an electrical current is shunted. The resistance of a shunt is precisely known and will be used to determine the intensity of the electric current that flows through this load by measuring the difference in voltage across it.

This measurement could have been obtained from any of the ADC included in the ESP32, nevertheless, we decide to use a device available in our lab stock, the **INA219** (link to datasheet). This decision was influenced by several reasons: first of all, this measurement is a highly exigent task which must be carried out constantly and with a high level of precision so having an external device to obtain this measurement will release the processor from some load. Secondly, the ADC can only read values from 3.3 V to 0 V without suffering damage, hence for a 6V battery like the one we are using, it would be necessary to make a voltage conversion. This might be done, for example, with a tension divider but it would lead to imprecision and errors in the measurements due to the tolerance in the value of the resistors.

The characteristic of the previously mentioned INA219 are shown in Table 5.9

Name	**INA219**
Type	Zero-Drift, Bidirectional Current/Power Monitor
Operating voltage	3 V ∼ 5.5 V
Maximum current consumption	1 mA
Maximum analog input	26 V
Accuracy	0.5% max error
Interface	I2C
Price	1.78 € URL

Table 5.9 – *INA219 technical characteristics*

Figure 5.21 – *INA219 module*

5.2.7 Other components

All the previously mentioned components are essential to comply with the functional requirements of the project, nevertheless, we have thought that **there are other components worth to be included** either for its usefulness during the debugging stage, to facilitate the communication between the user and the device or to add any other extra features. These are the following components:

5.2.7.1 The display

A display is always a useful component, especially in the testing and debugging stage. In it we can show any data for the user, the stage of the code in which the microcontroller is and any error if it would occur.

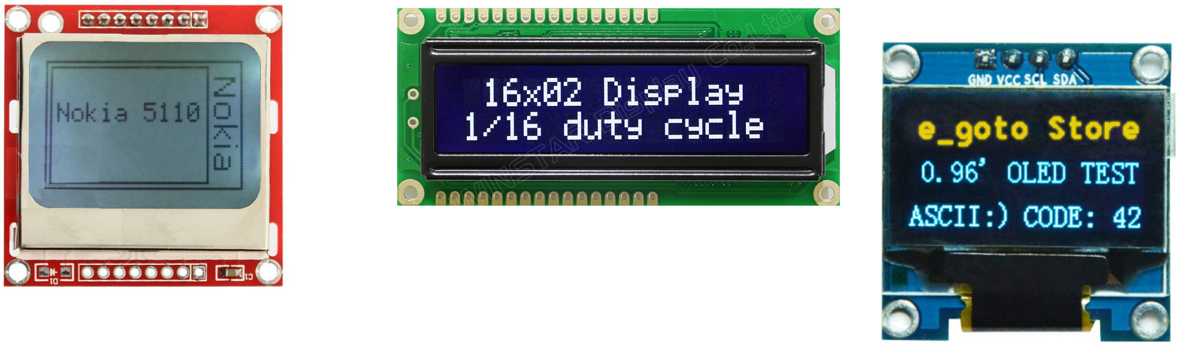
Although this data can be also shown in the programmer's computer through the USB port, the display is still worth to be included since we will hardly have access to a computer in the maritime environment where it will be tested and will operate.

A great variety of display technologies can be found in the market, although we have selected three of them as the possible best options. These along with theirs technical specifications can be seen in Table 5.10

Name	**Nokia 5110 LCD** Datasheet	**16×2 character LCD** Datasheet	**OLED SSD1306** Datasheet
Type	LCD	LCD	OLED
Operating Voltage	2.7V ∼ 3.3V	3.1V ∼ 3.5V	2.8 V ∼ 5.2 V
Consumption	80 mA	17.5 mA	20mA
Matrix size	84x84 pixels	16x2 characters 8x5 pixels per character	128x64 pixels
Color	monochrome	monochrome 8x5 pixels per character	yellow and blue
Number of pins	5(logic) + 2 (power)	7(logic) + 3 (power)	2(logic - I2C) + 2(power)
Price	2.24 € URL	1.96 € URL	1.79 € URL
Our choice	×	×	✓

Table 5.10 – *Display comparison*

After comparing the different technologies it becomes clear that the best option is the **OLED SSD1306** since it has a higher number of pixels, two colors, less pins are needed to control it, relatively low consumption and a lower price.



(a) *NOKIA 5110 LCD* **(b)** *16x2 Character LCD* **(c)** *OLED SSD1306*

Figure 5.22 – *Display comparison*

5.2.7.2 The Real time clock (RTC)

A real time clock is a device that measures the passage of time thanks to a crystal oscillator. They are usually found implemented along with a backup supply thanks to which the clock can keep measuring the time synchronized with the actual time even if the master device is off.

This device, although it is not essential, might be useful to predict arrival times and to calculate periods of time such as the time between the last charge of the battery which might be especially useful since it affects its state of charge as we will explain in depth later.

Our microcontroller, the ESP32, already has a crystal oscillator implemented inside the chip so a RTC can be done without needing an external device, however, the use of an external RTC presents two main advantages over using the internal:

- The fact of having a back-up supply thanks to which the clock will keep running and won't desynchronize. The ESP32 can be set at hibernation mode when is not being used so that it could keep measuring the time permanently, nonetheless, the hibernation mode of the microcontroller has a much higher power consumption (2.5 µA) than which such a simple device would have (on the order of hundreds of nA).

- Having an external device to obtain this measurement release the processor from some computational load so it can focus on more important tasks.

A RTC that can be found in our lab stock is the **DS1302** (link to datasheet) both on chip and on module, between them we decided to use the module for simplicity. Its technical characteristics are shown in Table 5.11

Name	DS1302
Type	I2C Real-Time Clock
Operating voltage	2 V ∼ 5.5 V
Backup supply voltage	2.0 V ∼ 3.5 V
Active current consumption	1.5 mA
Current consumption with backup supply	300 nA
Interface	Simple 3-Wire Interface
Price	0.30 € URL

Table 5.11 – *DS1302 technical characteristics*

Figure 5.23 – *DS1302 module RTC*

5.2.7.3 The LEDs

5

As previously mentioned in the functional requirements,4.1, the ship must fulfil the visibility requirements imposed by the navigation regulations. to achieve that boat already includes the necessary **5mm LEDs** as shown in Figure 2.4 as well as some extra that we won't use. So we will need to have 2 pin male headers on our board to feed them.

In addition, we will include some **LEDs to make visual indications**. The things we want to indicate are:

- If there is power supply.
- If any of the voltage regulators, which we will discuss later, are working.
- If the ESP32 is on.
- Indication within the program flow , for example, to show stages of the code or errors.

In addition, to avoid an excessive current flowing through the component some series resistors are needed as is shown in Figure 5.24.

The value of this resistor is obtained from the equation below, Equation 5.2.3, where V_{in} is the voltage at which they are powered up, V_f is the forward voltage of the LED, which depends on the color and architecture of the component as shown in Table 5.12, and the maximum current recommended by the manufacturer is approximately 20 mA for all of them (selle information).

$$R_{min} = \frac{V_{in} - V_f}{I_{max}} \tag{5.2.3}$$

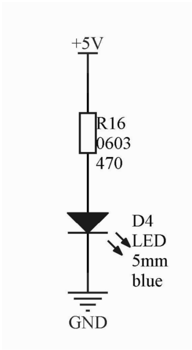


Figure 5.24 – *LED with series resistor*

	Vf	Rmin obtained	Price
Red 5mm	~1.8 V	160 Ω	-
Green 5mm	~2.1 V	145 Ω	-
Blue 5mm	~3.8 V	60 Ω	-
White 5mm	~3.1 V	95 Ω	-
Blue 0603 SMD	~3.1 V	10 Ω	- €
Green 0603 SMD	~2.1 V	60 Ω	- €

Table 5.12 – *LEDs characteristics*

We have just shown the equation from which we can calculate the minimum series resistance to avoid damaging the device, nonetheless, this doesn't mean that we must adjust their values to the minimum possible. That could be a wise decision if we would want to obtain the higher bright possible but, as the unique purpose of our LEDs is to make indications and as our device depends on the duration of a battery, we decided to oversize its value to obtain the same functionality with lower consumption. Considering all these arguments, we have selected a standard value of **470 Ω for the series resistor of all of them.**

5

Finally, to make indications within the program flow we will use two LEDs which, in principle, would need one pin each. To avoid that and so **reduce the number of pins needed** to light up the components we will implement the following circuit, Figure 5.25.

In the shown circuit, when one of them is lit the other is off and vice versa, being possible turning both off by setting an intermediate voltage value with a single pin, the **"2LED" pin**.

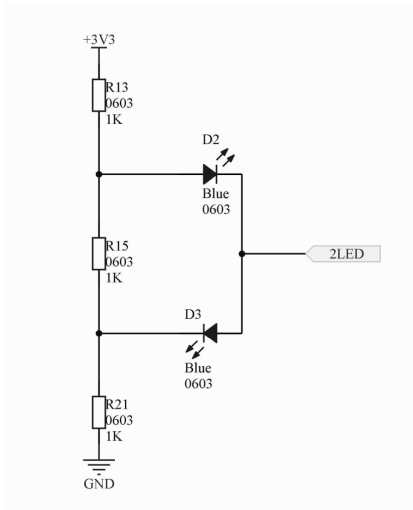


Figure 5.25 – *Double LED circuit*

5.2.7.4 The buttons

Buttons are elements worth to be included in most the devices to allow the user to move through the different menus or to set configurations in the device. We could have implemented simply a circuit with a pull-up resistor as in the reset and boot buttons, Figure 5.9, so that a pin per button would be needed, nonetheless, in the same way as in the previous subsection (subsubsection 5.2.7.3), we have determined that a **circuit to reduce the number of pins used** will be useful. As a result of that, the circuit shown in Figure 5.26 will be implemented.

5

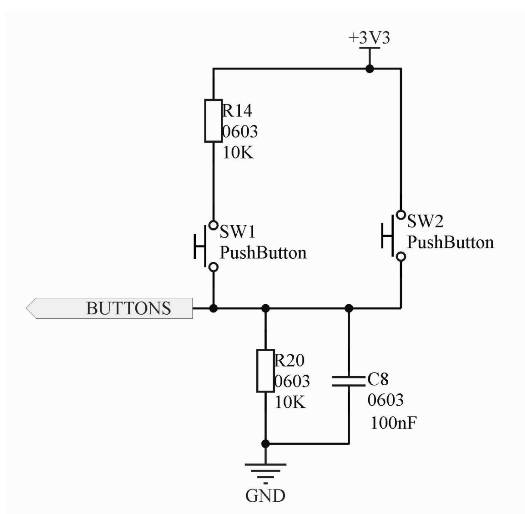


Figure 5.26 – *Double button circuit*

In this circuit, the initial value of the "Buttons" pin is GND as the push buttons are an open circuit. When "SW2" is pressed the output value changes to 3.3V, on the contrary, when "SW1" is pressed the value changes to 1.65 V a cause of the voltage divider, which generate a voltage that can be calculated from the following equation, Equation 5.2.4.

$$V_{Buttons} = \frac{V_{in} * R_{14}}{R_{14} + R_{21}} \tag{5.2.4}$$

By the same token, more buttons could have been added without needing more pins simply by adjusting the value of the resistors that form the voltage divider to avoid overlapping with the value that any of the previously implemented buttons would set if it was pressed. Additionally, a capacitor has been added as in the previously implemented buttons (see Figure 5.9) equally to avoid the switch bouncing effect.

The device selected for this circuit is a two-pins SMD button as the shown in Figure 5.27

Figure 5.27 – *2 pins SMD button*

Notwithstanding, another option to permit the user to interact with the microcontroller is a **rotary encoder**, whose functional principle is similar to that of a button.

A Rotary Encoder is an electromechanical device that converts the rotational movement of an axis into a signal and generates a signal proportional to the rotation of the said axis. It looks very similar to a potentiometer (see Figure 5.28), but unlike a potentiometer, it can be turned indefinitely both clockwise and counterclockwise. Additionally, it includes a button on its axis of rotation.

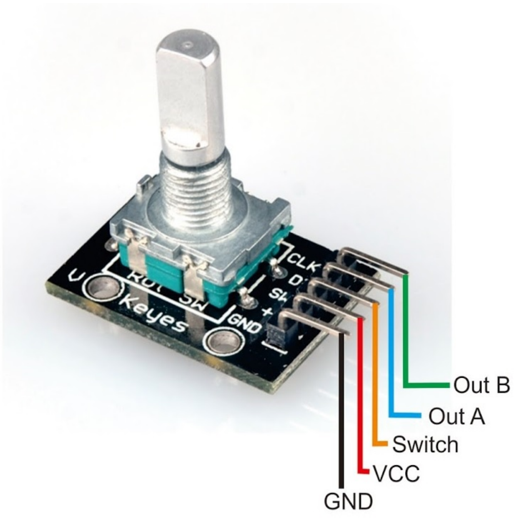


Figure 5.28 – *Rotary encoder*

Its operational principle is based in a shaft that rotates without limit and causes, with small microswitches, a digital signal for each step that rotates (a train of square pulses) similar to the diagram shown in Figure 5.29. By having two similar mechanical outputs, we obtain two different pulses that indicate the direction of rotation as well as the number of pulses that have been rotated as the signal from one output will necessarily be out of phase with respect to the other, since we first go through one step and then the other.[22]

5

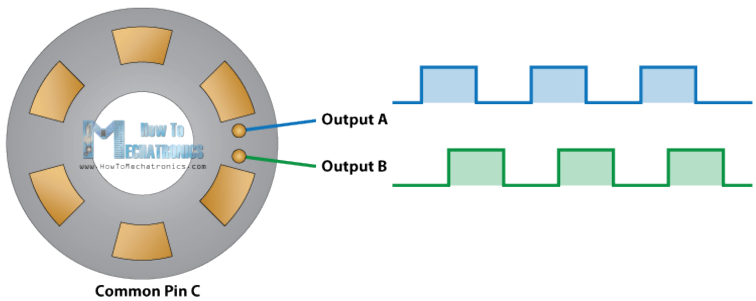


Figure 5.29 – *Rotary encoder working principle [22]*

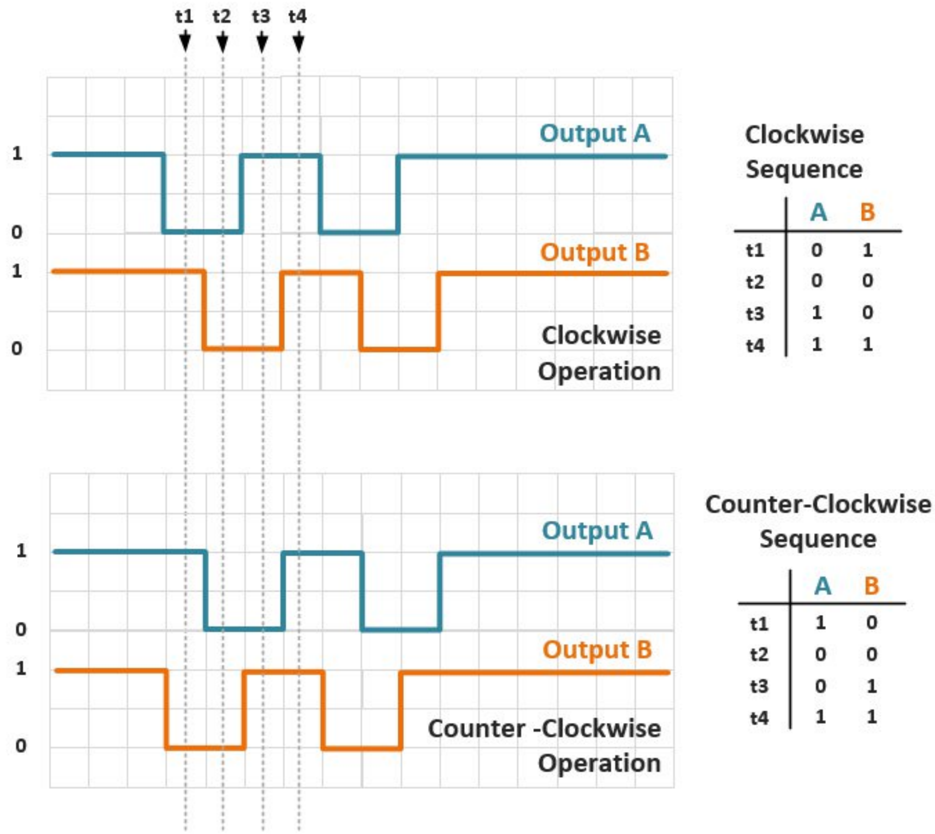


Figure 5.30 – *Rotary encoder functional diagram [22]*

In order to obtain the desired functionality the circuit shown in Figure 5.31 will be implemented.

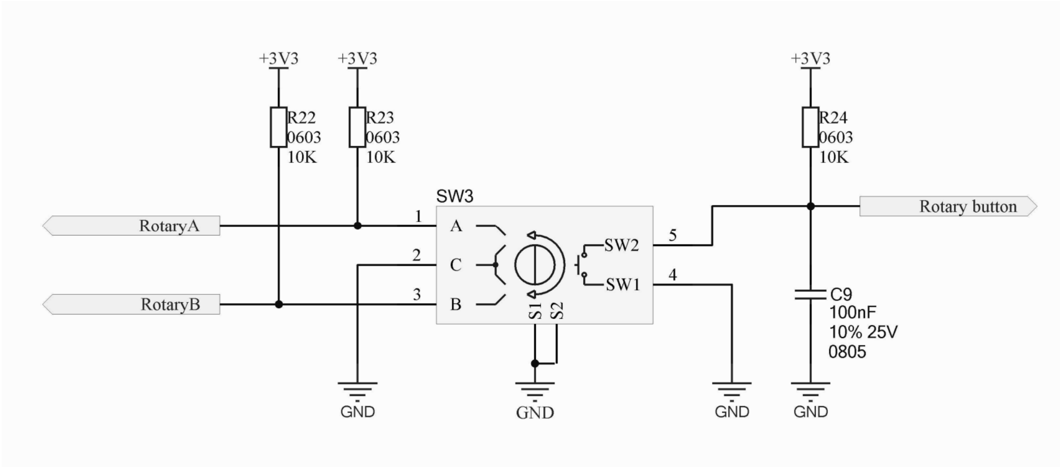


Figure 5.31 – *Rotary encoder schematic*

5.2.8 The power supply

As previously commented, the power supply of our project will be a battery, in this case a sealed lead-acid battery whose nominal voltage is 6V and 7 Ah of capacity (datasheet in [19]) owing to that it is the battery that the initial product includes. The technical characteristics of this product, which can be seen in Table 5.13, are very relevant information since this element is quite sensitive and requires precise care, specifically in terms of charge and discharge.

Name	MHB MS7-6 Sealed lead acid battery
Nominal voltaje	6V
Capacity	7Ah @20Hr(0.35A, 5.25V) , 4.54Ah @1Hr(4.54A, 5.25V)
Weight	1.17Kg
Number of cells	3 (2.0 V per cell)
Maximum discharge current	70A (5sec)
Maximum charge current	2.1A
Cycle use charge voltage	7.2 - 7.5V (-15mV/ºC)
Float use charge voltage	6.75 - 6.9V (-10mV/ºC)
Internal resistance	max. 20.0 Ω
Self discharge	2% of capacity per month (25ºC)
Price	- € URL

Table 5.13 – *Sealed lead-acid battery 6V 7Ah technical characteristics*

The behaviour of this battery is described by the discharge and charge curves extracted from the datasheet, which are shown in Figure 5.32.

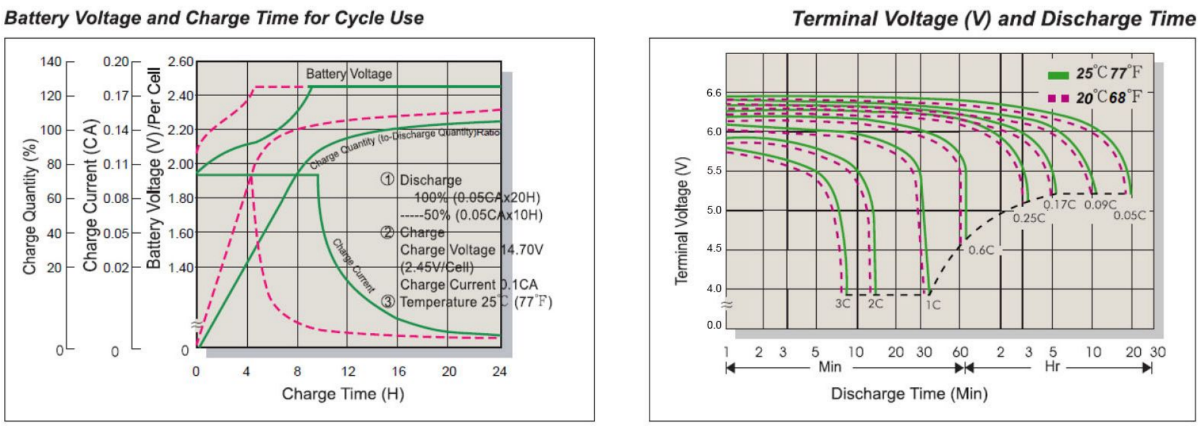


Figure 5.32 – *Discharge and charge curve of a 6V sealed lead-acid battery*

5.2.8.1 The power budget (see Appendix B)

As previously mentioned in the description of voltage and current sensor, subsubsection 5.2.6.1, rechargeable batteries of any kind must keep its voltage between a certain maximum and minimum value to prevent damages or a reduction in their capacity. Additionally, as can be observed in the previously mentioned figures, the voltage at which the battery is completely discharged, as well as the capacity, depends on the discharge current. This entails that a previous current consumption estimation would have to be carried out, which is exhibited in the power budget. However, the definition of a power budget is useful for several other reasons:

1. To estimate the terminal voltage.
2. To estimate the capacity of the battery.
3. To estimate the battery duration in terms of time of use.
4. To know how many voltage regulators are necessary to cover the supply needs of each component
5. To know how much current each voltage regulator must be able to deliver.
6. To estimate the power dissipation of each component.

This is exhibit in Table 5.14.

The average current consumption estimated from this data is around 3 A, which amounts to 0.44C. Comparing this value with the characteristic discharge curve shown in Figure 5.32 we will find out that from this value of current the voltage at which the battery is fully discharged is between 4.8 V and 5.2 V, from this we decided that 5.2 V would be an adequate value for safety. Although, since a consumption sensor is included, a dynamic estimation of the terminal voltage could be done by software based on the current measured, however, that would need a complex algorithm which might reduce the computing power of the microcontroller for other primary tasks.

Component	**Imax** (mA)	**Ityp** (mA)	**Vmin** (V)	**Vtyp** (V)	**Vmax** (V)	**Pmax** (mW)
ESP32	500	500	3	3.3	3.6	1650
OLED SSD1306	20	15	2.8	3.3	5.2	66
GY-NEO6MV2	115	50	3	3.3	5	379.5
Adafruit 10 dof	30	30	3	3.3	5	99
SIM800L	1500	350	3.4	4	4.4	6000
DS1302	1.5	1.5	3	3.3	5.5	4.95
INA219	1	1	3	3.3	5.5	3.3
DC motor	5000	2000	-	6	-	30000
Servomotor	500	100	-	5	-	2500
Led 5mm x 4	40	40	-	5	-	200
Led SMD 0603 x 1	5	5	-	6	-	30
Led SMD 0603 x 1	5	5	-	5	-	25
Led SMD 0603 x 1	5	5	-	4	-	20
Led SMD 0603 x 3	15	15	-	3.3	-	49.5
Total	**7667.5**	**3047.5**				**41001.75**

Table 5.14 – *Power budget*

5.2.8.2 State of charge estimation

All these calculations entail that the device must auto turn itself off when a voltage lower than 5.2 V is detected (the electrical scheme to achieve the said functionality will be described in the power switch, subsubsection 5.2.8.3).

Despite that, it might be bothersome and inconvenient if the device suddenly shut down, without any previous warning, in the middle of the sea. Due to this, as previously mentioned, we will make a state of charge estimation but, how will this be carried out?

One of the simplest and more used methods is the **integration of the instantaneous current** defined by the following equations:

The electrical current is defined as shown in Equation 5.2.5.

$$I = \frac{Q}{t} \rightarrow I(t) = \frac{dQ(t)}{dt} \tag{5.2.5}$$

Solving this equation we obtain Equation 5.2.6, the charge inside the battery, where Q_0 is the charge at t_0.

$$Q(t) = \int_0^{t_1} I(t)dt + Q_0 \tag{5.2.6}$$

This theoretical value must be adapted to a real project in which the value of the current flowing would not be constantly known but discretely depending on the baud rate of the sensor and the processing speed of the microcontroller. Due to that, this equation must be transformed in Equation 5.2.7, where k is the number of iterations and T the sampling rate.

$$Q(kT) = \sum_{k=1}^{n} I(kT) * T + Q_0 \tag{5.2.7}$$

This could not be the most accurate method to obtain this parameter, nevertheless, we will obtain an estimated approximation valuable enough to give indications about the battery state.

5.2.8.3 The power switch

Another important feature that must be included in our device is a power switch. Thanks to which the microcontroller is provided with the capability of being turned on or off by the user as well as by itself, in the same way that nowadays smartphones have a power button.

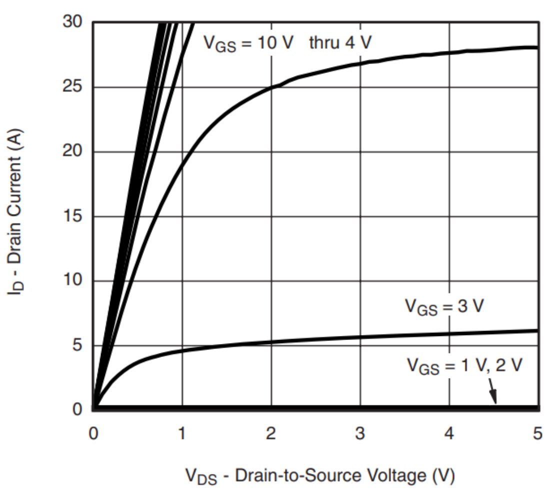
This functionality is included by implementing a pass transistor which will act as a switch by letting the current flow as desired. To control it we must implement a circuit that allows modifying its V_{sg} (voltage between source and gate), which behaviour is described by the Table 5.15 [28] [2].

Operation mode	**Voltage**	**Current**
Cut off	$V_{sg} < \lvert V_{th} \rvert$	$I_d = 0$
Linear	$V_{sg} > \lvert V_{th} \rvert, V_{sd} < V_{sg} - \lvert V_{th} \rvert$	I_d proportional to V_{sg} and V_{sd}
Saturation	$V_{sg} > \lvert V_{th} \rvert, V_{sd} > V_{sg} - \lvert V_{th} \rvert$	aprox. constant (I_d proportional to V_{sg})

Table 5.15 – *Operation modes of a P-channel MOSFET*

5

This operating modes can be described through the Figure 5.33, obtained from the chosen power pass transistor's datasheet.



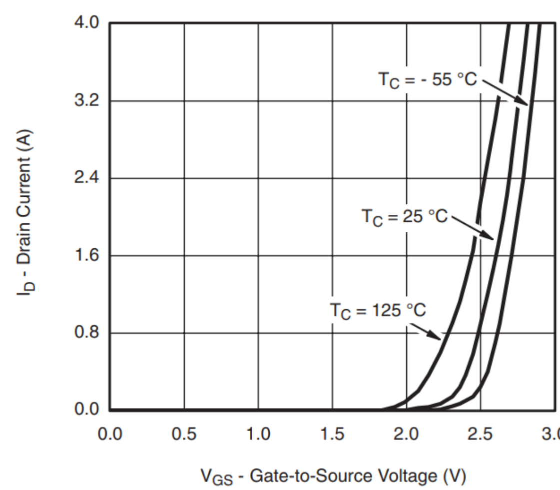


Figure 5.33 – *Output characteristics of a power pass transistor*

By analyzing these graphs we can see that, at the practical level, we will have a switch that will let the current flow if V_{sg} is higher than 2 - 3V.

As can be noticed, a **p-channel MOSFET has been chosen**, as usually in power pass transistors. This is due to that, while in a p-channel MOSFET current flows from the source to the drain, in an n-channel MOSFET current flows in the opposite direction, from the drain to the source. This means that for a n-MOSFET the source will have to be connected directly to the load.

This shows a clear problem: as the load will vary depending on the processor load, the actuators, etc, the source voltage will vary as well and so the current output will change too, creating irregularities in the delivered power. On the other hand, for a p-MOSFET the source will be fixed at the battery voltage which implies that V_{sg} will be fixed and so the power source will be stable.

As commented, the source voltage (V_s) will be the battery voltage, between 5 and 7V. For this reason, taking into consideration that a V_{sg} higher than 2 - 3V is needed to let the current flow, we will design a circuit to set in V_g 2 or fewer volts. We can achieve that with the circuit shown in Figure 5.34

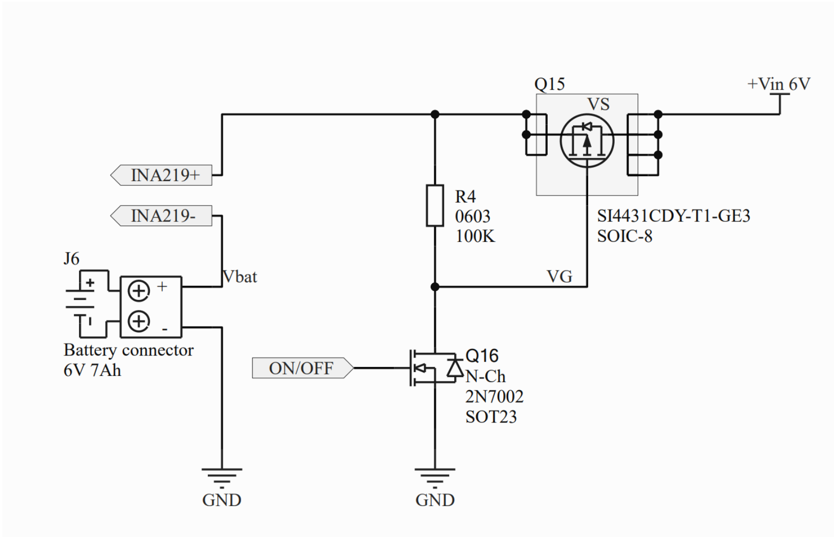


Figure 5.34 – *First version of the power switch*

Thanks to this circuit is possible to control V_{sg} so that if the *ON/OFF* pin is set at low level (0V) then V_g will equal to V_s, the battery voltage, and so $V_{sg} = 0$V. On the contrary, if the pin is set at high level (3.3V for our microcontroller) then V_g will approximately 0V and so V_{sg} will be the battery voltage, 6V, letting the current flow.

Nonetheless, the ***ON/OFF* pin must be modifiable by the ESP32 but also by the user** through a button. This means that we must keep completing the functionalities of the power switch. For that purpose we designed the circuit shown in Figure 5.35, in which we include a button with a pull-down resistance and a capacitor to avoid the switch bouncing.

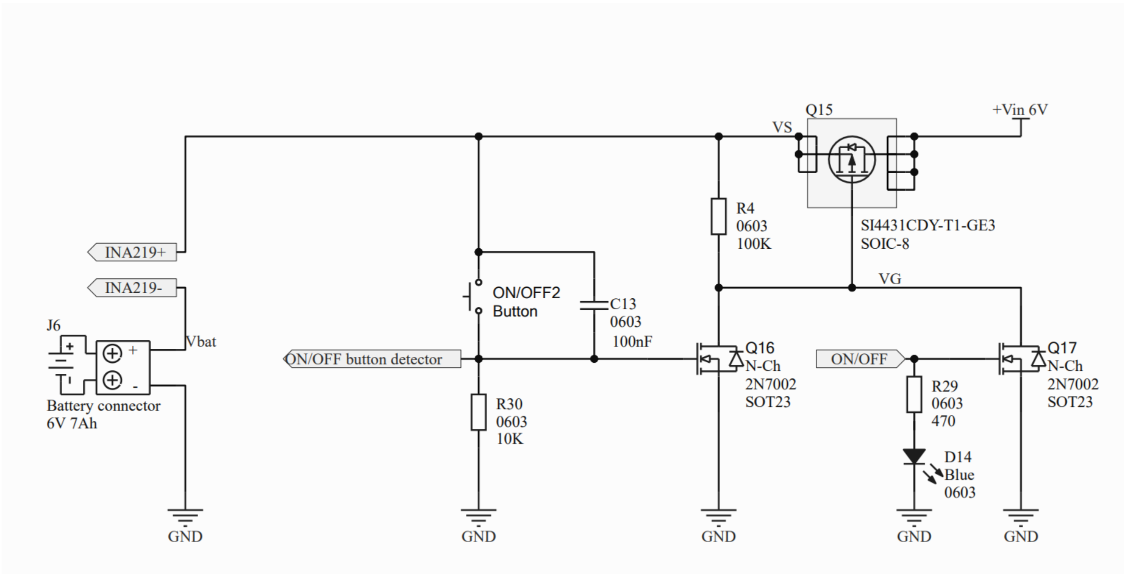


Figure 5.35 – *Second version of the power switch*

This button will let the current flow as long as it is pressed, however, if it is released the current flow would stop. This problem is solved keeping V_g at 0V by the microcontroller. This implies that this buttons must be pressed till the ESP32 initialize and set the *ON/OFF* pin at low level, after which the button can be released and the current will keep flowing.

In addition, once V_g is set at 0V by the microcontroller, the button could not stop the current flow by itself. Due to that the *ON/OFF button detector* pin has been added as an input in the ESP32 to detect it and to stop the current flow from the microcontroller when this event took place.

5

Finally, we noticed that the voltage at this *ON/OFF button detector* would be the battery voltage when pressed, a too high value for our microcontroller which could in fact suffer damages. For this reason a tension divider has been added to reduce this voltage. All this culminates with the circuit shown in Figure 5.36.

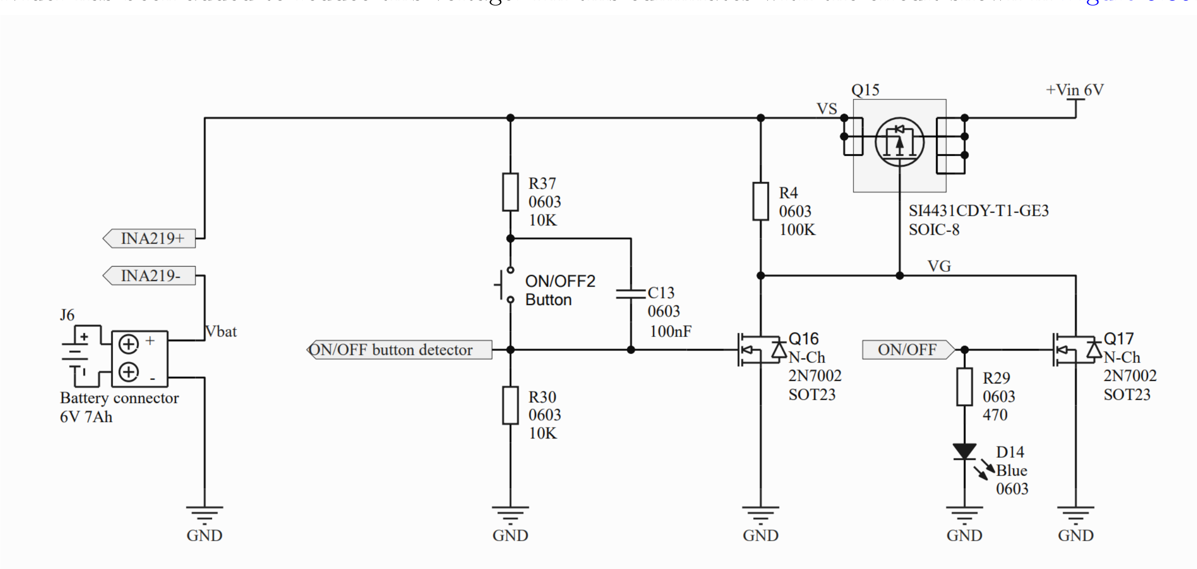


Figure 5.36 – *Final version of the power switch*

To verify that the functioning of the theoretical design carried out is correct **we decided to simulate its behaviour in a SPICE-like simulator**, which can be seen in Figure 5.37. The simulation goes through different stages:

1. Firstly, the ON/OFF button is pressed during 3.5ms. During this time V_{sg} grows up to 5V, letting the current flow through the pass transistor and so obtaining a little bit less than 6V in the output. In addition, during this time the ESP32 starts its initialization.

2. Secondly, after 3ms the ESP32 finalizes its initialization and set the ON/OFF pin at high level what allows the button to be released while keeping the V_{sg} at 5V.

3. Finally, the microcontroller decides to turn itself off by setting the ON/OFF pin at low level and so stopping the current flow.

