

# ÀLIGA-20

Un submarí controlat via mòbil



<https://youtu.be/mQgLerRcSrc>

## **ABSTRACT**

My project is about the design and creation of a remotely operated submersible vehicle control unit. The unit allows this submarine to be controlled by a mobile phone using Bluetooth technologies combined with an onboard microprocessor.

The aim of the project is to see if it's possible to put my concepts into practice by creating a fully functioning prototype. In order to lay out my ideas and concepts, the project has been divided into three parts.

The first part explains all of the theoretical concepts regarding the control and operation of the unit and focuses on the programming of communication between the mobile phone and the unit's onboard microprocessor and also the programming of the microprocessor's physical control functions.

The second part explains step by step how to create and mount the submersible's circuit board and propulsion mechanisms and also explains in detail the code which is stored and executed on the microprocessor.

The final part explains how I was able to put my ideas and concepts into practice thanks to the use of a working handmade submarine and demonstrates how my prototype complies with the expectations and objectives which were set out from the beginning.

## **ABSTRACT**

Mi trabajo de investigación trata sobre el diseño y la creación de una unidad de control de vehículo sumergible operada a distancia. El proyecto permite que este sumergible sea controlado por un teléfono móvil, utilizando tecnologías Bluetooth combinadas con un microprocesador integrado.

El objetivo del trabajo es ver si es posible poner en práctica mis conceptos creando un prototipo en pleno funcionamiento. Para exponer mis ideas y conceptos, el proyecto se ha dividido en tres partes.

La primera parte explica todos los conceptos teóricos sobre el control y funcionamiento de la unidad, se centra en la programación de la comunicación entre el teléfono móvil y el microprocesador integrado de la unidad y también la programación de las funciones de control físico del microprocesador.

La segunda parte explica paso a paso cómo crear y montar la placa de circuito del sumergible, los mecanismos de propulsión y también explica en detalle el código que se almacena y ejecuta en el microprocesador.

La parte final explica cómo pude poner en práctica mis ideas y conceptos gracias al uso de un submarino hecho a mano en funcionamiento y demuestra cómo mi prototipo cumple con las expectativas y objetivos que se plantearon desde el principio.

# ÍNDEX

## MEMÒRIA

1. INTRODUCCIÓ.....	2
1.1. Objectius.....	3
2. PART TEÒRICA.....	4
2.1. Components electrònics.....	4
2.1.1. Arduino UNO.....	4
2.1.2. Mòdul Bluetooth.....	6
2.1.3. Relé.....	7
2.1.4. Pont H.....	9
2.2. Conceptes bàsics de programació.....	12
2.2.1. Tipus de funcions.....	13
2.2.2. Creació de variables.....	14
3. PART PRÀCTICA.....	17
3.1. Hardware & Software.....	17
3.1.1. Procediment.....	18
3.1.2. Creació mitjançant pont H.....	31
3.2. Fabricació de l'estructura.....	33
3.2.1. Introducció.....	34
3.2.2. Procediment.....	34
3.3. Costos del projecte.....	42
4. CONCLUSIONS.....	48
5. WEBGRAFIES.....	52

## ANNEXOS

I. Circuit fet en digital.....	57
II. Programació del circuit format per relés .....	58
III. Programació del circuit format per ponts H.....	61
IV. Fotos del procés.....	63

## PLÀNOLS

# 1. INTRODUCCIÓ

Aquest treball tracta sobre com controlar un submarí utilitzant un dispositiu mòbil. Primerament el meu objectiu era crear un robot guia, el qual no es podia portar a la pràctica degut a la seva complexitat. És per això que mitjançant molta recerca i l'ajut del meu tutor vaig arribar a la conclusió que havia de canviar tant els meus objectius com el mateix treball i intentar començar amb unes metes més assequibles, és a dir, per començar s'han de realitzar les parts més bàsiques, i un cop són completades, el projecte pot anar ampliant i incrementant la seva complexitat.

El meu TDR s'estructura en dues parts, primerament, es troba la part teòrica, en què s'expliquen cada un dels components electrònics que seran emprats posteriorment i la manera en la que són utilitzats. Seguidament, s'observa un resum dels conceptes més bàsics relacionats amb la programació de l'Arduino. Per acabar, es defineix la creació de funcions i com s'elaboren de forma adient.

Tot seguit, hi ha la part pràctica, aquesta està repartida en un conjunt de passos. Els primers es centren en el fet de familiaritzar-te i conèixer la funcionalitat dels components electrònics; a mesura que s'avança, es va formant a poc a poc el que esdevindrà el circuit principal del futur submarí.

En relació amb la programació, s'ha de tenir en compte que durant els passos anteriors, de forma simultània, també s'aniran elaborant varis programes, encara que no serà fins al final, que s'acabarà de polir el programa definitiu, el qual quedarà emmagatzemat a l'Arduino durant la utilització del submarí.

Per últim, es troba, de forma pautada, la creació de l'estructura del submarí, la qual s'encarregarà de la protecció de tot el muntatge.

Finalment, podem observar tota la part econòmica, englobant així els costos del projecte per complet i tot el material necessari, la representació del programa final i els plànols del submarí.

## 1.1. OBJECTIUS

Els diferents objectius que m'he plantejat a l'hora de crear el meu treball es poden resumir en els punts següents:

### Objectius principals

- Realitzar la construcció d'un prototip que funcioni correctament, mitjançant la creació de pautes bàsiques, aquestes es van creant i tornant més complexes a mesura que les anteriors van sent assolides.
- Que a partir del meu treball de recerca, qualsevol altra persona pugui seguir desenvolupant tant el circuit com el programa que he utilitzat, per tal d'augmentar el gran ventall d'accions que es poden arribar a dur a terme, ja que tots aquests tipus de moviments es basen a fer girar un motor, sigui completa o parcialment, o donar electricitat a un circuit. Aquests fets se centren a controlar un relé o un pont H, de la manera i en el moment que vulguem.

### Objectius parcials

- Reciclar diferents tipus de materials per poder elaborar l'estructura del submarí.
- Poder augmentar els meus coneixements de programació i familiaritzar-me amb la manipulació de diferents circuits elèctrics i de microcontroladors com l'Arduino Uno.

## **2. PART TEÒRICA**

### **2.1. COMPONENTS ELECTRÒNICS**

Els components electrònics són els dispositius que fan possible la creació de diferents tipus de circuits electrònics, a causa que cada un s'utilitza per a una funció en particular.

És per això que fent passar electricitat per aquests, hi ha milers de projectes que es poden arribar a crear i desenvolupar. Els que jo he utilitzat per a la creació del meu projecte són:

- Primerament, un microcontrolador anomenat Arduino Uno. Emmagatzema tota la informació i és el cervell del submarí.
- Tot seguit, el mòdul Bluetooth és el pont de comunicació entre el dispositiu mòbil i el submarí.
- I finalment, els relés i els ponts H s'encarreguen de controlar el pas d'electricitat cap als motors i de variar-los el sentit de gir.

#### **2.1.1. ARDUINO UNO**

L'Arduino UNO és un tipus de microcontrolador, el qual pot ser programat per tal que les accions que volem realitzar, siguin enviades cap als diferents components electrònics que utilitzarem. Aquesta placa està formada per 14 pins digitals (0-13) i 6 d'analògics. Aquests pins digitals són els que, posteriorment seran utilitzats, ja que els podem programar de tal manera que treguin 5 V cada un. Aquests 5 V podrien ser utilitzats per fer funcionar els motors de forma que quan el pin fos activat el motor aniria a 5 V, però, mitjançant aquest procediment, la tensió per la qual els motors són alimentats, estaria limitada. És per això que aquests volts només els farem servir com a senyal i no com a font d'energia. Per aquest projecte es poden utilitzar altres plaques Arduino i microcontroladors d'altres tipus, però l'Arduino UNO és un dels més complets, amb un preu molt assequible, i posseeix moltes sortides, qualitat molt útil tenint en compte que per cada motor haurem d'utilitzar 2 pins.

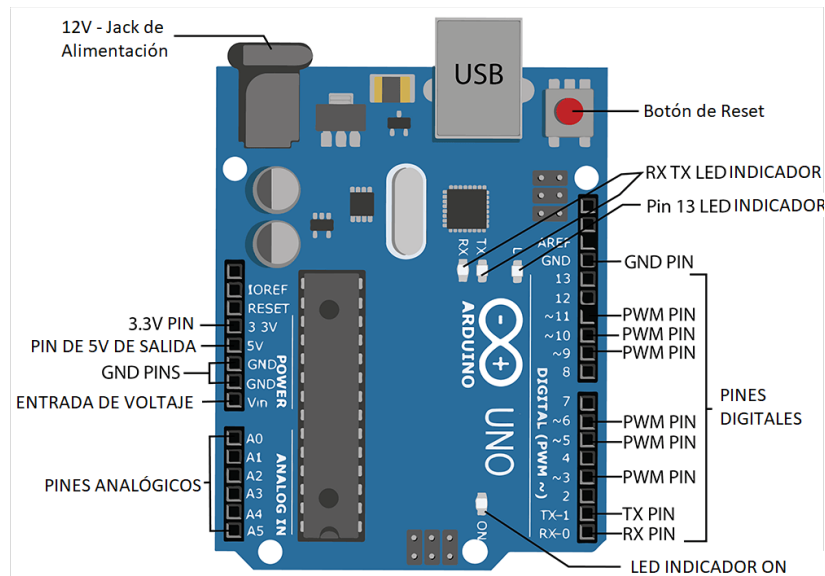


Una de les altres qualitats que té, a part de la seva memòria capaç d'emmagatzemar el codi pujat des de l'ordinador i després aplicar-lo, és el fet que hi hagi més d'una manera d'alimentar-lo. Hi ha una entrada de 5 V i una altra que té la capacitat d'admetre tensions diferents, el rang que accepta és des de 7 V fins als 12 V, cosa que aprofitarem degut a que la bateria que utilitzarem serà d'aquesta mateixa tensió. Amb l'entrada de 5 V ja és operatiu tot i que la recomanada és la segona (7 V-12 V). Aquest Arduino també serà utilitzat per alimentar tant els relés que utilitzarem, com el mòdul Bluetooth amb els 5 V que poden ser extrets.

Una altra part de l'Arduino que cal destacar són les sortides RX i TX, les quals ocupen els pins digitals 0 i 1 respectivament. Aquests dos ports sèrie tenen la capacitat d'enviar i rebre informació. (El RX és el pin d'entrada i el TX el pin de sortida.)

En relació amb la programació cal utilitzar un llenguatge de C++ i per tant, hem d'utilitzar el programa específic que proporciona Arduino, aquest pot ser descarregat de la xarxa i ens servirà per poder reprogramar la nostra placa a voluntat.





Imatge 1: Parts de l'Arduino UNO. Font: <https://www.pinterest.es/pin/754071531350592322/>  
(13/08/20)

### 2.1.2. MÒDUL BLUETOOTH

El mòdul Bluetooth que he utilitzat és el HC-06. Una de les seves característiques principals és un LED que es troba encès sempre que estigui connectant dos dispositius correctament, parpellejant sempre i quan estigui preparat per enllaçar-se. Per tal de poder-lo utilitzar hem de connectar els quatre pins d'entrada que són presents:

**V** = El primer pin és el que ha de ser utilitzat per alimentar al nostre dispositiu, en un principi aquest tipus de mòdul accepta un interval des de 3,3 V fins als 6 V, permetent-nos alimentar-lo amb els 5 V que l'Arduino és capaç d'extreure.

**GND** = El segon pin és el negatiu i ha d'estar obligatòriament connectat a la mateixa font d'energia d'on traguem el voltatge, en el nostre cas, l'Arduino. En cas que no ho féssim, el circuit elèctric no estaria complet, cosa que no ens permetria obtenir un correcte funcionament del dispositiu.

**RX i TX** = Els dos últims dos pins que hi ha s'encarreguen de l'intercanvi de dades. Hem de connectar el TX (emissor) del mòdul amb el RX (receptor) de la nostra placa

i a l'inrevés, per tal de poder completar un circuit on les dades poden viatjar de forma bidireccional.

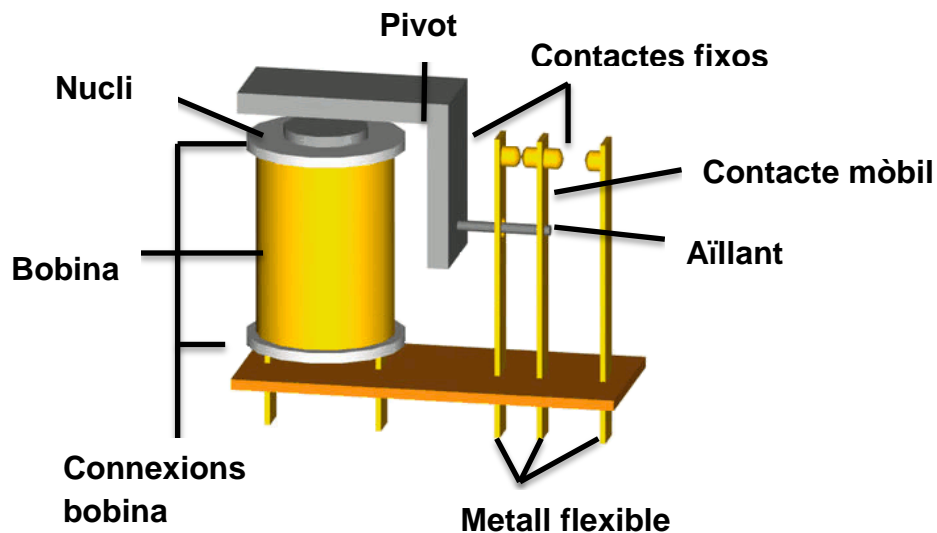
El fet de connectar aquestes dues entrades al nostre microcontrolador provoca la inhabilitat de poder carregar nou software a la placa, ja que l'entrada i sortida d'informació ja és ocupada. Aquest petit entrebanc té una fàcil solució, la qual consisteix a desconnectar el nostre mòdul mentre estiguem carregant nous programes. I un cop aquests són pujats, el tornem a connectar.

Un cop hem configurat el nostre dispositiu, és possible que ens demani un codi, el qual és (1234). Aquest codi està predeterminat i la seva variació és possible per tal de protegir, en aquest cas, al nostre submarí de l'accés públic al seu control.



### **2.1.3. RELÉ**

Un relé és un component electrònic que, mitjançant l'ús de l'energia elèctrica i de la força magnètica, produeix un moviment mecànic. Aquest dispositiu electromagnètic està format per la bobina i l'electroimant, dos contactes fixos als extrems i un de mòbil, el qual es mou quan es crea un camp elèctric al voltant de la bobina. Gràcies a aquests contactes podem obtenir una sortida del relé que sempre tanqui el circuit (quan el relé està apagat) i una ho faci només quan el contacte mòbil es desplaça (quan el relé és accionat).