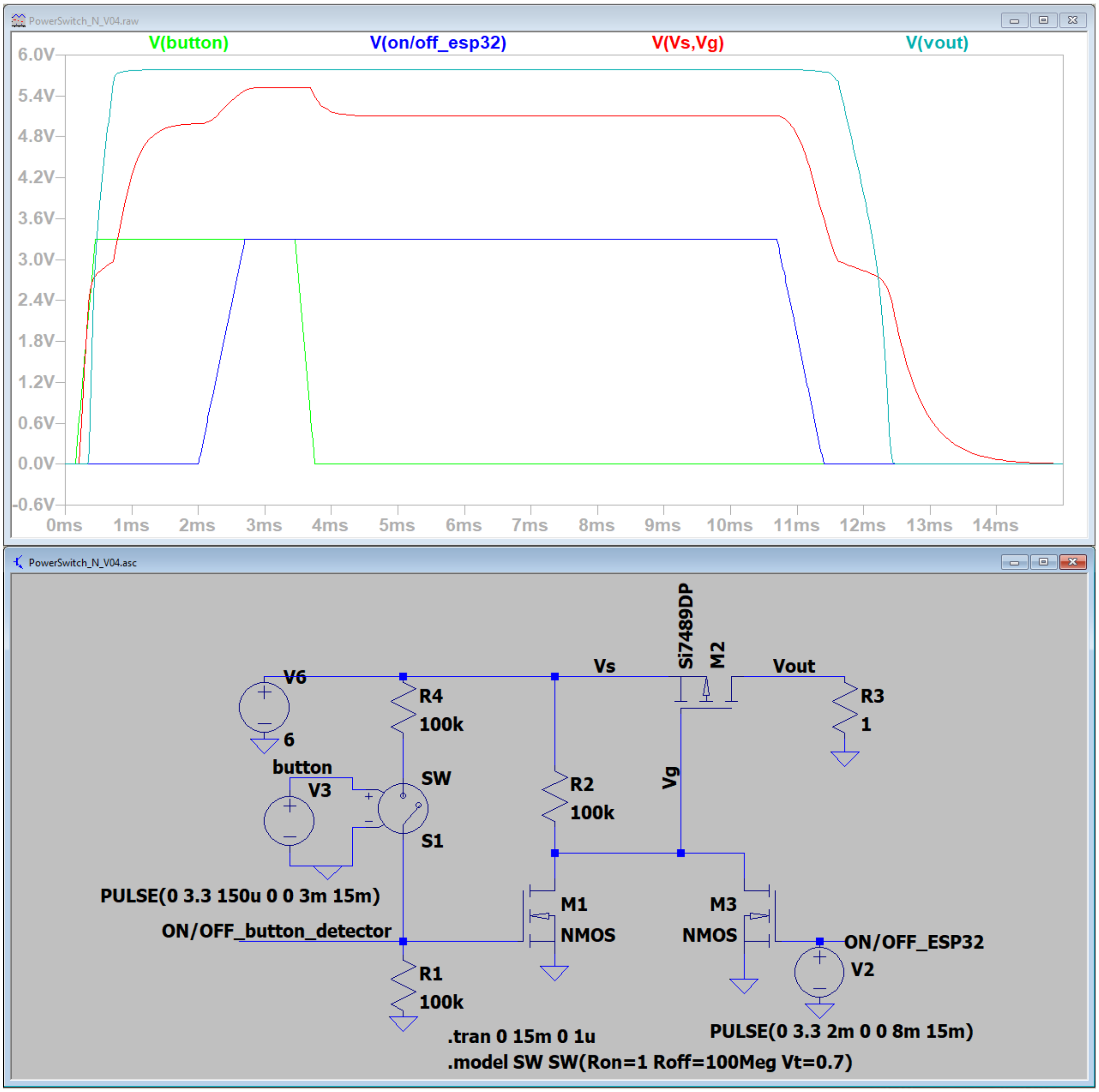
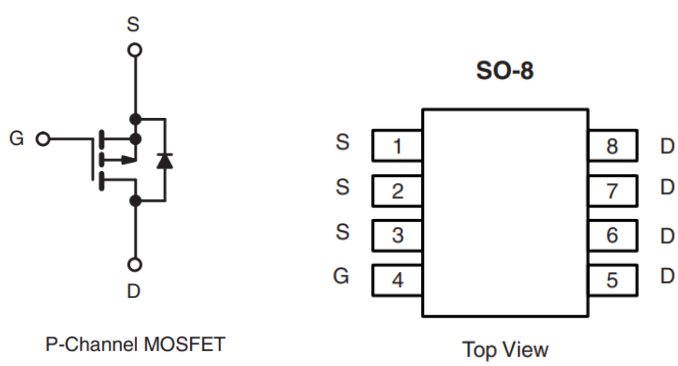


Figure 5.37 – *Simulation of the power switch*

All is already defined regarding the schematic, nonetheless we must **select the components** that will live up this schematic. The transistors that control the pass transistor's gate are the default low power transistors for our project, already previously mentioned in the programming interface,subsection 5.2.2, the **2N7002**. Regarding the pass transistor we have selected the **Si4431CDY** (datasheet in [32]) which is available in the lab and adapts perfectly to our needs. Its characteristics are the previously shown in Figure 5.33 and in Table 5.16.

Name	**Si4431CDY**
Type	MOSFET
Channel	P-channel
Vds max	-30 V
Id max	9 A
Max. power disspation	4.2 W
Rds max	0.05 Ω @Vgs = -4.5V
Packaging	SO-8
Price	0.90 € URL

Table 5.16 – *Si4431CDY P-MOSFET*



(a) *Packaging* **(b)** *Functional diagram*

Figure 5.38 – *Si4431CDY P-MOSFET*

5.2.8.4 The battery charger [5]

After that, we decided to embark on the design of our own battery charger for the project. To begin with, the first thing we need to know is how a lead-acid battery is charged.

Although most of charger provide a constant current phase and a constant voltage phase, not all of them are equal and their functioning depend on the type of battery. Each battery can be used in 2 different modes This modes will define the working conditions and so the charge and discharge parameters implement. They are:

- **Standby mode**: In this mode the battery is used within a narrow voltage range either charging or discharging it. In online Uninterruptible Power Supply (UPS) or relevant devices, batteries are used in standby mode. In this way, the battery is continuously charging, and at the same time, it is discharging just like a capacitor. Here this 6V battery's datasheet indicates that in standby mode, the terminal voltage should be 6.8 V. Otherwise, the battery may get heated and dried inside or even the battery plates may get damaged. Nevertheless, for our specific case, this mode of operation will not be implemented as we will be working with a vehicle disconnected from the network.

- **Cycle mode**: In cycle mode, the battery is charged and then disconnected from the charger. Afterward, the battery is only discharged (down to a certain level) when it is required. Once the battery is discharged, we need to charge that battery again to the full charge level. This is cycle use. This operation mode will be the employed in our project. [18]

Regarding the charging process, for a lead-acid battery the charge is based in three states which will be described for a cycle use. They are:

1. Firstly, **a constant current phase**, that the manufacturer recommends setting between 0.05C and 0.3C which makes the battery voltage grow gradually until it reaches the overcharge voltage (Voc). This Voc is usually set as 2.45 V per cell.

2. Then, **a constant voltage phase** starts at the overcharge voltage. In this state the current input decrease slowly as the battery starts to act as it had a higher input resistance and not to accept more charge. This phase ends when the current descends to the overcharge terminate current (Ioct), usually C/50.

3. Finally, The **float stage** commences, phase in which the charge voltage is reduced to approximately 2.25 V per cell and held constant, while the current is reduced to less than 1% of battery capacity. This mode is specially useful to maintain a fully charged battery indefinitely since this kind of battery has a high auto-discharge rate.

In addition, some chargers add a trickle phase previous to starting the charge. The output current in this phase is limited to a low-level until the battery reaches a specified voltage, preventing a high charging current if the battery is shorted or damaged. This information can be observed in the battery datasheet [19] as well as in most of lead-acid charger datasheets as will be shown later.

This stages can be described by the following graph, Figure 5.39 obtained from the datasheet [29] of a battery charger IC, the UC3906.

5

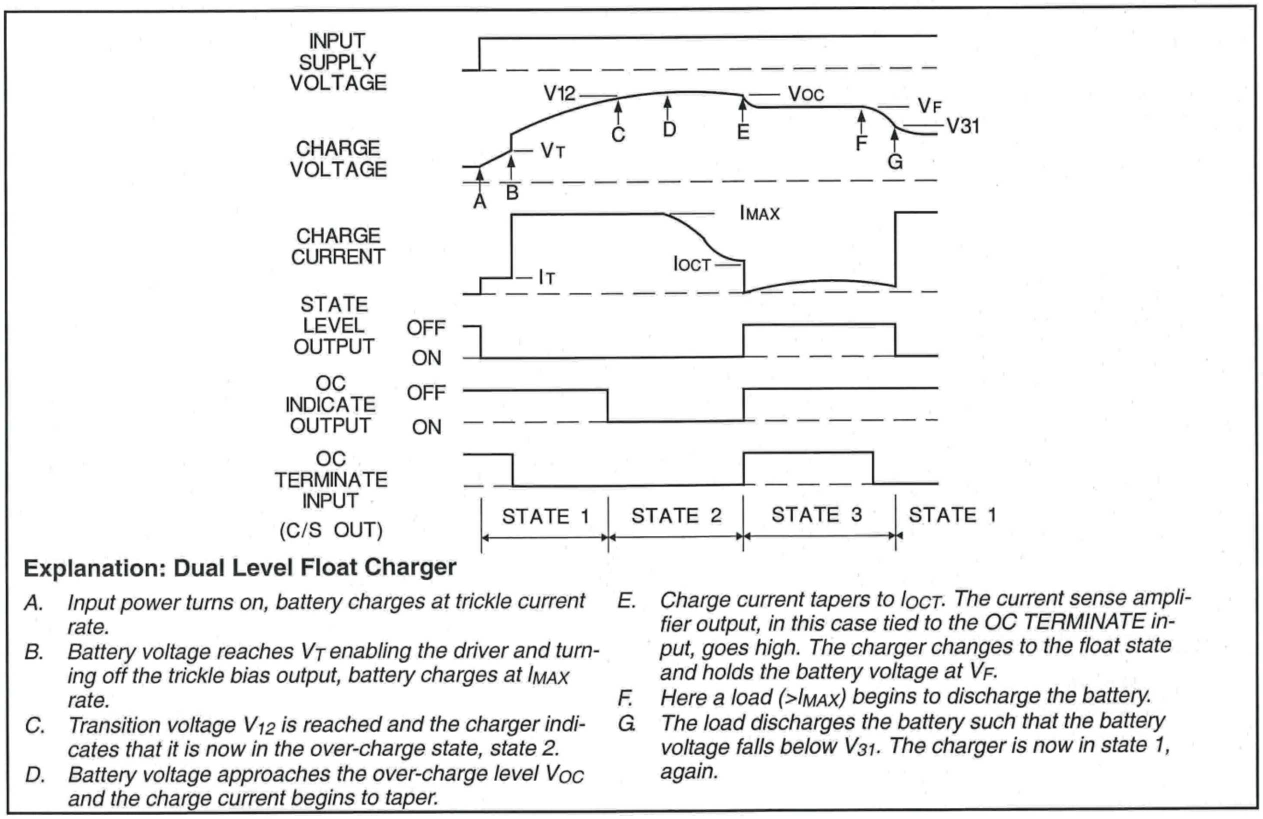


Figure 5.39 – *Lead-acid battery charging process*

Taking into consideration these theoretical aspects of the charging process and the technical characteristics of our battery we can now define the parameters of our charger:

- Start-up trickle current (It): 10 mA.
- Start-up voltage (Vt): 5.0 V .
- Current during constant current phase (Imax): 1.05 A (0.15C).
- Overcharge voltage (Voc): 7.35 V (2.45 V/cell).
- Overcharge terminate current (Ioct): 140 mA (C/50).
- Float stage voltage (Vf): 6.75 (2.25 V/cell).

To create these voltage and current outputs it exists several configurations but they will be defined by the controller IC:

Name	BQ25723 Datasheet	LT1512 Datasheet
Type	Buck-boost converter based	Buck-boost converter based
Switching frequency	800-kHz/1.2-MHz programmable	500kHz
Input range	3.5 V ∼ 26 V	< 30 V
Output current	up to 16.1 A	1.5A
Number of cells	1 ∼ 4	any
Extra features	I2C	-
Price	4.03 € URL	9.98 € URL
Our choice	×	×

Table 5.17 – *Battery charger ICs*

Name	UC3906 Datasheet	BQ24450 Datasheet
Type	pass-transistor based	pass-transistor based
Input range	<40V	<40 V
Output current (to control transistor's base)	25 mA	40 mA
Extra features	state indicator outputs	-
Price	11.95 € URL	11.06 € URL
Our choice	✓	×

Table 5.18 – *Battery charger ICs*

The **BQ25723** is a very modern device that includes the latest technology for battery charging with new features as I2C, which allows the management of the charge and discharge of the battery, a specially useful characteristics. Despite this, this IC is built in a package too small to be manufactured in out lab due to that we had to dismissed it.

The other components selected are simpler devices which fulfills its function with almost any special feature. We must take into account that in an suitable charger we should have LEDs to indicates whether the device is fully charged or not and if there is an input power supply. Due to that, we decided to choose the **UC3906** which has state indicator outputs. This outputs can be attached to a comparator, creating a circuit to turn on a "fully charged" LED as shown the typical application that can be seen in Figure 5.40, obtained from an application note provided by the manufacturer [30].

5

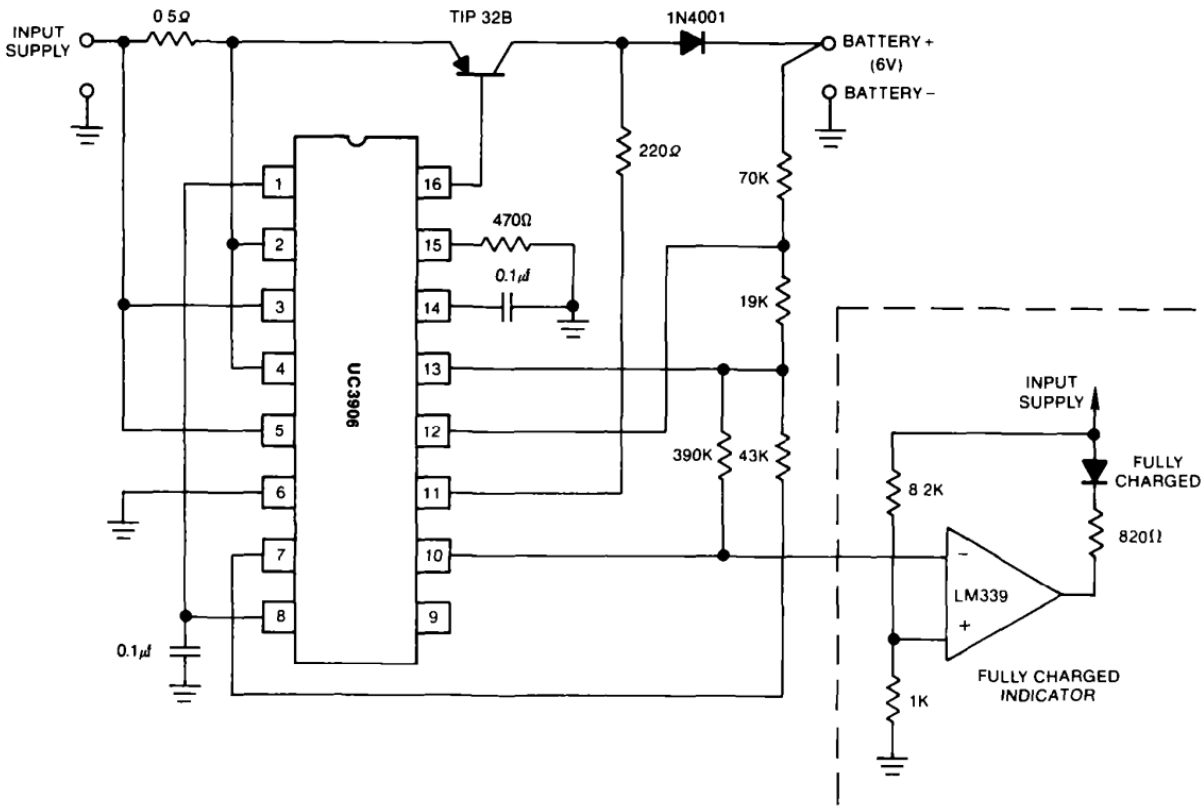


FIGURE 13. This dual level float charger was designed for a 6V (three 2V cells) 2.5AH battery. A separate “fully charged” indicator was added for visual indication of charge completion.

Figure 5.40 – *UC3906 typical application with a "fully charged" LED*

After that we proceeded to the design phase according to the datasheet and the application note information, which are shown in Figure 5.41 and Figure 5.42.

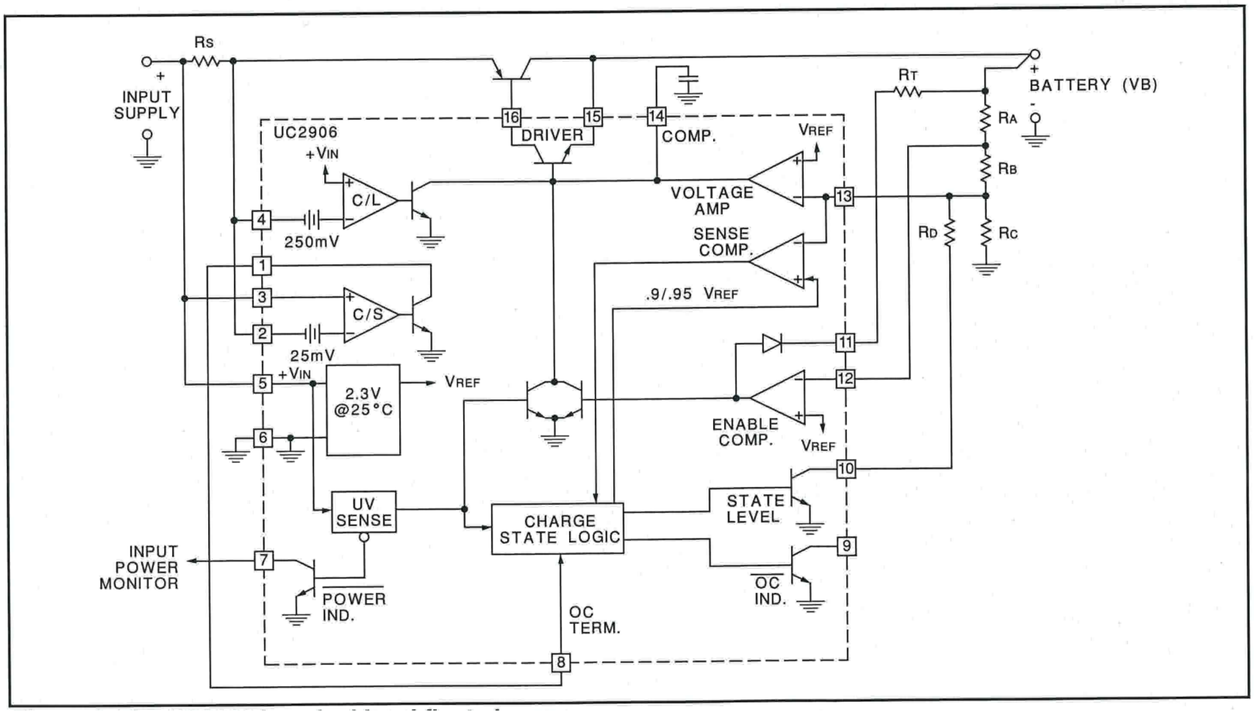


Figure 5.41 – *UC3906 typical application*

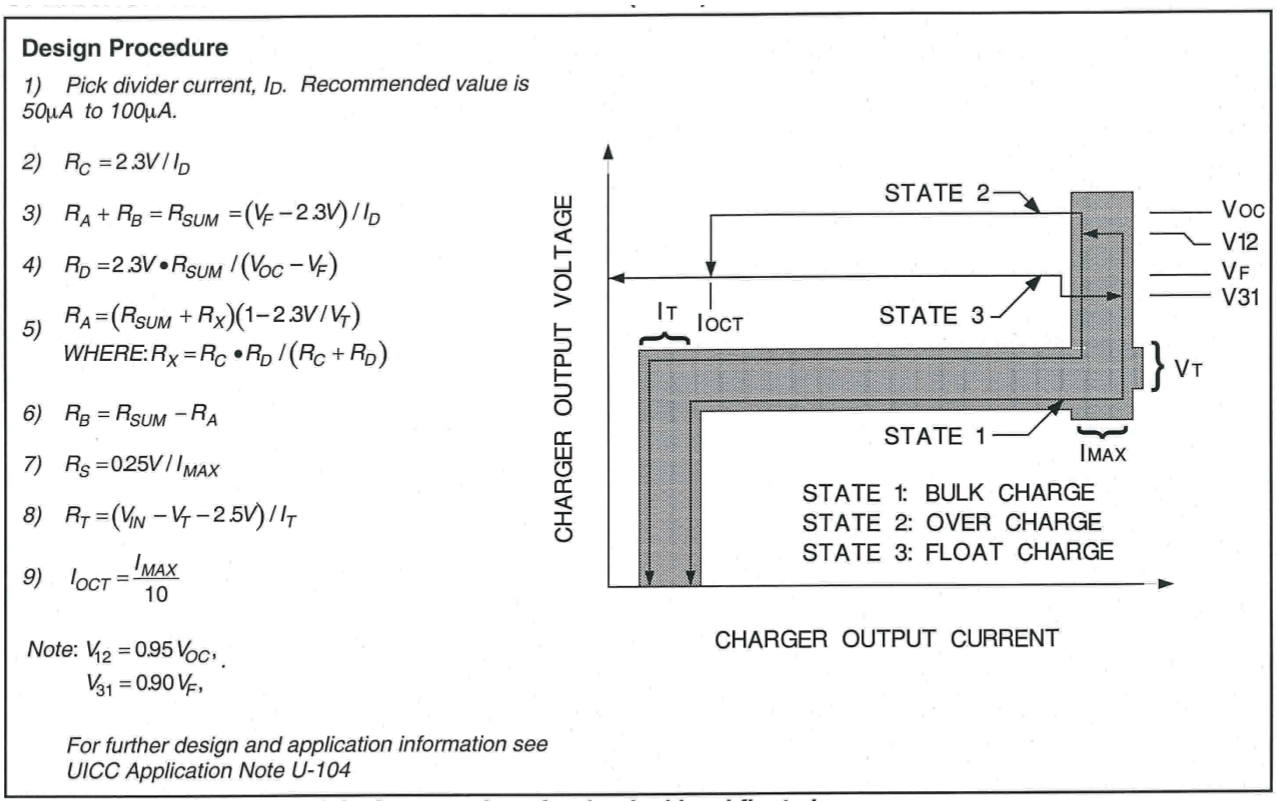


Figure 5.42 – *State diagram and design equations for UC3906*

Once the design was finished we decided to order the components, however, we ran into a problem. We suddenly noticed that these devices selected due to their simplicity and size to be manufactured in GranaSAT are also old-fashioned devices which are no longer used on current devices due, in fact, to their excessive simplicity and size for cutting-edge projects. For this reason, these devices are not available in almost any supplier stock and those few that still have it in stock sell it at a pretty high price for this type of device owing to the low demand.

This made us rethink the design and we decided to implement a similar circuit based on a pass transistor but this time, as we have at our disposal a powerful microprocessor and a current and voltage sensor for the battery, we decided to do the regulation of the charging process with the ESP32 through its firmware.

To achieve that we need the following characteristics:

- **An external voltage source** which should be higher than the maximum voltage value that the battery might reach, this is the overcharge voltage (Voc): 7.35 V. Conversely, it shouldn't be too high to facilitate the power regulation through the pass transistor and to avoid damaging the battery. this means that it should be between 7.5 V and 12 V approximately.
- **An input voltage sensor** thanks to which the ESP32 will be able to identify when the charger has been connected to initialize the charge. This has been implemented through a simple tension divider as shown in Figure 5.43

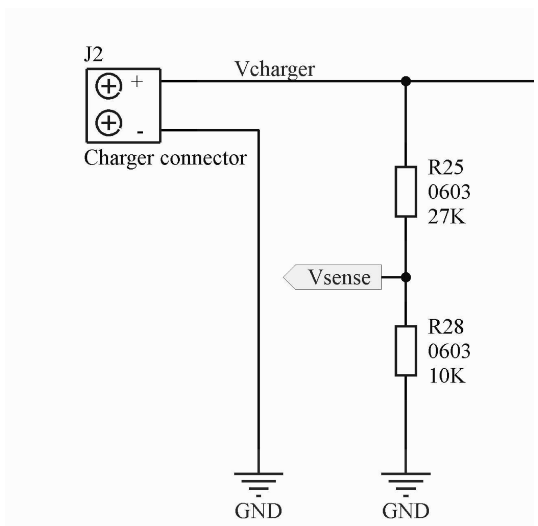


Figure 5.43 – *Tension divider for the voltage sensor*

The resistor value has been chosen for an important reason. The ESP32 only support inputs from 0 V to 3.3 V while a proper input voltage to charge the battery must be higher, as previously mentioned, so we designed the voltage divider to transform those 12 V into a maximum value of 3.3 V as follows.

On the basis of the tension divider equation, Equation 5.2.4, we obtain:

$$\frac{V_{sense-max}}{V_{in-max}} = \frac{3.3}{12} = 0.275 = \frac{R_{25}}{R_{25} + R_{28}} \tag{5.2.8}$$

From where we get:

$$R_{25} = 2.63 * R_{28} \tag{5.2.9}$$

To fulfil this equation, the resistor values might be $R_{25} = 10k\Omega$, $R_{28} = 26.3k\Omega$. Although, as $26.3k\Omega$ is not an standard value, we used the most similar standard value: $R_{28} = 27k\Omega$. With these value the maximum input voltage will be 12.2 V.

- **A pass transistor** thanks to which we will control the input current to the battery and voltage making use of a configuration such as a common drain. This device will be the same as the one used for the power switch owing that its available in the lab and fulfills all our requirements: **Si4431CDY** [32] (Table 5.16).

To verify if the design of the charger is correct **we simulated it in a SPICE-like simulator**, whose results can be seen in Figure 5.44.

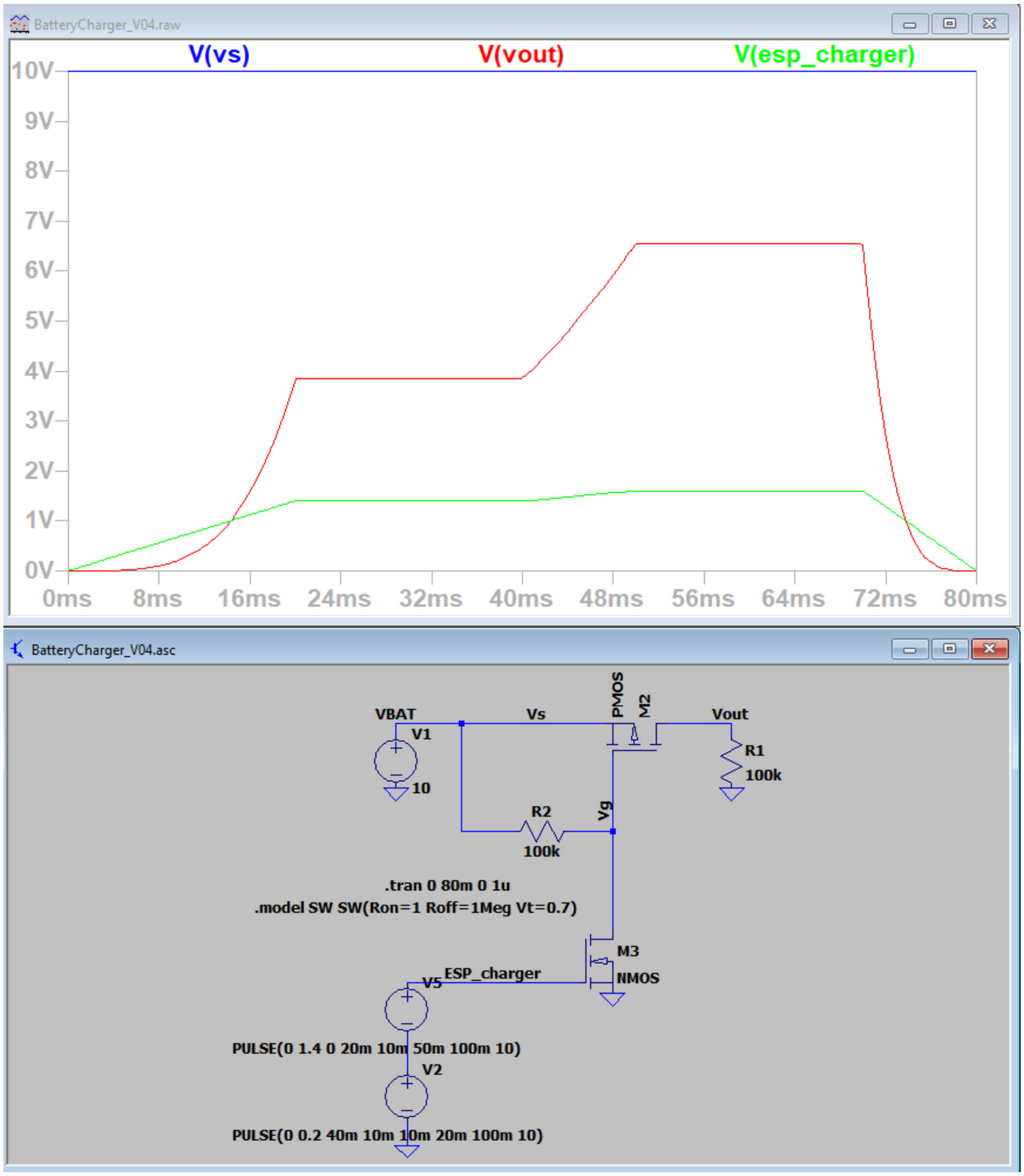


Figure 5.44 – *Charger circuit via ESP32 - SPICE simulation*

As can be seen, we can modify the output voltage through the pass transistor which is controlled with voltages between 0V and 3.3V from the "Charger" pin of the ESP32.

Finally, both the charger and the power switch schematic will be unified in a single schematic sheet as is shown in Figure 5.45 as well as in the Altium Designer® files, Appendix E. To make this unification without creating any conflict between the circuits several power diodes have been implemented.

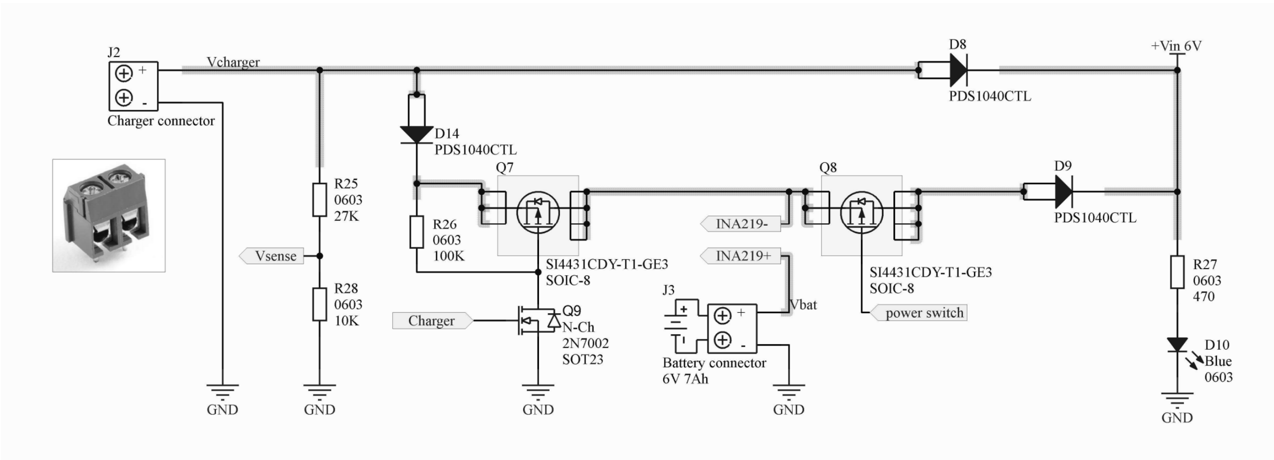


Figure 5.45 – *Power supply - battery charger and power switch*

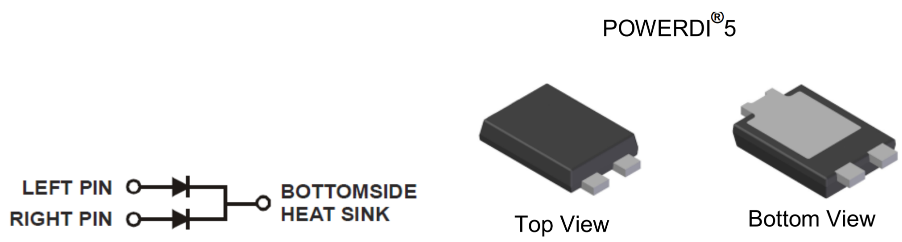
These diodes mainly fulfill two protective functions, on the one hand, they protect the rest of the circuit against an inverse polarization in case the battery or the charger is connected with its polarity inverted by error. On the other hand, they prevent the current to flow from the battery to the charger or to other parts of the circuit while it is been charged which would affect the charging process.

5

The diodes selected for this purpose are the **PDS1040CTL** (link to datasheet), whose technical characteristics can be seen in Table 5.19.

Name	**PDS1040CTL**
Type	SCHOTTKY diode
Vf max	0.5 V @If = 5 A, Ts = +25ºC
If max	10 A
Packaging	POWERDI5
Price	1.57 € URL

Table 5.19 – *PDS1040CTL SCHOTTKY diode*



(a) *Pinout* **(b)** *Packaging (POWERDI)*

Figure 5.46 – *PDS1040CTL SCHOTTKY diode*

This device was chosen owing to that it can deliver up to 10 A, which is clearly enough for our project, and that it is available in our lab stock.

5.2.8.5 The voltage regulators

As shown and commented in the power budget, subsubsection 5.2.8.1, the different devices that will compose our PCB have different needs regarding the power supply, due to that the following voltage supplies are needed:

- A **3.3V** source to feed:

Component	**Imax** (mA)	**Ityp** (mA)	**Pmax** (mW)
ESP32	500	500	1650
OLED SSD1306	20	15	66
GY-NEO6MV2	115	50	379.5
Adafruit 10 dof	30	30	99
DS1302	1.5	1.5	4.95
INA219	1	1	3.3
Led SMD 0603 x 3	15	15	49.5
Total	**681**	**611**	**2247,3**

Table 5.20 – *3.3V working components*

- A **4V** source to feed:

Component	**Imax** (mA)	**Ityp** (mA)	**Pmax** (mW)
SIM800L	1500	350	6000
Led SMD 0603 x 1	5	5	20
Total	**1505**	**355**	**6020**

Table 5.21 – *4V working components*

- A **5V** source to feed:

Component	**Imax** (mA)	**Ityp** (mA)	**Pmax** (mW)
Servomotor	500	100	2500
Led 5mm x 4	40	40	200
Led SMD 0603 x 1	5	5	25
Total	**545**	**155**	**2725**

Table 5.22 – *5V working components*

- A **6V** source, this will be the primary source. This voltage is already obtained from the battery, consequently, no voltage regulators will be necessary in this stage. Nevertheless, as previously mentioned, we must have into consideration that this voltage will vary between 5 and 7V.

Now that we know the requirements of every component we can move on to the election of the voltage regulators, but first to all, what is a voltage regulator?

A voltage regulator is a device that is able to maintain a stable voltage source independently of the device load and within a wide range of input voltage. This features must be implemented in most electronic devices since they are usually designed to be powered at predetermined voltages.

In this field it exists three main architectures of voltage regulators which are going to be discussed:

1. **Linear regulators**: are simple transistor-based devices packaged as ICs. Their internal circuitry uses differential amplifiers to control output voltage against a reference voltage. They can provide a fixed output or an adjustable value, usually modifiable through a potentiometer or variable resistor. In addition, owing to their operating principle, they are step-down converter so by definition the output voltage is always below the input voltage. Among their main advantages we can find:

   - A Low cost.
   - A Negligible output voltage ripple.
   - A high reliability
   - A smaller size.
   - They are generally easy to design.

   However, these regulators also have a few disadvantages:

   - They typically require the input voltage to be at least 1V above the output voltage.
   - A quite low efficiency.

   The voltage drop of a linear regulator is comparable to a voltage drop across a resistor. Due to their operational principle, these regulators fix the output voltage at the same time that delivers almost the same current in the output and the input. This implies that, while in the power input there will be $V_{in} * I_{out}$, in the output will be $V_{out} * I_{out}$, creating a power loss that depends on the voltage difference between the input and the output as described in Equation 5.2.10.

$$\eta(efficiency) = \frac{P_{in} - P_{loss}}{P_{in}} * 100\% = \frac{P_{out}}{P_{in}} * 100\% = \frac{V_{out} * I_{out}}{V_{in} * I_{out}} * 100\% = \frac{V_{out}}{V_{in}} * 100\% \quad (5.2.10)$$

This efficiency is usually between 30% and 70%, a rather low value. For example for our project, considering to use it as a 3.3V regulator the efficiency will be the shown in the following Equation 5.2.11:

$$\eta = \frac{V_{out}}{V_{in}} * 100\% = \frac{3.3V}{6V} * 100\% = 55\% \tag{5.2.11}$$

2. **Linear low-dropout regulator (LDO):** these regulators are actually included in the linear regulators category, nevertheless, they count on a special characteristic which makes them different to other linear regulators. It is that they can keep properly working even when the supply voltage is very close to the output voltage, up to 100 mV and, therefore, this involves a quite lower power loss. Although they are slightly more expensive than regular lineal regulators, they include the same other advantages such as size, low output ripple, etc. Despite this, they are still not as efficient as the switching regulator.

3. **Switching regulators** [16]: are based in a circuit that uses a power switch, an inductor, and a diode to transfer energy from input to output. The power switch is turned on and off by a switching controller IC that monitors the output of the switching regulator in a feedback control loop. Switching regulators can be step-down converter, step-up converter, or a combination of both, which makes them more versatile than a linear regulator. This principle implies a series of advantages:

   - Very high efficiency, from 80% up to 99%.
   - More flexibility, allowing to increase or reduce the input voltage.
   - They don't need a certain voltage difference between input and output.
   - Due to the higher efficiency they can usually deliver a higher output current.
   - Their output voltage is always adjustable.

   And, of course, they have several additional disadvantages:

   - Higher cost.
   - Not Negligible output voltage ripple.
   - Require several external components which implies a higher size and a higher difficulty of implementation.
   - Higher complexity to control and to design due to that requires selecting external component values, tuning control loops for stability, and careful layout design.
   - The switching usually produces noise.