Els responsables del funcionament del relé, a part de les peces que el formen, són els sis pins que han de ser connectats, tres d'entrada i tres de sortida. Els dos primers pins que trobem són el positiu i el negatiu, i els connectarem, juntament amb el mòdul Bluetooth, als 5 V de l'Arduino i al GND d'aquest. L'electricitat que circula per aquests dos pins és la que causa que el contacte mòbil sigui desplaçat i el relé passi d'utilitzar la sortida normalment tancada a la normalment oberta.

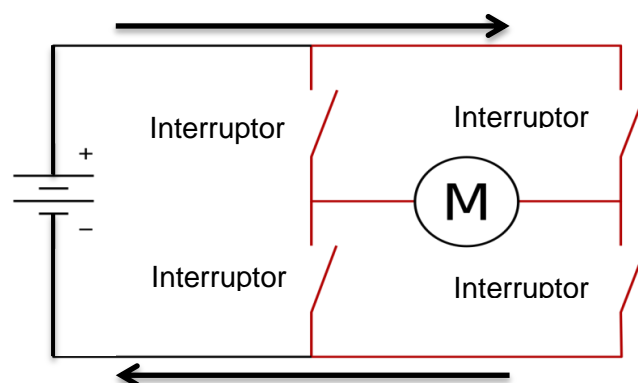
L'últim pin d'entrada que tenim és el del senyal, aquest ha de ser connectat a una de les sortides digitals de la nostra placa, ja que serà la que controli l'accionament del relé. És a dir, sempre que el pin digital envii 5 V (com a senyal) el relé s'accionarà. Per tant, es podria dir que aquest pin és el que fa d'interruptor electrònic i evita que hàgim de canviar la sortida del circuit de forma manual.

Amb relació al circuit que serà controlat pel relé hi ha tres pins que hem de tenir en compte, el COM (comú), el qual sempre serà part del circuit, i tot seguit el NO (normalment obert), i finalment el NC (normalment tancat). Aquestes tres sortides poden formar part de circuits que posseeixin un voltatge molt elevat, a diferència dels tres d'entrada que han de funcionar amb els cinc volts de l'Arduino. Un relé per les sortides NC, NO, i COM pot aguantar una intensitat de 10 Ampers i d'una tensió màxima de 250 Volts.



#### 2.1.4. PONT H

Un “H-bridge” (pont H), és un tipus de circuit que s'utilitza per poder canviar la polaritat d'un motor elèctric DC, és a dir, per aconseguir variar el sentit del seu gir.



Per arribar a aquest objectiu necessitem combinar quatre interruptors, de tal manera que sempre connectarem dos d'aquests provocant el tancament del circuit. Per tant per fer funcionar el motor cap a un cantó hauríem de connectar l'interruptor 1 i el 4, i per fer l'invers, el 2 i el 3. Aquests són les úniques dues combinacions que podem dur a terme, a part de deixar tots els interruptors oberts, de tal forma que el motor no es mouria.

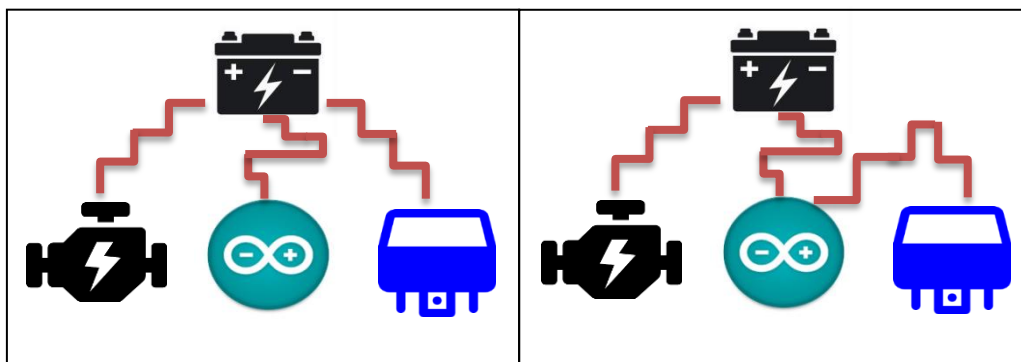
En cas de tancar tots els interruptors o connectar els dos parells o els dos senars, produiríem un curtcircuit, ja que l'electricitat no passaria pel motor i els electrons d'un pol anirien cap a l'altre.

Per evitar la utilització d'interruptors mecànics podem utilitzar dos relés per cada un dels motors que volem controlar, ja que el relé té dues sortides diferents, i el podem

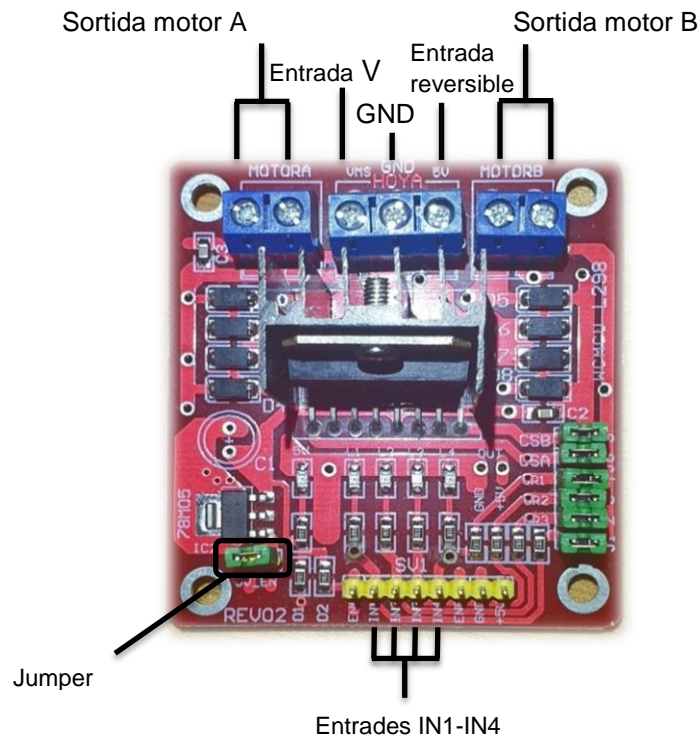
controlar de forma digital. Un relé contindria l'interruptor 1 i 2 i l'altre el 3 i el 4. Per exemple, el NO seria l'interruptor 1 i el NC seria el 2, i del segon relé el NO seria l'interruptor 4 i el NC el 3.

Sempre hem de vigilar el fet que si utilitzem aquest tipus de circuit, per tal que els motors estiguessin parats sempre hi ha d'haver un relé connectat, i sempre que el controlador que s'utilitzi (Arduino, Raspberry...) estigui apagat, els relés també ho estaran, això vol dir que el motor utilitzat seguirà en funcionament.

Per evitar això el que podem fer és utilitzar la mateixa font per alimentar tant el controlador, com els motors, com els relés. Per tant sempre que els motors tinguin electricitat per funcionar, el microcontrolador també en tindrà i a la vegada els relés també permetent d'aquesta forma l'aturada dels motors, i si cap d'aquests té corrent, els motors romandran aturats. Si alimentem els relés a través de la nostra placa Arduino i aquesta utilitza la mateixa font d'obtenció d'electricitat que els motors, aquesta solució segueix sent vàlida.



Una altra forma de fer servir aquest tipus de circuit capaç de canviar la polaritat dels motors és mitjançant uns components electrònics els quals també reben el nom de H bridge, ja que aquesta és la seva única funció. Si es fan servir, et pots estalviar el muntatge, que implica la combinació de dos relés i per tant reduir l'espai necessari que el circuit necessita i reduir tant costos econòmics, com temps emprat.



Aquests components poden controlar entre un i dos motors a la vegada, i per fer-los funcionar hem de connectar, per començar, el voltatge d'entrada i el GND. Si aquest voltatge es troba entre 5 i 12 volts, el mòdul funcionarà amb normalitat i la tercera entrada que tenim la podem fer servir com a sortida de 5 V. En canvi, en cas que volguéssim alimentar els nostres motors amb una tensió superior a aquests 12 volts, hauríem de retirar el "jumper regulador" del mòdul. Degut a l'alta tensió utilitzada, l'entrada del voltatge aniria directament a proporcionar energia als motors en comptes de desviar una part cap al mòdul com passava anteriorment, i per tant hauríem d'alimentar al nostre pont H amb 5 V mitjançant la tercera entrada que tenim.

Com en aquest cas el pont H és doble, tenim dos parells de sortides, el parell A i el parell B. Del primer parell de sortides, una de les dues anirà cap a un pol del motor i l'altra sortida cap a l'altre, i el mateix amb l'altre parell. Un cop hem fet això el circuit dirigit al motor està complet, només falta connectar els pins digitals de l'Arduino cap als quatre pins presents al mòdul H bridge. Els pins 1 i 2 del pont H controlen el motor A i els pins 3 i 4 el motor B. Per poder manejar a voluntat el motor A, hi ha tres possibles combinacions que podem fer amb els pins 1 i 2.

En primera posició tindríem la possibilitat de tenir el motor aturat, i això es faria amb el simple fet de no enviar cap senyal cap als pins 1 i 2.

La següent opció que podríem tenir seria enviar corrent cap al pin 1, fent així que el motor girés en un sentit. Finalment si el corrent fos enviada al pin 2, en comptes de pel pin 1, aconseguiríem l'altre sentit de gir.

Una última opció seria fer córrer el corrent pels dos pins a la vegada, cosa que produiria un curtcircuit, ja que la tensió de la bateria que alimenta als motors passaria pels dos pols d'aquest. Si això es fes amb altes tensions, ja que per un H bridge poden circular fins a 35 V, podria arribar a ser un fet perillós i pot arribar a produir un escalfament del cablejat fins al punt de produir una flama, i en un cas extrem, un incendi.

## 2.2. CONCEPTES BÀSICS DE PROGRAMACIÓ

Abans de poder començar la programació del nostre submarí, hi ha varis conceptes que hem de conèixer. També hem de saber que el programa que fem quan utilitzem els relés serà diferent del que escriurem amb la utilització del pont H mitjançant el mòdul L298N. Els conceptes que seran properament explicats són únicament vàlids pel programa d'Arduino que està amb llenguatge de C++.

El programa es divideix en tres grans blocs. Primerament, en l'obrir una nova pàgina per programar, ens apareix les funcions “`void setup`” i “`void loop`”. Tot el que sigui escrit per sobre d'aquestes dues funcions serà tota la informació que l'Arduino necessita saber per poder complir les posteriors indicacions. En cas d'escriure alguna cosa després del “`void loop`”, aquesta, no seria llegida i per tant, és possible que l'Arduino no pugui complir totes les ordres que li són assignades.

El següent gran bloc és tot l'interval que es troba dintre de la funció “`void setup`”. Tot l'escrit en el seu interior serà el que l'Arduino executarà just quan s'encengui. Un exemple seria fer que els motors es posin en posició d'atur en cas que s'estiguin movent.

Finalment tenim l'últim bloc, el qual es troba dintre la funció “**void loop**”. Tot el codi que estigui escrit en el seu interior és el que serà constantment repetit, és per això que aquest bloc ha d'estar situat a la part més baixa del programa. En aquest apartat, un cop l'Arduino hi arriba es manté en bucle i és per això que aquí hem de col·locar el que l'Arduino ha d'estar contínuament comprovant, és per això que aquí és on escriurem tot el conjunt de condicionals.

```
/*Tot l'escrit en aquest espai és la informació  
necessària que posteriorment serà utilitzada */  
  
void setup() {  
  
    /*El que conté aquest bloc és el que s'inicia  
    tant bon punt l'Arduino rep corrent*/  
  
}  
  
void loop() {  
  
    /*Finalment aquí es col·loca tot el que ha de  
    ser constantment repetit*/  
  
}
```

### 2.2.1 TIPUS DE FUNCIONS

Per començar, parlarem de les funcions que es troben ja definides dintre el programa d'Arduino.

Primerament, hi ha les que s'encarreguen de definir el tipus de dades que seran enviades. Aquests són gairebé imprescindibles, ja que en cas de no especificar com o què és la informació l'Arduino no podrà processar el que li arriba i el nostre programa no funcionarà.

Nosaltres hem utilitzat la funció “**int**” la qual especifica que les dades del que fiquem just després són nombres sencers, és a dir, no són ni lletres ni nombres amb decimals.



```
int valorDelBoto;    /*int S'utilitza per declarar que les dades
                     que arriben són nombres sencers/*/
```

Hi ha altres tipus de variables que poden ser utilitzades segons ens convingui, les més conegudes són “byte, int, long, float, boolean i char”. Cada un d'aquests ha de ser utilitzat per una ocasió diferent. Els quatre primers, és a dir, des del “byte” fins al “long”, són molt semblants, l'únic que canvia és el rang de nombres que accepten, per exemple el byte, el qual està codificat en 8 bits, només accepta nombres des del zero, fins al 255. I en canvi, el “long”, accepta un nombre que es troba entre el  $\pm 2.147.483.647$ , però està codificat en 32 bits i per tant ocupa més espai en el nostre programa, i per tant el nostre programa serà més pesat.

La següent variable que tenim és el “float”, i a part d'acceptar un interval superior a l'anterior, admet l'intercanvi de dades amb nombres que posseeixen decimals.

El “boolean” tracta d'un tipus de variable que només comprèn dues possibilitats, verdader (true) o fals (false). Aquest tipus de variable pot ser utilitzada per diversos tipus de programes, un fàcil exemple el qual es pot utilitzar o per entendre la seva utilització és mitjançant un programa d'exàmens o tests. Aquest estaria codificat per tal que, si hi ha varies opcions i només hi ha una de certa, en cas de clicar la correcta doncs rebríem un “true”, i en qualsevol altre cas rebríem un “false”.

Finalment, tenim la variable la qual ens pot servir per arribar a emmagatzemar lletres, si més no, mitjançant un procés que pot resultar una mica complicat. Aquesta complicació es deu al fet que cada caràcter té una representació numèrica o per exemple el nombre 97 és la lletra **a**.

### 2.2.2. CREACIÓ DE VARIABLES

A causa de la gran quantitat d'ordres que s'han de dur a terme cada cop que l'Arduino rep un nombre, el bloc on es troben les condicions es veu molt ampli i en cas de voler realitzar un canvi posteriorment, seria molt complicat. Per tal de

solucionar el nostre problema d'organització ho hem de fer mitjançant a creació de noves variables. El fet de tenir un programa ben organitzat no sembla un fet primordial a primera vista, però un cop el codi es comença a estendre i/o s'ha de fer un canvi després d'haver transcorregut molt de temps, en cas de no tenir una bona organització el fet de buscar on s'ha de fer el canvi o el d'haver de recordar-se de cada una de les línies escrites, pot causar severos problemes de funcionament i inclús haver de refer el programa o parcialment o per complet.

Aquestes variables han de ser col·locades abans del “void setup” i del “void loop”, ja que, com he dit anteriorment, tota la informació que l'Arduino necessitarà durant el bloc del “void loop” s'ha de trobar abans d'aquest i fora del “void setup”, per tant, a principi del codi.

Per poder crear aquestes variables el que hem de fer és ficar “void” i a continuació, el nom de la variable la qual volem crear. Per exemple, en cas de voler fer una variable per avançar/ tirar cap a davant, ho fariem de la següent manera:

```
void davant()
```

I tot seguit col·locar dintre les claus ({ }), tot el conjunt d'ordres que volem que es duguin a terme.

```
void davant() {  
    digitalWrite(8, LOW);  
    digitalWrite(9, LOW);  
    digitalWrite(10, LOW);  
    digitalWrite(11, LOW);  
}
```

El fet d'utilitzar un nom com **davant** ens ajuda a saber que les ordres escrites just després són les que necessitem per tal que el nostre submarí avanci.

Finalment hem de col·locar el nom de la variable cada cop que volem que s'utilitzin les ordres que li corresponen, i això seria durant el bloc dels condicionals.

```
if(Serial.read==X) { davant(); }
```

En aquest petit fragment de codi hem escrit que: en cas que el “**Serial.read**” (nombre que rep l’Arduino) sigui igual al número que correspon a la tecla d’avançar, dugui a terme la variable ***davant***.

### 3. ELABORACIÓ DEL PROJECTE

#### 3.1. HARDWARE & SOFTWARE

El primer que vaig realitzar va ser pensar com hauria de fer el meu futur submarí. Aquest aniria dirigit des del mòbil, via Bluetooth fins a l'Arduino. A continuació vaig decidir que la millor manera de fer-ho era amb el dispositiu HC-06. Aquest mòdul Bluetooth permet enviar i rebre informació des de l'Arduino, fins a un dispositiu que estigui connectat a aquest mòdul i a l'inrevés. Els components electrònics Relés van ser el següent que vaig triar. Gràcies a aquests podia controlar el pas d'electricitat des d'unes bateries fins als motors amb la utilització dels pocs volts que surten de l'Arduino. La major part de les peces les vaig comprar per Amazon i llocs similars i la resta, en una botiga d'electrònica.

Abans de poder començar tot el procés també vaig haver de descarregar l'aplicació oficial per poder programar l'Arduino.

El material necessari per poder realitzar la primera part del projecte:

- Bateria 12 V (encara que per fer proves no cal tant de voltatge)
- Arduino UNO
- Sis relés (hi ha diversos models, els indicats són els que posseeixen 6 pins, tres d'entrada i tres de sortida, com els anomenats anteriorment)
- Diferents tipus de cables (mascle/ mascle, mascle/ femella, femella/ femella, aquests s'utilitzaran depenent el que ens convingui)
- Quatre motors
- LED i unes resistències
- “Breadboard” per tal de poder unir varies components tots junts i per poder fer proves amb els LED

### 3.1.1. PROCEDIMENT

#### Pas 1

Un cop tenim totes les peces comprades hem de començar a fer proves amb el material i comprovar el seu correcte funcionament. Aquestes es poden basar a connectar corrent cap al relé per tal d'articular la seva part mòbil, enviar electricitat cap als motors o comprovar que els LED tant de l'Arduino com del mòdul Bluetooth s'encenguin quan són connectats a una font d'alimentació o a l'ordinador.

A causa que hem d'utilitzar quatre motors per poder realitzar aquest treball, haurem de mirar que no n'hi hagi cap de defectuós o amb un moviment inusual. D'aquests quatre, dos estan situats als laterals i l'altre parell sota el submarí.

I finalment, amb relació als relés necessitarem comprovar el funcionament dels dos que controlaran els relés de la part inferior del submarí, i els quatre que regularan motors laterals, degut a que cada un d'aquests dos ha de ser monitorat de forma diferent.

#### Pas 2

Inicialment, hem de connectar el dispositiu Bluetooth amb la placa Arduino. En comptes de fer-ho directament, enviarem tant els 5 V com el GND que es poden extreure del microcontrolador, cap a una "Breadboard", d'aquesta forma, posteriorment podem utilitzar aquests 5 V de tensió per altres components electrònics, sempre que els connectem en paral·lel.

El mòdul Bluetooth té quatre sortides, les dues primeres són el TX i el RX, les quals hem d'unir amb els pins 0 i 1 de l'Arduino respectivament, és a dir, el TX del mòdul amb el RX del Arduino i el RX del primer amb el TX del segon.

Els altres dos pins del mòdul són el positiu, on hem d'enviar els 5 V, i el negatiu, el qual ha de ser ajuntat amb el GND. Cada un dels pins d'aquest dispositiu té el seu nom de forma que sabem quin és quin.

### Pas 3

En relació amb l'aplicació que utilitzarem, hi ha moltes diferents que ens poden ser útils, ja que totes tenen la mateixa finalitat, i aquesta és intercanviar ordres i dades entre el telèfon mòbil i aquest mòdul, tot i que hem de vigilar, ja que la que utilitzarem també ha de ser compatible amb la partició de pantalla que ofereixen els mòbils de marca Samsung, entre d'altres. Per poder mirar que és el que passa quan es prenen els botons de l'aplicació s'ha d'utilitzar un petit programa molt simple que el que fa és dir-li a l'Arduino que envii la informació que rep, cap al monitor serial. Per fer això s'ha de tenir certes precaucions, ja que, si no les tenim en compte, ens hi podem passar molt de temps intentant que funcioni sense aconseguir-ho. Aquesta precaució es basa en el fet que, si el mòdul Bluetooth, el qual ocupa els pins RXD i TXD, està encès, no es poden carregar programes a l'Arduino, ja que aquests dos pins són emprats per enviar i rebre informació, i en cas de ser ocupats, l'Arduino tria com a preferència les dades del Bluetooth en comptes de les de l'ordinador. És per això que, abans de poder carregar qualsevol programa, hem de tenir en compte que el dispositiu de Bluetooth estigui apagat o desconnectat d'aquests dos pins.



Una de les aplicacions que es poden utilitzar

### Pas 4

El programa que hem de fer per poder veure la informació que rep l'Arduino, del dispositiu Bluetooth, tracta sobre dir-li a l'Arduino que obri el port serial i envii les dades a una velocitat determinada. Per poder fer això utilitzem la funció **“Serial.begin”**, aquesta la posarem dintre del **“void setup”**, i li direm que envii les dades a 9600 bits per segon. Tot seguit, ficarem dues altres funcions dintre del **“void loop”** per tal que es repeteixin tota l'estona, la **“Serial.println”** i la funció **“Serial.read”**,

aquesta última la col·locarem en l'interior de la primera per poder aconseguir que les dades que hi ha en la segona funció es puguin veure en el monitor serial. Aquestes dades que podrem veure seran les que l'Arduino va rebent del mòdul Bluetooth.

```
void setup() {Serial.begin(9600);  
}  
/*Aquesta és la part que serveix per determinar  
la velocitat en que l'Arduino fa el intercanvi de  
dades*/  
  
void loop() {Serial.println(Serial.read());  
}  
/*Aquest tros del programa és el que estarà  
constantment repetint-se i que enviarà les dades  
de l'Arduino al monitor serial*/
```

### Pas 5

Un cop tinguem aquest programa fet i carregat a l'Arduino, hem d'entrar al Monitor Serie, el qual està situat a l'apartat de "Herramientas", per poder configurar l'aplicació emprada amb el programa que farem, ja que depenent l'aplicació, els nombres enviats cap a l'Arduino, quan els botons són presos, poden variar. També hem de tenir en compte que els botons de l'aplicació triada han d'enviar un senyal quan deixin d'estar presos, i és per això que s'ha de vigilar quina s'escull. Un cop es pot veure què succeeix quan es prenen els botons, es pot apreciar que quan un botó és pressionat, l'Arduino rep un nombre en concret i quan aquest botó es deixa anar, n'envia un altre. Aquestes dades han de ser guardades per utilitzar-les posteriorment.

```
13:55:02.381 -> -1  
13:55:02.381 -> -1  
13:55:02.381 -> -1  
13:55:02.381 -> 90  
13:55:02.381 -> -1  
13:55:02.381 -> -1  
13:55:02.381 -> -1
```

La primera part d'aquesta imatge, és a dir, els nombres de l'esquerra són l'hora exacte en què l'Arduino rep un senyal. A la part dreta de la fletxa, s'aprecien una sèrie de nombres, els quals representen la comunicació entre el telèfon mòbil i l'Arduino. Quan no hi ha intercanvi d'informació es rep el -1, però en el moment en què prenem un dels botons de l'aplicació, aquest nombre s'altera, en aquest cas, amb el número 90, a les 13 h 55 min i 2,3 segons. D'aquesta forma podem saber quins nombres necessitem utilitzar, i quins van relacionats amb cada botó.

### Pas 6

A continuació, el que hem de fer és començar el nostre futur programa principal. Abans de tot, hem de tornar a utilitzar la funció **Serial.begin** i col·locar-la dintre el **void setup**, ja que hem de seguir especificant la velocitat en què l'Arduino fa l'intercanvi de dades.

La primera part tindrà com a funció treure 5 V d'un pin determinat sempre que l'Arduino rebi el nombre que nosaltres volem. Per exemple, en el cas que volguéssim encendre un LED cada cop que prenem un botó, que és el nostre objectiu actualment, hauríem d'agregar les següents funcions. La primera és la **pinMode**, l'hem de col·locar dintre del **void setup** i la utilitzarem perquè l'Arduino envii els 5 V desitjats a un pin determinat, ficant el nombre del pin i la funció **OUTPUT** separats per una coma. Després ficarem un condicional dintre del **void loop** i l'utilitzarem per dir que cada cop que l'Arduino rebi el nombre que volem, faci una acció en concret. Aquesta funció es representa a través de **if** i en el seu interior hi col·loquem el **Serial.read**.

Aquest ha d'anar igualat al nombre que rebem cada cop que premem el botó d'encendre. Aquest nombre el sabem gràcies al pas anterior, on podíem veure quin nombre rebia l'Arduino, cada vegada que el botó d'encendre era pres.



```

void setup() {Serial.begin(9600);

  pinMode(10, OUTPUT);      /*/Aquesta funció serà emprada per tal
                             que el pin escollit, en aquest cas el
                             10, pugui treure un voltatge de 5 V/*/
}

void loop() {

  if (Serial.read()==X){    /*/ En aquesta part del programa és on
                             hem de substituir la incògnita X pel
                             nombre rebut quan prenem el botó. I un
                             cop el nombre rebut és igual al que hem
                             col·locat, fa la següent funció./*/

    digitalWrite(10,HIGH); /*/En aquest fragment és on decidim si
                             el pin 10 (en aquest exemple) envia els
                             5 V que necessitem o no. (En cas que no
                             volguéssim es posaria LOW en comptes de
                             HIGH)/*/

  }
}

```

---

### Pas 7

Un cop ja hem fet el pas anterior, hem de fer el mateix procediment per poder apagar el LED quan prenem un altre botó. Les úniques diferències que hi ha és que en el “**digitalWrite**” del següent condicional hem de posar “**LOW**”, per tal que el pin no envii cap senyal elèctric i ficar el nombre que l’Arduino rep, quan el botó d’apagar és pressionat. (Aquest nombre es pot saber mitjançant el pas 4 i 5).

```

void loop() {

  if (Serial.read()==X){    /*/Aquesta incògnita (X) l’hem de
                             substituir per el nombre que rebem quan
                             prenem el botó d’encendre/*/

    digitalWrite(10,HIGH)
  }

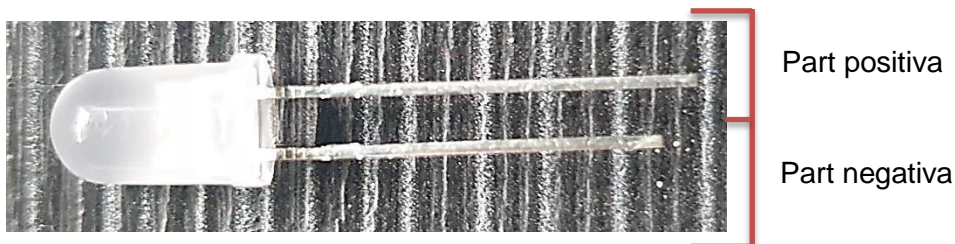
  if (Serial.read()==Y){    /*/ I en aquesta, ha de ser el valor que
                             obtenim mitjançant el botó d’apagar/*/

    digitalWrite(10,LOW)
  }
}

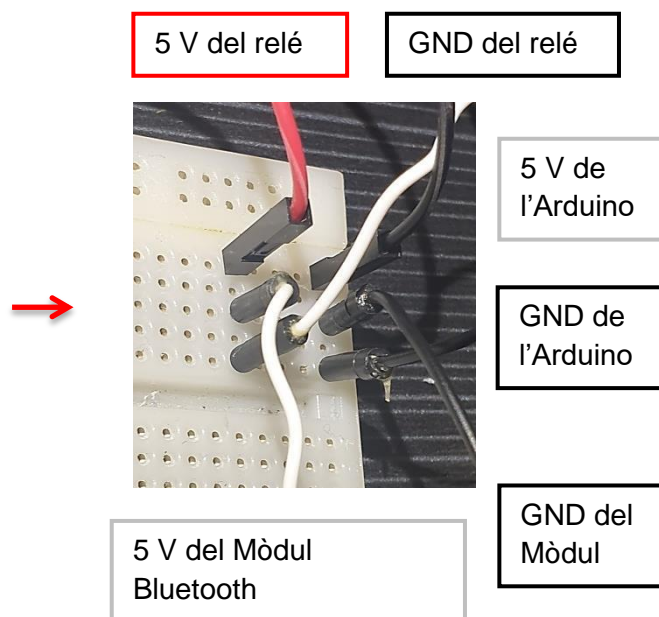
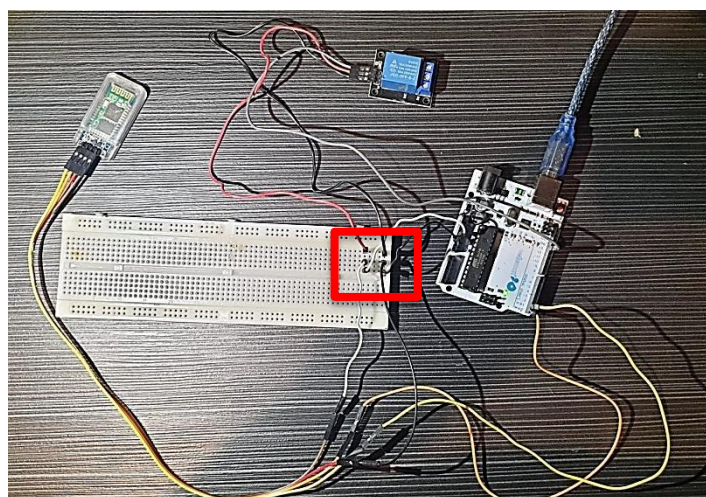
```

### Pas 8

Un cop ja tenim el programa preparat hem de connectar la sortida utilitzada amb la part positiva del LED, aquesta es reconeix fàcilment, ja que té una llargària superior a l'altre. A continuació unirem l'extrem restant amb el GND de l'Arduino.