[20]

Taking into account all this consideration, we have enough knowledge to select the different voltage regulators for our project. By extrapolation, we can notice that for a project like ours, which depends on a battery, **the efficiency is one of the main factors to consider**, for this reason we initially selected **switching regulators**.

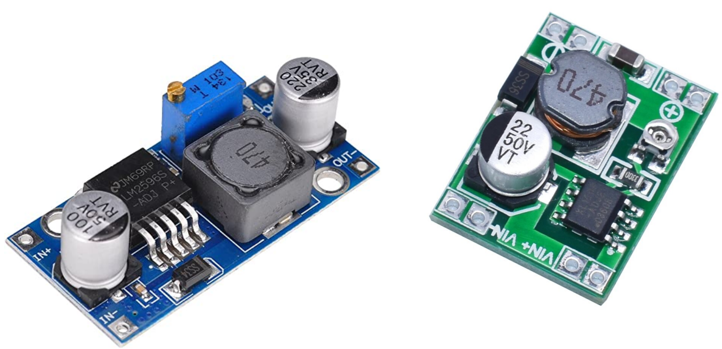
Moreover, requiring the input voltage to be at least 1V above the output voltage might probably be an important constraint. In particular, for the 5V regulator it surely would be a problem since the battery voltage would vary between 5 and 7V. Due to that, for this voltage output the use of a switching voltage regulator becomes indispensable.

To simplify the design process we decided to use modules which already include all the necessary components to make the regulation:

Name	LM2596 DC-DC module	XL1509 DC-DC module
Type	switching regulator	switching regulator
Conversion	step-down converter	step-down converter
Input voltage	4V ∼ 35V	4.5 ∼ 40V
Output voltage	1.23V ∼ 30V	1.27 ∼ 37V
Output current	3A (max)	2A (max)
Conversion efficiency	92% (max)	96% (max)
Output voltage ripple	<30mV	<30mV
Commutation frequency	150 KHz	150 KHz
Size	43mm * 21mm * 14mm	22.5mm * 16.75mm * 4mm
Price	0.80 € (+1.33€ of shipping cost) URL	0.94 € URL
Our choice	×	✓

Table 5.23 – *Switching regulator comparison*

The **XL1509 DC-DC module** was chosen for its smaller size, lower price and higher efficiency. Although the maximum output current is smaller, 2A are enough for the power supply needs of our components.



(a) *LM2596 DC-DC* (b) *XL1509 DC-DC*

Figure 5.47 – *Switching regulator comparison*

After that and despite the previously mentioned reasons, we considered whether it would be **appropriate to use a linear regulator or not**. By analysing the voltage source previously commented we realised that it could be possible and advantageous to use a linear regulator for the 4V source since the voltage difference between input and output is low. Besides that, the SIM800L is a device that has consumption peaks when the signal is being transmitted but that it has a quite low consumption the most part of the time in consequence the power loss in the voltage regulator will be low.

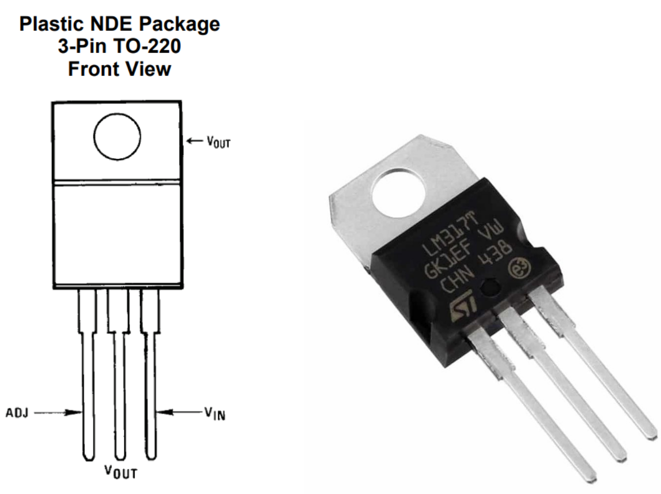
This was our initial thought, although later we find out that the voltage drop in the pass transistor, the shunt resistor and the diode implemented after the battery could be decisive since it could transform the voltage difference in the regulator from the 1V needed to a lower value and therefore hinder it from working properly. Hence, it becomes necessary to analayse the voltage drop and the voltage difference that the regulator needs more closely.

As the 4V needed is not a standard value, there isn't any fixed linear regulator with such output voltage. For this reason, we move on to consider adjustable linear regulators. In the lab stock we could find the

LM317 (datasheet in [31]), the same used in the previously done prototype (see subsection 2.2.1). Its technical characteristics are shown in Table 5.24.

Name	LM37
Type	Adjustable linear regulator
Conversion	step-down converter
Input voltage	4.25V ∼ 40V
Output voltage	1.25V ∼ 37V
Output current	2.2A @(Vin - Vout < 15V)
Conversion efficiency	highly dependent on V_{in} and V_{out} ($\frac{4V}{6V} * 100 = 66.6\%$ for our application)
Output voltage ripple	none
Accuracy	1%
Size	(TO-220) 19.25mm * 9.9mm * 4.5mm
Price	0.388 € URL

Table 5.24 – *LM317A technical characteristics*



(a) *Pinout* **(b)** *Packaging (TO-220)*

Figure 5.48 – *LM317A*

Nonetheless, the information provided by the datasheet regarding the dropout voltage needed didn't seem precise enough for this application. For this reason, and since we have a prototype which already includes this module, we simply tested it experimentally. In this way we discovered that the SIM800L module stops working correctly from an input voltage in the LM317 of 4.8 V or less.

Even so, we still have the question of how much voltage there would be in V_{in} after the voltage drop in the diode, the pass transistor and the shunt resistor that connects to the battery. For these reasons and since we are talking about a prototype design, we finally decided to make a double implementation to check its operation in real conditions. This means that **we will have both regulators, the LM317 and the**

XL1509 DC-DC, on the PCB of our device to obtain a 4 V output.

A diagram of all the previously commented components regarding the power budget and their interconnections is shown in Appendix B which surely will clarify the aspect of the design to the reader.

5.2.9 Effect of the regulators in the consumption

Finally, we must take into account that the previously mentioned **switching voltage regulators won't have the same input current and output current**, in fact, as they are step-down converter, the input current will be lower so **the overall current consumption will be lower**. This current can be calculated by the principle that the input power multiplied by the efficiency must be equal to the output power, this is shown in Equation 5.2.13.

$$V_{in} * I_{in} * \eta = V_{out} * I_{out} \tag{5.2.12}$$

$$I_{in} = \frac{V_{out} * I_{out}}{V_{in} * \eta} \tag{5.2.13}$$

We will estimate a efficiency of 90% in the worst case, so knowing the input voltage, the output voltage, the output current and this parameter we can proceed to estimate the input current consumption of each switching voltage regulator.

Vin (V)	**Vout** (V)	**Iout max** (mA)	**Iout typ** (mA)	**Iin max** (mA)	**Iin typ** (mA)
6	**3.3**	681	611	416	373
6	**4**	1505	355	1115	263
6	**5**	545	155	505	144

Table 5.25 – *Switching regulators input current consumption*

5

And from those values we can know calculate the overall maximum consumption:

Component	**Imax** (mA)	**Ityp** (mA)	**Pmax** (W)
DC motor	5000	2000	30
Led SMD 0603 x 1	5	5	0.030
3.3V components	416	373	1.372
4V components	1115	263	4.460
(with linear regulator)	(1505)	(355)	(6.020)
5V components	505	144	2.525
Total	**7041**	**2785**	**38.387**
(with linear regulator)	(7431)	(2877)	(39.947)

Table 5.26 – *Overall current consumption*

5.2.10 Schematic overview

It is still worth mentioning various aspects to take into account once considering the project as a whole and so the interrelation between the different parts.

The I2C communication needs a pull-up resistor in each one of the two wires that allow the communication (SDA and SCL) which are recommended to be of 10 KΩ. As we are working with modules, the three devices that use this type of communication, which are the IMU (Adafruit 10-dof), the display (oled ssd1306) and the voltage and current sensor (INA219); already include these resistors. This means that the resistors are in parallel which would divide its value in 3 obtaining an equivalent value of 3.3 KΩ so to ensure an adequate functioning we should desolder two of the three pars of resistors in these modules.

In addition, it is convenient to emphasise that the adjustable switching voltage regulators, which are tuned through a potentiometer, must be set at its desired value before its implementation to avoid damaging any component while doing the tuning.

All the previously related information is finally transformed into **16 schematics sheets**, which are shown in Appendix E. Through them all the concepts are unified to create a functional design. **We recommend paying special attention to the first sheet** which shows a diagram of the overall project as well as the interconnection between the different parts.

In addition, a **BOM has been created to list all components** with which our project can be implemented, this is shown in Appendix A.

5.3 PCB design

Finally, to finish with the hardware, a PCB design must be made to transform all the ideas previously shown in the schematic design into a real product.

5.3.1 The board shape

To begin with, the first aspect we have to look at is how much space we have available inside our boat and how will the PCB be integrated within that space.

The boat has a hollow interior in which there is enough space available, however, we must access to it through an opening located in the upper part of the same as shown in Figure 5.49, which reduces the suitable space.

On top of that, the battery must be placed in the same space and for this purpose the chassis includes some sockets so that a plate to hold the battery can be screwed (see Figure 5.50).

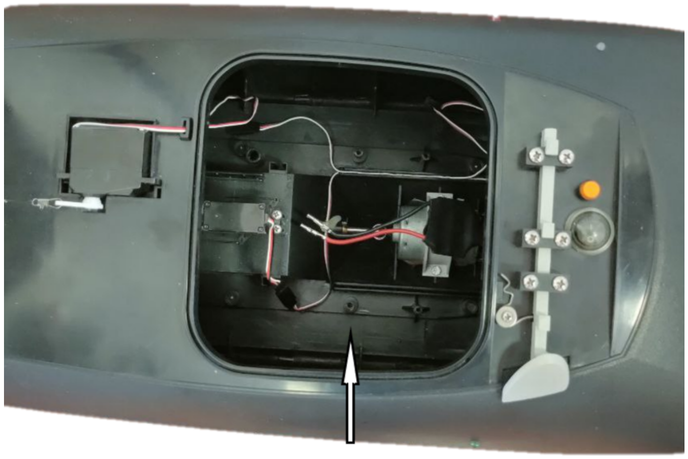
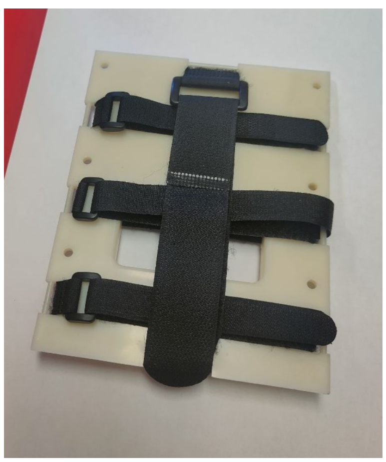


Figure 5.49 – *Boat opening view with sockets*



(a) *Without battery*

(b) *With battery*

Figure 5.50 – *Plate to place the battery*

5

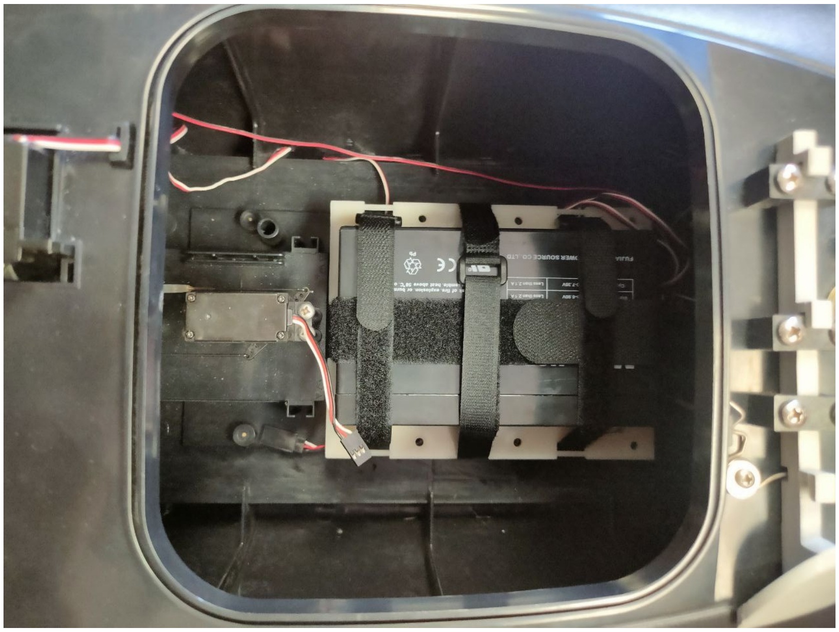


Figure 5.51 – *Battery placement inside the boat*

Once already inside the subject of the placement of the plate, we must take into account that the battery must be accessible to replace it if it was needed. This means that placing the PCB over the battery won't be a suitable option. In addition, as the board must be fixed to the boat, we should try to make use of the included sockets to facilitate the implementation.

Due to these constraints we decided to **place it vertically on one side of the boat**, just next to the battery. On the contrary, the sockets are vertically placed as well, therefore, to solve this inconvenience, we decided to implement **two boards creating a single "L" shaped board** composed od a horizontal board fixed to the sockets and a vertical board fixed to the previous board. It is easier to be shown than described so it is exhibited in Figure 5.54.

For the definition of its limits, the interior of the ship has been measured in such a way that the maximum space available and accessible inside the ship is taken into account. Later, starting from this maximum board size, we would implement the schematics reducing its size as much as possible, always within the limits.

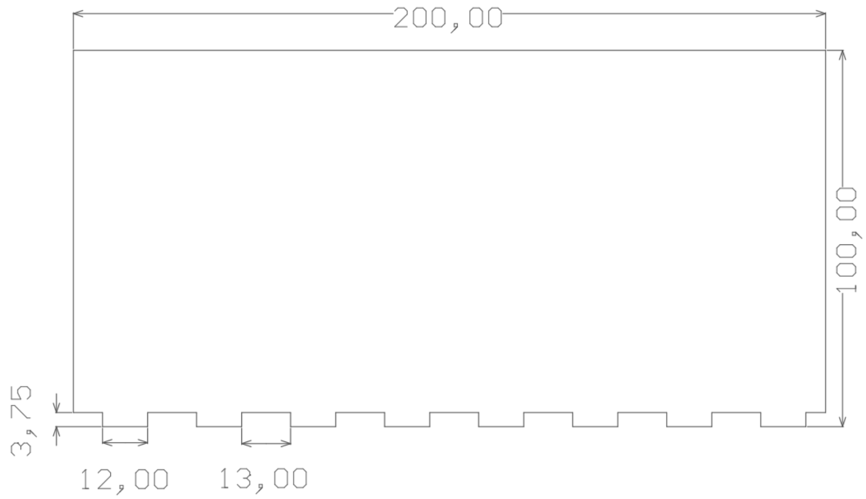


Figure 5.52 – *Vertical board (control board)*

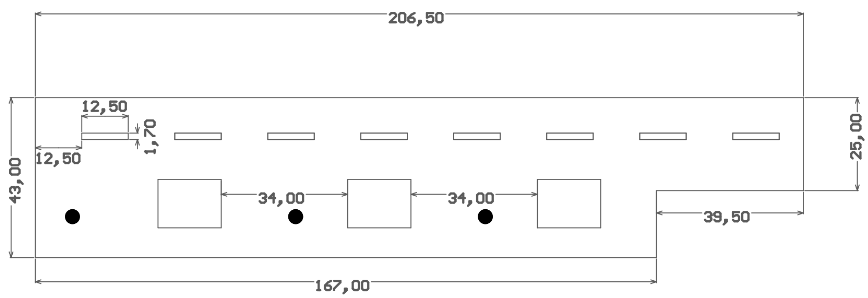


Figure 5.53 – *Horizontal board (auxiliary board)*

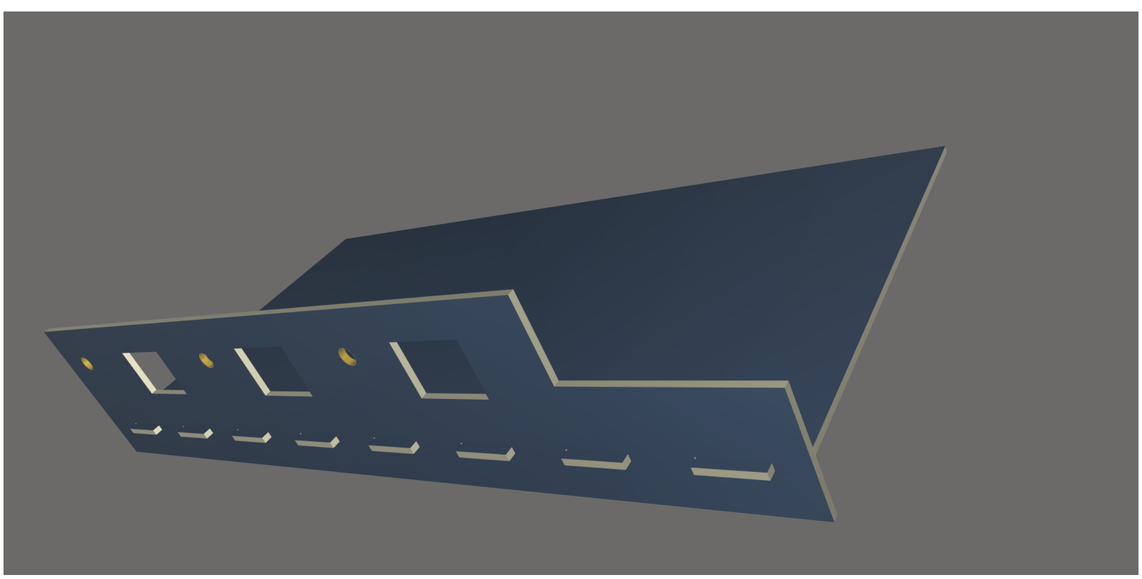


Figure 5.54 – *3D assembly of both boards*

As both boards would have a copper layer we will solder the junction between both to increase the mechanical resistance.

5.3.2 Placing the components

To initiate the component placing, **we first must have the footprint and 3d model of all of the chosen components**. This task, although it might seem simple, was one of the most time-demanding tasks within the PCB design since our device includes a wide variety of components from SMD devices or through-hole to modules. In fact, sometimes the footprint and the 3d model could not be found in the same web page so, in the end, we had to browse a wide variety of pages such as GrabCAD, 3D ConcentCentral, SnapEDA, Octopart, etc. And even after that, some of them could not be found and had to be made by ourselves such as, for example, the XL1509 buck regulator module or the LM317, which con be found but none of the models had the pins bent at a 90º angle as our project required so we had to modify them with the help of SolidWorks.

This is probably the most important part of the PCB design. The placement will define the subsequent track routing, if it is made improperly it could lead to an unroutable layout, to manipulable components located in inaccessible areas or to obtain a design bigger than needed which will mean a higher material and manufacturing cost. Due to that we must keep these constraints in mind:

1. The following components must be placed on the sides or the top to be able to be manipulated or visualized:
   - All the indicators LEDs.
   - The display.
   - The battery, motors and charger terminal connectors.
   - The two-pins connectors for the chassis LEDs.
   - The USB port.
   - The buttons must not only be placed properly but also they must be surrounded by enough free space to be pushed with the fingers.
   - The rotary.
2. The ESP32 must be placed on the top of the board with the antenna pointing up to increase its range and sensibility. In addition, nothing can be placed or routed where the antenna is placed, in any of both sides of the PCB, to avoid interferences.
3. Screws should not have tracks below since, as they are conductors, they could create a shortcut between tracks. Although regarding the spacers, these can be found in plastic versions so we would not encounter the same problem.
4. We must consider how components on one face could affect components on the other, this refers to through-hole components or screwed components where certain parts will rise above the board. To reduce the board size we find out a way to overcome this constraint, this is by rising the device whenever it was necessary by the use of spacers or pin headers though this can only be done with screwed modules. In addition, as the modules include holes for 3mm screws which are quite big compared to the total size of our board, we decided to use fewer screws per module in such a way that they stay fixed thanks to one or two screws and the pin headers.

The placing process is completely related to the routing process so it is always necessary to do big or small modifications to facilitate the track routing. For example, we initially placed the 4 V voltage regulators next to the SIM800L but far from the battery terminal, this implied a longer power track to supply this regular.Thus, as the power tracks are quite wide as we will explain later, it became quite hard to route so we decided to change its position to move it closer to the battery terminal.

5

5.3.3 Routing the components

Finally, once the placing has been finished, we can move on to route the components. Nevertheless, the connection diagram is not the only necessary information to begin the routing, we must also know the size of the different elements of the PCB that will interconnect the implemented devices. Due to that we firstly will begin calculating the size of the tracks.

What we are interested in is knowing the calculation to determine the width that the trace must have to support a certain current. This trace will dissipate heat, generating a thermal increase. This heating effect is actually what matters to us for practical sizing purposes. This is why, to calculate a current carrying capacity, it must be analyzed in terms of temperature increase, setting the maximum allowable temperature increase.

Some of the parameters that alter or modify the thermal behavior of the track, affecting the increase in temperature are:

- Electrical current
- Type of the base material
- Track section
- Thickness of the copper laminate
- PCB thickness
- Presence of ground planes or large areas of copper.
- Application environment (housing, air circulation by fans, vacuum, etc.)

The calculation is complicated due to that there are many elements involved. Therefore, the standard was set by means of tests and presented in the form of curves. Thanks to these empirical data, with few data, the width of the runways can be determined fairly accurately. We must bear in mind that the width values obtained are minimal and it is always advisable to oversize everything that the design allows us.

To calculate the minimum width of a certain track we need to know three pieces of information:

- Maximum current that might flow through the track.
- Maximum temperature increase that track can withstand. Allow at least a 10°C temperature rise in any application.
- Copper thickness of the material used for the track.

The maximum temperature increase allowed is expressed in degrees centigrade. The calculation is based on setting a maximum admissible temperature variation. Thermal variation is defined as an increase in temperature above the initial temperature experienced by the conductor (track). The initial temperature of the printed circuit is considered equal to the temperature of the working environment, as long as the power dissipation of the components is low and the circuit operates in a normal environment. If this is not the case, the working temperature of the printed circuit must be taken into account, which is what will be taken as ambient temperature.

It is important to make it clear that the initial temperature is not the environment surrounding the PCB, but rather the working temperature of the material itself.

Therefore, the maximum temperature increase allowed always refers to the ambient temperature. Thus, if we design our circuit to work at an ambient temperature of 50ºC and we want the track temperature to never exceed 60ºC, the maximum temperature increase allowed must be 10ºC.

Regarding the third and last piece of information, we must bear in mind that we must not confuse the terms thickness or thickness and track width. In a printed circuit we normally have tracks of different widths, but all of them are of the same thickness. The thickness refers to "the height" of the track relative to the material that serves as the base on the printed circuit board.

In the Figure 5.55 one the track width is shown while in the Figure 5.56 tracks with the same width but different thicknesses are shown.

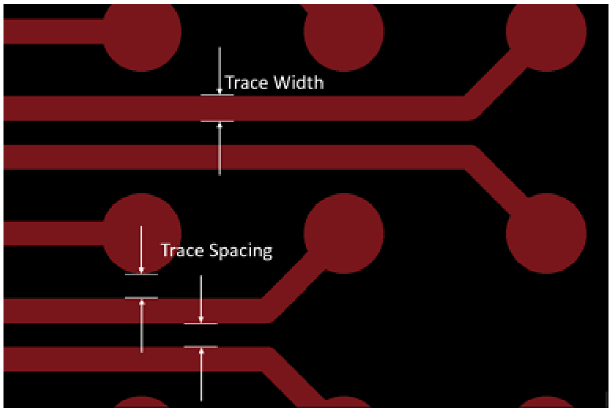


Figure 5.55 – *Track width*

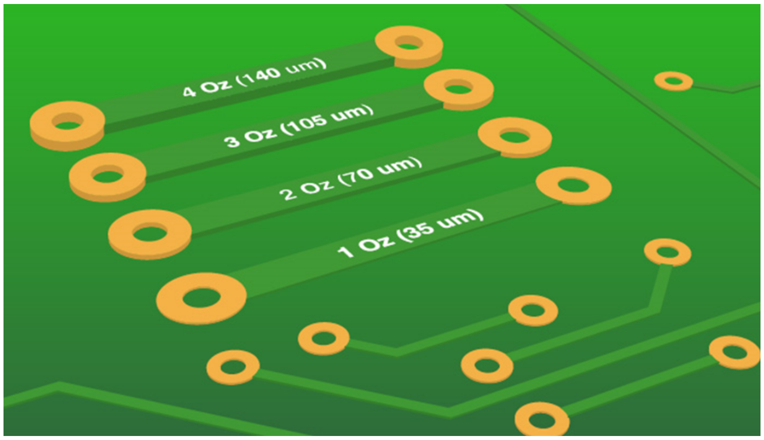


Figure 5.56 – *Track thickness*

There are four thickness standards used in the commercialization of the plates: 18µm (0.5oz/ft2), 35µm (1oz/ft2), 70µm (2oz/ft2), 105µm (3oz/ft2) and 140µm (4oz/ft2), among them the one found in our laboratory materials is **18µm (0.5oz/ft2)** . Another variable is the base material thickness or plate thickness which includes both the fiber and the copper. Some common commercial values are 1.6mm and 2mm, in this case we will be using **1.6mm**. Regarding the temperature on the surface of the PCB, it could be measured through a thermal camera to obtain a precise value however we will estimate that in working conditions it would be 30ºC, with a temperature increment of 30ºC to avoid exceeding 80ºC, temperature at which some IC might suffer damage. [12] [3]

5

After, knowing all the parameters that define the minimum width we proceed to calculate its value. To achieve this we will use online tools specialized in the calculation of this parameter, specifically, a calculator provided by Digikey.es from which we acquire the data information shown in Table 5.27. However, the routing size of a track is defined not only by its typical value but also by its maximum and minimum acceptable value, this will allow a higher flexibility since we could adapt the size to wider or more narrow zones without increasing excessively the track impedance.

Current	Width	Standardized width	Name used in nets
5A	2.84 mm (111.8 mils)	2.8 mm (110 mils)	POWER-5
1.5A	0.54 mm (21.2 mils)	0.5 mm (20 mils)	POWER-1.5
500mA	0.12 mm (4.66 mils)	0.127 mm (5 mils), 0.2 mm in lab (8 mils)	-

Table 5.27 – *track width as a function of the current*

All this information must be contrasted with the capabilities of the manufacturer to verify its fabricability, in case of sending the design to a professional, for example to **JLPCB** which design constraints can be found in jlpcb/capabilities. Here, according to the discussed topic, the minimum track width can be found which value is 5mil (0.127mm) with a spacing of 5mil (0.127mm) as well. Comparing this value with the previously calculated in Table 5.27, this implies that for tracks with currents of 500 mA or lower the track width must be the mentioned minimum.

On the contrary, if the design is made in the GranaSAT lab, some of the most important constraints are:

- Minimum track width of 8 mils (0.2 mm).
- Minimum spacing of 8 mils (0.2 mm)
- Drill size for vias of 0.7 - 1.2 mm, with steps of 0.1 mm.
- Inability to manufacture an overlay so the silkscreen won't be shown.

For this overlay we will use the previously mentioned JLPCB capabilities, which are:

- Minimum line width of 6 mils (0.153mm).
- Minimum text height of 40 mils (1.0mm).
- Minimum pad to silkscreen distance of 0.15mm.

These constraints will have to be configured in Altium Designer® to be able to perform a design rule check which will verify that our design is correct. They can be set by hand, as it would be needed for the GranaSAT lab constraints, or imported through a ".rul". In the case of JLCPB they can be imported from the rule files which can be found in jlpcb/how-to-export-altium-pcb-to-gerber-files.

All the previously mentioned theoretical aspects of the routing process have been obtained from class notes of the subject "Prototipado y test electrónicos" from the degree in industrial electronic engineering and from the subject "Printed circuit technology" from the degree in telecommunications engineering.

Knowing all the constraints and recommendations for a proper design we move ahead to design our PCB and after up to 6 different board versions, **we obtained our final design, which is shown in the appendix, Appendix F.**

To verify if this final design fulfills the space constraints inside the boat chassis **we decided to print the PCB on paper to create a copy of the model in cardboard, the so-called pretotype** so we could test it experimentally and so we could see that it was correct as can be seen in Figure 5.57, Figure 5.58 and Figure 5.59.

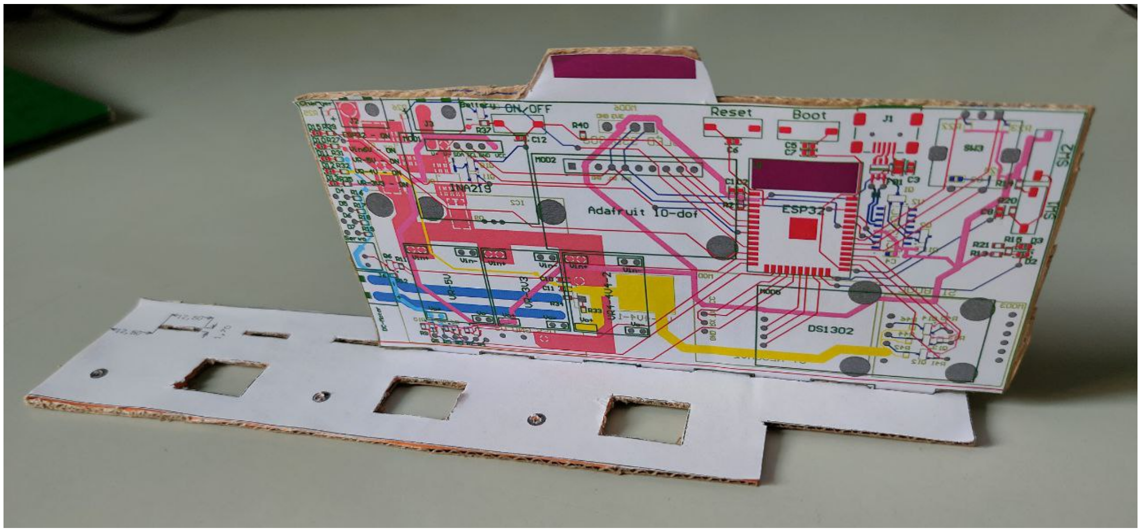


Figure 5.57 – *Pretotype*

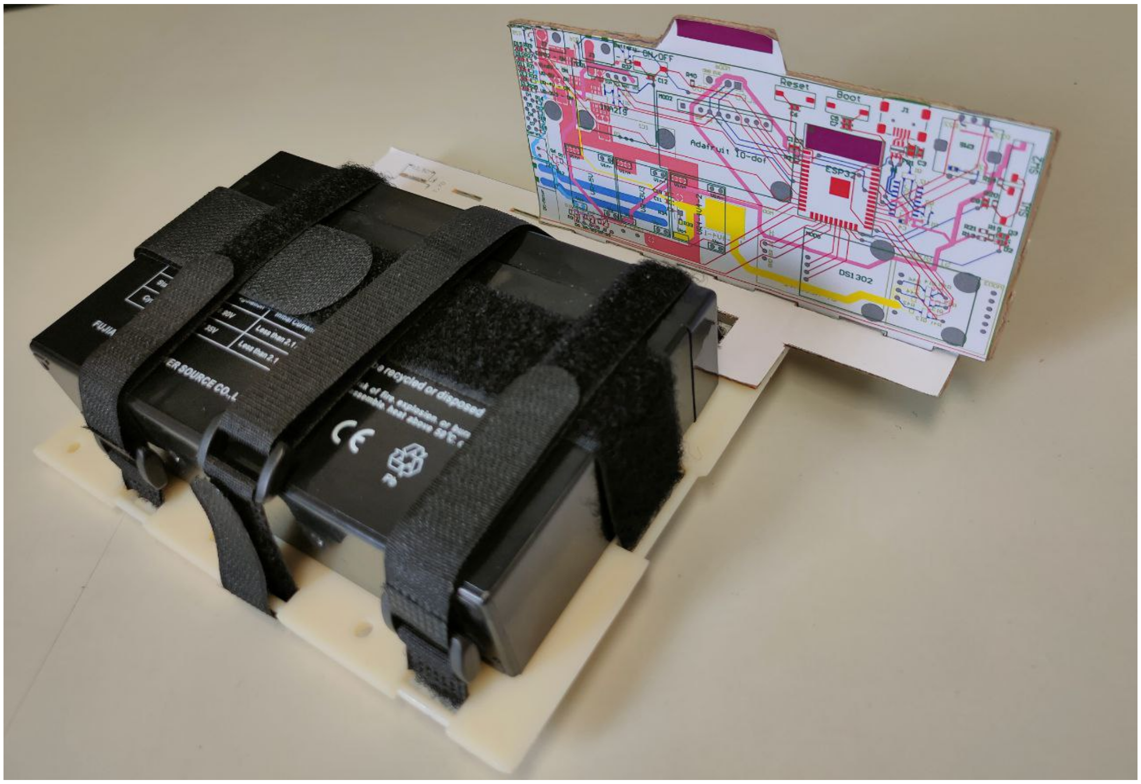


Figure 5.58 – *Pretotype along with the battery*

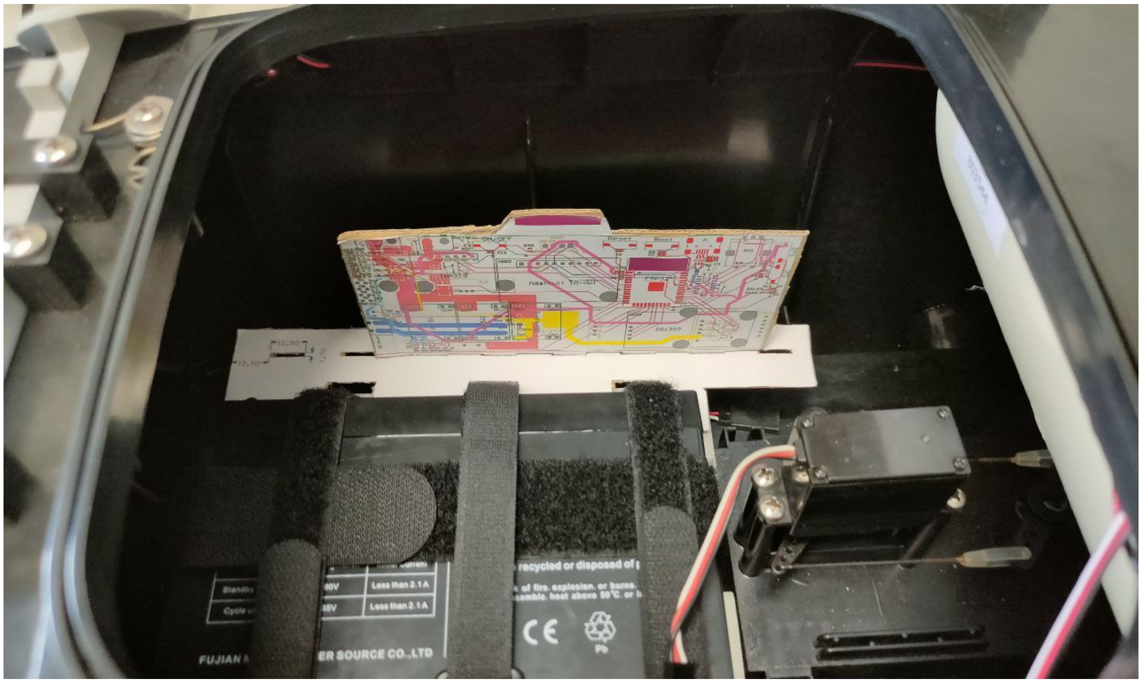


Figure 5.59 – *Pretotype inside the boat*

5.3.4 PCB manufacturing

After finishing the design the next step is manufacturing the board, however we have not had enough time to carry out this task. Therefore, purely for demonstrative purposes we will show how to request the product to be manufactured online, in this case by JLPCB.

Firstly we have to export the *GERBER* and *NCdrill* files from the Altium project. These will determine the copper layers manufacturing and the drills that must be made to connect the through-hole devices and the screws respectively. After that, we head to the JLPCB's web page, jlcpcb.com, and click in the *"Order now"*. From this, the web page shown in Figure 5.60 will appear.