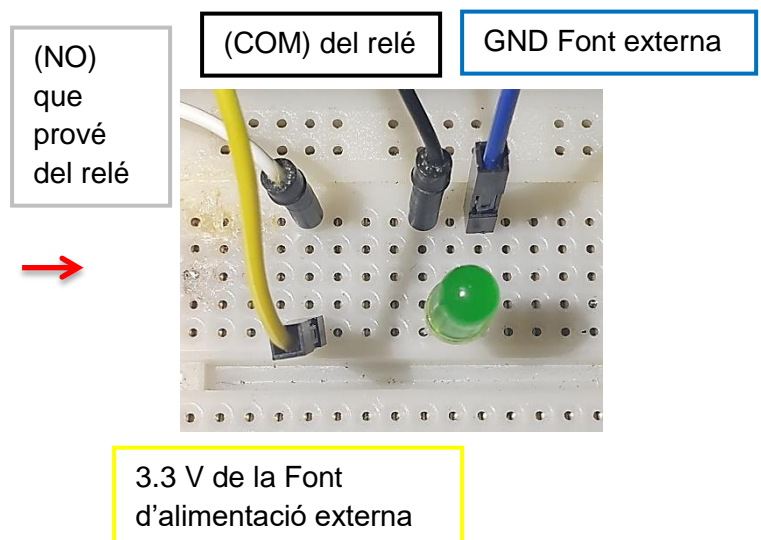
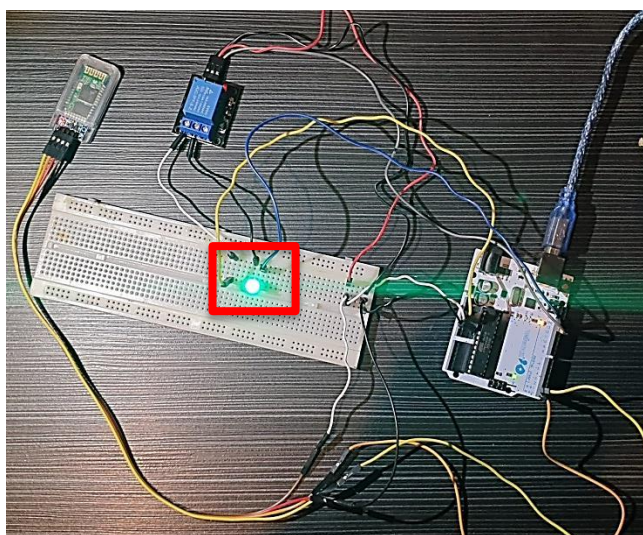
Quan ja podem controlar un LED de forma voluntària, ho provarem amb un relé. Els relés que utilitzem tenen tres entrades i tres sortides. Dues de les primeres tres entrades serveixen per alimentar al relé, el GND i els 5 V, els quals poden ser subministrats pel mateix Arduino. L'altra entrada s'utilitza per poder controlar el relé com ens interressi. Aquesta última entrada la connectarem a un pin de l'Arduino, per tal que, quan enviem un senyal elèctric des d'aquell mateix pin, el relé sigui accionat. I quedarien lliures les tres sortides restants, la del mig que és la comuna (COM), les dels extrems que són la NO (normalment oberta) i la NC (normalment tancada).



En aquesta primera imatge es pot veure a l'Arduino proporcionant 5 V tant al mòdul com al relé mitjançant la "Breadboard". També es pot apreciar la connexió dels pins TX i RX i del cable que va des del pin 10 fins al relé.

### Pas 9

Quan ja hem completat el pas anterior, prosseguim amb ajuntar un LED al relé connectat a l'Arduino. Utilitzem LED per tal que les proves que fem puguin ser verificades de manera visual i ràpida. Per tant, en comptes d'encendre el LED mitjançant els 5 V que provenien del pin de l'Arduino, el que farem és alimentar el LED amb una font d'energia externa, ja que així és com posteriorment alimentarem els motors. Una vegada totes les entrades del relé són ocupades, connectem un extrem del LED al comú. L'altre extrem anirà connectat a la font d'alimentació i finalment, connectarem el NO del relé amb l'altre punt restant de l'alimentació. (Depenen del LED també hem d'utilitzar una resistència). Connectem les sortides del relé de tal manera que, quan la part mòbil del seu interior sigui accionada mitjançant l'aplicació, el LED rebrà l'electricitat necessària com per il·luminar. (La font d'alimentació externa pot ser la sortida de 3,3 V que té l'Arduino amb un altre GND)



### Pas 10

Per avançar un pas més, el que hem de fer és canviar el LED per un motor i programar un fragment que tingui com a finalitat fer que el motor deixi de girar sense la necessitat d'utilitzar dos botons diferents. Això ho aconseguirem de forma molt senzilla, de la mateixa manera que hem aconseguit encendre un LED mitjançant el

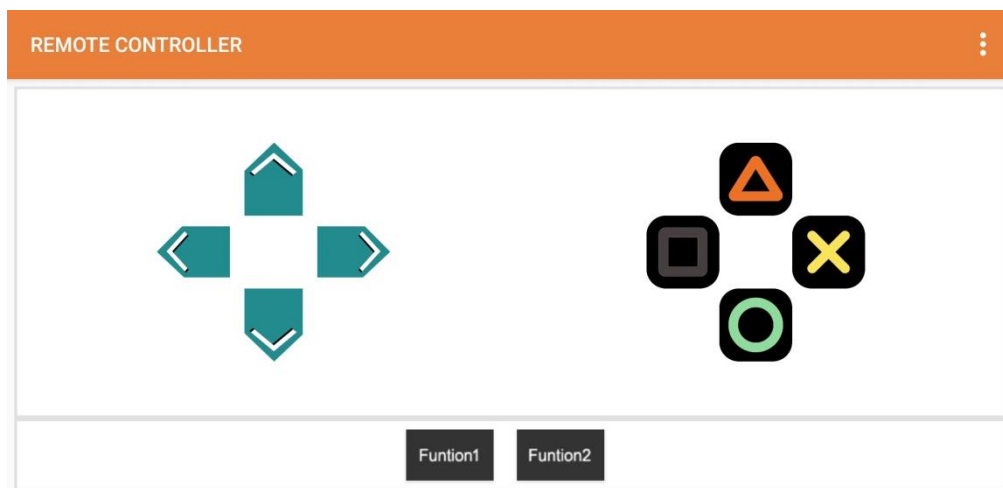
nombre que rebem quan un botó és pressionat, el que farem és fixar-nos en el nombre que és enviat cap a l'Arduino quan el botó es deixa de pressionar.

(Aquest nombre el podem aconseguir en el pas 4 i 5)

```
if (Serial.read()==90) {  
  
    digitalWrite(10,LOW);}  
}
```

En cas que el nombre que rebem sigui el 90, hauríem de col·locar-lo com una ordre més, és a dir, en un condicional i especificant l'ordre i el pin desitjats.

Degut a que el mateix nombre és emprat per mitja part dels controls i un altre per l'altre part, podem programar tots els botons de la part esquerra o tots els de la dreta amb una línia per cada cantó, per tant, mai podrem avançar i retrocedir a la vegada, cosa que evita un munt de problemes, però sense evitar l'habilitat d'avançar, retrocedir o girar a la mateixa vegada que pugem o baixem.



### Pas 11

Després de realitzar el pas anterior, procedirem amb el canvi de polaritat, mantenint el fet de fer que el motor pari de fer voltes quan cap botó és pressionat. Per tant, a continuació utilitzarem dos relés per un sol motor per a aconseguir el canvi de sentit en el gir. (Cada relé ha d'anar connectat en un pin diferent)

Aquest objectiu s'aconseguirà a través de comunicar la part NO d'un relé amb la NC de l'altre, i en aquest mateix punt i posarem un dels dos pols de la bateria que alimentarà al motor. L'altre pol de la bateria anirà connectat de la mateixa manera però amb els altres extrems dels relés a la inversa, és a dir, si del primer relé hem utilitzat el NO, ara utilitzarem el NC. Un cop tenim aquesta part del hardware muntada, ja hem ocupat totes les entrades i sortides excepte una de cada relé. Aquestes sortides que queden són les COM (la sortida comuna) de cada relé. I aniran connectades una a cada pol del motor. Un cop aquesta part del circuit està muntada, hem de repetir el procés tants cops com sigui necessari, que en el nostre cas és dues vegades més, per tal de controlar a voluntat les tres agrupacions de motors que tenim.

## Pas 12

A hores d'ara, necessitem una font d'alimentació per l'Arduino i una altra per als motors, ja que si utilitzem l'Arduino per obtenir energia, només podem arribar a extreure un màxim de 5 V. Aquesta electricitat seria insuficient tenint en compte tot el pes del submarí, s'avançaria a velocitats molt baixes, o potser ni s'avançaria. És per això que necessitem utilitzar una bateria que ens proporcioni un voltatge major.

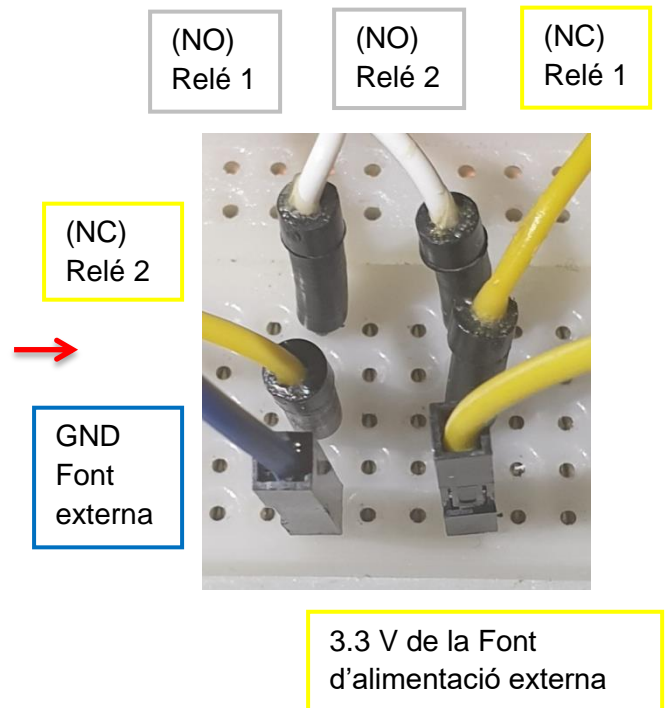
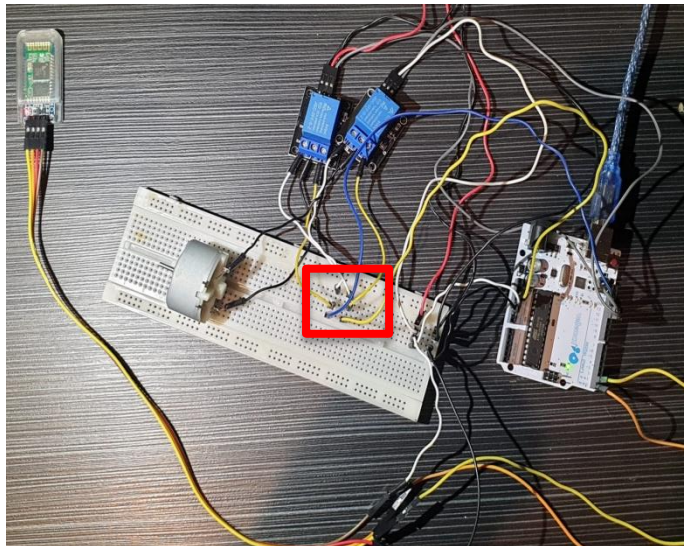
El problema d'afegir una nova font d'alimentació de tensió superior, és que l'Arduino només accepta 5 V, i necessitaríem dues bateries de diferent tensió, excepte en el cas que utilitzéssim la segona entrada que posseeix l'Arduino, la qual s'ha de fer servir amb una electricitat d'entre 7 i 12 volts.

Per tant, el que hem de fer és extreure l'electricitat d'una bateria de 12 V i fer-la córrer per una "Breadboard", tot seguit connectarem aquesta tant cap a la segona entrada de l'Arduino, com cap al circuit independent que es forma entre els motors i les sortides dels relés. D'aquesta manera, amb una sola font d'alimentació proporcionem energia als motors i l'Arduino, que a la vegada proporciona 5 V als relés i al mòdul Bluetooth.<sup>1</sup>

---

<sup>1</sup> El circuit final es troba fet en digital en el punt I. dels annexos.





### Pas 13

Per poder fer funcionar aquest nou muntatge electrònic hem d'adaptar el programa que teníem. Anteriorment, només havíem d'enviar 5 V amb un dels pins de sortida, pel fet que només utilitzàvem un sol relé, però com actualment n'utilitzem dos, hem de jugar amb aquests per tal d'enviar les ordres que volem. En el programa anterior, per fer girar el motor, havíem d'escriure “HIGH” al cantó del pin corresponent, el qual es troba dintre del “digitalWrite”. En canvi, actualment, per fer girar el motor hem de posar una altra “digitalWrite” que serà la que controlarà el relé afegit.

Un cop tinguem aquestes dues línies fetes hem de tenir en compte que cada cop que vulguem fer girar el motor hem d'encendre els dos relés o apagar-los tots dos, això dependrà del sentit que volem aconseguir, és a dir, sempre que escriguem “HIGH” a les dues línies girarà cap a un cantó i cada cop que escriguem “LOW” cap a l'altre.

I finalment, per parar els motors hem de ficar una de les línies en “HIGH” i l'altre en “LOW”.

#### Pas 14

Un cop hem repetit el pas 9 per tal que els nostres quatre motors puguin ser controlats, ja que n'utilitzem dos per avançar, retrocedir i girar, i dos més per tal que el submarí pugui ascendir i descendir, el hardware ja estaria preparat, llavors hem de continuar amb la programació. Degut a que hem d'utilitzar diferents vegades uns condicionals els quals són executats simultàniament, hem de fer servir un altre tipus de funció. En cas que ho féssim només amb la funció "if" de forma repetida, tindriem problemes degut a que, la utilització constant d'aquesta, provoca un mal funcionament del nostre programa. Aquest segon condicional és el "else if".

El procediment que hem de seguir per poder realitzar correctament l'ús d'aquest condicional és fent servir la funció "if", tot seguit utilitzar tants "else if" com siguin necessaris, i finalment acabar amb un "else". Aquesta última funció no s'ha de posar obligatòriament, però si més endavant es vol fer algun canvi, aquesta modificació no té tanta dificultat. La utilització del "else if" és la mateixa que la de la funció "if", és a dir, es col·loca la funció, seguidament, en el seu interior, has d'igualar el "Serial.read" amb el nombre que reps quan pitges el botó desitjat. Aquesta igualació es fa amb dos símbols d'igual.

```
        if (Serial.read==57) {                //pujar

        digitalWrite(12, HIGH);
        digitalWrite(13, HIGH);

    }

        else if (Serial.read==56) {          //baixar

        digitalWrite(12, LOW);
        digitalWrite(13, LOW);
    }
```

### Pas 15

A continuació hem de col·locar les funcions “`int Serial.read`” a la primera línia, per aclarir a l'Arduino el tipus de dades que s'estan intercanviant.

I finalment, el que hem de fer és col·locar una sèrie d'ordres dintre del “`setup`”. Aquestes consisteixen a fer que tots els motors es parin quan s'executa el programa, ja que en cas de perdre la connexió per alguna raó i recuperar-la poc després, els motors es detindrien en comptes de continuar girant.

```
digitalWrite(8, HIGH);  
digitalWrite(9, LOW);  
digitalWrite(10, HIGH);  
digitalWrite(11, LOW);  
digitalWrite(12, HIGH);  
digitalWrite(13, LOW);  
}
```

Aquest fragment  
serveix perquè els  
motors estiguin parats  
quan s'encén l'Arduino

### Pas 16

Tot i que el programa ja funciona correctament, hi ha una sèrie de millores que proporcionen una millor efectivitat i una major velocitat en l'intercanvi de dades entre el telèfon i l'Arduino. La primera que hem de fer és igualar, dintre del “`void loop`”, el “`Serial.read`”) amb una funció que tingui un nom de fàcil comprensió, com per exemple “`valorDelBoto`”. D'aquesta forma, evitariem un ús excessiu del “`Serial.read`”, que en alguns casos, pot arribar a causar lentitud i un mal funcionament del programa.

La segona és facilitar la comprensió cap a qualsevol lector, cosa que actualment, pot ser un pèl complicada o inclús cap a un mateix després d'un temps de desús, és per això que, el que hem de fer és igualar tot el conjunt d'indicacions que els relés han de seguir cada vegada que un botó és pres. Això es fa mitjançant la creació de noves funcions, la qual cosa ha estat anteriorment explicada en l'apartat teòric. En aquestes hem de posar el conjunt d'ordres que volem per fer una acció determinada com la d'avançar, tot seguit escrivim “`void`” i just després el nom de la funció. I tal com s'explica en la teoria, aquestes funcions s'han de col·locar abans que el “`void setup`”.



```

void davant() {
    digitalWrite(8, LOW);          /*el conjunt d'ordres que
    digitalWrite(9, LOW);          posteriorment utilitzarem
    digitalWrite(10, LOW);         mitjançant les funcions*/
    digitalWrite(11, LOW);
}

```

I per tant cada vegada que vulguem emprar aquest conjunt d'accions només ens caldrà ficar el nom que posseeix la funció (davant), en aquest cas.

I per concloure, hem de col·locar totes les condicions dins d'una, la qual ordeni que, en cas que, el nombre que l'Arduino rep, és a dir la funció “valorDelBotó”, no sigui major a -1, que l'Arduino no faci res. Com que -1 és el nombre enviat quan cap botó està sent pres, no té cap utilitat el fet que el microcontrolador pateixi un sobreesforç i disminueixi tant la seva velocitat de treball com el seu període de vida útil.

Aquesta condició l'hem d'escriure amb la funció (if), dins d'aquesta, hi col·loquem el “valorDelBoto” i finalment posem el signe > (més gran que) seguit del -1, i acabant amb un (else), el qual no contingui res.<sup>2</sup>

```

if(valorDelBoto > -1){

    if(valorDelBoto==57){ pujar();}
    else if(valorDelBoto==56){ baixar();}
    else if(valorDelBoto==49){ dreta();}
    else if(valorDelBoto==50){ esquerra();}
    else if(valorDelBoto==48){ pararUltimMoviment();}
    else if(valorDelBoto==111){ pararUltimMovimentX2();}
    else if(valorDelBoto==51){ davant();}
    else if(valorDelBoto==52){ darrera();}
    else{}

}
else{}

```

---

<sup>2</sup> El programa es troba acabat per complet al punt II. dels annexos.

### 3.1.2. CREACIÓ MITJANÇANT EL PONT H

És cert que, mitjançant el procediment anterior, els nostres “hardware & software” serien completament funcionals, tot i que, aquest mateix procediment podria ser executat mitjançant els ponts H, els quals aporten un punt de simplicitat un cop es tenen clars tots els conceptes.

El procediment per tal de muntar el circuit electrònic a través del component pont H és el següent:

Per començar hem de connectar la font d'alimentació al pont H, en cas que aquesta superi els 12 V hauríem de retirar un “Jumper” per tal que els ponts H puguin aguantar aquesta gran tensió i alimentar-los des d'una altra font de 5 V.

Degut a que els ponts H que utilitzem són dobles, amb un sol d'aquests components podem controlar 2 motors. És per això que el següent que hem de fer és connectar el primer motor en la parella A i el següent en la parella de pins B i el tercer motor ha d'anar connectat en un altre pont H en la parella A.

Igual que amb els relés necessitem dos pins que surtin de l'Arduino per cada una de les agrupacions de motors, en aquest cas, necessitem sis pins. Els dos primers pins que procedeixen de l'Arduino i tenen, com a finalitat, controlar la primera agrupació de motors, han de col·locar-se en el IN1 i IN2, ja que són els pins que controlen la parella de pins A.

A continuació els següents dos pins procedents de l'Arduino els hem de connectar en les sortides IN3 i IN4, ja que controlen la parella B. Aquest fragment ha de ser repetit tantes vegades com sigui necessari, és a dir, un cop per cada agrupació.

Un cop aquesta part està acabada, hem d'adaptar la part de programació, aquesta només varia en un aspecte respecte al codi dels relés. Aquesta petita diferència és que en el programa que ja tenim escrit, per tal de fer que el nostre submarí avanci hem d'escriure un (LOW) en el primer pin de sortida de l'Arduino i (HIGH) en la segona, invertir el (LOW) amb el (HIGH) per variar el sentit de gir i finalment posar

els dos en (LOW) per parar els motors, a diferència del codi dels relés, en el que per girar en una direcció o altra tant el (LOW) com el (HIGH) havia de coincidir, i combinar-los per fer parar els motors.<sup>3</sup>

---

<sup>3</sup> El programa acabat es troba al punt III. dels annexos.

### 3.2. FABRICACIÓ DE L'ESTRUCTURA

El material necessari per realitzar la fabricació de l'estructura és el següent:

- Tub de PVC en forma de T de 16 cm de diàmetre.
- Complements per poder roscar i ampliar la longitud del tub principal
- Un tub d'un metre de llargària i de 5 cm de diàmetre.
- Tubs rectes, en forma de T i corbats de 3 cm de diàmetre.
- Dos tubs forma cilíndrica, un de 13 cm de diàmetre i 22 cm de llargària i l'altre d'11 cm de diàmetre i 18 de llargària, amb les seves respectives rosques i tapes.
- Una peça metàl·lica amb forma de con amb la puntera rodona, i amb una obertura per permetre l'entrada de l'aigua
- Reixa metàl·lica de forma circular i reixa de PVC amb forma de bombeta, les dues per dosificar l'entrada d'aigua i evitar l'accés a brossa
- Abraçadora
- Cilindre de PVC de 4 cm de diàmetre i 5 cm d'allargada per protegir els motors
- Motors submergibles
- Go pro
- Adhesius Go pro

D'altra banda, les eines adients per poder manipular els diversos materials són:

- Tornavís (de diferents dimensions)
- Claus i caragols
- Trepant amb les diferents broques
- Cola de PVC
- Sicaflex juntament amb una pistola per tal que pugui ser aplicat.
- Alicates
- Cable principal el qual conté sis més en el seu interior
- Rebladora i els seus reblons corresponents

### **3.2.1. INTRODUCCIÓ**

En vers a la creació de l'estructura del submarí, els tres grans reptes que hem de resoldre són:

- La impermeabilitat
- La flotació
- L'estabilitat

És per això que, encara que, el posteriorment utilitzarem, sembli estanc, ens hem d'assegurar, ja que una sola gota d'aigua en contacte amb el hardware, podria causar un curt-circuit que inutilitzaria el correcte funcionament del submarí.

El primer repte l'intentarem resoldre mitjançant la utilització de dues càpsules diferents, en l'interior d'aquestes hi col·locarem, la bateria de 12 V i tot el circuit electrònic, tot i que, posteriorment, reforçarem les tapes de les càpsules per tal d'evitar l'entrada d'aigua.

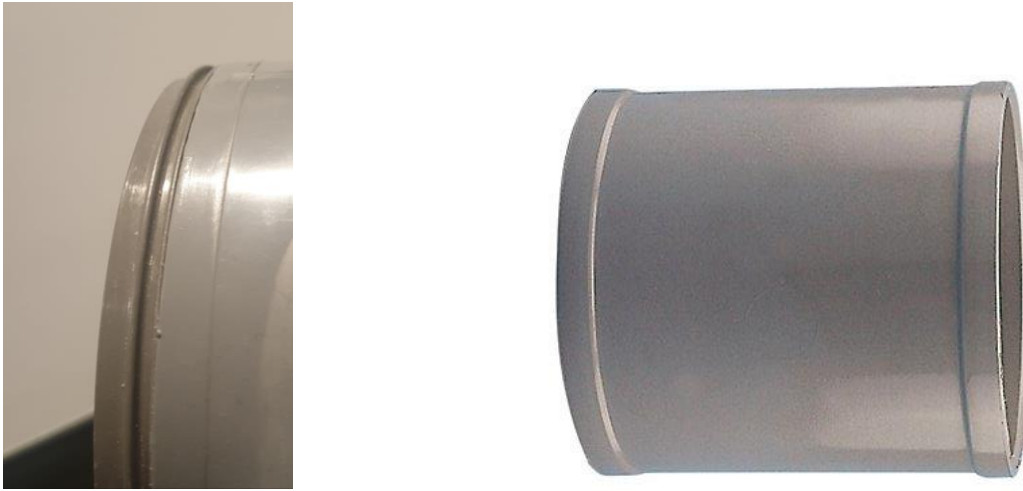
### **3.2.2. PROCEDIMENT**

#### Pas 1

El primer que hem de fer és col·locar el tub de PVC en forma de T de tal manera que el forat del mig quedi cap amunt. En aquest forat li hem de muntar una rosca i tot seguit una tapa. Aquesta tapa ens permetrà obrir i tancar aquest forat de forma còmoda, degut a que en el seu interior hi posarem la càpsula de la bateria, la qual ha de tenir un fàcil accés per poder carregar-la quan aquesta s'hagi esgotat. Aquest procés el repetirem amb un dels dos forats restants possibilitant així, que la part del davant tingui un fàcil accés.

A continuació hem d'adjuntar-li una extensió a l'últim forat, que ens permetrà, posteriorment, col·locar al seu interior la càpsula que contindrà els circuits, a diferència de la càpsula de la bateria, aquesta té un accés més difícil, degut a que

un cop el circuit és muntat correctament, no cal que sigui manipulat, encara que sempre hem de poder arribar-hi en cas que hi hagi un imprevist.<sup>4</sup>



## Pas 2

Tot seguit utilitzarem tant el tub de 13 cm de diàmetre i 22 cm de llargària com el d'11 cm de diàmetre i 18 de llargària. Amb els dos s'aplicarà el mateix procés, el qual tracta d'unir les peces amb rosca als dos extrems del tub cilíndric amb l'ús de la cola de PVC. Aquest producte ha de ser aplicat amb molta cura, ja que en cas de no fer-ho, es podria filtrar aigua cap a la bateria o cap al circuit.

Un cop la substància emprada s'hagi assecat, procedim amb la col·locació de les tapes hermètiques. A causa de la perillositat de l'entrada d'aigua dins les càpsules formades, el que hem de fer és assegurar l'estanquitat d'aquestes mitjançant el Sicaflex i la pistola que es necessita per emprar aquest material. És molt important que ens assegurem de sellar correctament totes les juntes i amb l'ajut d'una barreja d'aigua amb sabó aquest Sicaflex es torna mal·leable, és per això que el podem aplicar de la forma i en el lloc que ens convingui.

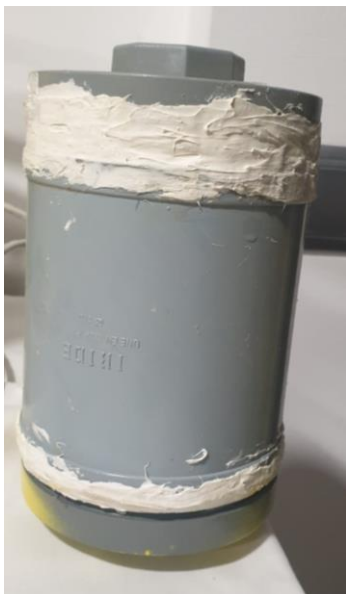
---

<sup>4</sup> Els plànols de les diferents parts de l'estructura en el seu apartat corresponent.

### Pas 3

Després d'haver completat el pas anterior, el que hauríem de fer és passar els cables des del circuit cap a la bateria i cap a cada un dels motors. A causa d'haver de fer un forat a través de les càpsules i després cobrir-lo per evitar el pas d'aigua, la millor opció és utilitzar un sol cable, que en conté sis en el seu interior. Degut a això, només haurem de fer passar un sol cable de major diàmetre per un sol forat en comptes de tots els petits cables que necessitem.