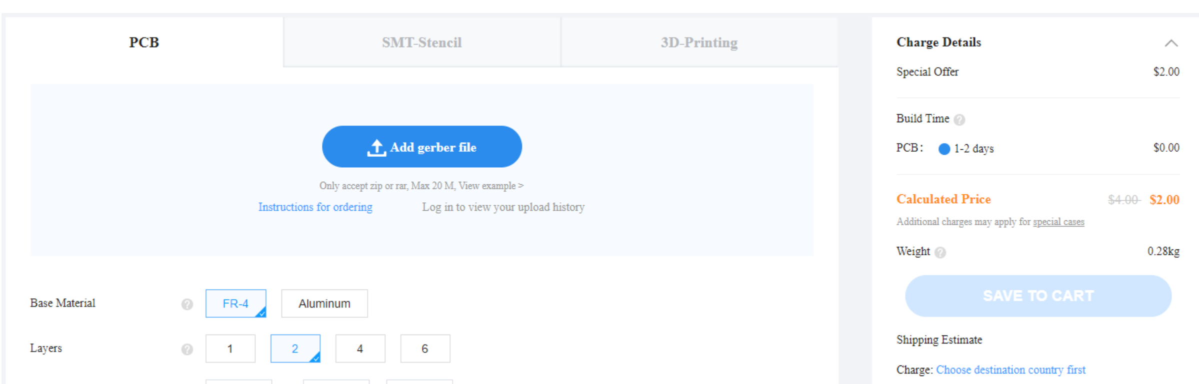


Figure 5.60 – *JLPCB order tab*

Here, we must upload our *GERBER* and *NCdrill* files without modifying the manufacturing parameters since the default parameters of JLPCB actually are the same used in our board design. From this we obtained the board price that can be seen in Figure 5.61 and Figure 5.62

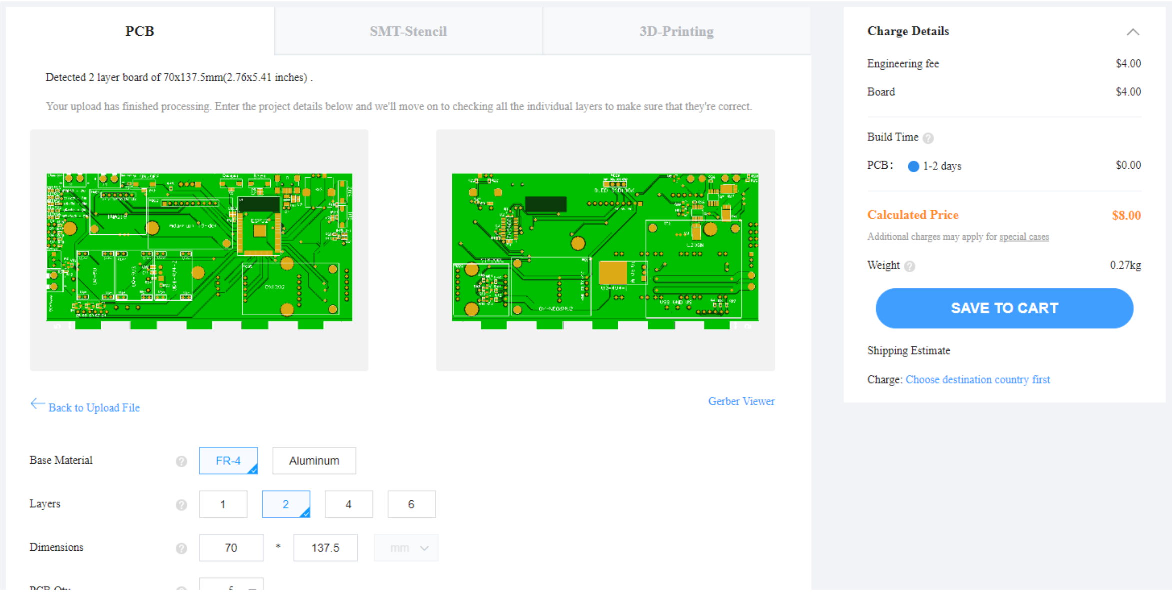


Figure 5.61 – *JLPCB - control board price*

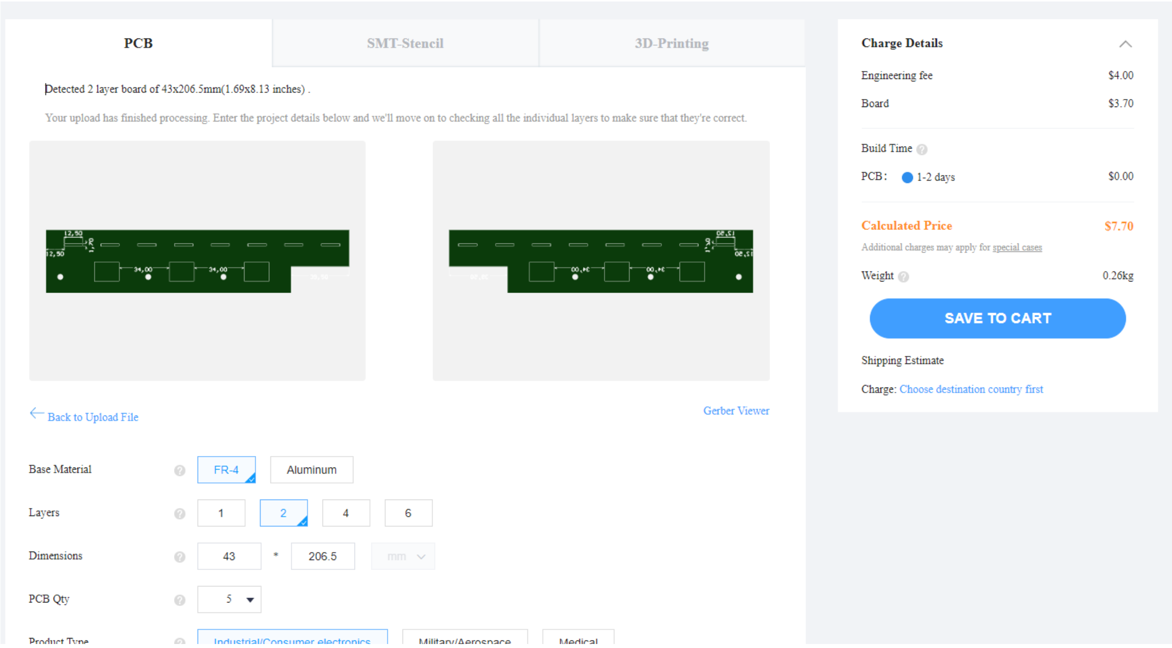


Figure 5.62 – *JLPCB - auxiliary board price*

5

5.4 Project budget

To end up with the project, once the schematic and PCB design is already finished, we must create a project budget in the same that it would be done in a professional project.

All the costs related to the work executed to design an upgraded version of the *Pondskater ORCA* bait boat in GranaSAT are shown in this chapter. They are split into the control board related costs, Table 5.28, and the total costs, Table 5.29.

Item	Cost per unit (€)	Quantity	Total cost (€)
PushButton	0.15	5	0.75
Capacitor 0603	0.16	12	1.92
USBLC6-2SC6	0.63	1	0.63
Blue LED 0603	0.33	6	1.98
Green LED 0603	0.33	1	0.33
5MM LED (2 male headers)	0.011	4x2	0.088
PDS1040CTL	1.57	3	4.71
Ferrite Bead Chip Fair-Rite 2508056007Y3	0.03	1	0.03
L298N	1.47	1	1.47
USB-MINI-B Female	0.32	1	0.32
Charger connector	0.57	1	0.57
Battery connector	0.57	1	0.57
Motor output connector	0.57	1	0.57
Servomotor connector (3 male headers)	0.011	3x1	0.033
INA219	1.78	1	1.78
Adafruit 10DOF	8.35	1	8.35
SIM800L	1.6	1	1.6
GY-NEO6MV2	2.9	1	2.9
DS1302 RTC	0.3	1	0.3
OLED SSD1306	1.79	1	1.79
2N7002	0.35	12	4.2
SI4431CDY-T1-GE3	0.9	2	1.8
Resistor 0603 SMD	0.05	41	2.05
Rotary encoder PEC11H	3.01	1	3.01
ESP32-WROOM-32D	2.21	1	2.21
CH340C	0.4	1	0.4
XL1509 buck regulator	0.94	3	2.82
LM317	0.39	1	0.39
Total			**47.571**

Table 5.28 – *Control board - components budget*

5

Item	Cost per unit (€)	Quantity	Total cost (€)
Control PCB - components	47.57	1	47.57
Control PCB - manufacturing	7.5 /5units	1	7.5
Auxiliary PCB - manufacturing	7.22 /5units	1	7.22
Pondskater ORCA	250	1	250
Manpower	20	300	6000
Altium Designer® license	304.85	1	304.85
Total			**6617.14**

Table 5.29 – *Project budget*

Chapter 6

Conclusions, future work and lessons learned

6.1 Conclusions

Throughout the development of the project the author has followed the same process that is adopted in the professional world for the development of competitive electronic products, this includes carrying out tasks such as schematic design, schematic simulation, component comparison and selection, component modelling, PCB design taking into account all the necessary constraints to be manufactured, project budget, etc.

The fact of working on these subjects has allowed the author to approach the theoretical knowledge learned throughout the bachelor years to the practical field as well as to enlarge our understanding in other related fields not imparted in our degree such as the usage of one the most used in the engineering industry, Altium Designer®, which especially required a lot of research time in the first development stages of the project in which the lack of experience slowed down the development speed and productivity.

Moreover, not only technical skills have been acquired thanks to the completion of this final project but also other quite important skills such as managing time in a long-term project, knowing to look for reliable sources from which to learn all the necessary concepts, making proper use of the laboratory instrumentation, communicating with coworkers to help each other whenever it has been necessary, and so on.

In summary, during this project essential skills and knowledge were garnered, which will be for sure a great asset when seeking a job in a leading technology company. For this reason, although the last stages of the product development are still missing such as verification or manufacturing, the author is highly satisfied with the work accomplished and the results presented within the available time, which is thought as rigorous and complete.

6.2 Future work

This project was born already bearing in mind that it would not be possible to finish it completely. The nature of the remaining work is varied. While the actions unable to perform are minimal, other remaining tasks may well be large enough to conduct a related bachelor's thesis. For example:

- Electronics:
  - To carry out the manufacturing process of these boards.
  - To finish all the PCB and component verifications that were unable to perform.
  - To verify that the design fits the space constraints within the boat.
  - To verify the real consumption of each component.
  - To design a second version of the board, where the identified mistakes and inaccuracies are corrected.
  - To design other versions of the board where further improvements are made.
  - To replace the USB mini-b port by a type C since the new law approved recently by the European Parliament establish this last type as the new industry standard from 2024 for small and medium-sized electronics.
- Firmware:
  - To design all the functions and routines for the whole system.
  - To create a code able to gather the information from all the sensors.
  - To specify an optimal BMS and state of charge estimation based on the commented principles.
  - To design the code to carry out the charging process following the parameters specified.
  - To create a system to steer the ship to the desired location by controlling the DC motor and the servomotor based on the data obtained from the sensors.
  - To be able to communicate the boat through mobile networks to an android application in order to be controlled by the user.

6.3 Applications

This design, which in principle would only be suitable for the boat for which it was designed, could be useful for a lot of similar projects based on the ESP32 since it includes a wide variety of sensors which, when they are properly programmed, can be used for many other purposes.

In fact, we already were bearing this in mind and so we added any functionality that might be slightly useful always trying to avoid any restriction in the design that might be an inconvenient for other projects. This can be observed in some decisions taken such as the reduction of the number of pins used by the LEDs or the buttons when there are still some free pins in the microcontroller, or the placement of the ESP32's antenna in the top part of the PCB leaving a free space surrounding to avoid interference when we already know that it would be unused for our project.

Due to that, it could be also used entirely or with minor modifications for other purposes such as to control any autonomous vehicle, to do a GPS tracking of another device, the BMS can be implemented in other designs, etc.

6

Bibliography

[1] Adafruit. Adafruit 10-dof imu breakout - l3gd20h + lsm303 + bmp180. https://www.adafruit.com/product/1604 Accessed May 2022.

[2] Adel S. Sedra and Kenneth C. Smith. *Microelectronic Circuits*, international edition. sixth edition, oxford university press, usa. ed. 1987.

[3] Aranda, A. R. *Class notes from the subject "Tecnología de circuitos impresos" imparted in the GIT of the UGR*. 2021-2022.

[4] Atmel. *ATmega328p datasheet*, 2015. https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf.

[5] Bolívar, S. R. *Class notes from the subject "Electrónica para acondicionamiento energético" imparted in the GIEI of the UGR*. 2021-2022.

[6] Diodes incorporated. *2N7002 datasheet*, 2013. https://www.diodes.com/assets/Datasheets/ds11303.pdf.

[7] Electrical4u. Crystal oscillator: Circuit, frequency and working principle. https://www.electrical4u.com/crystal-oscillator/ Accessed May 2022.

[8] ELETimes. Microprocessor vs microcontroller: What is the difference?, April 2021. https://www.eletimes.com/microprocessor-vs-microcontroller-what-is-the-difference#:~:text=Ultimately%2C%20microcontrollers%20and%20microprocessors%20are,that%20connects%20to%20external%20peripherals. Accessed May 2022.

[9] Espressif System. *ESP32-WROOM-32 datasheet*, 2018. https://www.mouser.com/datasheet/2/891/esp-wroom-32_datasheet_en-1223836.pdf.

[10] Espressif System. *ESP8266EX datasheet*, 2020. https://espressif.com/sites/default/files/documentation/0a-esp8266ex_datasheet_en.pdf.

[11] Espressif Systems. *ESP32 Series datasheet*, 2021. https://www.overleaf.com/project/62738f6fd1c1dddcbca0a65d.

[12] Fernández, P. G. *Class notes from the subject "Prototipado y test electrónicos" imparted in the GIEI of the UGR*. 2021-2022.

[13] Granasat. *Granasat BEXUS Student Experiment Documentation*. Granasat, January 2014. v5-0, Available at http://rexusbexus.net/wp-content/uploads/2015/07/BX19_GRANASAT_SED_v5-0_15Jan15-reduced.pdf Accessed July-2016.

[14] Hangzhou ZhongKe Microelectronics CO. *ATGM336H-5N user manual*. https://www.icofchina.com/d/file/xiazai/2016-12-05/b5c57074f4b1fcc62ba8c7868548d18a.pdf Accessed May 2022.

[15] HONEYWELL. *HMC5883L datasheet*, 2013. https://www.digikey.es/es/datasheets/honeywellmicroelectronicsprecisionsensors/honeywell-microelectronics-precision-sensors-hmc5883l.

[16] INTEGRATED, M. Glossary definition for switching regulator. https://www.maximintegrated.com/en/glossary/definitions.mvp/term/Switching%20Regulator/gpk/298 Accessed May 2022.

[17] KOYANAGI, F. Esp32: Sim800l and barrier sensor. https://www.instructables.com/ESP32-SIM800L-and-Barrier-Sensor/ Accessed November 2021.

[18] LAB PROJECTS BD, MKDAS. 6v lead-acid battery charger circuit, 2021. https://labprojectsbd.com/2021/08/06/6v-lead-acid-battery-charger-circuit/.

[19] MHB BATTERY. *MS7-6 SLA 6V 7AH battery*, 2012. http://www.baterije.org/specifikacijebaterija/MS7-6.pdf.

[20] MONOLITHICPOWER.COM. Voltage regulator types and working principles. https://www.monolithicpower.com/en/voltage-regulator-types Accessed May 2022.

[21] NAUTICPEDIA. Luces de navegación obligatorias según embarcación. http://como.nauticpedia.com/luces-de-navegacion-obligatorias-segun-embarcacion/ Accessed May 2022.

[22] PROMETEC. Rotary encoders. https://www.prometec.net/rotary-encoders/ Accessed May 2022.

[23] REXUS/BEXUS. Rocket & balloon experiments for university students, 2013. http://www.rexusbexus.net/ Accessed July-2016.

[24] SIMCOM. *SIM800L datasheet*, 2013. https://www.filipeflop.com/img/files/download/Datasheet_SIM800L.pdf.

[25] STMICROELECTRONICS. *L78xx series positive voltage regulator datasheet*, 2018. https://www.mouser.es/datasheet/2/389/cd00000444-1795274.pdf.

[26] STMICROELECTRONICS. *USBLC6-2 datasheet*, 2021. https://www.st.com/resource/en/datasheet/usblc6-2.pdf.

[27] SYNACORP. *GY-NEO6MV2 GPS Module*. https://www.epitran.it/ebayDrive/datasheet/NEO6MV2.pdf Accessed May 2022.

[28] TEJADA, J. A. J. *Class notes from the subject "Componentes electrónicos" imparted in GIEI the UGR*. 2018-2019.

[29] TEXAS INSTRUMENT. *UC3906/UC2906 sealed lead acid battery charger datasheet*, 1996. https://www.ti.com/lit/ds/symlink/uc3906.pdf?ts=1654324735418&ref_url=https%253A%252F%252Fwww.google.com%252F.

[30] TEXAS INSTRUMENT. *UC3906 battery charger application note*, 1999. https://www.ti.com/lit/an/slua115/slua115.pdf?ts=1654347065509&ref_url=https%253A%252F%252Fwww.google.com%252F.

[31] TEXAS INSTRUMENTS. *LM317 Accurate Adjustable Voltage Regulator datasheet*, MARCH 2015–REVISED JUNE 2020. https://www.ti.com/lit/ds/symlink/lm317a.pdf?ts=1653961488090&ref_url=https%253A%252F%252Fwww.ti.com%252Fproduct%252FLM317A%253Futm_source%253Dsupplyframe%2526utm_medium%253DSEP%2526utm_campaign%253Dnot_alldatasheet%2526DCM%253Dyes%2526dclid%253DCMWs8o6riPgCFSeClQIdSnwBsg.

[32] VISHAY. *Si4431CDY P-Channel 30-V (D-S) MOSFET datasheet*, 2009. https://www.mouser.es/datasheet/2/427/VISH_S_A0002474142_1-2568417.pdf.

[33] WCH. *CH340C datasheet*. https://www.mpja.com/download/35227cpdata.pdf.

[34] WIKIPEDIA. Eeprom. https://en.wikipedia.org/wiki/EEPROM Accessed June 2022.

[35] Wikipedia. General purpose input output. https://en.wikipedia.org/wiki/General-purpose_input/output Accessed May 2022.

[36] Wikipedia. Gprs. https://en.wikipedia.org/wiki/General_Packet_Radio_Service Accessed May 2022.

[37] Wikipedia. Gsm. https://en.wikipedia.org/wiki/GSM Accessed May 2022.

[38] Wikipedia. H-bridge. https://en.wikipedia.org/wiki/H-bridge Accessed May 2022.

[39] Wikipedia. Inertial measurement unit. https://en.wikipedia.org/wiki/Inertial_measurement_unit Accessed May 2022.

[40] Wikipedia. Pulse width modulation. https://en.wikipedia.org/wiki/Pulse-width_modulation Accessed May 2022.

[41] Wikipedia. Serial peripheral interface. https://en.wikipedia.org/wiki/Serial_Peripheral_Interface Accessed May 2022.

[42] Wikipedia. Universal asynchronous receiver transmitter. https://en.wikipedia.org/wiki/Universal_asynchronous_receiver-transmitter Accessed May 2022.

[43] Wikipedia. Usb. https://en.wikipedia.org/wiki/USB Accessed May 2022.

Appendix A

Electronics BOM

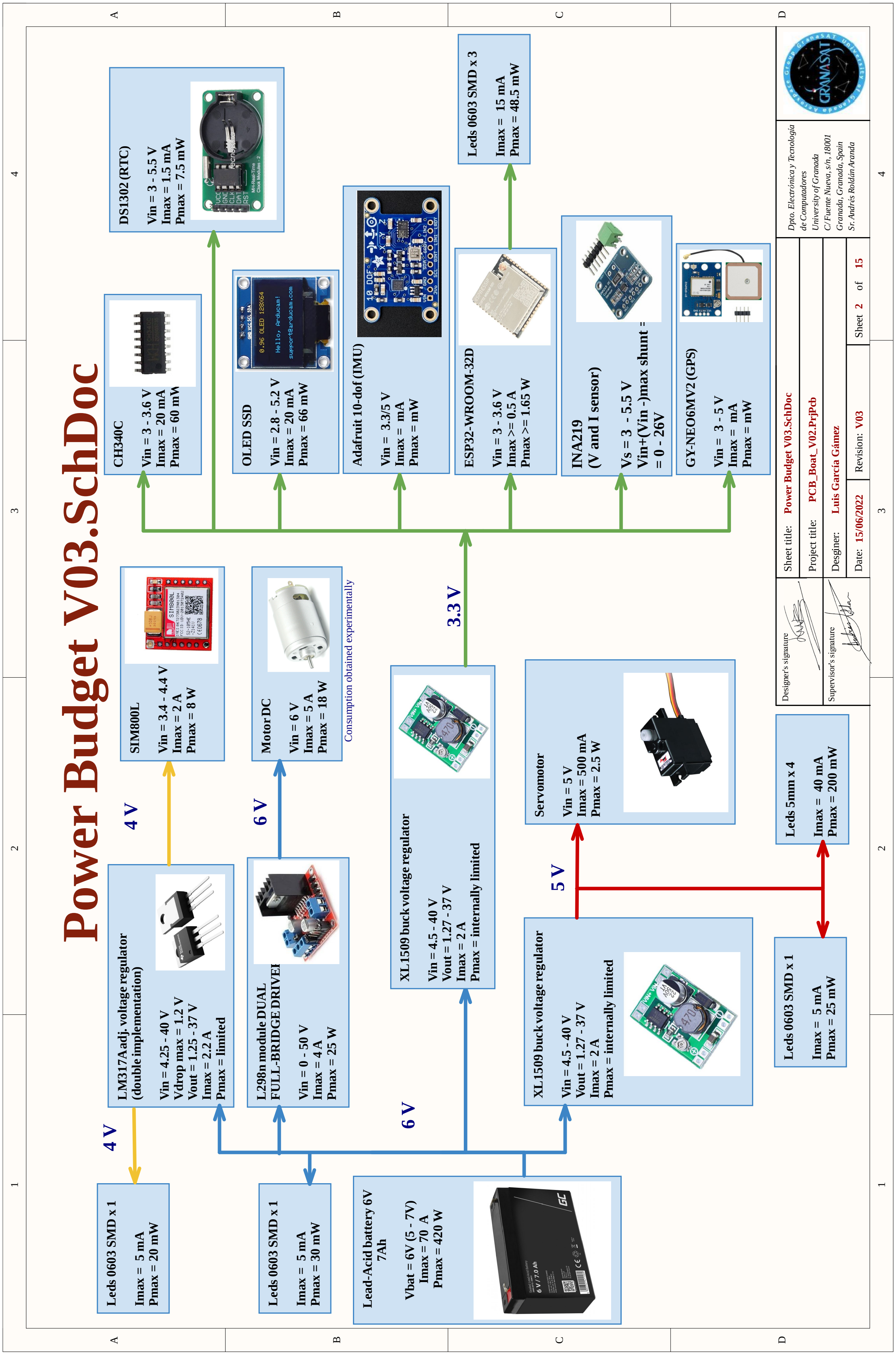
In this chapter you can see a list with all the necessary components to manufacture this product, Table A.1.

Item	Designator name	Quantity
PushButton	Boot, ON/OFF, Reset, SW1, SW2	5
Capacitor 0603	C1 - C12	12
USBLC6-2SC6	D1	1
Blue LED 0603	D2, D10, D11, D12, D13, D15	6
Green LED 0603	D3	1
5MM LED connector (2 male header)	D4, D5, D6, D7	4
PDS1040CTL	D8, D9, D14	3
Ferrite Bead Chip Fair-Rite 2508056007Y3	FB1	1
USB-MINI-B Female	J1	1
Charger connector	J2	1
Battery connector	J3	1
Motor output connector	J6	1
Servomotor connector (3 male header)	Servo	1
MOD INA219	MOD1	1
Adafruit 10DOF	MOD2	1
SIM800L	MOD3	1
GY-NEO6MV2	MOD4	1
DS1302 RTC	MOD5	1
OLED SSD1306	MOD6	1
L298N	MOD7	1
2N7002	Q1 - Q6, Q9 - Q14	12
SI4431CDY-T1-GE3	Q7, Q8	2
Resistor 0603 SMD	R1 - R41	41
BUT Rotary encoder PEC11H	SW3	1
ESP32-WROOM-32D	U1	1
CH340C	U2	1
XL1509 buck regulator	VR4-4V4-2, VR-3V3, VR-5V	3
LM317	VR-4V4-1	1

Table A.1 – *Control board - Bill of materials*

Appendix B

Power Budget


Power Budget V03.SchDoc
Leds 0603 SMD x 1
Imax = 5 mA
Pmax = 20 mW
4 V
LM317A adj. voltage regulator
(double implementation)
Vin = 4.25 - 40 V
Vdrop max = 1.2 V
Vout = 1.25 - 37 V
Imax = 2.2 A
Pmax = limited
4 V
SIM800L
Vin = 3.4 - 4.4 V
Imax = 2 A
Pmax = 8 W
Leds 0603 SMD x 1
Imax = 5 mA
Pmax = 30 mW
L298n module DUAL
FULL-BRIDGE DRIVER
Vin = 0 - 50 V
Imax = 4 A
Pmax = 25 W
6 V
Motor DC
Vin = 6 V
Imax = 5 A
Pmax = 18 W
Consumption obtained experimentally
Lead-Acid battery 6V
7Ah
Vbat = 6V (5 - 7V)
Imax = 70 A
Pmax = 420 W
6 V
XL1509 buck voltage regulator
Vin = 4.5 - 40 V
Vout = 1.27 - 37 V
Imax = 2 A
Pmax = internally limited
3.3 V
XL1509 buck voltage regulator
Vin = 4.5 - 40 V
Vout = 1.27 - 37 V
Imax = 2 A
Pmax = internally limited
5 V
Servomotor
Vin = 5 V
Imax = 500 mA
Pmax = 2.5 W
Leds 0603 SMD x 1
Imax = 5 mA
Pmax = 25 mW
Leds 5mm x 4
Imax = 40 mA
Pmax = 200 mW
CH340C
Vin = 3 - 3.6 V
Imax = 20 mA
Pmax = 60 mW
DS1302 (RTC)
Vin = 3 - 5.5 V
Imax = 1.5 mA
Pmax = 7.5 mW
OLED SSD
Vin = 2.8 - 5.2 V
Imax = 20 mA
Pmax = 66 mW
Adafruit 10-dof (IMU)
Vin = 3.3/5 V
Imax = mA
Pmax = mW
ESP32-WROOM-32D
Vin = 3 - 3.6 V
Imax >= 0.5 A
Pmax >= 1.65 W
Leds 0603 SMD x 3
Imax = 15 mA
Pmax = 48.5 mW
INA219
(V and I sensor)
Vs = 3 - 5.5 V
Vin+(Vin -)max shunt =
= 0 - 26V
GY-NEO6MV2 (GPS)
Vin = 3 - 5 V
Imax = mA
Pmax = mW
Designer's signature
Supervisor's signature
Sheet title: Power Budget V03.SchDoc
Project title: PCB_Boat_V02.PrjPcb
Desgner: Luis García Gámez
Date: 15/06/2022
Revision: V03
Sheet 2 of 15
Dpto. Electrónica y Tecnología de Computadores
University of Granada
C/ Fuente Nueva, s/n, 18001
Granada, Granada, Spain
Sr. Andrés Roldán Aranda

Appendix C

Firmware

C.1 Code to make a LED blink

```
#define ONBOARD_LED 2
//LED integrated in the ESP32 board

void setup() {
  pinMode(ONBOARD_LED, OUTPUT);
  Serial.begin(9600);
}

void loop() {
  delay(500);
  digitalWrite(ONBOARD_LED, HIGH);
  Serial.println("LED on");
  delay(500);
  digitalWrite(ONBOARD_LED, LOW);
  Serial.println("LED off");
}
```

Listado C.1 – *Code to make a LED blink*

C.2 Code to send and receive SMS and calls with the SIM800L

This code is based on the code to control a barrier sensor with the SIM800L found in [17]

```
#include <SPI.h>
#define TINY_GSM_MODEM_SIM800
#include <TinyGsmClient.h>        // Library with GSM commands

//SIM800L serial communication object
HardwareSerial SerialGSM(1);

//library object with GSM functions
TinyGsm modemGSM(SerialGSM);

//Serial speed of both the SIM800L and the serial monitor
```

```
const int BAUD_RATE = 9600;

//variables used to count time without crashing the loop function
long int millisRefCon;

//flag that indicates the time count (used by the 'timeout' function
 )
bool flagCon = false;

// RX and TX pins where the SIM800L will be connected
const int RX_PIN = 2, TX_PIN = 4;

//mobile operator APN
const char *APN = "inet.es";
//if it does not exist, leave it empty
const char *USER = "";
//if it does not exist, leave it empty
const char *PASSWORD = "";

//const char *APN = "ac.vodafone.es";
//const char *USER = "vodafone";
//const char *PASSWORD = "vodafone";

//const char *APN = "orangeworld";
//const char *USER = "orange";
//const char *PASSWORD = "orange";


bool wantToCall = true;
bool wantToSendSMS = true;
// Number of the phone to call or send a SMS
const String number = "+34—————————";
const String message = "Hola mundo";

//Send AT command and wait until a response is obtained
String sendAT(String command)
{
  String response = "";
  SerialGSM.println(command);
  while (!SerialGSM.available());
  response = SerialGSM.readString();
  return response;
}

void setupGSM()
{
  Serial.println("Setup GSM...");
  //initialize serial SIM800L
  SerialGSM.begin(BAUD_RATE, SERIAL_8N1, RX_PIN, TX_PIN, false);
  delay(3000);

  //display modem info on serial monitor
  Serial.println(modemGSM.getModemInfo());
```

```

  if (!modemGSM.restart())
  {
    Serial.println("Restarting GSM\nModem failed");
    delay(10000);
    ESP.restart();
    return;
  }
  Serial.println("Modem restart OK");

  if (!modemGSM.waitForNetwork())
  {
    Serial.println("Failed to connect\nto network");
    delay(10000);
    ESP.restart();
    return;
  }
  Serial.println("Modem network OK");

  if (!modemGSM.gprsConnect(APN, USER, PASSWORD))
  {
    //display.setTextColor(ST7735_RED);
    Serial.println("GPRS Connection\nFailed");
    delay(10000);
    ESP.restart();
    return;
  }
  Serial.println("GPRS Connect OK");

 //Define SMS mode for text (0 = PDU mode, 1 = Text mode)
  if (sendAT("AT+CMGF=1").indexOf("OK") < 0)
  {
    Serial.println("SMS Txt mode Error");
    delay(10000);
    ESP.restart();
    return;
  }
  Serial.println("SMS Txt mode OK");

//Delete all stored SMS
  sendAT("AT + CMGD=1,4");
}

bool timeout(const int DELAY, long *previousMillis, bool *flag)
{
  if (*flag)
  {
    *previousMillis = millis();
    *flag = false;
  }

  if ((*previousMillis + DELAY) < millis())
  {
```

```
    *flag = true;
    return true;
  }

  return false;
}

void call(String number)
{
  Serial.print("Calling...");
  Serial.println(number);
  bool answer= modemGSM.callNumber(number);

  if (answer)
    Serial.println(" OK");
  else
    Serial.println(" fail");

  if (answer)
  {
    answer = modemGSM.callHangup();
    Serial.print("Hang up: ");
    if (answer)
      Serial.println("OK");
    else
      Serial.println("fail");
  }
}


//Function that check if the SIM800L has disconnected, if yes, try
 to reconnect
void verifyGPRSConnection()
{
  Serial.print("GPRS: ");

  if (modemGSM.isGprsConnected())
    Serial.println("Connected");
  else
  {
    Serial.println("Disconnect");
    Serial.println("Reconnecting...");

    if (!modemGSM.waitForNetwork())
    {
      Serial.println("GPRS Connection Failed");
      delay(5000);
    }
    else
    {
      if (!modemGSM.gprsConnect(APN, USER, PASSWORD))
      {
        Serial.println("GPRS Connection Failed");
```

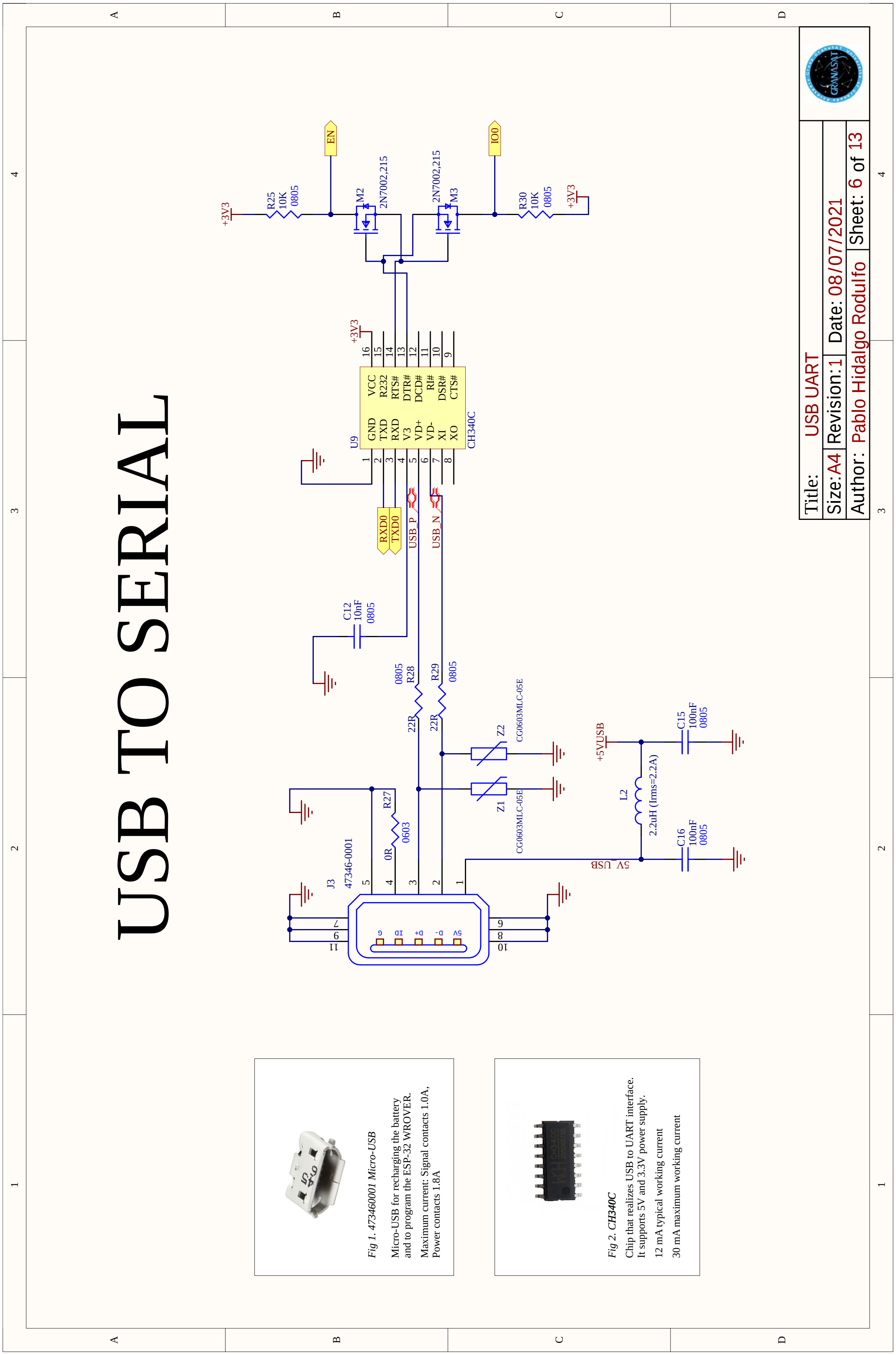
```
        delay(5000);
      }
      else
      {
        Serial.println("GPRS connection correct");
      }
    }
  }
}

//
------------------------------------------------------------------------
SETUP
------------------------------------------------------------------------

void setup()
{
  Serial.begin(BAUD_RATE);
  Serial.println("Starting...");
  millisRefCon = millis();

  // Function that starts and configures the SIM800L
  setupGSM();
  Serial.println("GPRS: Connected1");
}

//
------------------------------------------------------------------------
LOOP
------------------------------------------------------------------------
void loop()
{

  if (timeout(5000, &millisRefCon, &flagCon))
    verifyGPRSConnection();

  // se o SIM800L está conectado
  if (modemGSM.isGprsConnected())
    {
      if (wantToCall)
        {
          Serial.println("Calling to number " + String(number));
          call(number);
        }
      if (wantToSendSMS) {
          Serial.println("Sending sms '" + message + "'");
          if (!modemGSM.sendSMS(number, message)){
            Serial.println("Fail to send SMS");
            }
         else{
            Serial.println("SMS sent");
            }
        }
```

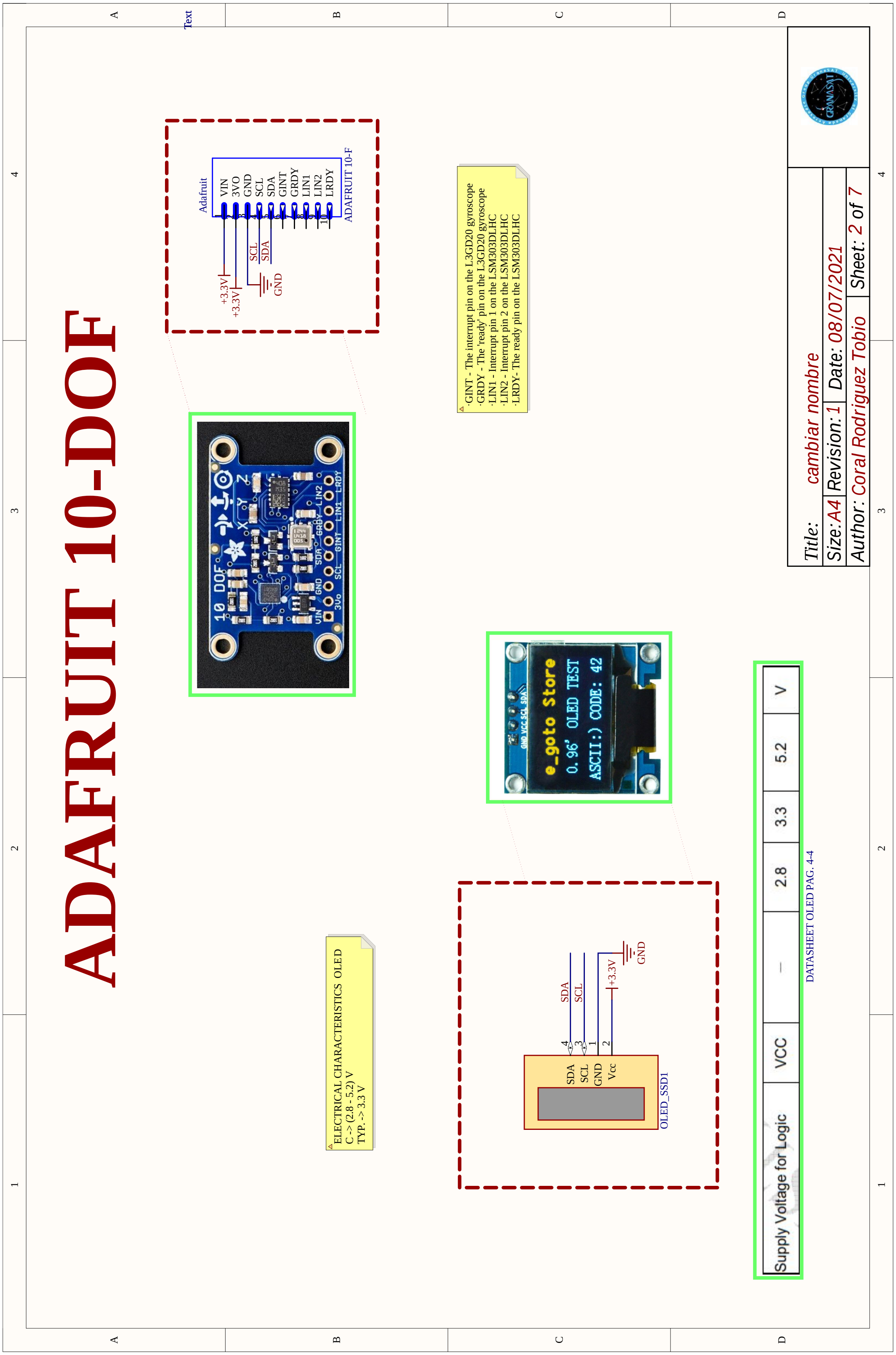
```
    }
  else {
    Serial.println("Disconnected");
  }

  delay(10);
}
```

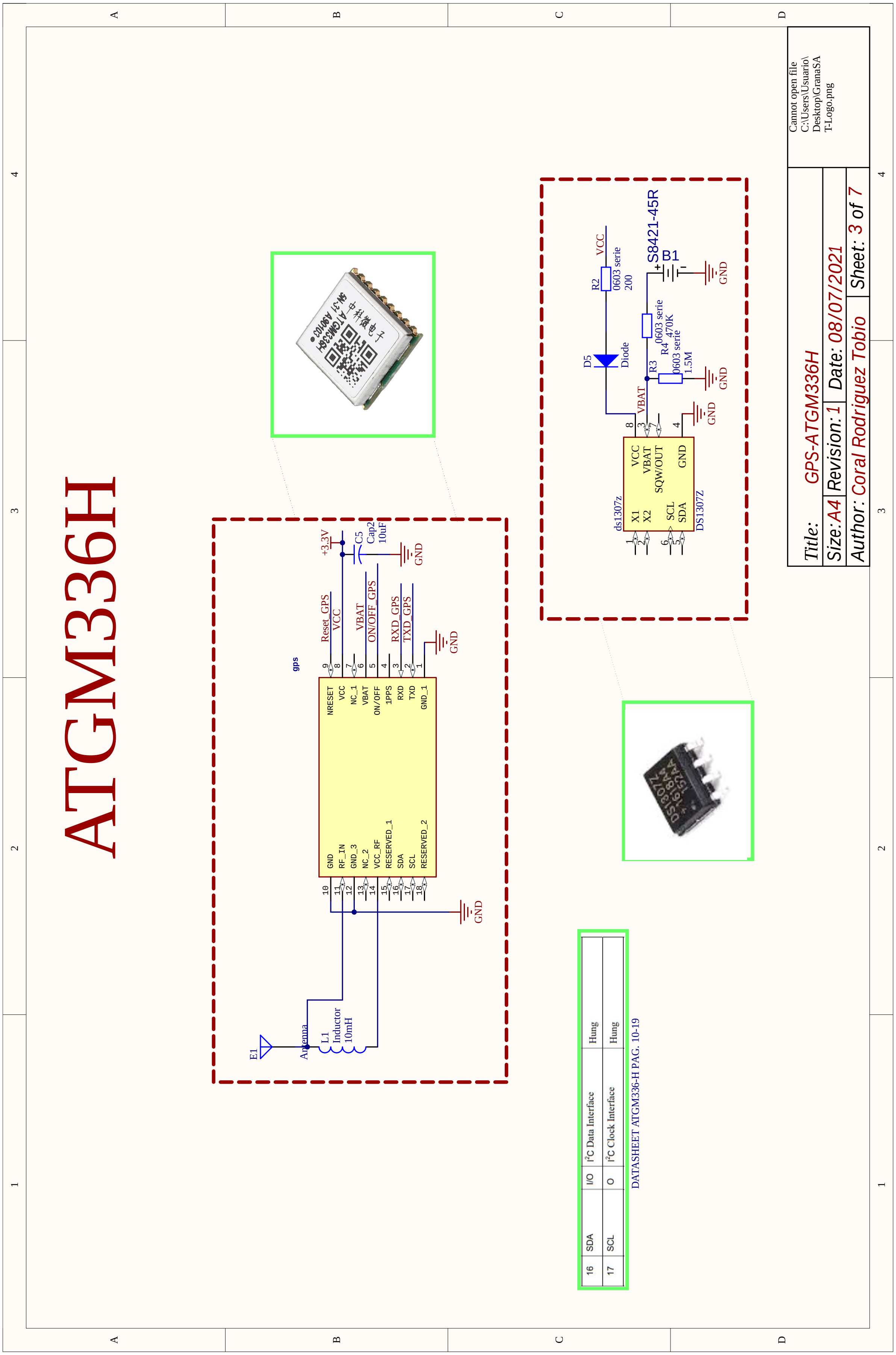
Listado C.2 – *Code to make calls and send SMS with the SIM800L*

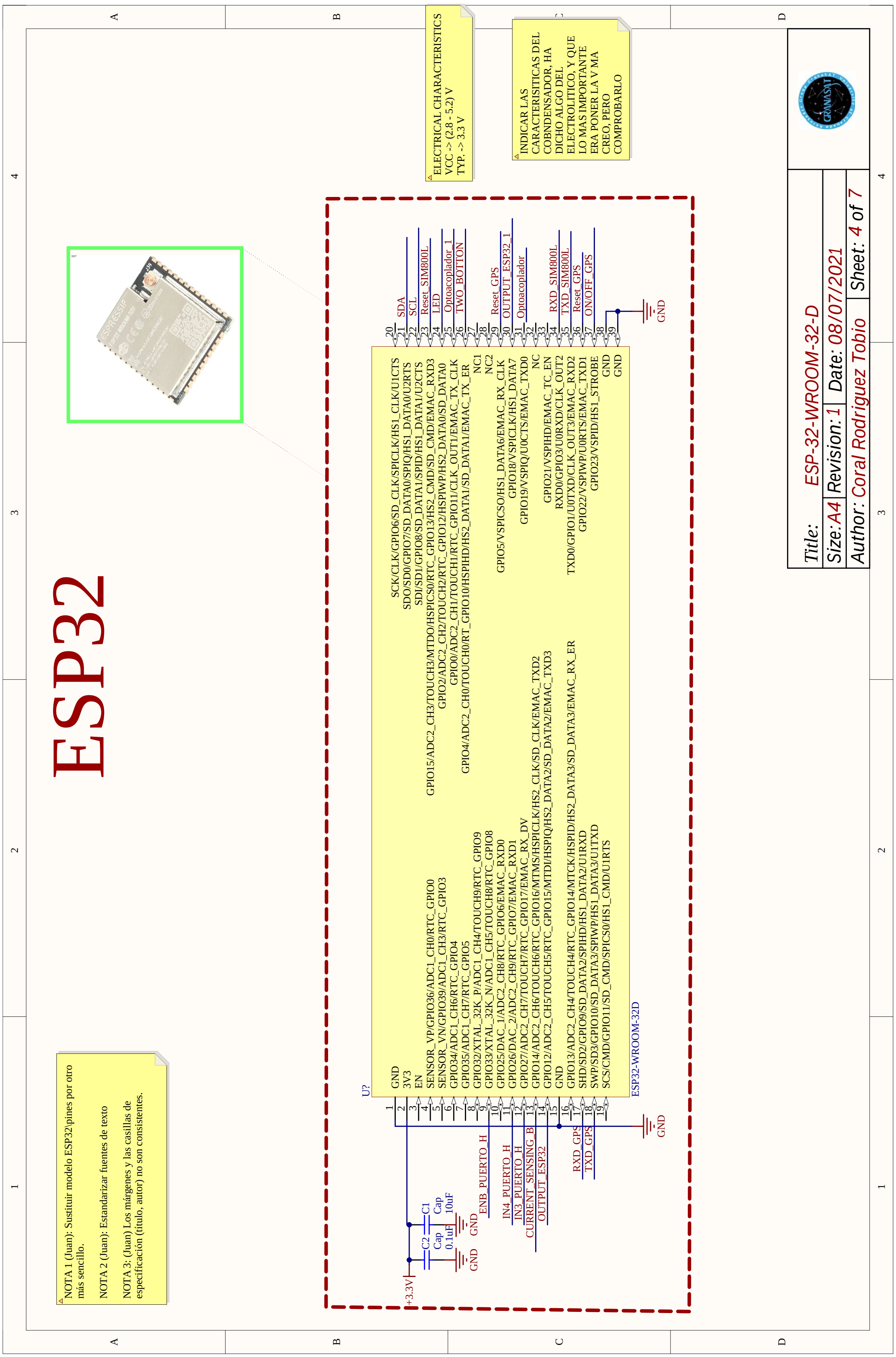
Appendix D

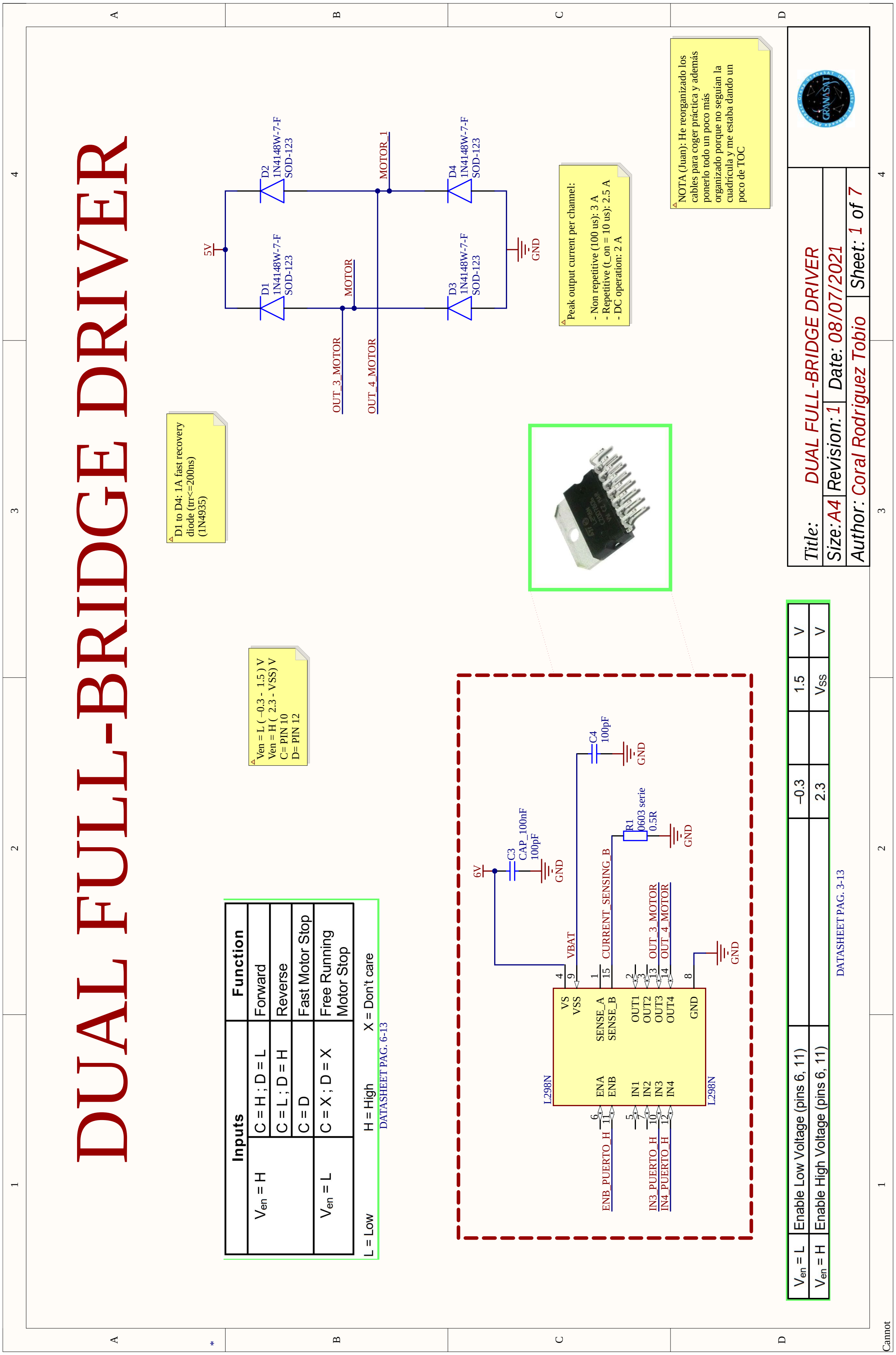
Previous work: Altium files

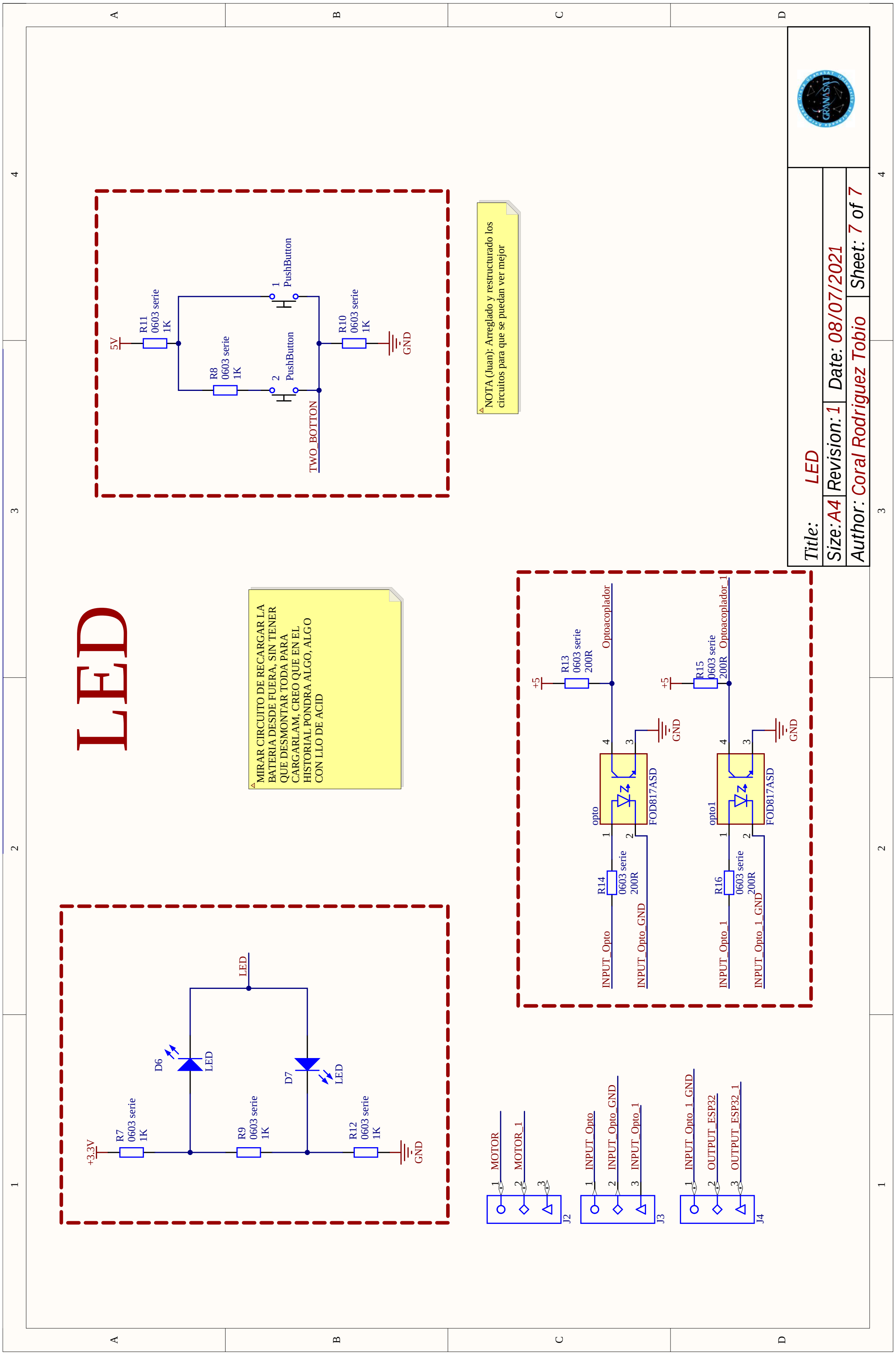

USB TO SERIAL
J3
47346-0001
R27
0R
0603
R28
22R
0805
R29
22R
0805
C12
10nF
0805
Z1
CG0603MLC-05E
Z2
CG0603MLC-05E
L2
2.2uH (Irms=2.2A)
C15
100nF
0805
C16
100nF
0805
+5VUSB
5V_USB
USB_P
USB_N
RXD0
TXD0
U9
CH340C
GND
TXD
RXD
V3
VD+
VD-
XI
XO
VCC
R232
RTS#
DTR#
DCD#
RI#
DSR#
CTS#
+3V3
R25
10K
0805
R30
10K
0805
M2
2N7002,215
M3
2N7002,215
EN
IO0
Fig 1. 473460001 Micro-USB
Micro-USB for recharging the battery
and to program the ESP-32 WROVER.
Maximum current: Signal contacts 1.0A,
Power contacts 1.8A
Fig 2. CH340C
Chip that realizes USB to UART interface.
It supports 5V and 3.3V power supply.
12 mA typical working current
30 mA maximum working current
Title: USB UART
Size: A4
Revision: 1
Date: 08/07/2021
Author: Pablo Hidalgo Rodulfo
Sheet: 6 of 13

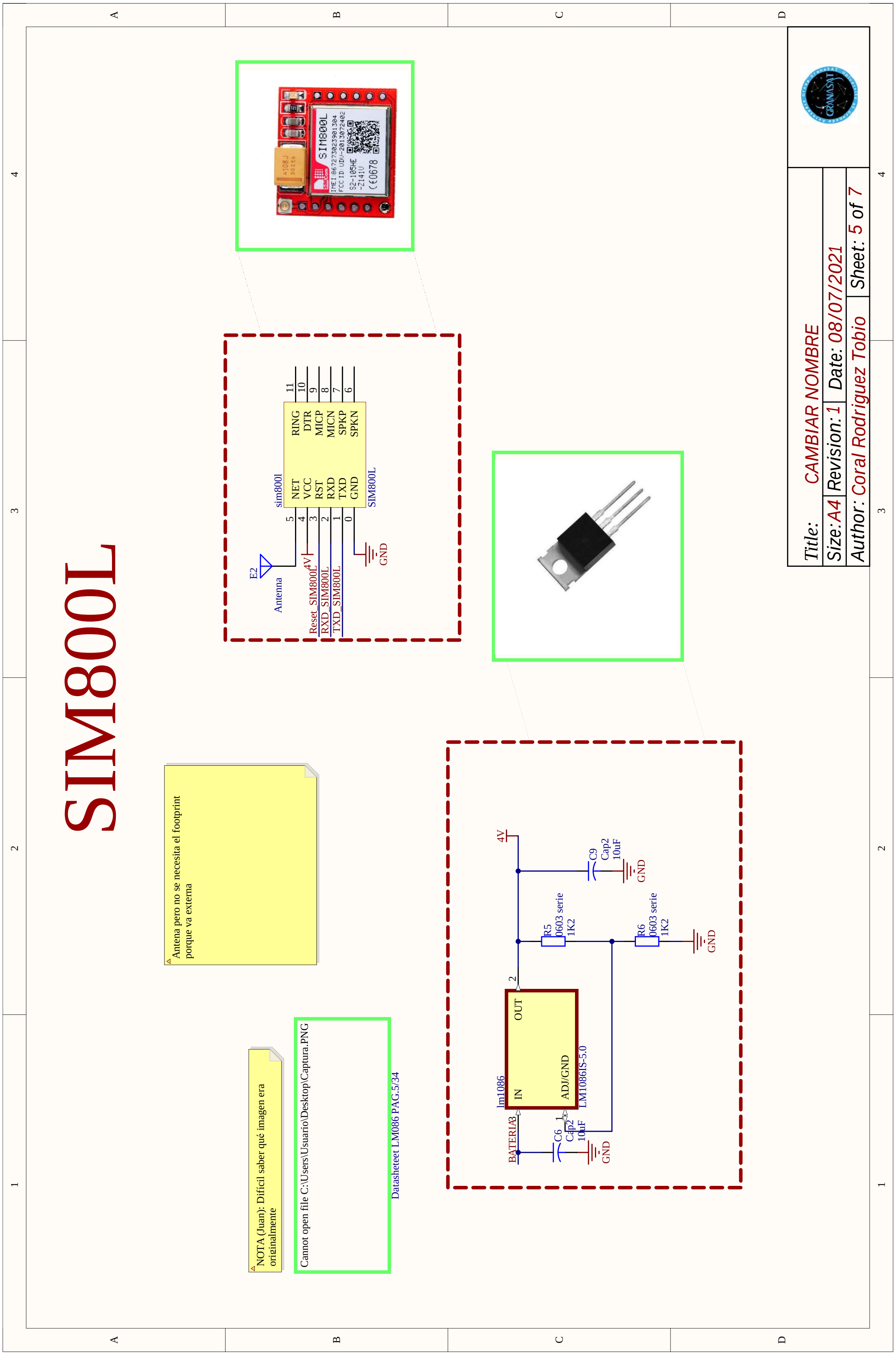

ADAFRUIT 10-DOF
Adafruit
VIN
3VO
GND
SCL
SDA
GINT
GRDY
LIN1
LIN2
LRDY
ADAFRUIT 10-F
+3.3V
+3.3V
SCL
SDA
GND
·GINT - The interrupt pin on the L3GD20 gyroscope
·GRDY - The 'ready' pin on the L3GD20 gyroscope
·LIN1 - Interrupt pin 1 on the LSM303DLHC
·LIN2 - Interrupt pin 2 on the LSM303DLHC
·LRDY- The ready pin on the LSM303DLHC
ELECTRICAL CHARACTERISTICS OLED
C-> (2.8 - 5.2) V
TYP.-> 3.3 V
SDA
SCL
GND
Vcc
OLED_SSD1
+3.3V
GND
e_goto Store
0.96' OLED TEST
ASCII:) CODE: 42
Supply Voltage for Logic
VCC
—
2.8
3.3
5.2
V
DATASHEET OLED PAG. 4-4
Title: cambiar nombre
Size: A4
Revision: 1
Date: 08/07/2021
Author: Coral Rodriguez Tobio
Sheet: 2 of 7
Text


ATGM336H
gps
NRESET
VCC
NC_1
VBAT
ON/OFF
1PPS
RXD
TXD
GND_1
GND
RF_IN
GND_3
NC_2
VCC_RF
RESERVED_1
SDA
SCL
RESERVED_2
Reset_GPS
VCC
VBAT
ON/OFF_GPS
RXD_GPS
TXD_GPS
+3.3V
C5
Cap2
10uF
GND
E1
Antenna
L1
Inductor
10mH
ds1307z
DS1307Z
X1
X2
SCL
SDA
VCC
VBAT
SQW/OUT
D5
Diode
R2
0603 serie
200
R3
0603 serie
470K
R4
0603 serie
1.5M
B1
S8421-45R
16 SDA I/O I²C Data Interface Hung
17 SCL O I²C Clock Interface Hung
DATASHEET ATGM336-H PAG. 10-19
Cannot open file
C:\Users\Usuario\Desktop\GranaSA
T-Logo.png
Title: GPS-ATGM336H
Size: A4
Revision: 1
Date: 08/07/2021
Author: Coral Rodriguez Tobio
Sheet: 3 of 7


ESP32
NOTA 1 (Juan): Sustituir modelo ESP32\pines por otro más sencillo.
NOTA 2 (Juan): Estandarizar fuentes de texto
NOTA 3: (Juan) Los márgenes y las casillas de especificación (título, autor) no son consistentes.
ELECTRICAL CHARACTERISTICS
VCC -> (2.8 - 5.2) V
TYP. -> 3.3 V
INDICAR LAS CARACTERISTICAS DEL COBNDENSADOR, HA DICHO ALGO DEL ELECTROLITICO, Y QUE LO MAS IMPORTANTE ERA PONER LA V MA CREO, PERO COMPROBARLO
U?
ESP32-WROOM-32D
+3.3V
C2 Cap 0.1uF
C1 Cap 10uF
GND
ENB_PUERTO_H
IN4_PUERTO_H
IN3_PUERTO_H
CURRENT_SENSING_B
OUTPUT_ESP32
RXD_GPS
TXD_GPS
SDA
SCL
Reset_SIM800L
LED
Optoacoplador_1
TWO_BOTTON
Reset_GPS
OUTPUT_ESP32_1
Optoacoplador
RXD_SIM800L
TXD_SIM800L
Reset_GPS
ON/OFF_GPS
Title: ESP-32-WROOM-32-D
Size: A4
Revision: 1
Date: 08/07/2021
Author: Coral Rodriguez Tobio
Sheet: 4 of 7

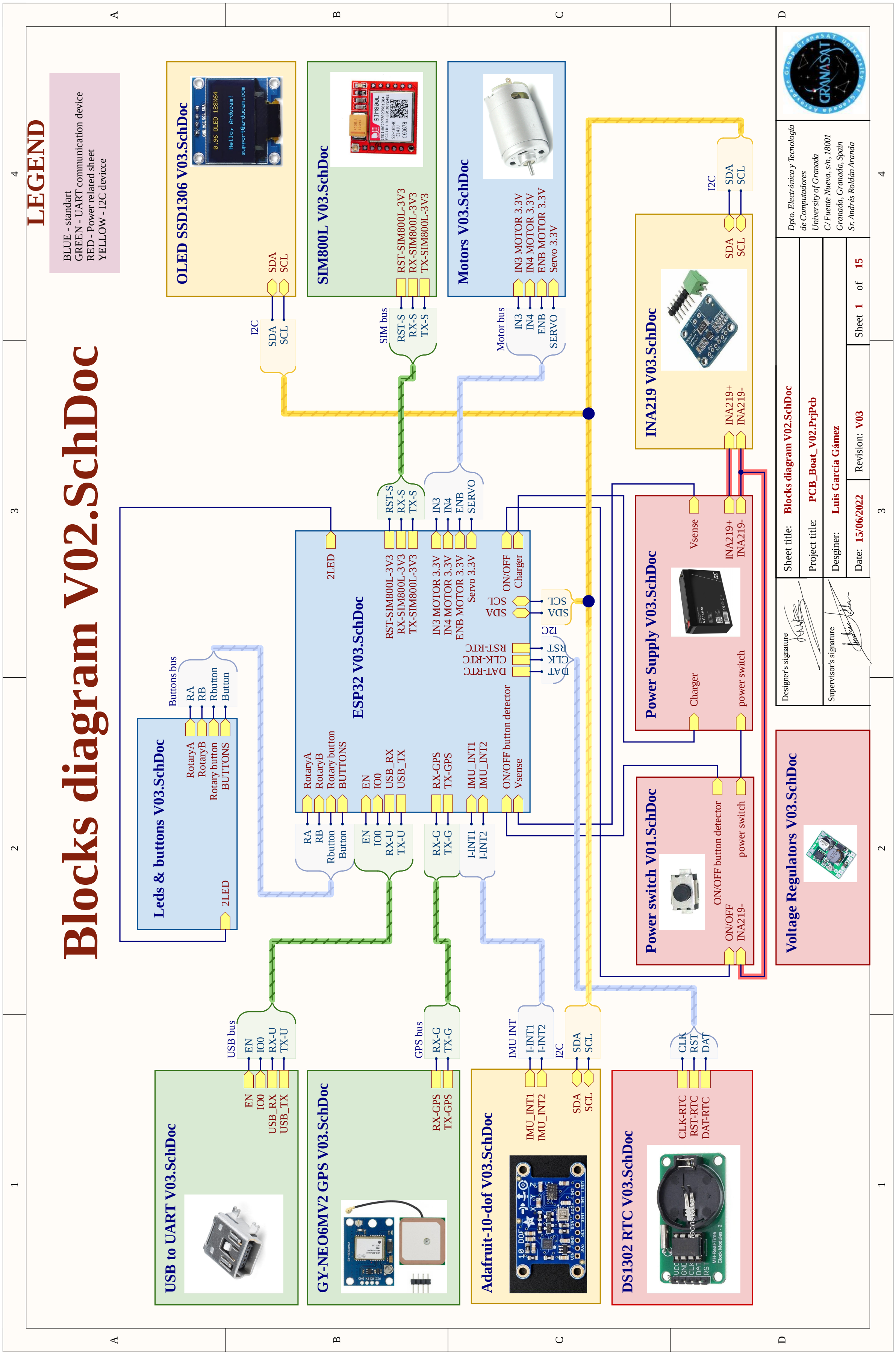

DUAL FULL-BRIDGE DRIVER
Inputs
Function
Ven = H
C = H ; D = L
Forward
C = L ; D = H
Reverse
C = D
Fast Motor Stop
Ven = L
C = X ; D = X
Free Running Motor Stop
L = Low
H = High
X = Don't care
DATASHEET PAG. 6-13
D1 to D4: 1A fast recovery diode (trr<=200ns) (1N4935)
Ven = L (−0.3 - 1.5) V
Ven = H (2.3 - VSS) V
C= PIN 10
D= PIN 12
5V
D1 1N4148W-7-F SOD-123
D2 1N4148W-7-F SOD-123
D3 1N4148W-7-F SOD-123
D4 1N4148W-7-F SOD-123
MOTOR
MOTOR_1
OUT_3_MOTOR
OUT_4_MOTOR
GND
Peak output current per channel:
- Non repetitive (100 us): 3 A
- Repetitive (t_on = 10 us): 2.5 A
- DC operation: 2 A
NOTA (Juan): He reorganizado los cables para coger práctica y además ponerlo todo un poco más organizado porque no seguían la cuadrícula y me estaba dando un poco de TOC
6V
C3 CAP_100nF 100pF
C4 100pF
R1 0603 serie 0.5R
VBAT
CURRENT_SENSING_B
OUT_3_MOTOR
OUT_4_MOTOR
ENB_PUERTO_H
IN3_PUERTO_H
IN4_PUERTO_H
L298N
VS
VSS
SENSE_A
SENSE_B
OUT1
OUT2
OUT3
OUT4
GND
ENA
ENB
IN1
IN2
IN3
IN4
Ven = L Enable Low Voltage (pins 6, 11) −0.3 1.5 V
Ven = H Enable High Voltage (pins 6, 11) 2.3 Vss V
DATASHEET PAG. 3-13
Title: DUAL FULL-BRIDGE DRIVER
Size: A4
Revision: 1
Date: 08/07/2021
Author: Coral Rodriguez Tobio
Sheet: 1 of 7
GRANASAT


LED
MIRAR CIRCUITO DE RECARGAR LA BATERIA DESDE FUERA, SIN TENER QUE DESMONTAR TODA PARA CARGARLAM, CREO QUE EN EL HISTORIAL PONDRA ALGO, ALGO CON LLO DE ACID
NOTA (Juan): Arreglado y restructurado los circuitos para que se puedan ver mejor
5V
R11 0603 serie 1K
R8 0603 serie 1K
R10 0603 serie 1K
PushButton
TWO_BOTTON
GND
+3.3V
R7 0603 serie 1K
R9 0603 serie 1K
R12 0603 serie 1K
D6 LED
D7 LED
MOTOR
MOTOR_1
J2
INPUT_Opto
INPUT_Opto_GND
INPUT_Opto_1
J3
INPUT_Opto_1_GND
OUTPUT_ESP32
OUTPUT_ESP32_1
J4
+5
R13 0603 serie 200R
R14 0603 serie 200R
R15 0603 serie 200R
R16 0603 serie 200R
Optoacoplador
Optoacoplador_1
opto
opto1
FOD817ASD
Title: LED
Size: A4
Revision: 1
Date: 08/07/2021
Author: Coral Rodriguez Tobio
Sheet: 7 of 7

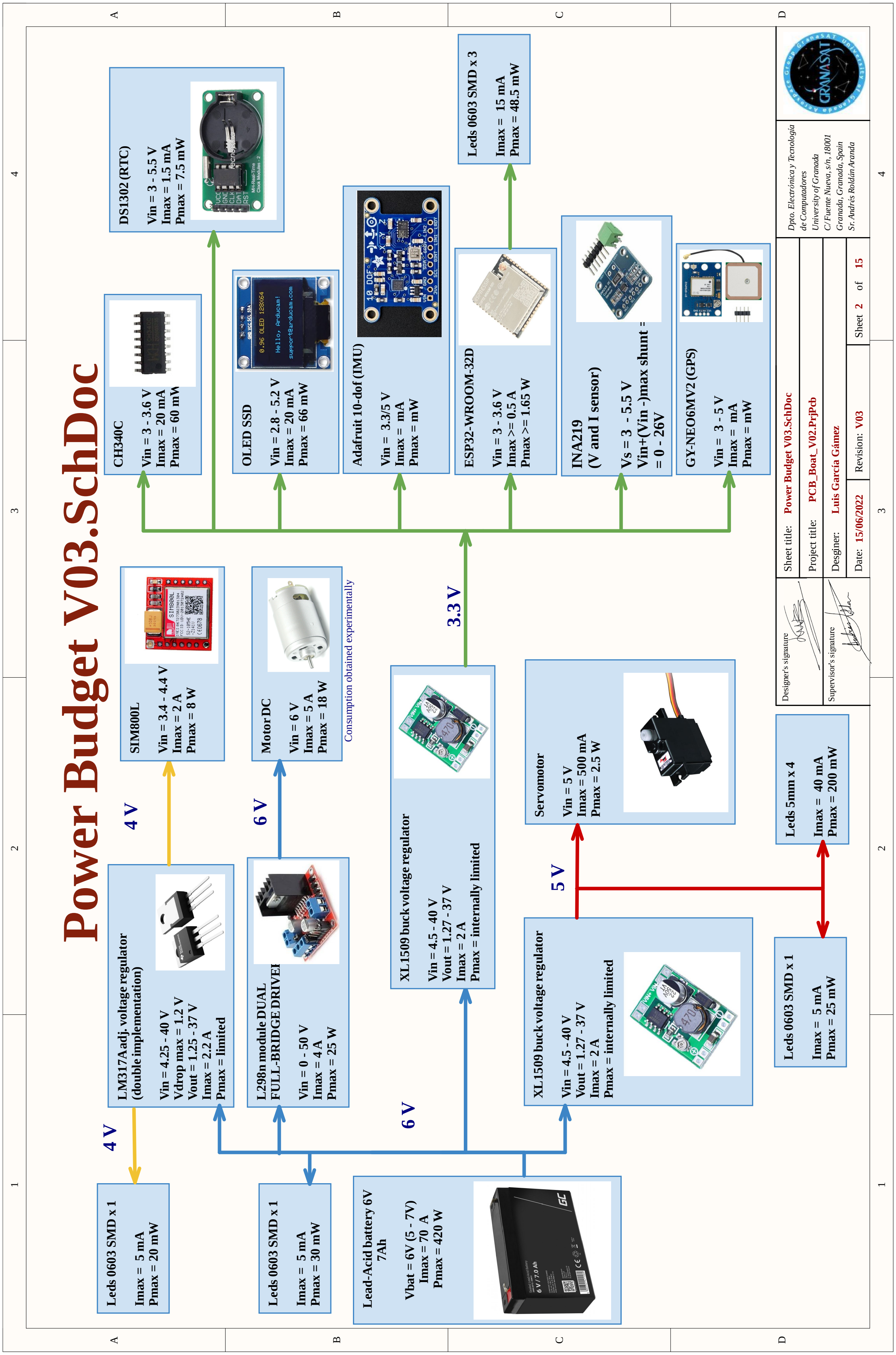

SIM800L
Antena pero no se necesita el footprint porque va externa
NOTA (Juan): Difícil saber qué imagen era originalmente
Cannot open file C:\Users\Usuario\Desktop\Captura.PNG
Datasheteet LM086 PAG.5/34
sim800l
NET
VCC
RST
RXD
TXD
GND
RING
DTR
MICP
MICN
SPKP
SPKN
SIM800L
E2
Antenna
4V
Reset_SIM800L
RXD_SIM800L
TXD_SIM800L
GND
lm1086
IN
OUT
ADJ/GND
LM1086IS-5.0
BATERIA
C6
Cap2
10uF
R5
0603 serie
1K2
R6
0603 serie
1K2
C9
Cap2
10uF
Title: CAMBIAR NOMBRE
Size: A4
Revision: 1
Date: 08/07/2021
Author: Coral Rodriguez Tobio
Sheet: 5 of 7