També hi ha l'opció d'utilitzar unes peces les quals eviten el pas d'aigua quan un cable els hi passa pel seu interior, tot i que, com ja he dit, la seva utilització és opcional, ja que amb la utilització del Sicaflex al voltant de la junta del cable amb el cilindre de la càpsula, ja no hi hauria circulació d'aigua.



### Pas 4

Aquest pas tractarà de crear una estructura que es col·locarà a la part inferior del submarí per tal de protegir els motors localitzats en aquesta zona. També evitarà que el submarí pateixi danys greus en cas que col·lideixi amb el fons del lloc on estigui navegant.

Per començar aquesta muntura hem d'utilitzar els tubs rectes, els de forma de T i els corbats, tots aquests de 3 cm de diàmetre. Tant uns com altres estan preparats

perquè puguin ser acoblats entre ells, i gràcies a això i mitjançant la cola de PVC podem començar amb la creació de l'estructura.

Primerament utilitzarem 6 tubs rectes i els tallarem per tal que mesurin 14 cm de llarg cada un. A continuació, incorporarem tres d'aquests mitjançant dos peces corbades formant un quadrat sense un dels seus cantons. Unit als dos extrems restants, col·locarem una peça en forma de T a cada un i juntament amb aquesta un altre tub de 14 cm.

Per ara, tindriem un rectangle el qual li manca un dels cantons i la base de les dues peces en forma de T lliure. Tot seguit seguiríem allargant el rectangle amb una peça amb forma de T més, per banda, i un cop ja les hem unit, tallarem dues peces rectes, però aquest cop de 5 cm, i les afegirem a l'altre cantó d'aquestes peces T. Finalment, farem servir les últimes dues peces que tenen forma de colze i una recta de 14 cm, per tancar el rectangle.

Actualment tenim el rectangle completat i les quatre bases de les peces T lliures, aquestes les utilitzarem per fixar-les al submarí, i això ho farem amb un tub de 3,5 cm unit a cada una de les bases desocupades. A la vegada, unit a una altra base d'una peça T, per tal de tenir més superfície en contacte amb l'esquelet principal del submarí i poder augmentar la rigidesa de l'estructura. Aquesta última agregació la farem amb un rebló i no amb la cola de PVC que hem estat utilitzant fins ara, tot i que abans de fer-la el que podríem fer és fer els forats on posteriorment aniran posicionats els motors per evitar que l'estructura ho dificulti.<sup>5</sup>

---

<sup>5</sup> L'estructura es pot veure clarament detallada a l'apartat dels plànols.





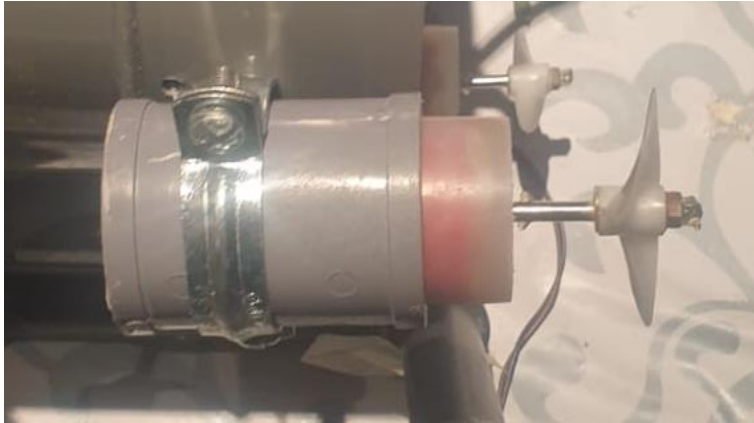
### Pas 5

Un cop hem fet els forats pels 2 motors de pujada i baixada a la part més baixa del submarí amb una distancia d'uns 20 cm, ara hem d'afegir els que ens serviran per avançar i retrocedir. Per que això pugui ser realitzat, hem d'utilitzar els cilindres de PVC de 4 cm de diàmetre i 5 cm d'allargada per protegir els motors.

Tot seguit, una abraçadora, que serà el punt d'unió entre el cilindre i l'esquelet del submarí. Aquestes abraçadores les col·locarem a la part de darrere del submarí i ho farem el més endarrere possible però lo suficientment endavant perquè les hèlices no arribin a tocar cap superfície. També hem de tenir en compte que s'ha de col·locar uns 3 cm per sobre de l'estructura formada durant el pas 4.

En l'interior d'aquests tubs es col·locaran els motors. És per això que abans de fixar els cilindres, hem de fer un petit forat tant en aquests com a la part de l'esquelet principal que es troba al cantó de l'abraçadora. La mesura d'aquest ha de ser prou gran perquè passin els dos cables del motor.

Un com els hem passat, podem omplir l'espai restant amb Sicaflex, d'aquesta manera evitem una desconexió dels cables amb el motor i a la vegada ens millorem la impermeabilitat d'aquest.



### Pas 6

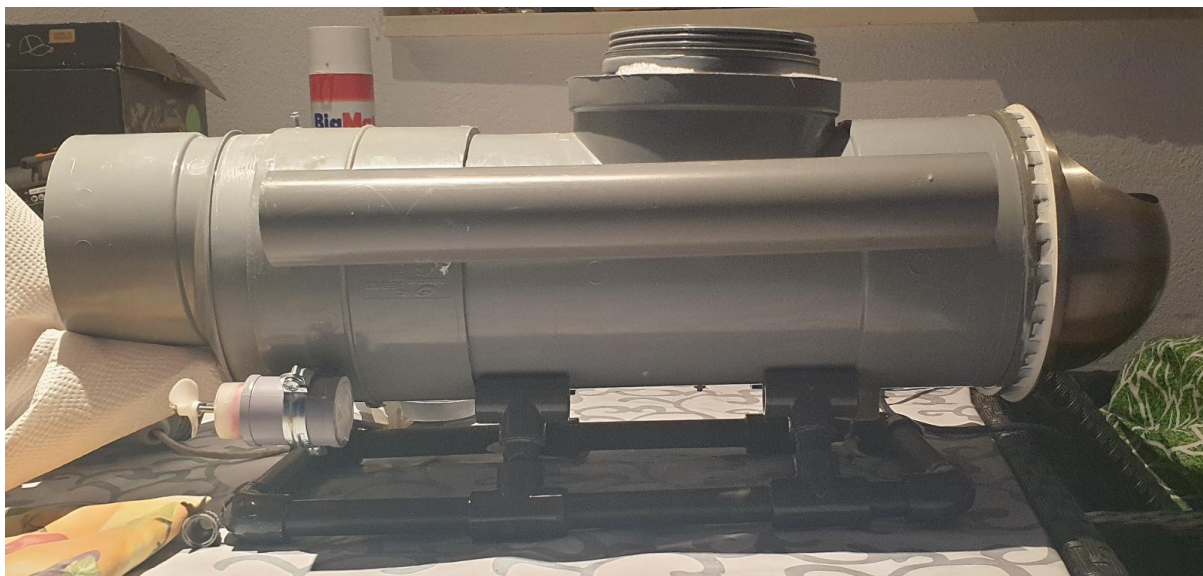
L'estructura principal del submarí està completa, tot i que a continuació hem d'acabar alguns detalls que poden ser fonamentals per mantenir l'equilibri i la flotabilitat del submarí. Un d'aquests és l'adjuntament de dos tubs laterals que estaran plens d'algun material que floti, en cas que el submarí s'enfonsi molt, o simplement deixar que corri l'aigua a través d'aquests en cas que no calgui més flotació. Aquests tubs de 5 cm de diàmetre i tallats amb una llargària de 38 cm, els hem de situar 7 cm per sobre els motors, ja que en cas que s'utilitzin com a flotadors, els motors s'han de trobar submergits de totes formes.

Finalment, hem de posar les dues càpsules a l'interior del tub principal, la que conté la bateria se situarà a l'interior de la base de peça principal, i pot anar fixada amb un clau per la part més baixa per augmentar la seva rigidesa. La segona càpsula estarà situada a la part del darrere, posada en horitzontal.

I per acabar, hem d'acoblar la peça de davant i la del darrere per tal de filtrar l'aigua que passa i evitar el pas de brossa.<sup>6</sup>

---

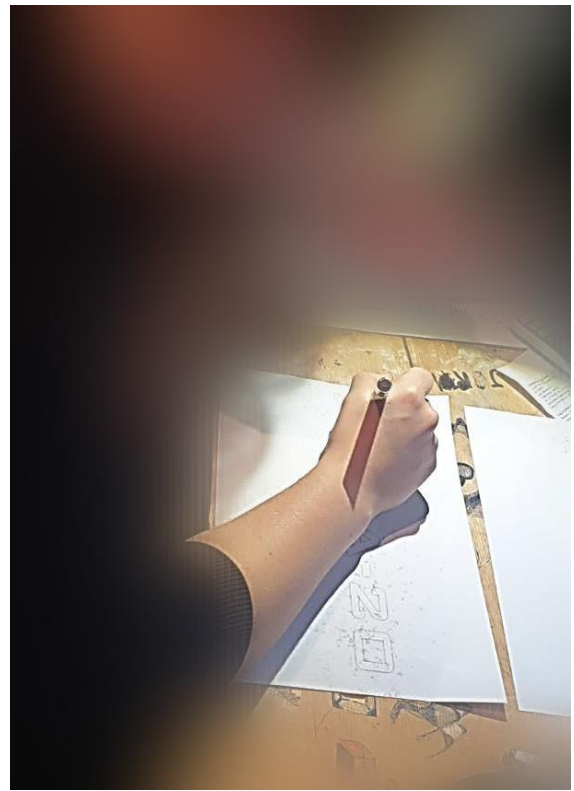
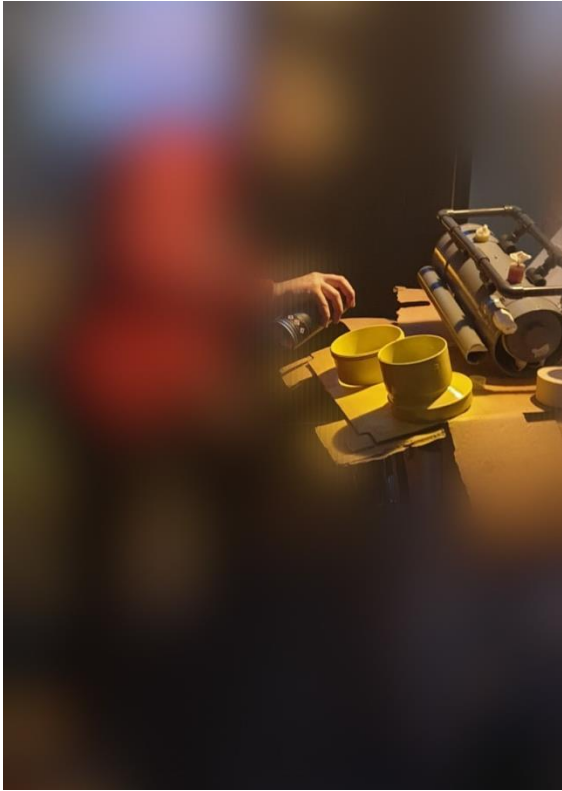
<sup>6</sup> En l'última part dels plànols situats als annexos, es pot veure la estructura completament acabada.







### Pas 7

Com a últim pas l'hem de pintar i un cop acabat aquest pas, hi ha la possibilitat d'afegir una Go pro al submarí, que via Bluetooth, enviaria les imatges en temps real. Aquestes les podríem visualitzar mitjançant l'ús de l'aplicació oficial Go pro, la qual també permet la divisió de pantalla, donant lloc al fet de controlar al submarí a través d'una visió en primera persona. L'adhesiu per instal·lar la càmera, pot anar col·locat segons en convingui, tot i que hi ha dues posicions en què li podem treure el màxim de profit. Per un cantó, el dispositiu el podem situar sobre la part del davant per tal que enfoqui cap al destinatari, però per altra banda, hi ha la possibilitat de col·locar la càmera per sota del tub principal, podent així obtenir una visió del fons on naveguem. En el cas que disposéssim la càmera en la primera posició, en la segona es podria muntar una llanterna submergible que s'accionés mitjançant un altre relé.





### 3.3 COSTOS DEL PROYECTO

PRODUCTE	FOTOGRAFIA	PREU	QUANTITAT	TOTAL
Components electrònics				
Arduino UNO		22.80€	2	45.60€
Relés (3 unitats)		6.49€	3	19.47€
Pont H		5.95€	3	17.85€
Basic starter kit		16.99€	1	16.99€

BreadBoard		8.90€	1	8.90€
Cables		3.99€	3	11.97€
Mòdul Bluetooth		9.99€	2	19.98€
Bateria 12V		23.2€	1	23.2€
Elements estructurals				
Tub de PVC (T) 16 cm de diàmetre		10.42€	1	10.42€
Rosca corresponent		9.29€	2	18.58€

Tub d'un metre i 5 cm de diàmetre		5.41€	2	10.82€
Tub 13 cm de diàmetre i 22 cm llargària		2.51€	1	2.51€
Tub 11 cm de diàmetre i 18 cm llargària		2.51€	1	2.51€
Registres corresponents (tapes i rosques)		3.38€	4	13.52€
Peça de PVC allargadora de 16 cm de diàmetre		2.28€	1	2.28€
Tub de PVC reductora		2.12€	1	2.12€
Tubs rectes, en forma de T i corbats de 3 cm de diàmetre. (12 u.)		19.46€	1	19.46€

Cable principal el qual conté sis més en el seu interior (10m)		11.99€	1	11.99€
Motor submergible		19.99€	4	79.96€
Joc de 6 tornavisos		15.95€	1	15.95€
Abraçadores + claus i cargols		2.50€	1	2.50€
Sicaflex		5.95€	3	17.85€
Pistola d'aplicació		8.99€	1	8.99€
Cola PVC		11.29€	1	11.29€



Alicates universals		9.99€	1	9.99€
Rebladora		15.29€	1	15.29€
Reblons (50 unitats)		6.29€	1	6.29€
Trepant + broques		59.99€	1	59.99€
Càmera Go pro (Hero 4)		399.99€	1	399.99€
Adhesius Go pro (10 unitats)		9.99€	1	9.99€
Peça metàl·lica amb forma de con i entrada de l'aigua, juntament amb una reixa.		22.99€	1	22.99€

Focus submergible		19.10€	2	38.20€
Pintura en esprai		4.95€	3	14.85€
Cinta de pintor		2.95€	1	2.95€
TOTAL				975.24€
TOTAL DESCONTANT EL MATERIAL DE PROPIETAT =				345.93€

## 4. CONCLUSIONS

Aquest treball ha englobat tres temes fonamentals, primerament tenim l'electrònica, és a dir, tot el conjunt de circuits i coneixements relacionats necessaris haver-lo dut a terme. Dins d'aquest apartat he pogut extreure les següents conclusions:

- ✓ Sempre s'ha de tenir en compte que estem treballant amb electricitat, la qual, en cas que es trobi en alts voltatges, com els que estem utilitzant, pot arribar a ser perillosa ja sigui amb una electrocució mitjançant un contacte directe o pel fet que un curt circuit pot fondre els cables arribant a la creació d'espurnes o fins i tot una flama.
- ✓ Amb relació al circuit electrònic, he observat que és una feina molt laboriosa i complexa, i s'ha de tenir present en tot moment on es vol arribar i el funcionament exacte de cada un dels components emprats. Ja que, un sol error, pot provocar el fet d'haver de repetir el circuit en la seva totalitat.
- ✓ Vigilar sempre l'estat dels components que estiguin sent utilitzats, ja que la no comprovació del correcte funcionament d'aquests, pot provocar que el circuit falli, encarar que estigui ben muntat.