Appendix E

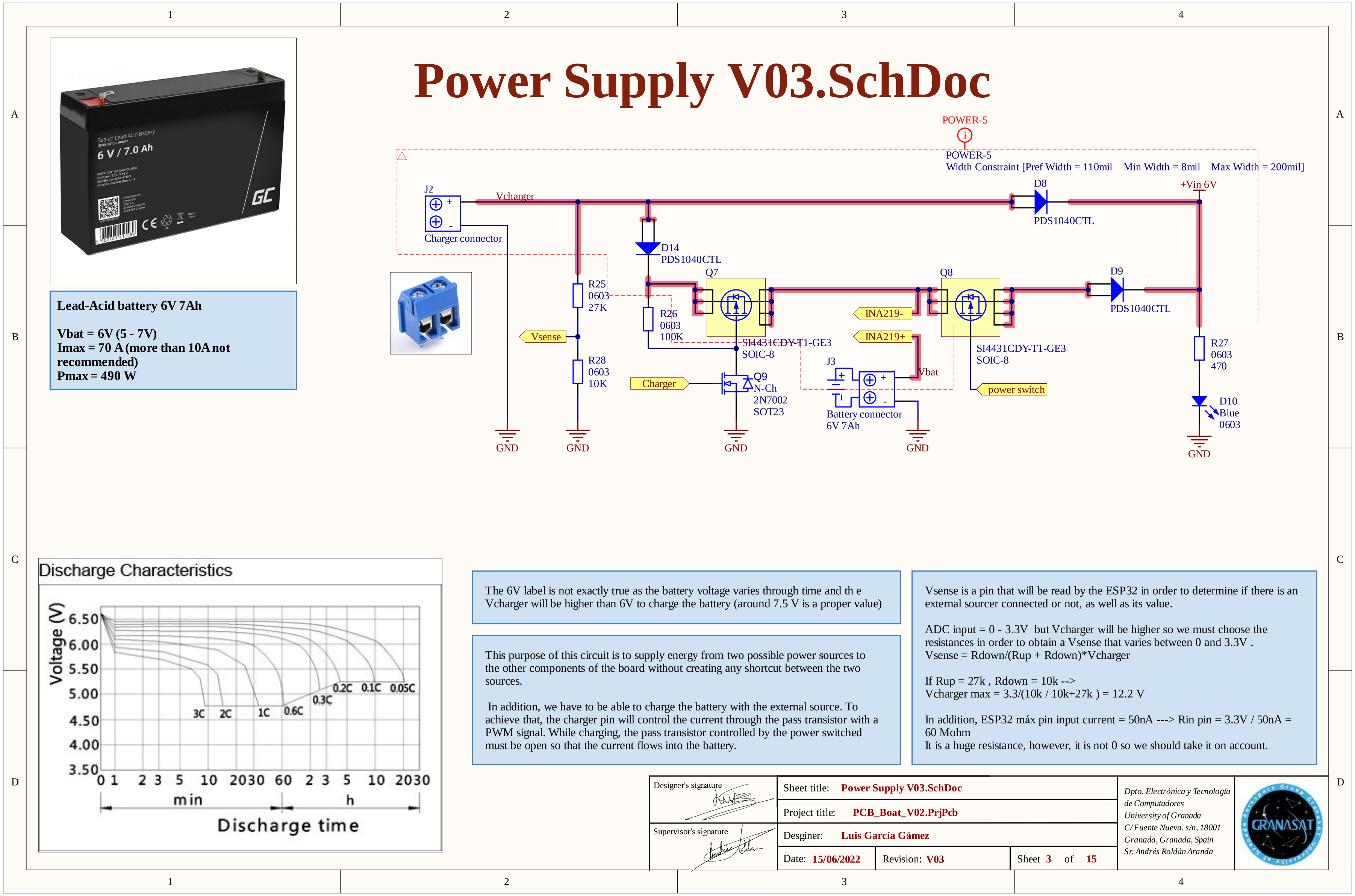
Electronics schematics - Altium files



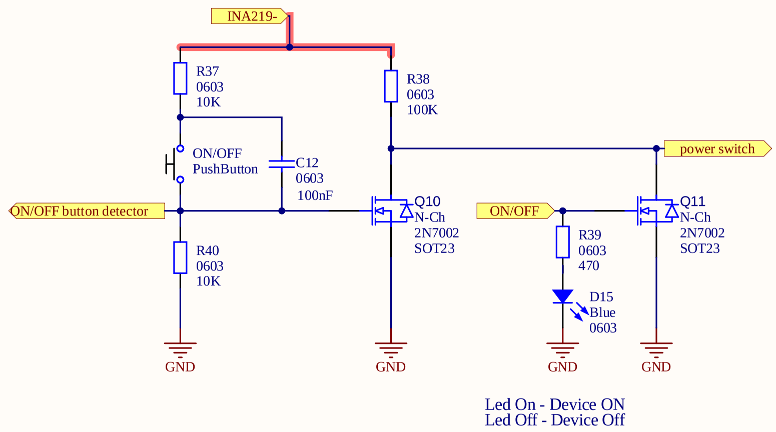
Blocks diagram V02.SchDoc
LEGEND
BLUE - standart
GREEN - UART communication device
RED - Power related sheet
YELLOW - I2C devicce
OLED SSD1306 V03.SchDoc
SIM800L V03.SchDoc
Motors V03.SchDoc
INA219 V03.SchDoc
ESP32 V03.SchDoc
Leds & buttons V03.SchDoc
Power Supply V03.SchDoc
Power switch V01.SchDoc
Voltage Regulators V03.SchDoc
USB to UART V03.SchDoc
GY-NEO6MV2 GPS V03.SchDoc
Adafruit-10-dof V03.SchDoc
DS1302 RTC V03.SchDoc
I2C
SIM bus
Motor bus
Buttons bus
USB bus
GPS bus
IMU INT
Sheet title: Blocks diagram V02.SchDoc
Project title: PCB_Boat_V02.PrjPcb
Desginer: Luis García Gámez
Date: 15/06/2022
Revision: V03
Sheet 1 of 15
Dpto. Electrónica y Tecnología de Computadores
University of Granada
C/Fuente Nueva, s/n, 18001
Granada, Granada, Spain
Sr. Andrés Roldán Aranda
Designer's signature
Supervisor's signature

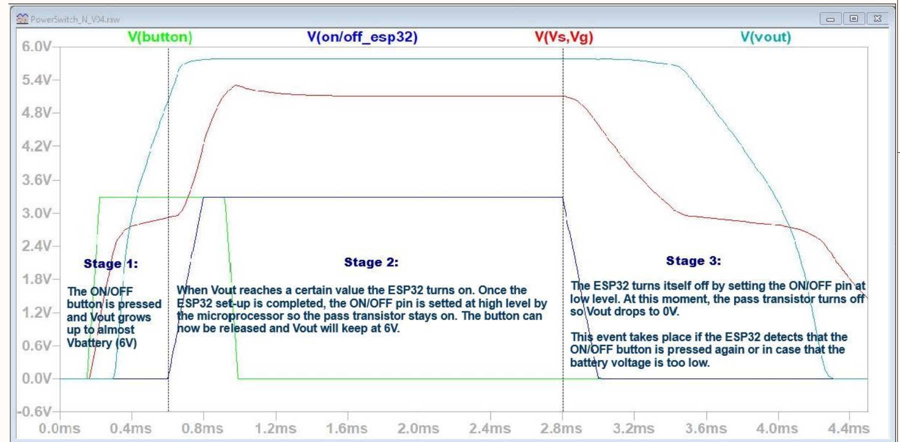


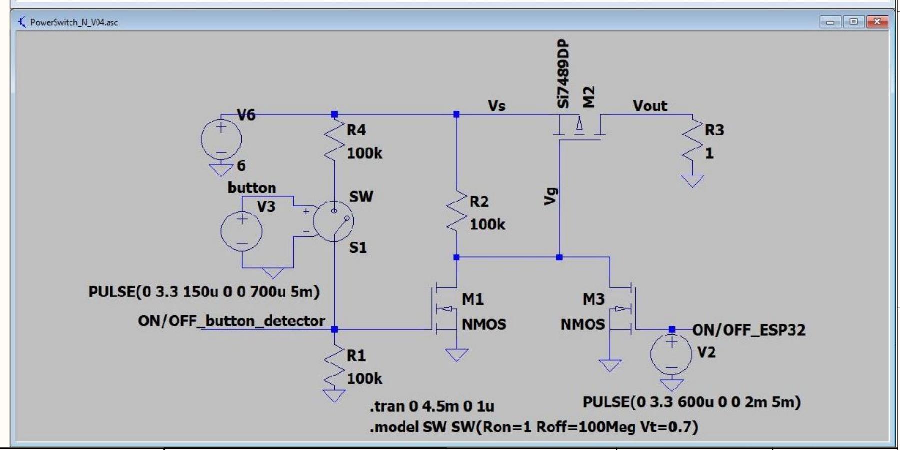
Power Budget V03.SchDoc
DS1302 (RTC)
Vin = 3 - 5.5 V
Imax = 1.5 mA
Pmax = 7.5 mW
Leds 0603 SMD x 3
Imax = 15 mA
Pmax = 48.5 mW
CH340C
Vin = 3 - 3.6 V
Imax = 20 mA
Pmax = 60 mW
OLED SSD
Vin = 2.8 - 5.2 V
Imax = 20 mA
Pmax = 66 mW
Adafruit 10-dof (IMU)
Vin = 3.3/5 V
Imax = mA
Pmax = mW
ESP32-WROOM-32D
Vin = 3 - 3.6 V
Imax >= 0.5 A
Pmax >= 1.65 W
INA219
(V and I sensor)
Vs = 3 - 5.5 V
Vin+(Vin -)max shunt =
= 0 - 26V
GY-NEO6MV2 (GPS)
Vin = 3 - 5 V
Imax = mA
Pmax = mW
3.3 V
SIM800L
Vin = 3.4 - 4.4 V
Imax = 2 A
Pmax = 8 W
Motor DC
Vin = 6 V
Imax = 5 A
Pmax = 18 W
Consumption obtained experimentally
4 V
6 V
LM317Aadj. voltage regulator
(double implementation)
Vin = 4.25 - 40 V
Vdrop max = 1.2 V
Vout = 1.25 - 37 V
Imax = 2.2 A
Pmax = limited
L298n module DUAL
FULL-BRIDGE DRIVER
Vin = 0 - 50 V
Imax = 4 A
Pmax = 25 W
XL1509 buck voltage regulator
Vin = 4.5 - 40 V
Vout = 1.27 - 37 V
Imax = 2 A
Pmax = internally limited
Servomotor
Vin = 5 V
Imax = 500 mA
Pmax = 2.5 W
5 V
XL1509 buck voltage regulator
Vin = 4.5 - 40 V
Vout = 1.27 - 37 V
Imax = 2 A
Pmax = internally limited
Leds 5mm x 4
Imax = 40 mA
Pmax = 200 mW
Leds 0603 SMD x 1
Imax = 5 mA
Pmax = 25 mW
4 V
Leds 0603 SMD x 1
Imax = 5 mA
Pmax = 20 mW
Leds 0603 SMD x 1
Imax = 5 mA
Pmax = 30 mW
6 V
Lead-Acid battery 6V
7Ah
Vbat = 6V (5 - 7V)
Imax = 70 A
Pmax = 420 W
Designer's signature
Supervisor's signature
Sheet title: Power Budget V03.SchDoc
Project title: PCB_Boat_V02.PrjPcb
Designer: Luis García Gámez
Date: 15/06/2022
Revision: V03
Sheet 2 of 15
Dpto. Electrónica y Tecnología de Computadores
University of Granada
C/ Fuente Nueva, s/n, 18001
Granada, Granada, Spain
Sr. Andrés Roldán Aranda
A
B
C
D
1
2
3
4


Power Supply V03.SchDoc
Lead-Acid battery 6V 7Ah
Vbat = 6V (5 - 7V)
Imax = 70 A (more than 10A not recommended)
Pmax = 490 W
6 V / 7.0 Ah
POWER-5
POWER-5
Width Constraint [Pref Width = 110mil Min Width = 8mil Max Width = 200mil]
J2
Charger connector
Vcharger
D8
PDS1040CTL
+Vin 6V
D14
PDS1040CTL
Q7
SI4431CDY-T1-GE3
SOIC-8
Q8
SI4431CDY-T1-GE3
SOIC-8
D9
PDS1040CTL
R25
0603
27K
R26
0603
100K
R28
0603
10K
R27
0603
470
D10
Blue
0603
Vsense
Charger
Q9
N-Ch
2N7002
SOT23
INA219-
INA219+
Vbat
J3
Battery connector
6V 7Ah
power switch
GND
GND
GND
GND
GND
Discharge Characteristics
Voltage (V)
6.50
6.00
5.50
5.00
4.50
4.00
3.50
3C 2C 1C 0.6C 0.3C 0.2C 0.1C 0.05C
0 1 2 3 5 10 2030 60 2 3 5 10 2030
min
h
Discharge time
The 6V label is not exactly true as the battery voltage varies through time and th e Vcharger will be higher than 6V to charge the battery (around 7.5 V is a proper value)
This purpose of this circuit is to supply energy from two possible power sources to the other components of the board without creating any shortcut between the two sources.
In addition, we have to be able to charge the battery with the external source. To achieve that, the charger pin will control the current through the pass transistor with a PWM signal. While charging, the pass transistor controlled by the power switched must be open so that the current flows into the battery.
Vsense is a pin that will be read by the ESP32 in order to determine if there is an external sourcer connected or not, as well as its value.
ADC input = 0 - 3.3V but Vcharger will be higher so we must choose the resistances in order to obtain a Vsense that varies between 0 and 3.3V .
Vsense = Rdown/(Rup + Rdown)*Vcharger
If Rup = 27k , Rdown = 10k -->
Vcharger max = 3.3/(10k / 10k+27k) = 12.2 V
In addition, ESP32 máx pin input current = 50nA ---> Rin pin = 3.3V / 50nA = 60 Mohm
It is a huge resistance, however, it is not 0 so we should take it on account.
Designer's signature
Supervisor's signature
Sheet title: Power Supply V03.SchDoc
Project title: PCB_Boat_V02.PrjPcb
Desginer: Luis García Gámez
Date: 15/06/2022
Revision: V03
Sheet 3 of 15
Dpto. Electrónica y Tecnología de Computadores
University of Granada
C/ Fuente Nueva, s/n, 18001
Granada, Granada, Spain
Sr. Andrés Roldán Aranda
GRANASAT

Power switch V01.SchDoc







Since the ESP32 have to keep the pass transistor on to power itself, when it is been programmed it won't have a power supply other than the external source of Vcharger (or keep the button pressed all the time until the program is loaded and the ESP32 is running it).

The purpose of this circuit is to control the gate of the PMOS transistor that will connect or disconnect the battery from the other devices when it is being charged.

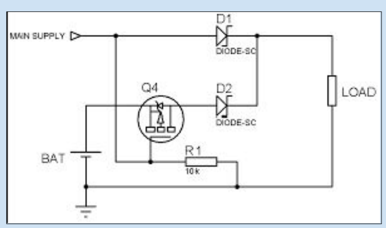
The PMOS will let the current flow whenever Vsg > Vth (aprox 1V). And Vs = Vbattery-6V

If any of the NMOS below (M7 and M8) are on, the voltage in "powerswitch" will drop to aprox. 0 V so Vsg = 6V so PMOS will be ON. Otherwise, "powerswitch will be Vbattery-6V" so Vsg = 0.

Taking the above into consideration, we can turn on the ESP32 with the button and keep it ON with the ON/OFF pin from the ESP32. In addition, we can turn off the microprocessor simply by setting ON/OFF at low.

The source switch could be done with a passive circuit as shown in the picture. However, We want to have control over the transistor in order to:

- turn on and off the device with a button
- isolate the battery from the circuit when it is being charged by an external source.



Designer's signature

Supervisor's signature

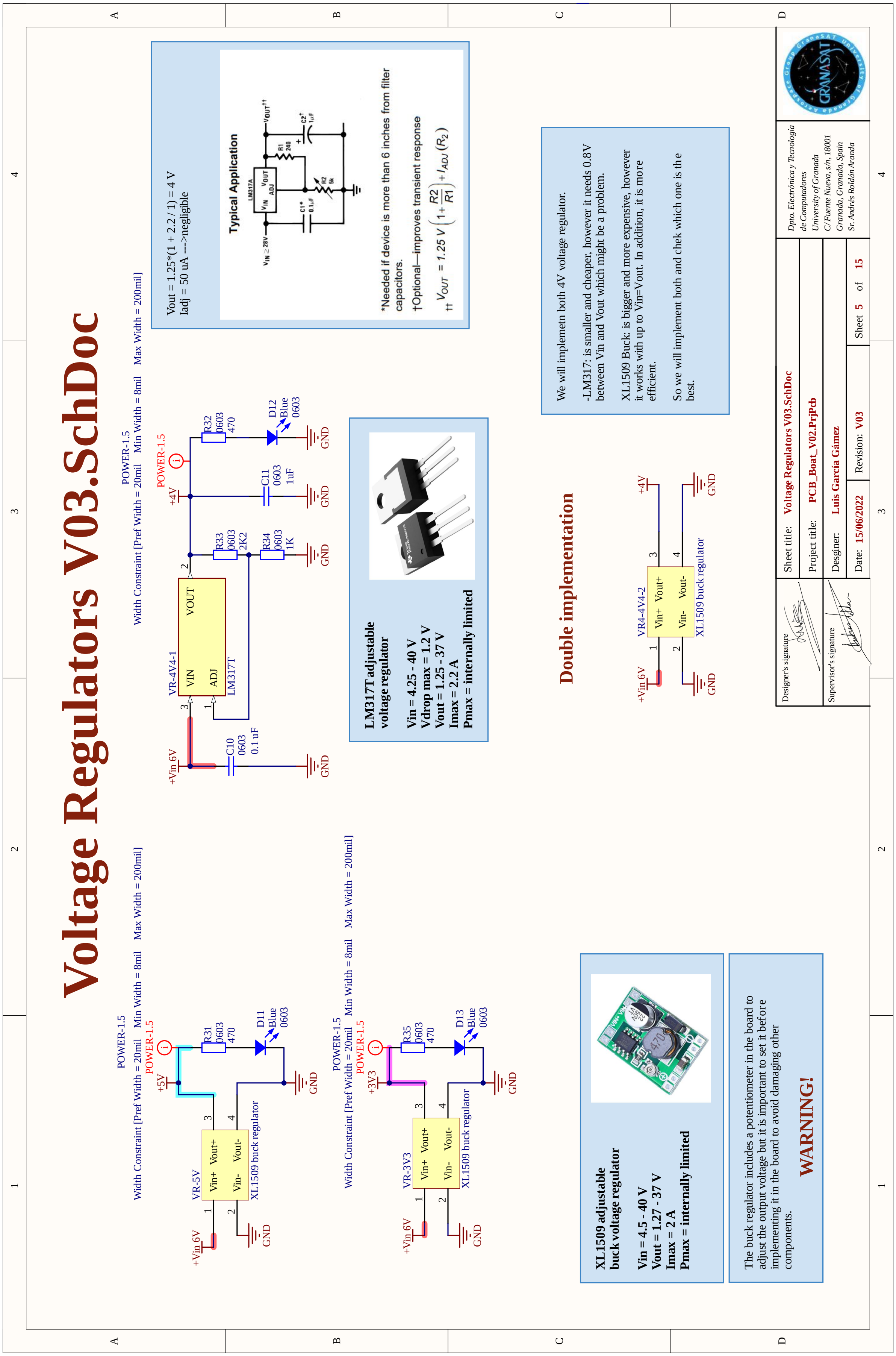
Sheet title: Power switch V01.SchDoc

Project title: PCB_Boat_V02.PrjPcb

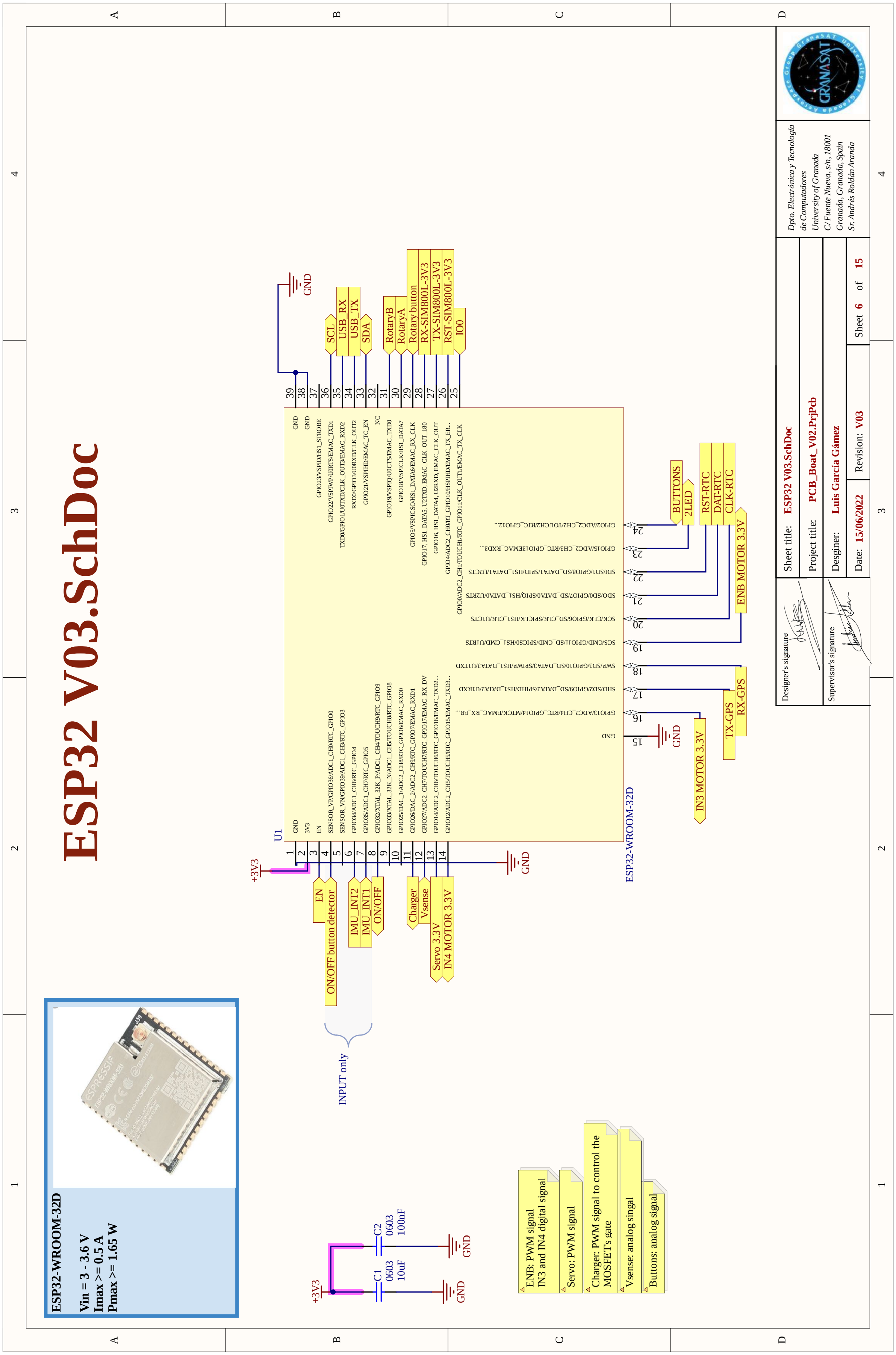
Desginer: Luis García Gámez

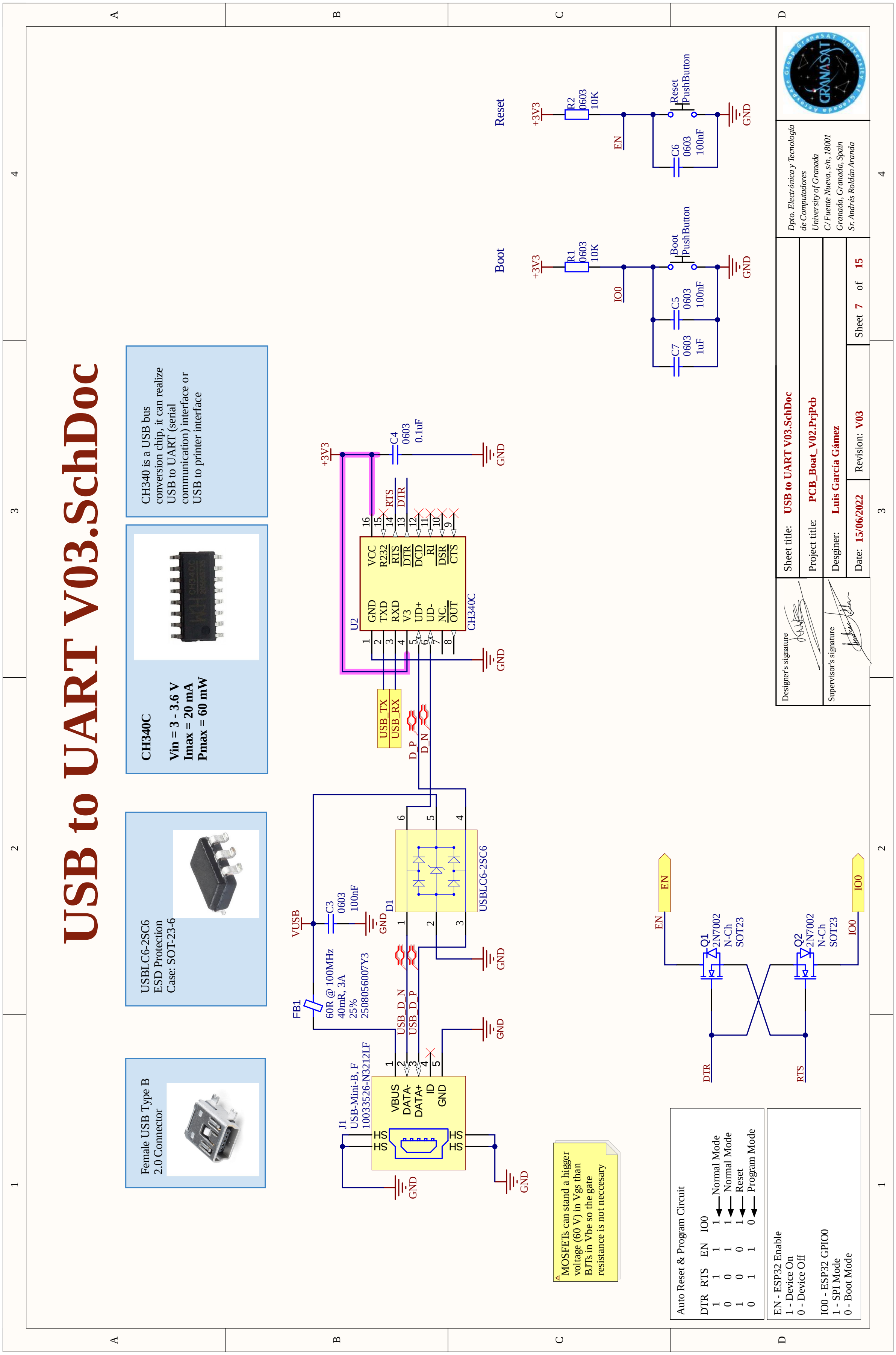
Date: 15/06/2022 Revision: Sheet 4 of 15

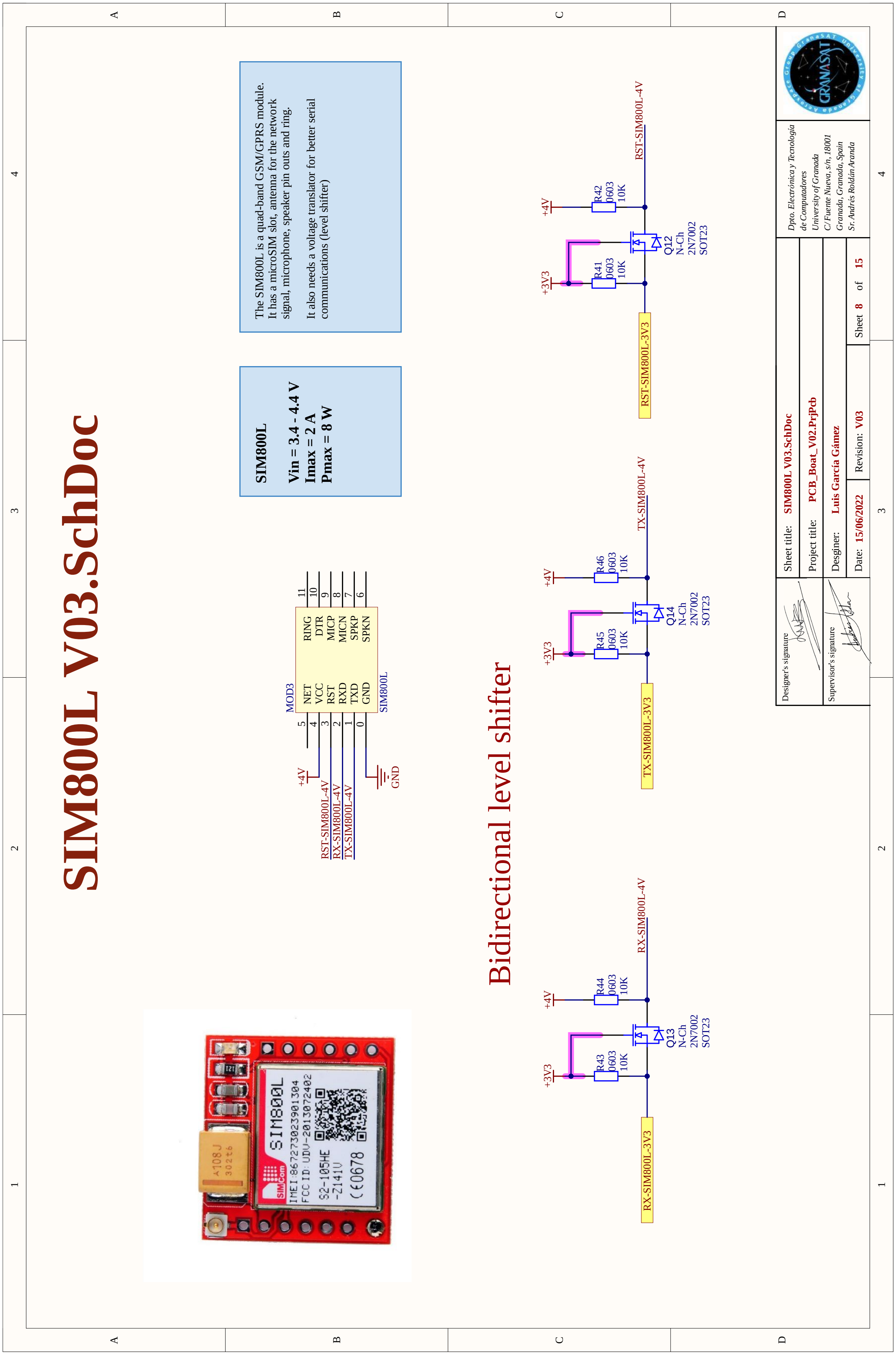
Dpto. Electrónica y Tecnología de Computadores
University of Granada
C/ Fuente Nueva, s/n, 18001
Granada, Granada, Spain
Sr. Andrés Roldán Aranda



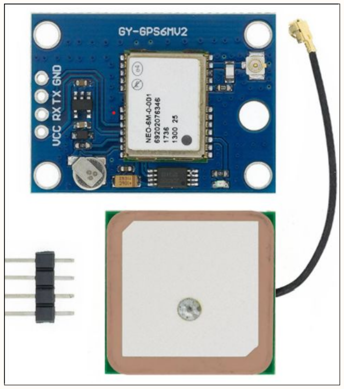
Voltage Regulators V03.SchDoc
Width Constraint [Pref Width = 20mil Min Width = 8mil Max Width = 200mil]
POWER-1.5
+Vin 6V
VR-5V
Vin+ Vout+
Vin- Vout-
XL1509 buck regulator
+5V
R31 0603 470
D11 Blue 0603
GND
VR-3V3
+3V3
R35 0603 470
D13 Blue 0603
XL1509 adjustable buck voltage regulator
Vin = 4.5 - 40 V
Vout = 1.27 - 37 V
Imax = 2 A
Pmax = internally limited
The buck regulator includes a potentiometer in the board to adjust the output voltage but it is important to set it before implementing it in the board to avoid damaging other components.
WARNING!
C10 0603 0.1 uF
VR-4V4-1
VIN VOUT ADJ
LM317T
R33 0603 2K2
R34 0603 1K
C11 0603 1uF
R32 0603 470
D12 Blue 0603
+4V
LM317T adjustable voltage regulator
Vin = 4.25 - 40 V
Vdrop max = 1.2 V
Vout = 1.25 - 37 V
Imax = 2.2 A
Pmax = internally limited
Vout = 1.25*(1 + 2.2 / 1) = 4 V
Iadj = 50 uA --->negligible
Typical Application
*Needed if device is more than 6 inches from filter capacitors.
†Optional—improves transient response
$V_{OUT} = 1.25\,V\left(1+\frac{R2}{R1}\right)+I_{ADJ}(R_2)$
Double implementation
VR4-4V4-2
We will implemen both 4V voltage regulator.
-LM317: is smaller and cheaper, however it needs 0.8V between Vin and Vout which might be a problem.
XL1509 Buck: is bigger and more expensive, however it works with up to Vin=Vout. In addition, it is more efficient.
So we will implement both and chek which one is the best.
Designer's signature
Supervisor's signature
Sheet title: Voltage Regulators V03.SchDoc
Project title: PCB_Boat_V02.PrjPcb
Designer: Luis García Gámez
Date: 15/06/2022
Revision: V03
Sheet 5 of 15
Dpto. Electrónica y Tecnología de Computadores
University of Granada
C/ Fuente Nueva, s/n, 18001
Granada, Granada, Spain
Sr. Andrés Roldán Aranda

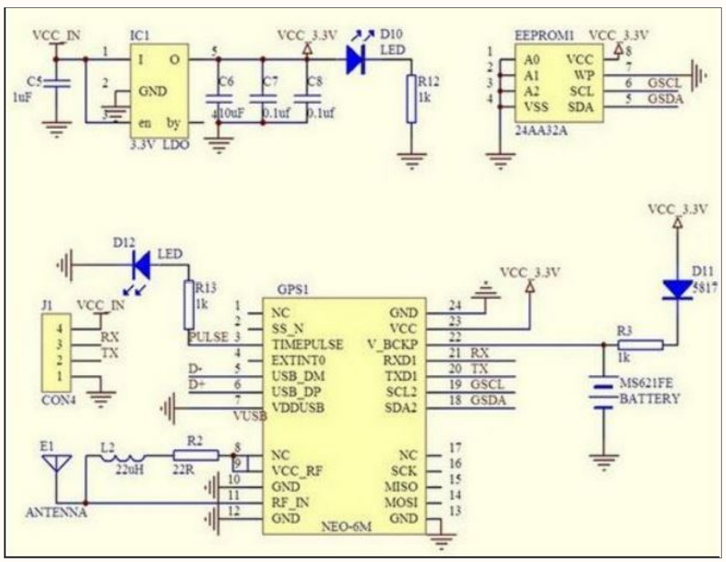

ESP32 V03.SchDoc
ESP32-WROOM-32D
Vin = 3 - 3.6 V
Imax >= 0.5 A
Pmax >= 1.65 W
U1
ESP32-WROOM-32D
+3V3
GND
EN
ON/OFF button detector
IMU_INT2
IMU_INT1
ON/OFF
Charger
Vsense
Servo 3.3V
IN4 MOTOR 3.3V
INPUT only
SCL
USB_RX
USB_TX
SDA
RotaryB
RotaryA
Rotary button
RX-SIM800L-3V3
TX-SIM800L-3V3
RST-SIM800L-3V3
IO0
BUTTONS
2LED
RST-RTC
DAT-RTC
CLK-RTC
ENB MOTOR 3.3V
TX-GPS
RX-GPS
IN3 MOTOR 3.3V
C1
0603
10uF
C2
0603
100nF
ENB: PWM signal
IN3 and IN4 digital signal
Servo: PWM signal
Charger: PWM signal to control the MOSFET's gate
Vsense: analog singal
Buttons: analog signal
Designer's signature
Supervisor's signature
Sheet title: ESP32 V03.SchDoc
Project title: PCB_Boat_V02.PrjPcb
Designer: Luis García Gámez
Date: 15/06/2022
Revision: V03
Sheet 6 of 15
Dpto. Electrónica y Tecnología de Computadores
University of Granada
C/ Fuente Nueva, s/n, 18001
Granada, Granada, Spain
Sr. Andrés Roldán Aranda


USB to UART V03.SchDoc
Female USB Type B
2.0 Connector
USBLC6-2SC6
ESD Protection
Case: SOT-23-6
CH340C
Vin = 3 - 3.6 V
Imax = 20 mA
Pmax = 60 mW
CH340 is a USB bus conversion chip, it can realize USB to UART (serial communication) interface or USB to printer interface
MOSFETs can stand a bigger voltage (60 V) in Vgs than BJTs in Vbe so the gate resistance is not neccesary
Auto Reset & Program Circuit
DTR RTS EN IO0
1 1 1 1 Normal Mode
0 0 1 1 Normal Mode
1 0 0 1 Reset
0 1 1 0 Program Mode
EN - ESP32 Enable
1 - Device On
0 - Device Off
IO0 - ESP32 GPIO0
1 - SPI Mode
0 - Boot Mode
J1
USB-Mini-B, F
10033526-N3212LF
FB1
60R @ 100MHz
40mR, 3A
25%
250805600Y3
C3
0603
100nF
D1
USBLC6-2SC6
U2
CH340C
C4
0603
0.1uF
Boot
Reset
R1
0603
10K
R2
0603
10K
C5
0603
100nF
C6
0603
100nF
C7
0603
1uF
Q1
2N7002
N-Ch
SOT23
Q2
2N7002
N-Ch
SOT23
Sheet title: USB to UART V03.SchDoc
Project title: PCB_Boat_V02.PrjPcb
Designer: Luis García Gámez
Date: 15/06/2022
Revision: V03
Sheet 7 of 15
Designer's signature
Supervisor's signature
Dpto. Electrónica y Tecnología de Computadores
University of Granada
C/Fuente Nueva, s/n, 18001
Granada, Granada, Spain
Sr. Andrés Roldán Aranda


SIM800L V03.SchDoc
SIM800L
Vin = 3.4 - 4.4 V
Imax = 2 A
Pmax = 8 W
The SIM800L is a quad-band GSM/GPRS module. It has a microSIM slot, antenna for the network signal, microphone, speaker pin outs and ring.
It also needs a voltage translator for better serial communications (level shifter)
Bidirectional level shifter
MOD3
SIM800L
NET
VCC
RST
RXD
TXD
GND
RING
DTR
MICP
MICN
SPKP
SPKN
+4V
GND
RST-SIM800L-4V
RX-SIM800L-4V
TX-SIM800L-4V
RST-SIM800L-3V3
TX-SIM800L-3V3
RX-SIM800L-3V3
+3V3
R41 0603 10K
R42 0603 10K
R43 0603 10K
R44 0603 10K
R45 0603 10K
R46 0603 10K
Q12 N-Ch 2N7002 SOT23
Q13 N-Ch 2N7002 SOT23
Q14 N-Ch 2N7002 SOT23
Sheet title: SIM800L V03.SchDoc
Project title: PCB_Boat_V02.PrjPcb
Designer: Luis García Gámez
Date: 15/06/2022
Revision: V03
Sheet 8 of 15
Designer's signature
Supervisor's signature
Dpto. Electrónica y Tecnología de Computadores
University of Granada
C/ Fuente Nueva, s/n, 18001
Granada, Granada, Spain
Sr. Andrés Roldán Aranda

GY-NEO6MV2 GPS V03.SchDoc





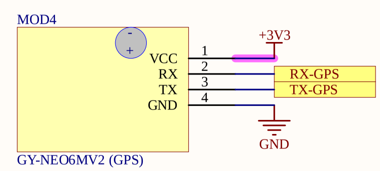
GY-NEO6MV2 (GPS)

Vin = 3 - 5 V
Imax = 10mA + 100mA (RF)
Pmax = mW

GY-NEO6MV2 board features the u-blox NEO-6M GPS module with antenna and built-in EEPROM.

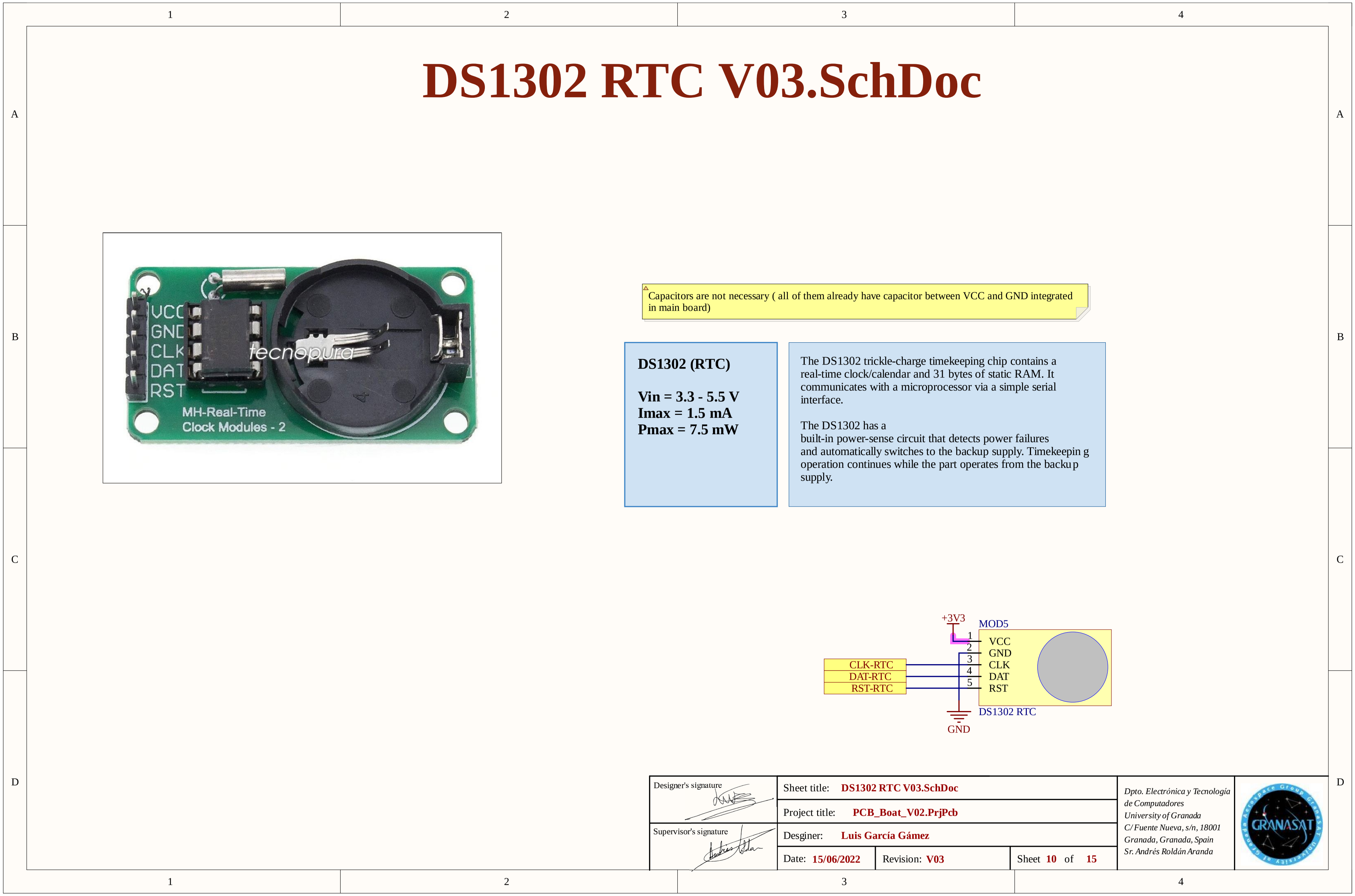
- Ceramic antenna
- EEPROM for saving the configuration data when powered off
- Backup battery
- LED signal indicator
- TTL level, compatible with 3.3V/5V system.

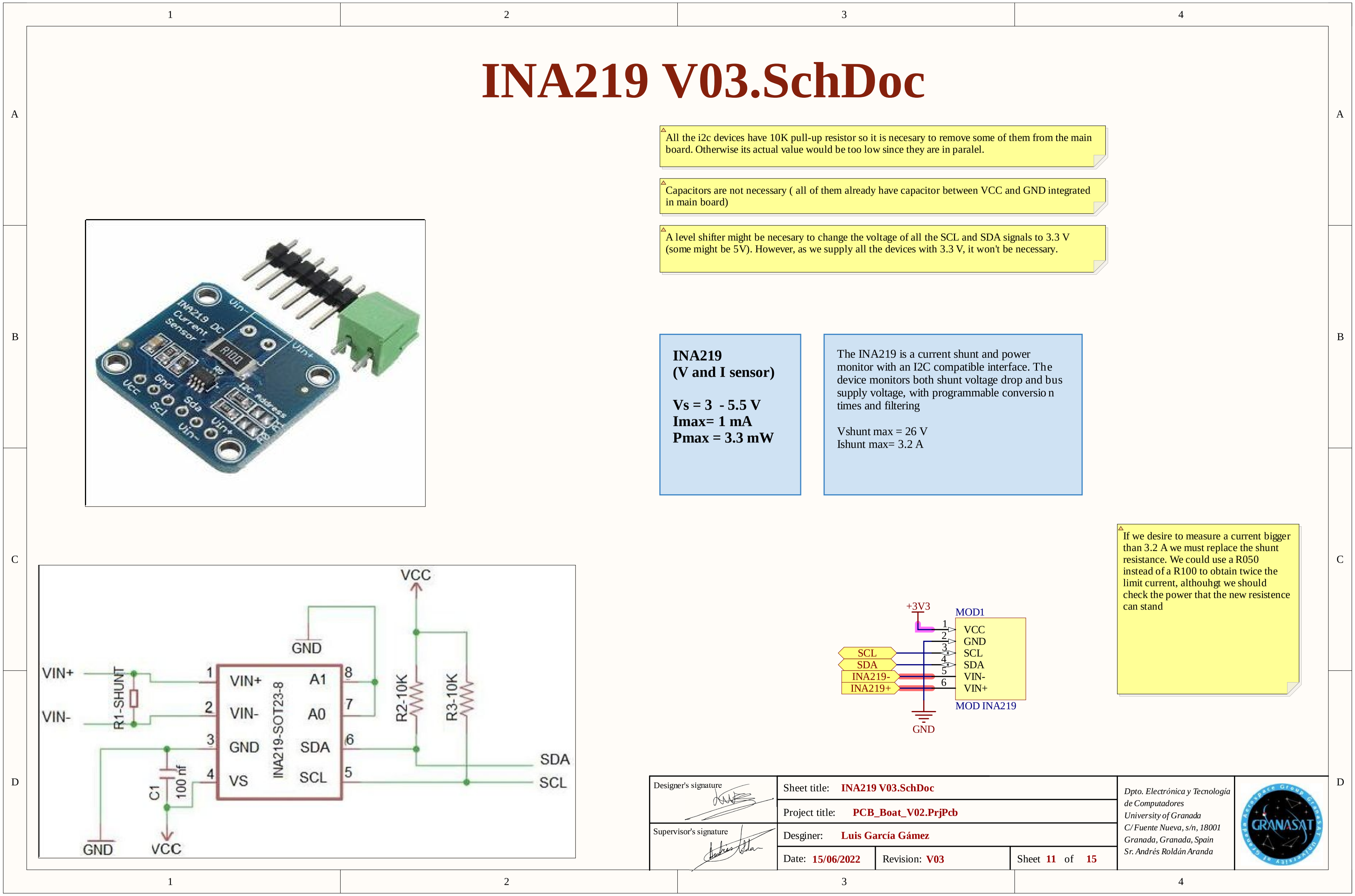
The backup battery is useful to keep the device connected to the GPS satellites. This connection might take up to 20 minutos to be established so keeping it active can save a lot of time when the device turn on



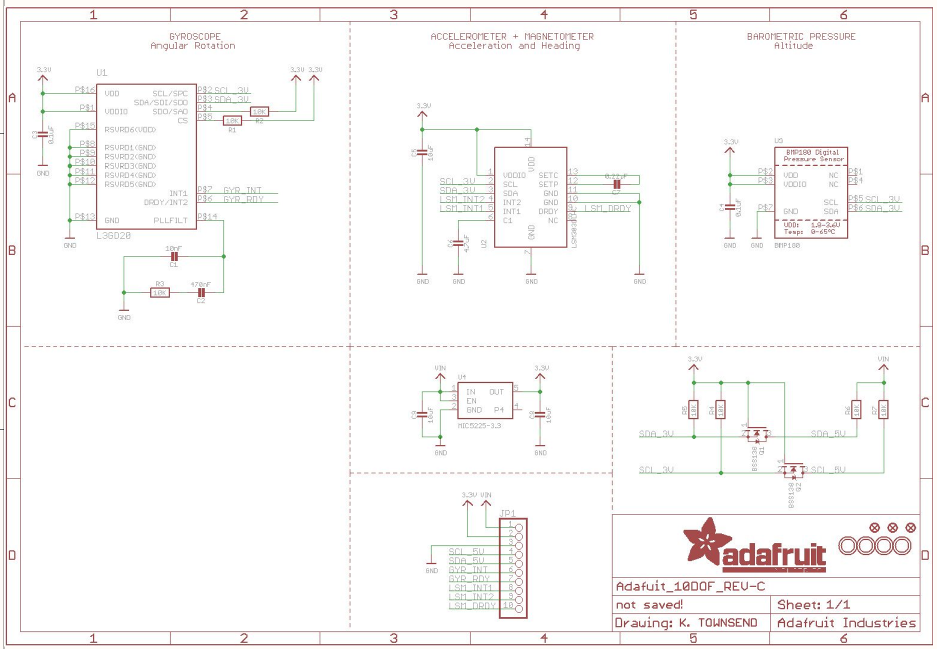
Designer's signature	Sheet title: GY-NEO6MV2 GPS V03.SchDoc
	Project title: PCB_Boat_V02.PrjPcb
Supervisor's signature	Desginer: Luis García Gámez
	Date: 15/06/2022 Revision: V03 Sheet 9 of 15

Dpto. Electrónica y Tecnología de Computadores
University of Granada
C/ Fuente Nueva, s/n, 18001
Granada, Granada, Spain
Sr. Andrés Roldán Aranda


DS1302 RTC V03.SchDoc
VCC GND CLK DAT RST
MH-Real-Time Clock Modules - 2
tecnopura
Capacitors are not necessary (all of them already have capacitor between VCC and GND integrated in main board)
DS1302 (RTC)
Vin = 3.3 - 5.5 V
Imax = 1.5 mA
Pmax = 7.5 mW
The DS1302 trickle-charge timekeeping chip contains a real-time clock/calendar and 31 bytes of static RAM. It communicates with a microprocessor via a simple serial interface.
The DS1302 has a built-in power-sense circuit that detects power failures and automatically switches to the backup supply. Timekeepin g operation continues while the part operates from the backu p supply.
+3V3
MOD5
1 VCC
2 GND
3 CLK
4 DAT
5 RST
CLK-RTC
DAT-RTC
RST-RTC
DS1302 RTC
GND
Designer's signature
Supervisor's signature
Sheet title: DS1302 RTC V03.SchDoc
Project title: PCB_Boat_V02.PrjPcb
Desginer: Luis García Gámez
Date: 15/06/2022
Revision: V03
Sheet 10 of 15
Dpto. Electrónica y Tecnología de Computadores
University of Granada
C/ Fuente Nueva, s/n, 18001
Granada, Granada, Spain
Sr. Andrés Roldán Aranda
GRANASAT


INA219 V03.SchDoc
All the i2c devices have 10K pull-up resistor so it is necesary to remove some of them from the main board. Otherwise its actual value would be too low since they are in paralel.
Capacitors are not necessary (all of them already have capacitor between VCC and GND integrated in main board)
A level shifter might be necesary to change the voltage of all the SCL and SDA signals to 3.3 V (some might be 5V). However, as we supply all the devices with 3.3 V, it won't be necessary.
INA219
(V and I sensor)
Vs = 3 - 5.5 V
Imax= 1 mA
Pmax = 3.3 mW
The INA219 is a current shunt and power monitor with an I2C compatible interface. The device monitors both shunt voltage drop and bus supply voltage, with programmable conversio n times and filtering
Vshunt max = 26 V
Ishunt max= 3.2 A
If we desire to measure a current bigger than 3.2 A we must replace the shunt resistance. We could use a R050 instead of a R100 to obtain twice the limit current, althouhgt we should check the power that the new resistence can stand
VIN+
VIN-
R1-SHUNT
GND
VCC
C1
100 nf
VIN+
VIN-
GND
VS
A1
A0
SDA
SCL
INA219-SOT23-8
R2-10K
R3-10K
SDA
SCL
+3V3
MOD1
VCC
GND
SCL
SDA
VIN-
VIN+
INA219-
INA219+
MOD INA219
Designer's signature
Supervisor's signature
Sheet title: INA219 V03.SchDoc
Project title: PCB_Boat_V02.PrjPcb
Desginer: Luis García Gámez
Date: 15/06/2022
Revision: V03
Sheet 11 of 15
Dpto. Electrónica y Tecnología de Computadores
University of Granada
C/ Fuente Nueva, s/n, 18001
Granada, Granada, Spain
Sr. Andrés Roldán Aranda

Adafruit-10-dof V03.SchDoc



All the i2c devices have 10K pull-up resistor so it is necesary to remove some of them from the main board. Otherwise its actual value would be too low since they are in paralel.

Capacitors are not necessary (all of them already have capacitor between VCC and GND integrated in main board)

A level shifter might be necesary to change the voltage of all the SCL and SDA signals to 3.3 V (some might be 5V). However, as we supply all the devices with 3.3 V, it won't be necessary.

Adafruit 10-dof (IMU)

Vin = 3.3/5 V
Imax = 6.1 mA
Pmax = 20,1 mW

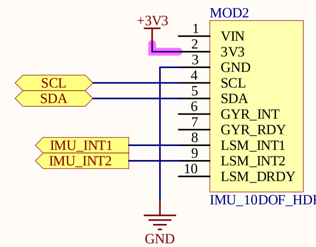
Adafruit's 10DOF (10 Degrees of Freedom) breakout board allows you to capture ten distinct types of motion or orientation related data.

It has several differents measurement devices:

LSM303DLHC - a 3-axis accelerometer (up to +/-16g) and a 3-axis magnetometer (up to +/-8.1 gauss) on a single die

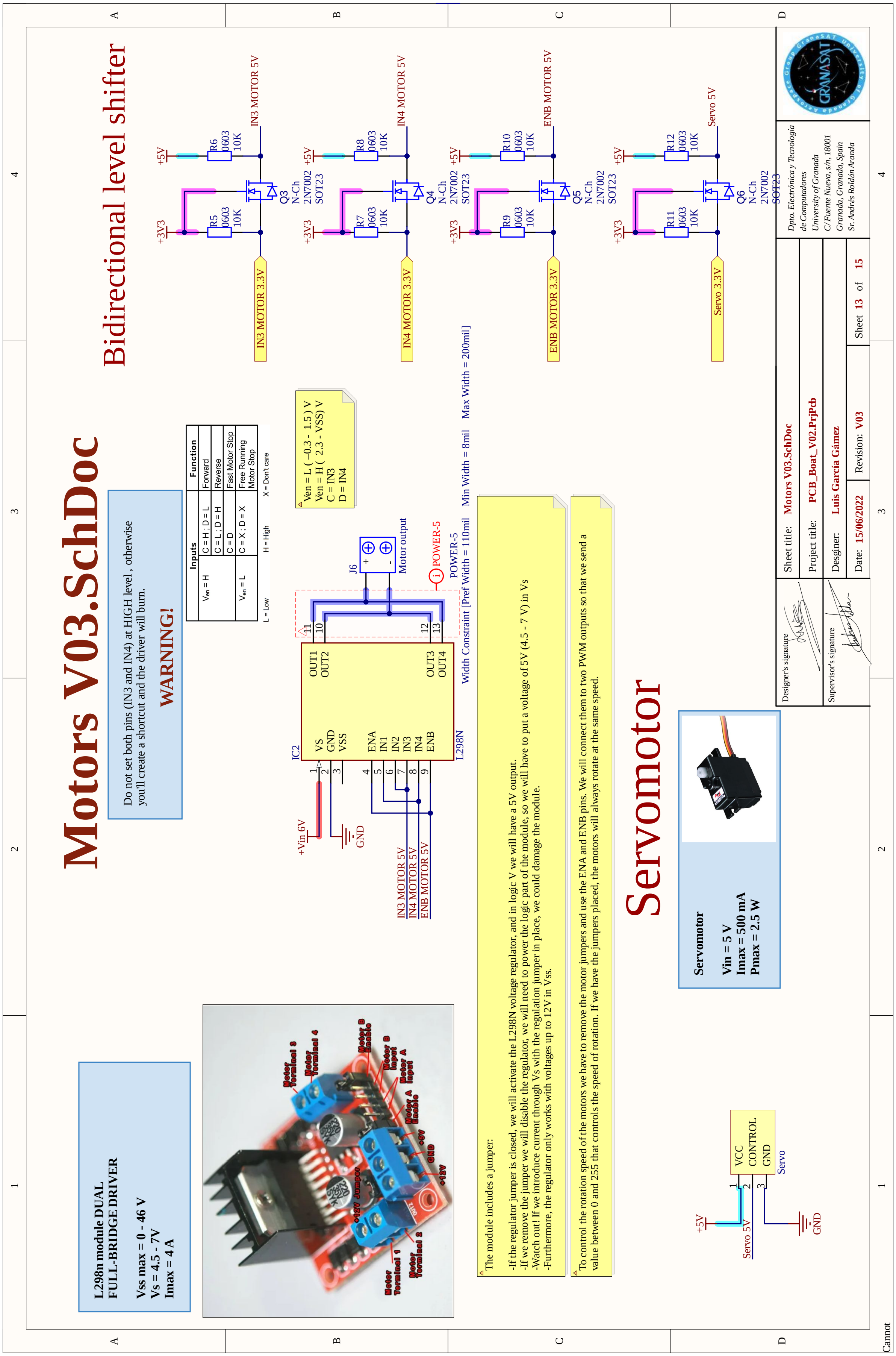
L3GD20 - a 3-axis gyroscope (up to +/-2000 dps)

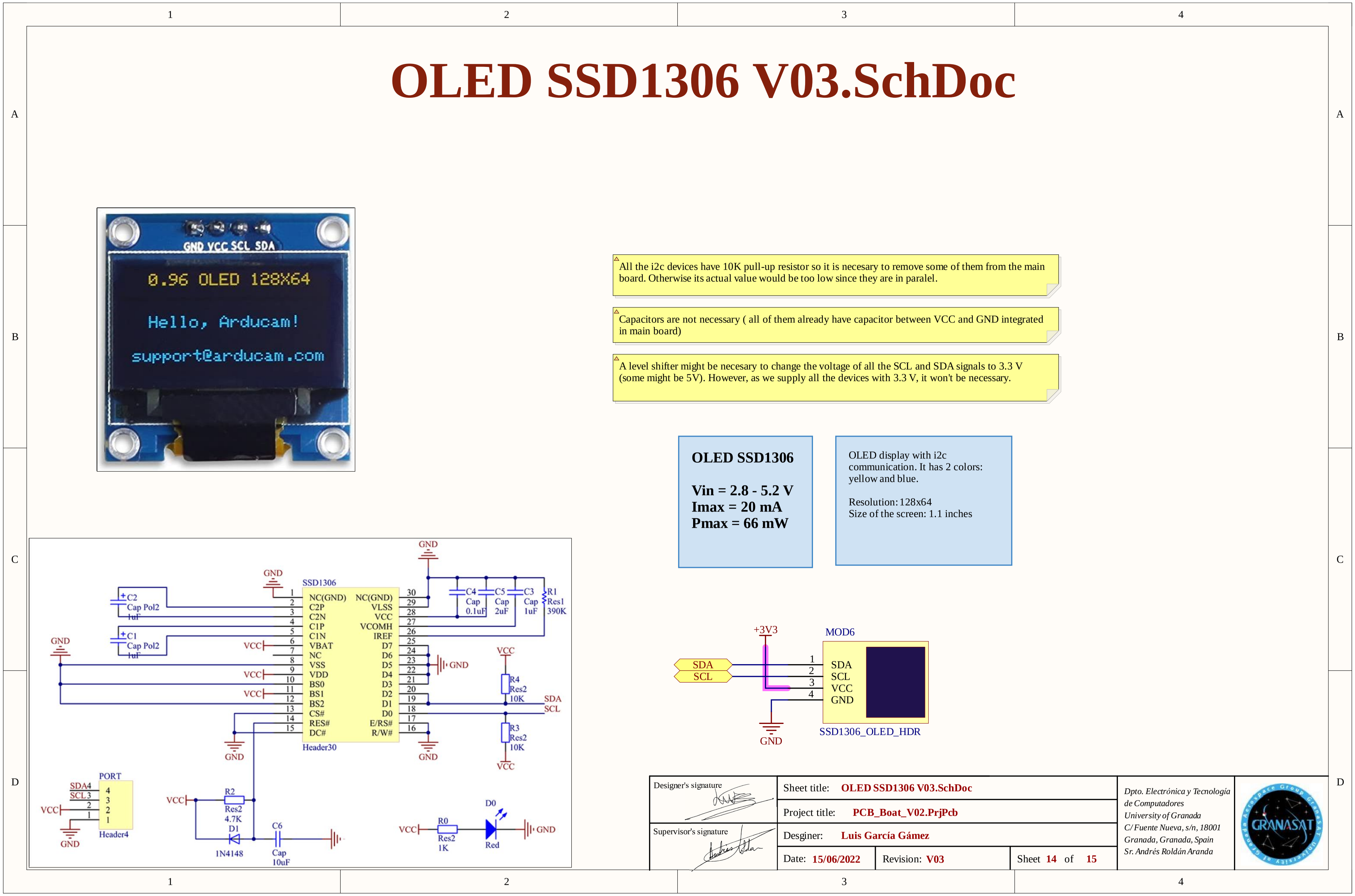
BMP180 - A barometric pressure sensor (300..1100 hPa) that can be used to calculate altitude, with an additional on-board temperature sensor



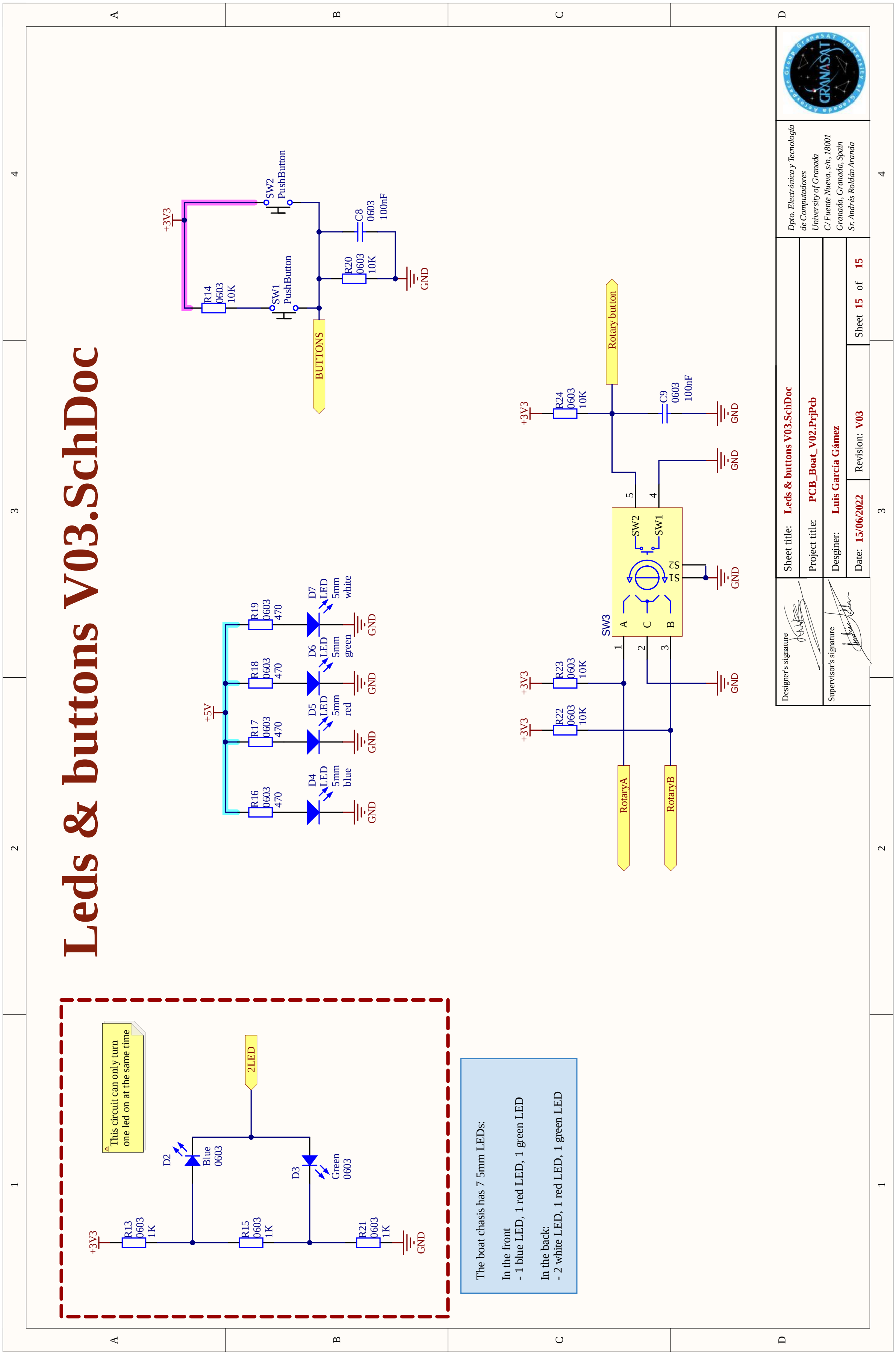
Add interruption pins if there is enough free pins in the ES32

Designer's signature	Sheet title: **Adafruit-10-dof V03.SchDoc**		*Dpto. Electrónica y Tecnología de Computadores*
	Project title: **PCB_Boat_V02.PrjPcb**		*University of Granada*
Supervisor's signature	Desginer: **Luis García Gámez**		*C/ Fuente Nueva, s/n, 18001*
	Date: **15/06/2022**	Revision: **V03** Sheet **12** of **15**	*Granada, Granada, Spain* *Sr. Andrés Roldán Aranda*


Motors V03.SchDoc
L298n module DUAL
FULL-BRIDGE DRIVER
Vss max = 0 - 46 V
Vs = 4.5 - 7V
Imax = 4 A
Do not set both pins (IN3 and IN4) at HIGH level , otherwise you'll create a shortcut and the driver will burn.
WARNING!
Inputs | Function
Ven = H | C = H ; D = L | Forward
C = L ; D = H | Reverse
C = D | Fast Motor Stop
Ven = L | C = X ; D = X | Free Running Motor Stop
L = Low H = High X = Don't care
Ven = L (-0.3 - 1.5) V
Ven = H (2.3 - VSS) V
C = IN3
D = IN4
Bidirectional level shifter
IC2
L298N
VS GND VSS ENA IN1 IN2 IN3 IN4 ENB
OUT1 OUT2 OUT3 OUT4
+Vin 6V
GND
IN3 MOTOR 5V
IN4 MOTOR 5V
ENB MOTOR 5V
J6
Motor output
POWER-5
Width Constraint [Pref Width = 110mil Min Width = 8mil Max Width = 200mil]
IN3 MOTOR 3.3V
IN4 MOTOR 3.3V
ENB MOTOR 3.3V
Servo 3.3V
Servo 5V
+3V3
+5V
R5 0603 10K
R6 0603 10K
R7 0603 10K
R8 0603 10K
R9 0603 10K
R10 0603 10K
R11 0603 10K
R12 0603 10K
Q3 N-Ch 2N7002 SOT23
Q4 N-Ch 2N7002 SOT23
Q5 N-Ch 2N7002 SOT23
Q6 N-Ch 2N7002 SOT23
The module includes a jumper:
-If the regulator jumper is closed, we will activate the L298N voltage regulator, and in logic V we will have a 5V output.
-If we remove the jumper we will disable the regulator, we will need to power the logic part of the module, so we will have to put a voltage of 5V (4.5 - 7 V) in Vs
-Watch out! If we introduce current through Vs with the regulation jumper in place, we could damage the module.
-Furthermore, the regulator only works with voltages up to 12V in Vss.
To control the rotation speed of the motors we have to remove the motor jumpers and use the ENA and ENB pins. We will connect them to two PWM outputs so that we send a value between 0 and 255 that controls the speed of rotation. If we have the jumpers placed, the motors will always rotate at the same speed.
Servomotor
Servomotor
Vin = 5 V
Imax = 500 mA
Pmax = 2.5 W
VCC
CONTROL
GND
Servo
Designer's signature
Supervisor's signature
Sheet title: Motors V03.SchDoc
Project title: PCB_Boat_V02.PrjPcb
Desginer: Luis García Gámez
Date: 15/06/2022 Revision: V03 Sheet 13 of 15
Dpto. Electrónica y Tecnología de Computadores
University of Granada
C/ Fuente Nueva, s/n, 18001
Granada, Granada, Spain
Sr. Andrés Roldán Aranda
Cannot



OLED SSD1306 V03.SchDoc
GND VCC SCL SDA
0.96 OLED 128X64
Hello, Arducam!
support@arducam.com
All the i2c devices have 10K pull-up resistor so it is necesary to remove some of them from the main board. Otherwise its actual value would be too low since they are in paralel.
Capacitors are not necessary (all of them already have capacitor between VCC and GND integrated in main board)
A level shifter might be necesary to change the voltage of all the SCL and SDA signals to 3.3 V (some might be 5V). However, as we supply all the devices with 3.3 V, it won't be necessary.
OLED SSD1306
Vin = 2.8 - 5.2 V
Imax = 20 mA
Pmax = 66 mW
OLED display with i2c communication. It has 2 colors: yellow and blue.
Resolution: 128x64
Size of the screen: 1.1 inches
SSD1306
Header30
NC(GND) C2P C2N C1P C1N VBAT NC VSS VDD BS0 BS1 BS2 CS# RES# DC#
NC(GND) VLSS VCC VCOMH IREF D7 D6 D5 D4 D3 D2 D1 D0 E/RS# R/W#
C2 Cap Pol2 1uF
C1 Cap Pol2 1uF
C4 Cap 0.1uF
C5 Cap 2uF
C3 Cap 1uF
R1 Res1 390K
R4 Res2 10K
R3 Res2 10K
PORT
Header4
R2 Res2 4.7K
D1 1N4148
C6 Cap 10uF
R0 Res2 1K
D0 Red
+3V3
MOD6
SDA SCL VCC GND
SSD1306_OLED_HDR
Designer's signature
Supervisor's signature
Sheet title: OLED SSD1306 V03.SchDoc
Project title: PCB_Boat_V02.PrjPcb
Designer: Luis García Gámez
Date: 15/06/2022
Revision: V03
Sheet 14 of 15
Dpto. Electrónica y Tecnología de Computadores
University of Granada
C/ Fuente Nueva, s/n, 18001
Granada, Granada, Spain
Sr. Andrés Roldán Aranda


Leds & buttons V03.SchDoc
This circuit can only turn one led on at the same time
The boat chasis has 7 5mm LEDs:
In the front
- 1 blue LED, 1 red LED, 1 green LED
In the back:
- 2 white LED, 1 red LED, 1 green LED
2LED
BUTTONS
Rotary button
RotaryA
RotaryB
SW1 PushButton
SW2 PushButton
SW3
GND
+3V3
+5V
Sheet title: Leds & buttons V03.SchDoc
Project title: PCB_Boat_V02.PrjPcb
Designer: Luis García Gámez
Date: 15/06/2022
Revision: V03
Sheet 15 of 15
Dpto. Electrónica y Tecnología de Computadores
University of Granada
C/Fuente Nueva, s/n, 18001
Granada, Granada, Spain
Sr. Andrés Roldán Aranda
Designer's signature
Supervisor's signature