Seguidament tenim l'apartat de programació, del qual he tret les següents conclusions:

- ✓ Tot i que tenia experiència en aquest camp, anterior al treball, m'ha suposat un gran esforç i una extensa recerca poder elaborar cada un dels programes que he necessitat, és a dir, per molt que sàpigues de programació, t'has d'informar del tipus de llenguatge que necessites segons el projecte que vulguis arribar a fer.
- ✓ Anar explicant cada pas a mesura que el programa va sent elaborat, ja que en cas que una altra persona volgués utilitzar part del meu programa, o ampliar-lo li seria més fàcil si per cada fragment de programa que escric,

explico el seu funcionament, també m'ha sigut molt útil per prosseguir amb l'escriptura del programa un cop la volia reprendre després de diversos dies.

L'últim dels temes fonamentals que abasta aquest treball, és la part mecànica, la qual he obtingut les següents conclusions:

- ✓ És molt complicat arribar a obtenir una flotabilitat 0, és a dir, que el submarí ni s'enfonsi ni suri més del conte. Per poder complir aquesta adversitat, s'han de fer moltes proves, fins a arribar-hi.
- ✓ Arribar a aconseguir que les càpsules siguin impermeables ha sigut més difícil de l'esperat, ja que les rosques emprades permetien un petit pas d'aigua que gairebé provoca un curtcircuit.
- ✓ S'ha de vigilar molt amb els materials que s'utilitzen, perquè si aquests són metàl·lics, com per exemple el plom, poden arribar a fer interferències en el senyal Bluetooth i provocar que el submarí acabi perdent la connexió.

He pogut complir la major part dels objectius que m'havia plantejat prèviament,

- ✓ En primer lloc he pogut realitzar la construcció d'un prototip que funcioni correctament, a causa que segueix cada una de les ordres enviades pel telèfon, té una flotabilitat 0, la qual li permet ascendir i descendir a voluntat i tant la bateria com el circuit es troben completament aïllats de l'entorn aquàtic.
- ✓ En segon lloc he pogut reutilitzar uns motors d'uns submarins que funcionaven via cable, i fer el suport protector dels motors inferiors mitjançant les restes d'aquests. També he reutilitzat una sortida d'un tub de claveguera per tal d'evitar que aigua amb brossa circuli per l'interior del submarí.

- ✓ He pogut adquirir coneixements de programació i de l'Arduino Uno, i a més a més he sigut capaç de realitzar diversos circuits elèctrics, cada un més complex que l'anterior.
- ✓ Amb relació l'últim objectiu, desconec si s'ha complet o no, degut a que encara, el meu treball no ha estat difós, tot i que el programa que he elaborat ja es troba a disposició de qui vulgui.

Per concloure, m'agradaria fer una reflexió personal sobre aquest treball. Aquesta inclourà diversos problemes que he patit durant l'elaboració d'aquest projecte, alguns més perillosos que d'altres, i com m'he anat sentit a mesura que anava avançant.

Degut a la meva inexperiència he hagut de repetir moltes vegades la completa realització del circuit, entre una d'aquestes, vaig tenir el problema més greu de tot el procés. Aquest va ocórrer quan estava treballant amb unes bateries de 20 V, les quals tenen un punt de perillositat, de sobte, mentre desmuntava un relé, un cable que anava connectat amb el pol positiu de la bateria, va caure amb un que anava cap al negatiu, provocant així un curt circuit que ràpidament va escalfar els cables a unes temperatures tan altes que es van fondre entre ells i va aparèixer molt de fum, el qual va acabar esdevenint en una gran flama.

Un altre dels problemes amb el que m'he trobat ha sigut que, al no haver comprovat el correcte estat dels materials que estava utilitzant, després de molt de temps i de moltes proves vaig poder descobrir que un dels components electrònics que havia comprat no funcionava correctament, tot i que abans d'arribar a aquesta conclusió vaig haver de muntar el circuit de moltes maneres diferents i canviar cada una de les parts que el formaven.

Tot aquest treball m'ha suposat la implicació de molt esforç i dedicació, no només econòmic, sinó també molta part psicològica, ja que he passat per moments de molta frustració i angonya, en els quals creia que no anava a poder acabar el treball.

Primerament em vaig haver d'enfrontar amb la part pràctica, encara que m'obligués a implicar un incommptable nombre d'hores, el vaig poder assolir gràcies a la proposició de moltes metes simples, que al final han acabat esdevinguin en un treball molt complex. La simplificació d'aquestes m'ha ajudat molt, encara que això no li restés dificultat al repte, em proporcionava molta seguretat i satisfacció poder aconseguir cada objectiu que em plantejava, i a poc a poc m'anava animant més per continuar.

## 5. WEBGRAFIA

Robòtica i electrònica (2016) Recuperat dia 25 de Gener del 2020

<http://elprofegarcia.com/>

Construcció d'un habitatge domòtic amb arduino (2016) Recuperat el dia 25 de Gener del 2020

<https://roboticabruquers.wordpress.com/2016/03/14/construccion-dun-habitatge-domotic-amb-arduino/>

Connectar arduino amb el telèfon mòbil via Bluetooth (2014) Recuperat el dia 25 de gener del 2020

<https://www.youtube.com/watch?v=j-cUDfmtq5g>

Entrades i sortides digitals i analògiques. Recuperat el dia 28 de Gener del 2020

[http://dfists.ua.es/~jpomares/arduino/page\\_10.htm](http://dfists.ua.es/~jpomares/arduino/page_10.htm)

Alberto Valero, Usando LEDs con sentencias condicionales if-else. Recuperat el dia 3 de febrer del 2020

<http://diwo.bq.com/usando-leds-con-sentencias-condicionales-if-else/>

Luis del Valle Hernández, Sentencia if con Arduino, controla la ejecución de tu código. Recuperat el dia 3 de Febrer del 2020

<https://programarfacil.com/blog/arduino-blog/if-else-arduino/>

Arduino y las esperes (2014) Recuperat el dia 6 de febrer del 2020

<http://www.studioseed.net/blog/arduino-y-las-esperas/>

If con varias condiciones (2018) Recuperat el dia 6 de febrer del 2020

<https://forum.arduino.cc/index.php?topic=562462.0>

Paco Pons, Instrucción condicional IF en Arduino. Recuperat el dia 6 de Febrer del 2020

<http://yomaker.com/condicional-if-en-arduino/>

Manuel Delgado Crespo, Arduino y su documentación en español. Recuperat el dia 6 de Febrer del 2020

<http://manueldelgadocrespo.blogspot.com/p/ifelse.html>

Sentencia IF anidada, Recuperat el dia 15 de febrer del 2020

<https://programacionfusa.wordpress.com/contenido/sentencia-if-anidada/>

Wait for input (2009). Recuperat el dia 15 de febrer del 2020

<https://forum.arduino.cc/index.php?topic=3567.0>

Variables (VOID) (2019). Recuperat el dia 15 de febrer del 2020

<https://www.arduino.cc/reference/tr/language/variables/data-types/void/>

Why is my HelloWorld function not declared in this scope? (2012) Recuperat el dia 16 de febrer del 2020

<https://stackoverflow.com/questions/8234484/why-is-my-helloworld-function-not-declared-in-this-scope>

Arduino y los relés. Recuperat el dia 21 de febrer del 2020

<https://www.prometec.net/reles/>

Módulos de relés y opto acopladores. Recuperat el dia 21 de febrer del 2020

<https://www.prometec.net/optoacopladores/>

Optoacoplador. Recuperat el dia 25 de febrer del 2020

<https://www.areatecnologia.com/electronica/optoacoplador.html>

Maxima tension e intensidad en un rele de 5 volts? Recuperat el dia 25 de febrer del 2020

<https://www.electronicafacil.net/foros/PNphpBB2-viewtopic-t-4444.html>

Programar Relay de 5V con Arduino (2016). Recuperat el dia 25 de febrer del 2020

<https://www.youtube.com/watch?v=Cyny5CnuHsk>

Wikipedia, Serial.available (2019). Recuperat el dia 12 de març del 2020

[https://arduwiki.perut.org/wiki/Serial.available\(\)](https://arduwiki.perut.org/wiki/Serial.available())

Arduino. Serial.available(), (2019). Recuperat el dia 12 de març del 2020

<https://www.arduino.cc/reference/en/language/functions/communication/serial/available/>

Arduino. Available() (2019) Recuperat el dia 12 de març del 2020

<https://www.arduino.cc/en/Serial.Available>

Arduino Tutorial: Serial Inputs (2017). Recuperat el dia 12 de març del 2020

<https://www.norwegiancreations.com/2017/12/arduino-tutorial-serial-inputs/>

Control del Arduino desde su teléfono Android (2014). Recuperat el dia 12 de març del 2020

[https://www.youtube.com/watch?v=ftXTtba\\_CjU](https://www.youtube.com/watch?v=ftXTtba_CjU)

Controle su Casa desde el Telefono (2015). Recuperat el dia 18 de març del 2020



<https://www.youtube.com/watch?v=2A3zqhkRjCs>

Encender un foco desde tu teléfono android (2017). Recuperat el dia 18 de març del 2020

<https://www.youtube.com/watch?v=USB-scEwknY>

Conectar 110V / 220V con Arduino (2015). Recuperat el dia 18 de març del 2020

<https://www.youtube.com/watch?v=Gif-vw9FtII&t=307s>

Designing In USB Type-C and Using Power Delivery for Rapid Charging (2017). Recuperat el dia 28 de març del 2020

<https://www.digikey.com/en/articles/designing-in-usb-type-c-and-using-power-delivery-for-rapid-charging>

Todo lo que no te han contado del cable USB Tipo C (2019). Recuperat el dia 28 de març del 2020

<https://www.youtube.com/watch?v=EdijTnehuj0>

10V desde el puerto USB (2013). Recuperat el dia 4 d'abril del 2020

<https://electronica.guru/questions/15396/10v-desde-el-puerto-usb>

Series and Parallel Battery Configurations (2019). Recuperat el dia 4 d'abril del 2020

[https://batteryuniversity.com/learn/article/serial\\_and\\_parallel\\_battery\\_configurations](https://batteryuniversity.com/learn/article/serial_and_parallel_battery_configurations)

Componentes electrónicos. Recuperat el dia 20 d'abril del 2020

<https://tuelectronica.es/componentes-electronicos/>

Miguel A. Chiarena. Componentes Electrónicos Básicos. Recuperat el dia 20 d'abril del 2020

<https://sites.google.com/site/electronica4bys/componentes-electronicos-basicos>

Descubre componentes electrónicos: Los más utilizados (2020). Recuperat el dia 20 d'abril del 2020

<https://www.electronicboard.es/componentes-electronicos-los-mas-utilizados/>

Yennifer Gomez, Conoce Distintos Tipos de Componentes Electrónicos. Recuperat el dia 20 d'abril del 2020

<https://siaguanta.com/c-tecnologia/tipos-de-componentes-electronicos/>

Tutorial Básico de Uso del Módulo Bluetooth HC-06 y HC-05. Recuperat el dia 2 de maig del 2020

[https://www.naylampmechatronics.com/blog/12\\_Tutorial-B%C3%A1sico-de-Uso-del-M%C3%B3dulo-Bluetooth-H.html](https://www.naylampmechatronics.com/blog/12_Tutorial-B%C3%A1sico-de-Uso-del-M%C3%B3dulo-Bluetooth-H.html)

MÓDULO BLUETOOTH HC-06. Recuperat el dia 2 de maig del 2020

<https://www.prometec.net/bt-hc06/>

Wikipedia. Relé. Recuperat el dia 13 de maig del 2020

<https://es.wikipedia.org/wiki/Rel%C3%A9>

Cesar Cinjordiz, Relé electromecánico (2016). Recuperat el dia 13 de maig del 2020

<https://www.infootec.net/rele-electromecanico/>

Salidas relé, transistor y control SSR. Recuperat el dia 29 de maig del 2020

<https://fema.es/news/salidas-rele-transistor-control-ssr-cuando-utilizar-una/>

¿Cómo saber los límites de voltaje de un relé? (2017). Recuperat el dia 29 de maig del 2020

<https://www.altecdust.com/blog/item/50-blog-rele-limites-voltaje-corriente>

INTRODUCCIÓN A LOS RELÉS. Recuperat el dia 2 de juny del 2020

<https://www.inventable.eu/introduccion-a-los-reles/>

Controlar motores de corriente continua con Puente H. Recuperat el dia 13 d'agost del 2020

<http://arteymedios.org/tutoriales/item/76-controlar-motores-de-corriente-continua-con-puente-h>

Como usar el driver controlador de motores L298N (2017). Recuperat el dia 13 d'agost del 2020

<https://www.youtube.com/watch?v=c0L4gNKwjRw>

Como usar el driver controlador de motores L298N | Arduino (2017). Recuperat el dia 13 d'agost del 2020

<https://www.youtube.com/watch?v=0bxqxp9EkVI>

¿Cuáles son los objetos que pueden hacer que tu conexión funcione mal? (2018). Recuperat el dia 25 d'agost del 2020