Appendix F

PCB images

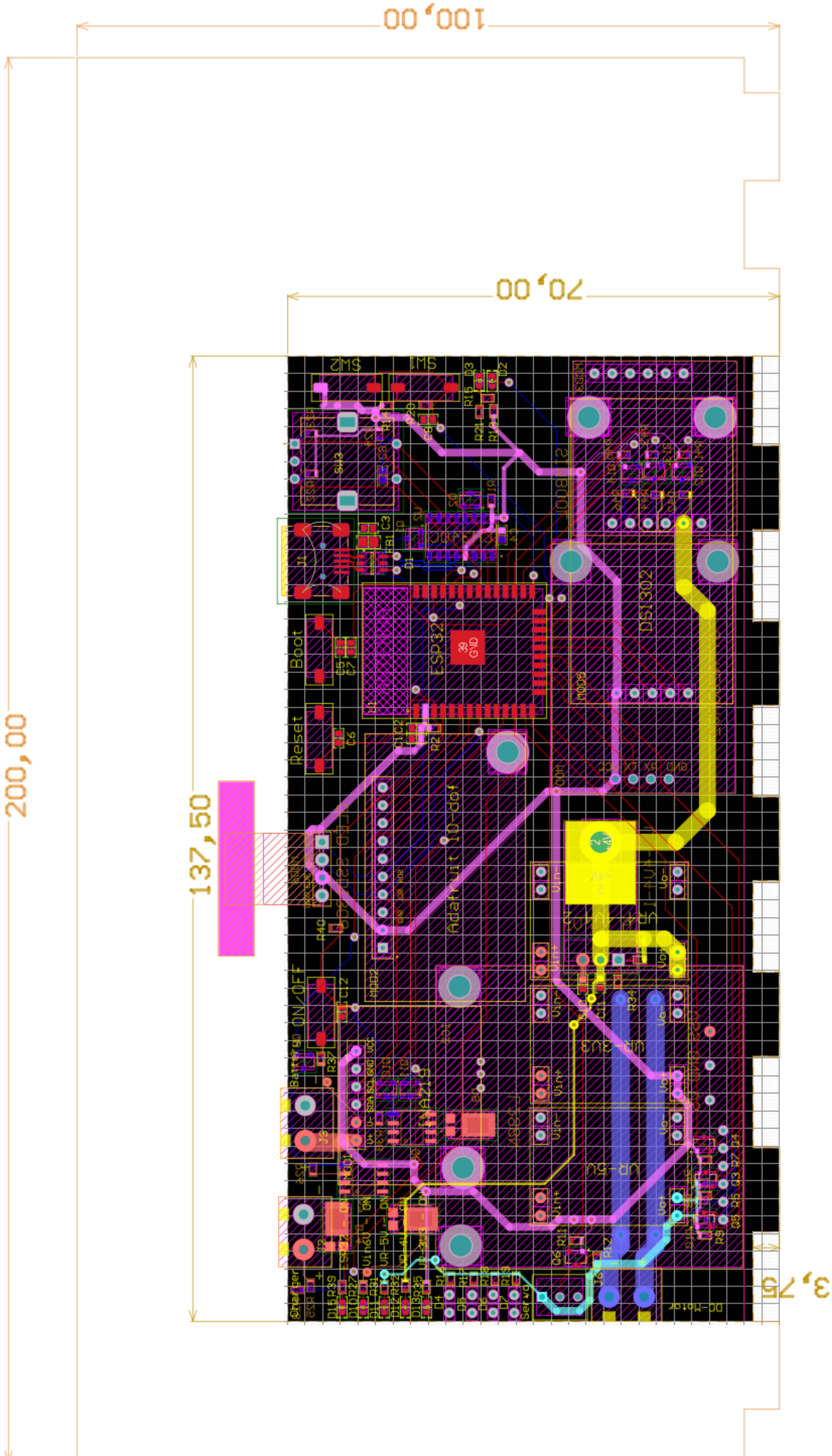


Figure F.1 – *Control PCB - size view*

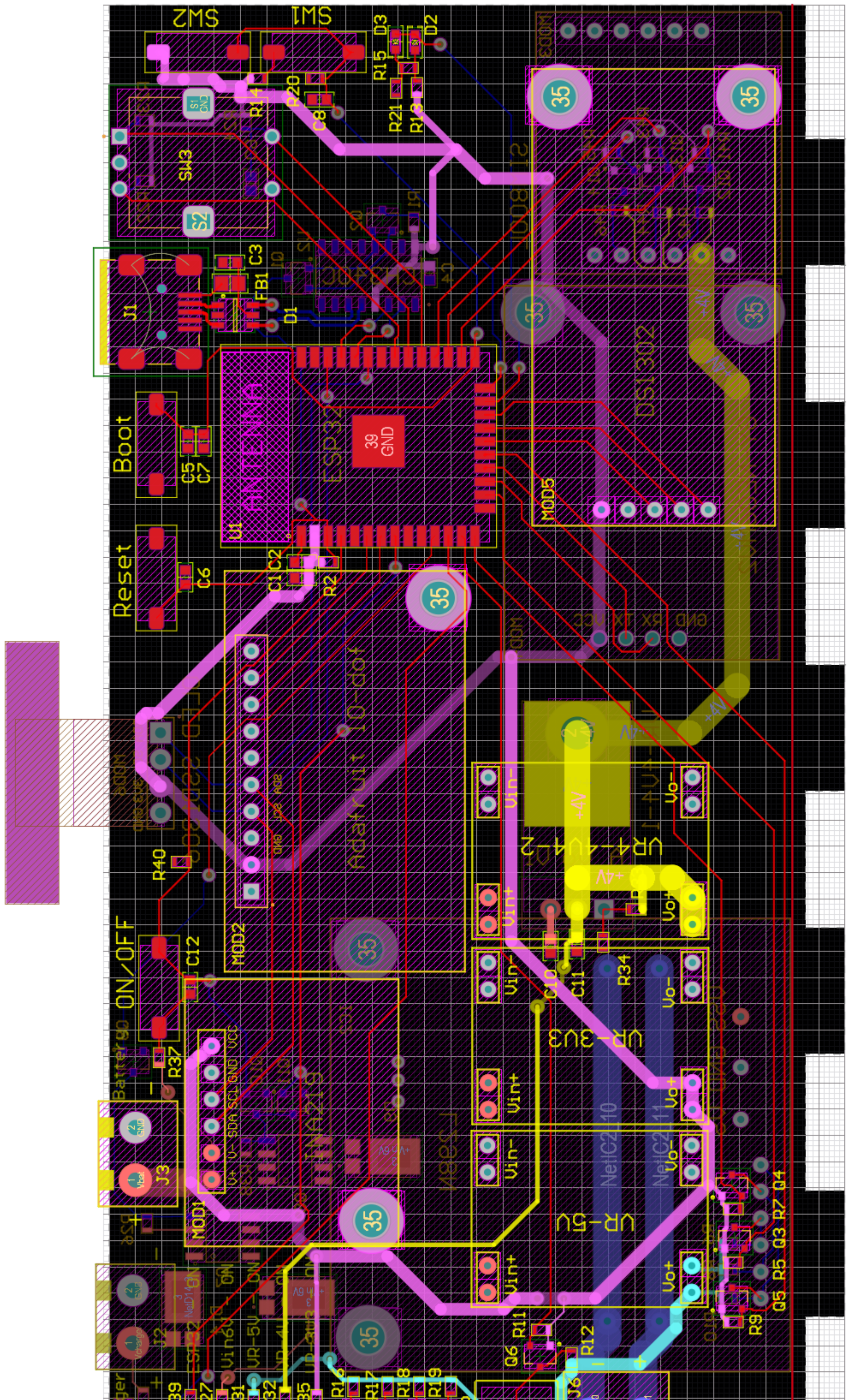


Figure F.2 – *Control PCB - top view highlighted*

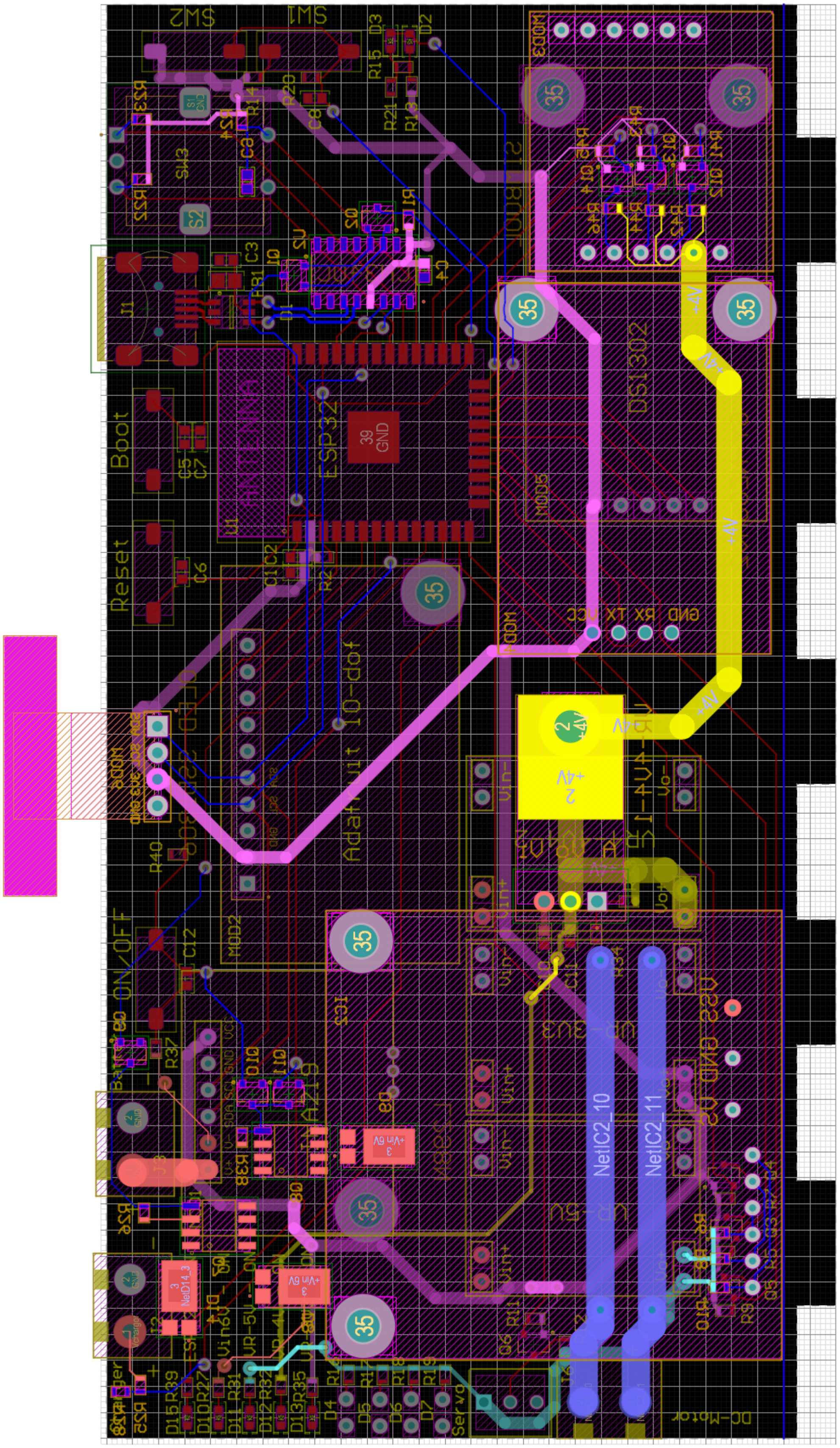


Figure F.3 – *Control PCB - bottom view highlighted*

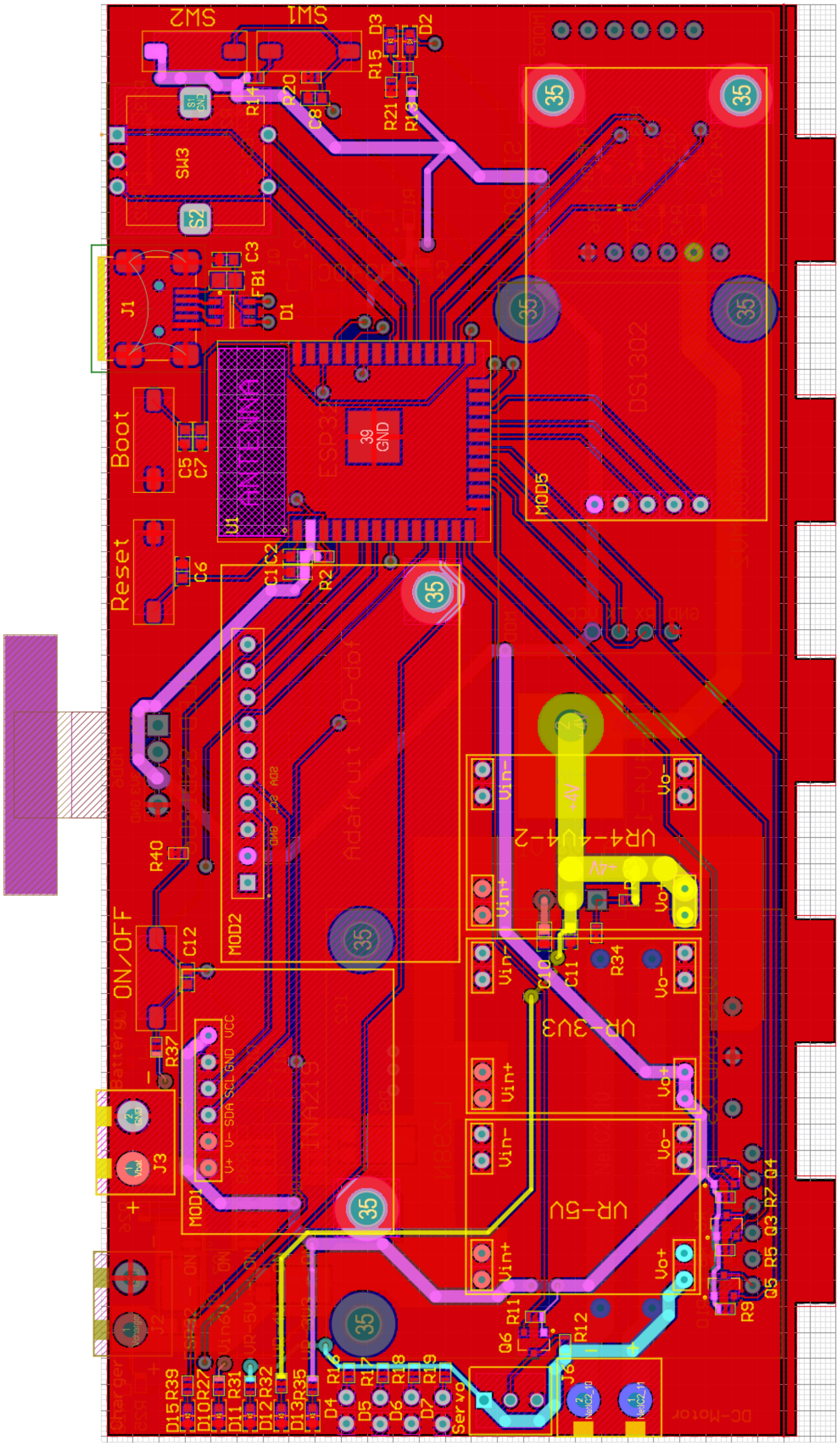


Figure F.4 – *Control* PCB *- top view with ground plane and polygons*

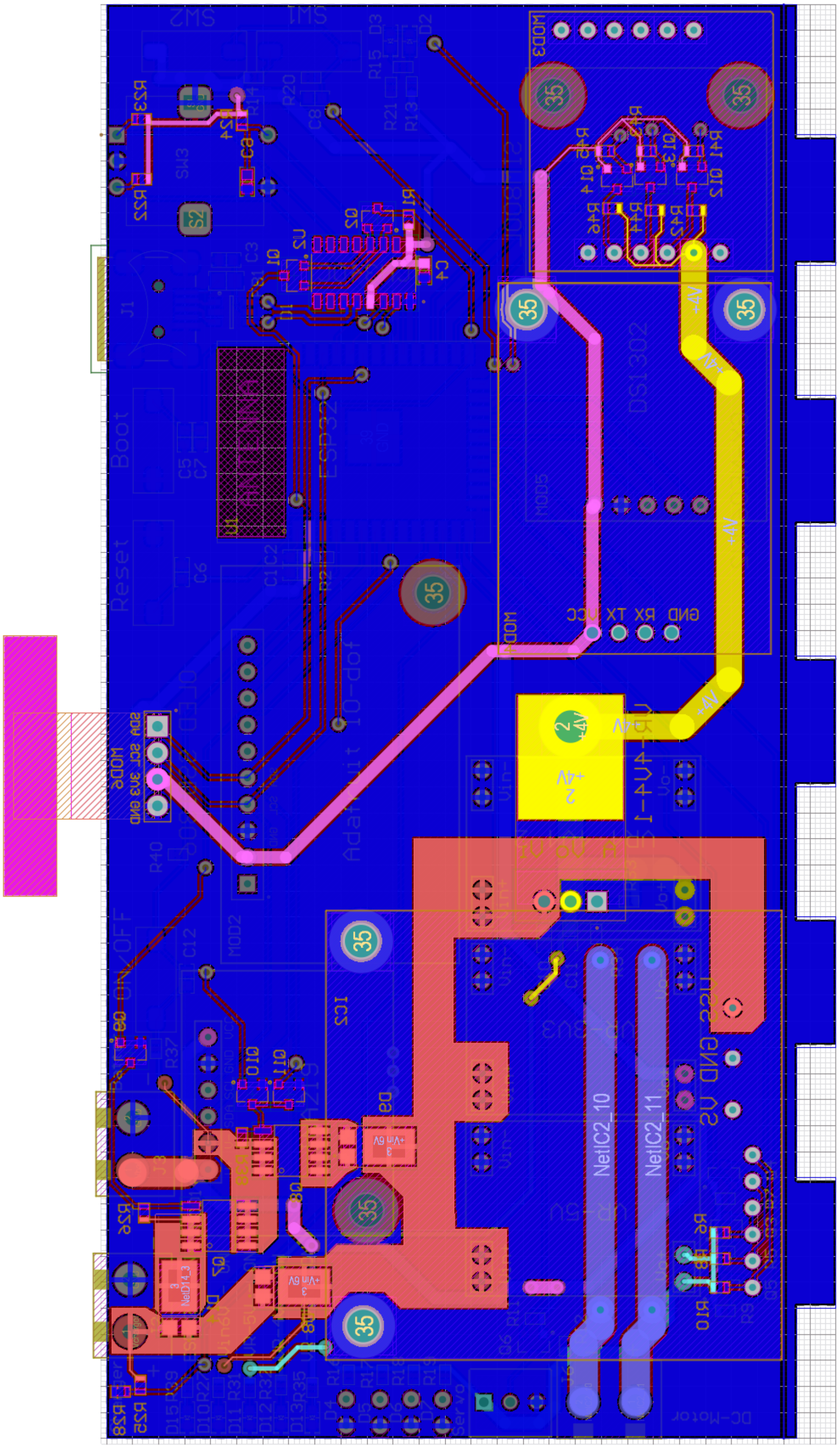


Figure F.5 – *Control PCB - bottom view with ground plane and polygons*

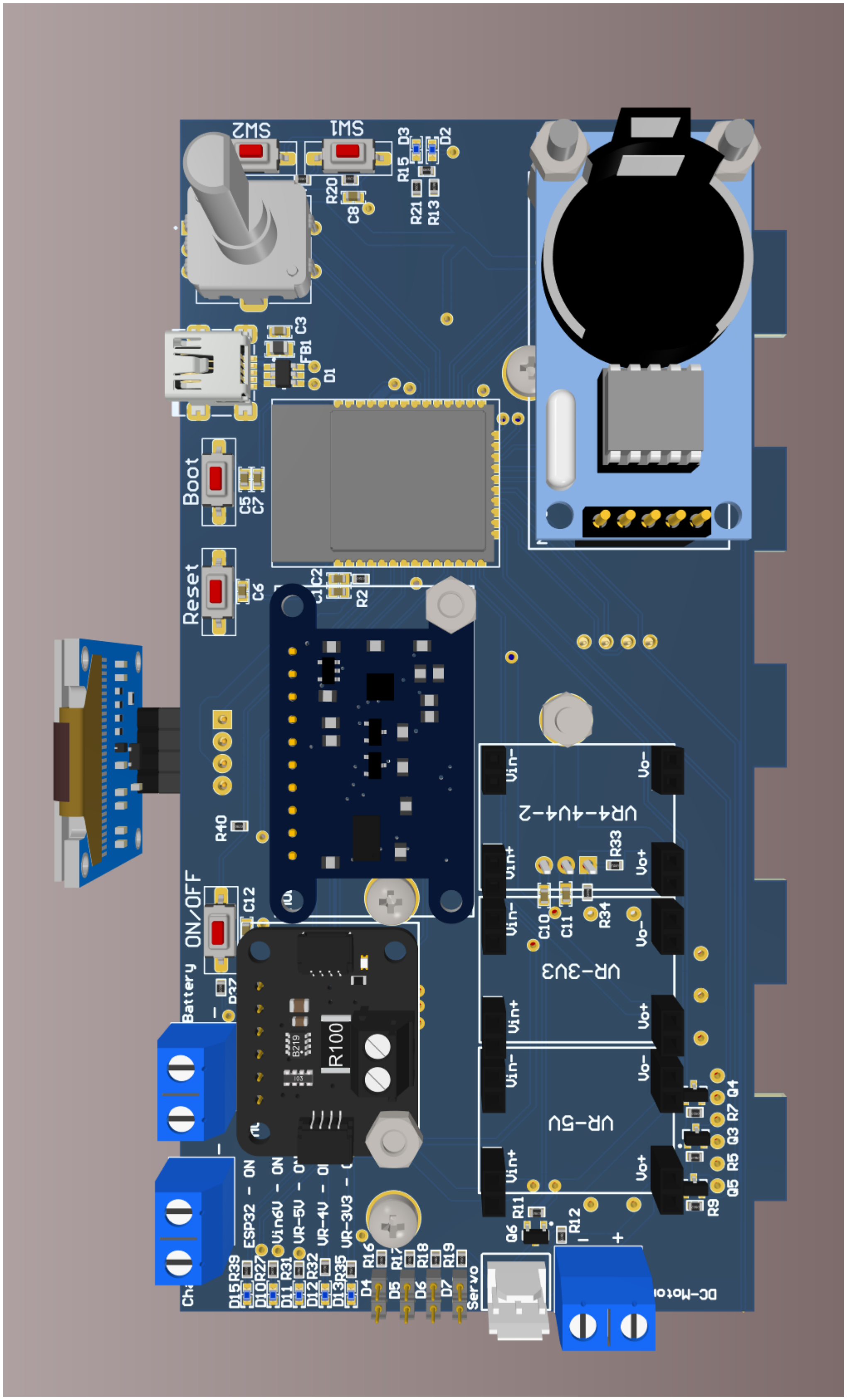


Figure F.6 – *Control PCB - 3D top view*

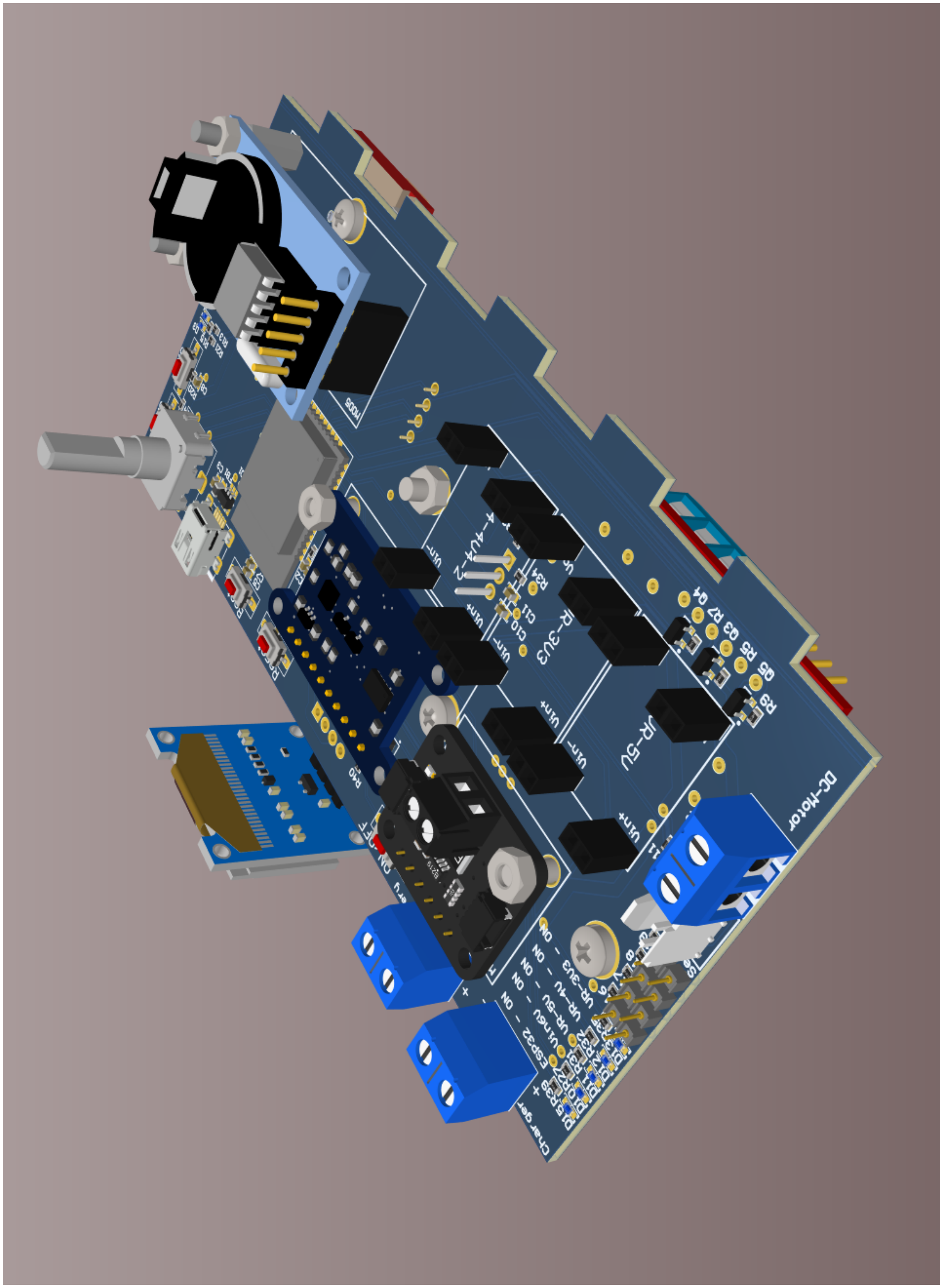


Figure F.7 – *Control PCB - 3D top view - 2*

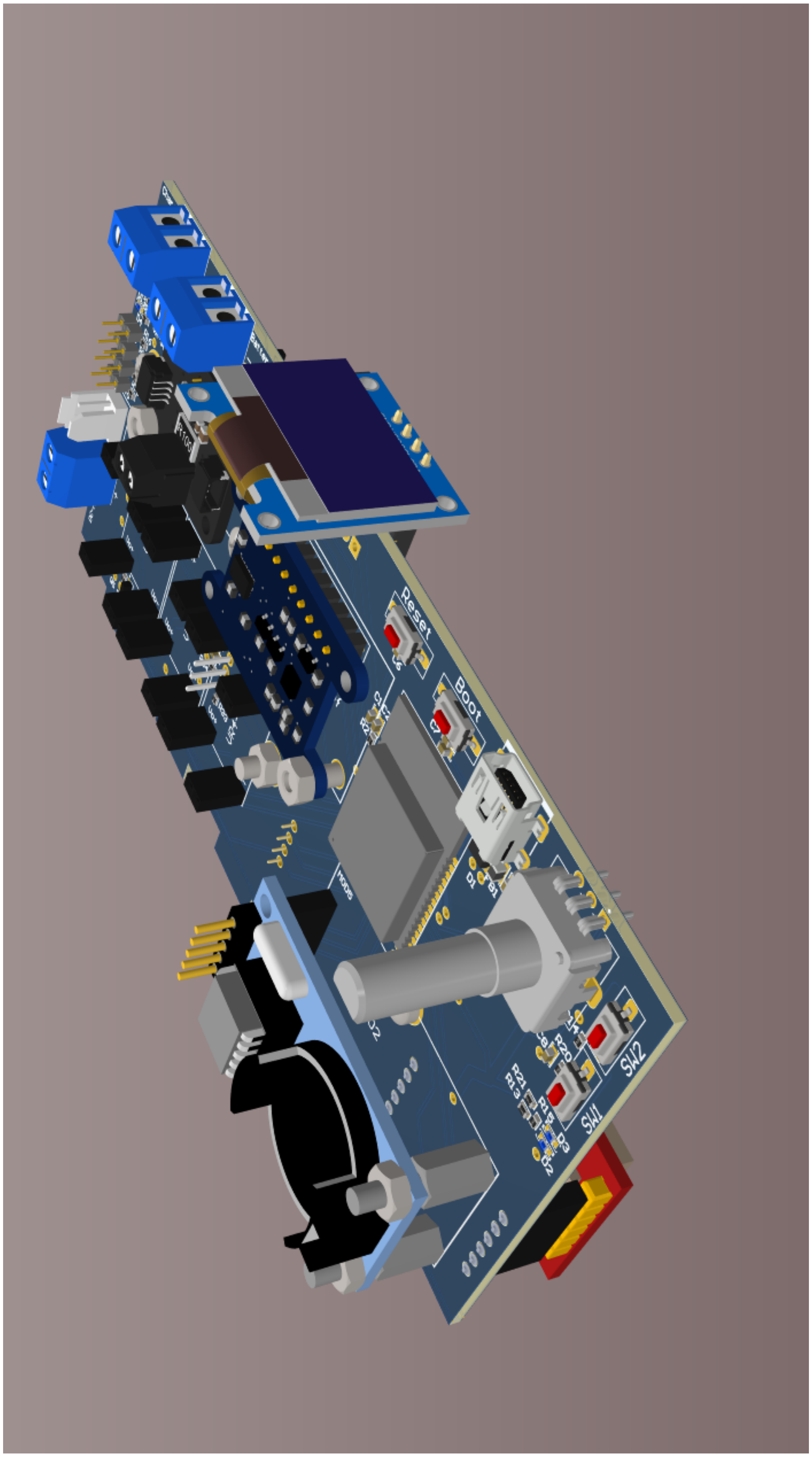


Figure F.8 – *Control PCB - 3D top view - 3*

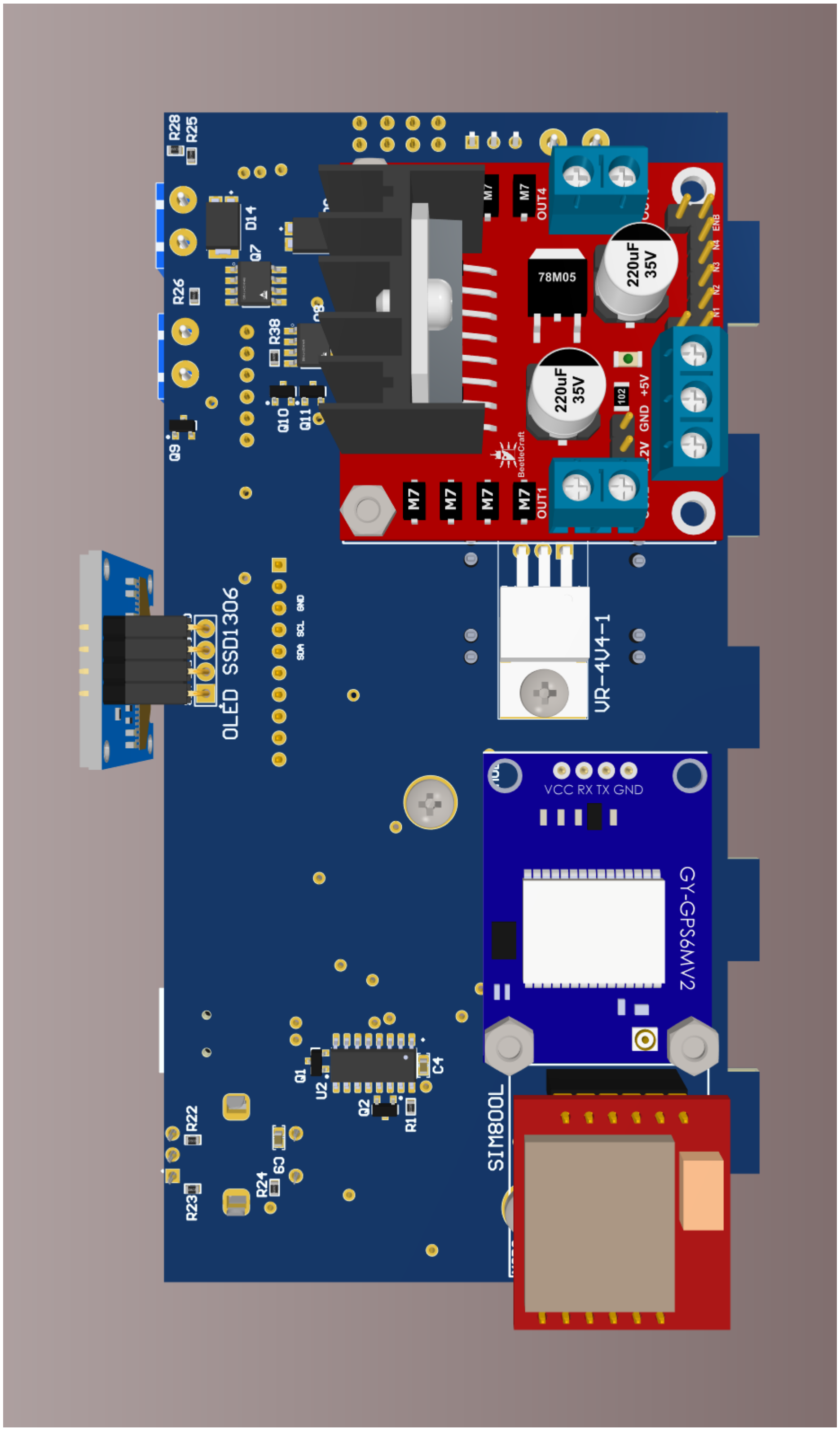


Figure F.9 – *Control PCB - 3D bottom view*

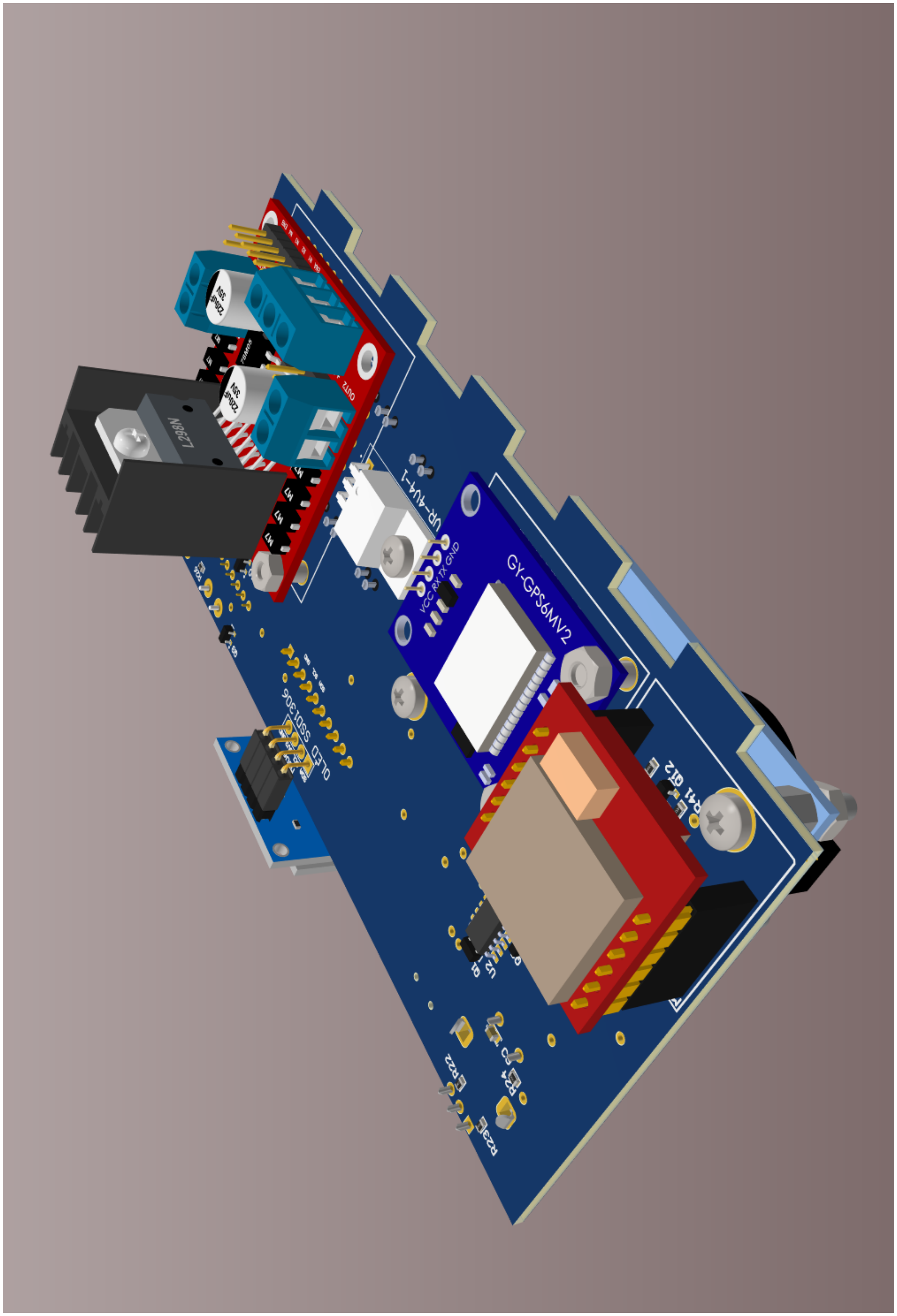


Figure F.10 – *Control PCB - 3D bottom view - 2*

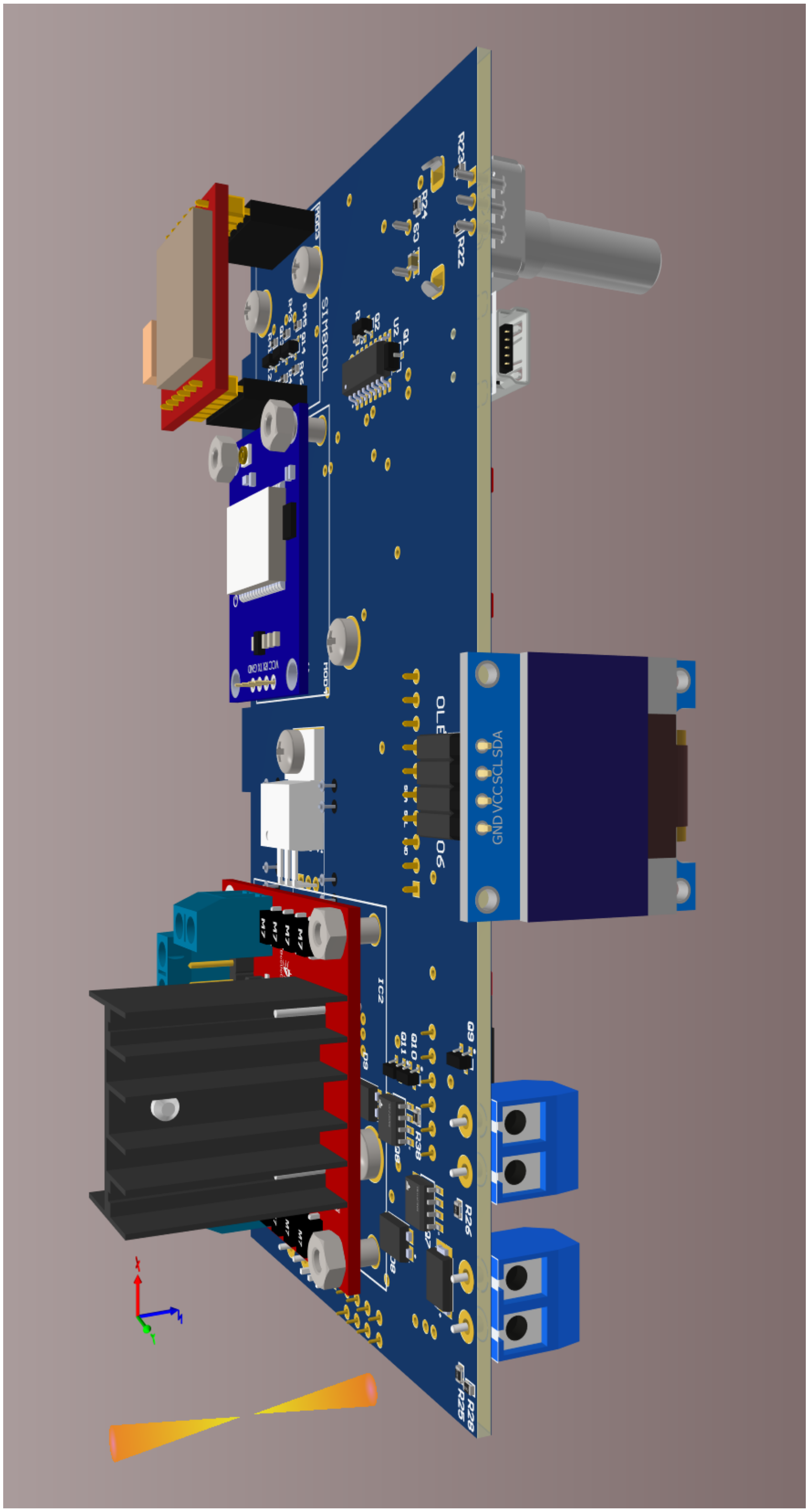


Figure F.11 – *Control PCB - 3D bottom view - 3*