<https://www.bbc.com/mundo/noticias-45420855>

What Is an H-Bridge? (2018). Recuperat el dia 19 de setembre del 2020

<https://www.build-electronic-circuits.com/h-bridge/>

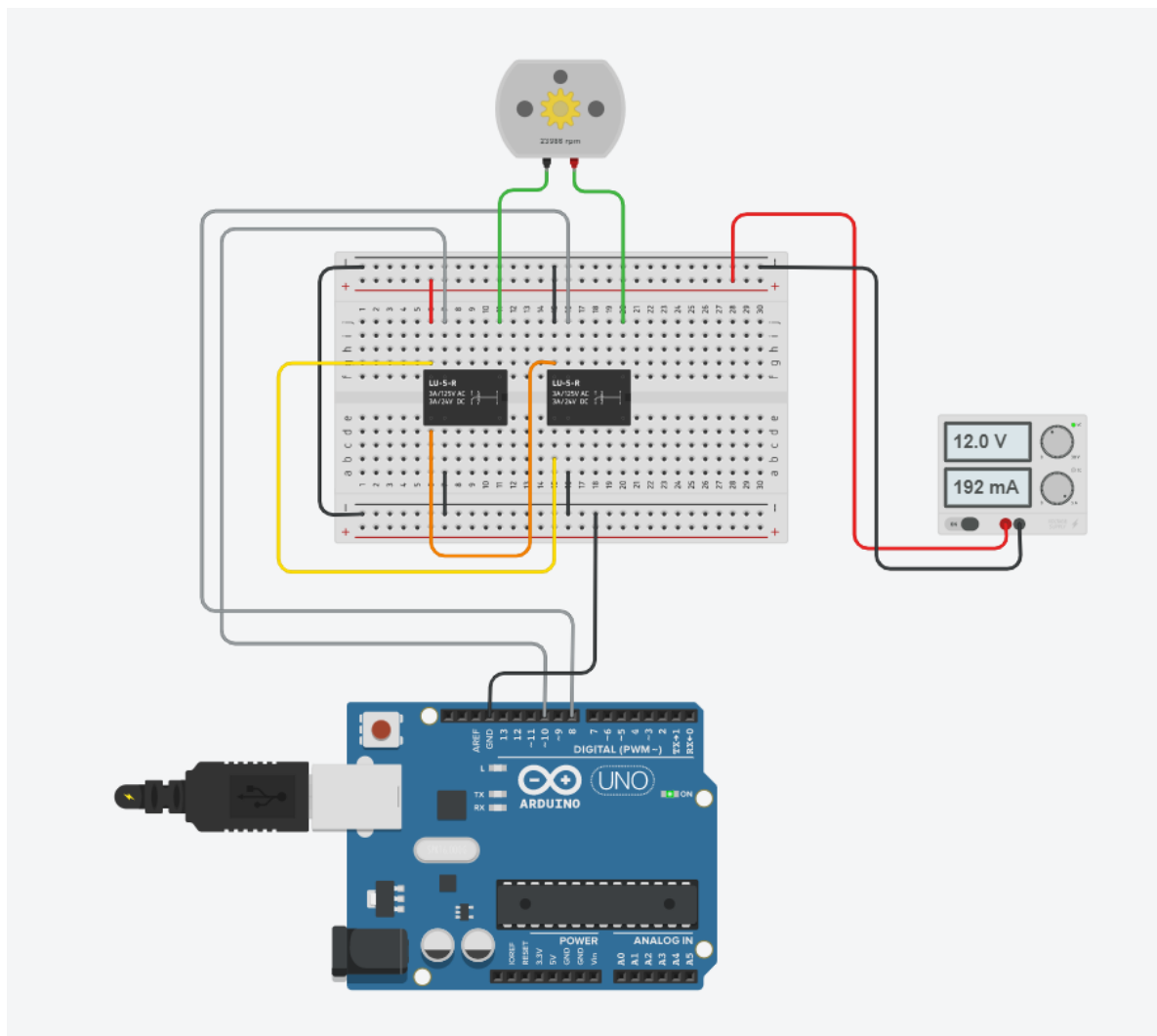
Diferencia entre componentes electrónicos pasivos y activos (30 de juliol del 2019). Recuperat el dia 26 de setembre del 2020

<https://www.surtel.es/entrada.php?slug=diferencia-entre-componentes-electronicos-pasivos-y-activos&p=1>

# ANNEXOS



## I. FORMACIÓ DEL CIRCUIT EN DIGITAL



Cable taronja: Connecta (NC) del relé 1 (esquerra) amb el (NO) del relé 2(dreta)

Cable groc: Connecta (NO) del relé 1 amb el (NC) del relé 2

Cable gris: Connecta cada relé amb el seu pin de l'Arduino corresponent

Cable verd: Connecta les sortides (COM) dels dos relés cap al motor

Cable vermell: Connecta (NO) del relé 1 amb el (NC) del relé 2

Cable negre: Connecta tots els (GND), tant els que s'uniran a l'Arduino com a la font d'energia externa.

## II. PROGRAMACIÓ DEL CIRCUIT FORMAT PER RELÉS

```
int valorDelBoto;                                /*int s'utilitza per declarar que les dades
                                                arriben són nombres sencers.
                                                S'ha d'especificar quin tipus de dada és*/

void davant() {
    digitalWrite(8, LOW);
    digitalWrite(9, LOW);                        //cada una de les ordres que posteriorment
    digitalWrite(10, LOW);                       //utilitzarem mitjançant les funcions
    digitalWrite(11, LOW);
}

void darrera() {
    digitalWrite(8, HIGH);
    digitalWrite(9, HIGH);
    digitalWrite(10, HIGH);
    digitalWrite(11, HIGH);
}

void dreta() {
    digitalWrite(8, LOW);
    digitalWrite(9, LOW);
    digitalWrite(10, HIGH);
    digitalWrite(11, HIGH);
}

void esquerra() {
    digitalWrite(8, HIGH);
    digitalWrite(9, HIGH);
    digitalWrite(10, LOW);
    digitalWrite(11, LOW);
}

void pujar() {
    digitalWrite(12, HIGH);
    digitalWrite(13, HIGH);
}

void baixar() {
    digitalWrite(12, LOW);
    digitalWrite(13, LOW);
}
```

```

void pararUltimMoviment(){
    digitalWrite(8, HIGH);
    digitalWrite(9, LOW);
    digitalWrite(10, HIGH);
    digitalWrite(11, LOW);
}

void pararUltimMovimentX2(){
    digitalWrite(12, HIGH);
    digitalWrite(13, LOW);

}

void setup()                                /*és el que executa quan encens l'Arduino*/
{
    Serial.begin(9600);                      //serveix perquè l'Arduino es comuniqui amb l'ordinador

    pinMode(8, OUTPUT);
    pinMode(9, OUTPUT);                      //perquè els pins enviïn 5 V
    pinMode(10, OUTPUT);
    pinMode(11, OUTPUT);
    pinMode(12, OUTPUT);
    pinMode(13, OUTPUT);

    digitalWrite(8, HIGH);
    digitalWrite(9, LOW);
    digitalWrite(10, HIGH);                  //perquè estiguin els motors parats
    digitalWrite(11, LOW);                  //quan s'encén l'Arduino
    digitalWrite(12, HIGH);
    digitalWrite(13, LOW);
}

void loop()                                //el que es repeteix tota l'estona
{

    valorDelBoto=Serial.read();

    if(valorDelBoto > -1){                    /*per tal que l'Arduino no treballi de
                                                forma innecessària que, quan no
                                                s'intercanvia cap senyal, apareix un -1 i
                                                d'aquesta forma, quan aquest hi aparegui,
                                                l'Arduino no farà cap mena d'esforç*/

```

```

if(valorDelBoto==57){ pujar();}
else if(valorDelBoto==56){ baixar();}      //conjunt de condicions les quals
else if(valorDelBoto==49){ dreta();}        //s'executaran un cop l'Arduino rebi
else if(valorDelBoto==50){ esquerra();}     //cada un dels nombres corresponents
else if(valorDelBoto==48){ pararUltimMoviment();}
else if(valorDelBoto==111){ pararUltimMovimentX2();}
else if(valorDelBoto==51){ davant();}
else if(valorDelBoto==52){ darrera();}
else{}                                     /*/ tant aquest "else{}" com el següent
                                           s'utilitza per tancar les condicions
                                           /*/
}
else{}

}

```

### III. PROGRAMACIÓ DEL CIRCUIT FORMAT PER PONTS H

```
int valorDelBoto;                                //int s'utilitza per declarar que les dades
                                                //que arriben són nombres sencers
                                                //S'ha d'especificar quin tipus de dada és

void davant() {
    digitalWrite(8, LOW);
    digitalWrite(9, HIGH);
    digitalWrite(10, LOW);
    digitalWrite(11, HIGH);
}

void darrera(){
    digitalWrite(8, HIGH);
    digitalWrite(9, LOW);
    digitalWrite(10, HIGH);
    digitalWrite(11, LOW);
}

void dreta(){
    digitalWrite(8, HIGH);
    digitalWrite(9, LOW);
    digitalWrite(10, LOW);
    digitalWrite(11, HIGH);
}

void esquerra(){
    digitalWrite(8, LOW);
    digitalWrite(9, HIGH);
    digitalWrite(10, HIGH);
    digitalWrite(11, LOW);
}

void pujar(){
    digitalWrite(12, LOW);
    digitalWrite(13, HIGH);
}

void baixar(){
    digitalWrite(12, HIGH);
    digitalWrite(13, LOW);
}

void pararUltimMoviment(){
    digitalWrite(8, LOW);
    digitalWrite(9, LOW);
    digitalWrite(10, LOW);
    digitalWrite(11, LOW);
}
```



```

void pararUltimMovimentX2() {
    digitalWrite(12, LOW);
    digitalWrite(13, LOW);
}

void setup()                                     /*és el que executa quan encens l'arduino*/
{
    Serial.begin(9600);                          //serveix perquè l'arduino es comuniqui amb l'ordinador

pinMode(8, OUTPUT);
pinMode(9, OUTPUT);                             //Perquè els pins enviïn 5V
pinMode(10, OUTPUT);
pinMode(11, OUTPUT);
pinMode(12, OUTPUT);
pinMode(13, OUTPUT);

digitalWrite(8, LOW);
digitalWrite(9, LOW);
digitalWrite(10, LOW);                          //perquè estiguin els motors parats quan s'encen l'arduino
digitalWrite(11, LOW);
digitalWrite(12, LOW);
digitalWrite(13, LOW);
}

void loop()                                     //el que es repeteix tota l'estona
{
    valorDelBoto=Serial.read();

    if(valorDelBoto > -1){

        if(valorDelBoto==57){ pujar();}
        else if(valorDelBoto==56){ baixar();}
        else if(valorDelBoto==49){ dreta();}
        else if(valorDelBoto==50){ esquerra();}
        else if(valorDelBoto==48){ pararUltimMoviment();}
        else if(valorDelBoto==111){ pararUltimMovimentX2();}
        else if(valorDelBoto==51){ davant();}
        else if(valorDelBoto==52){ darrera();}
        else{}

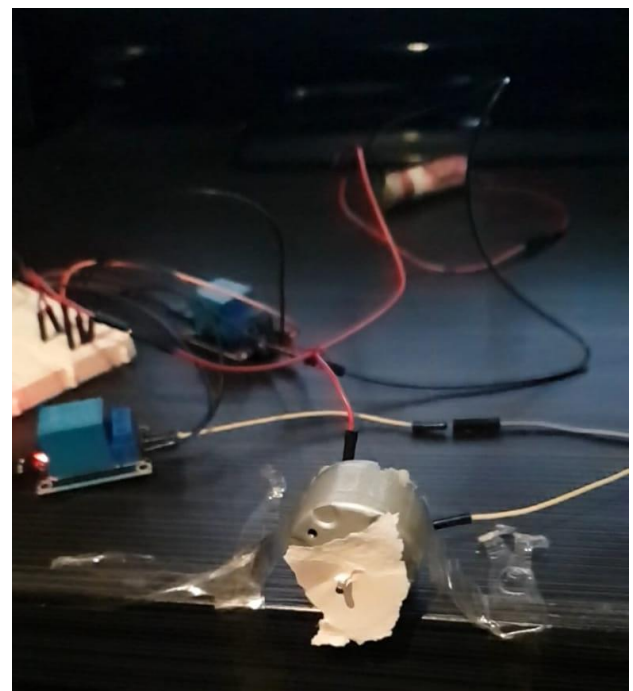
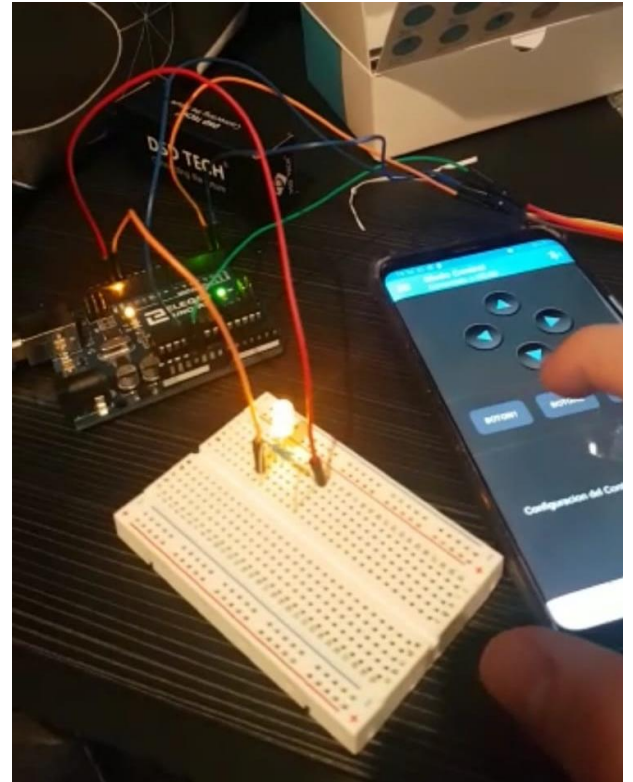
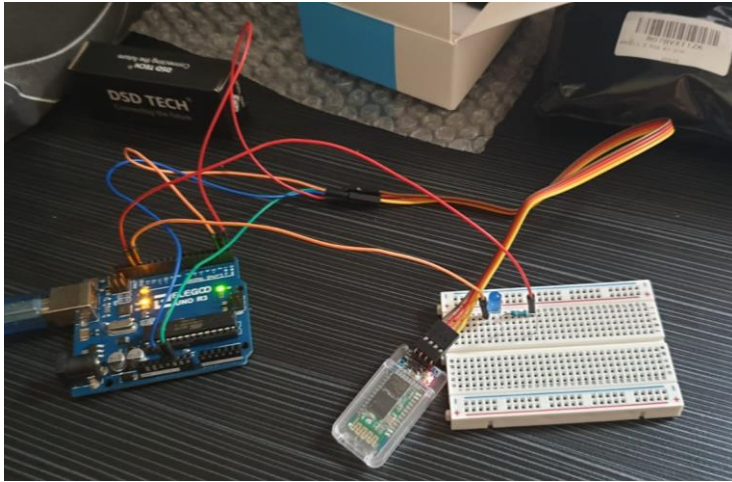
    }
    else{}

}

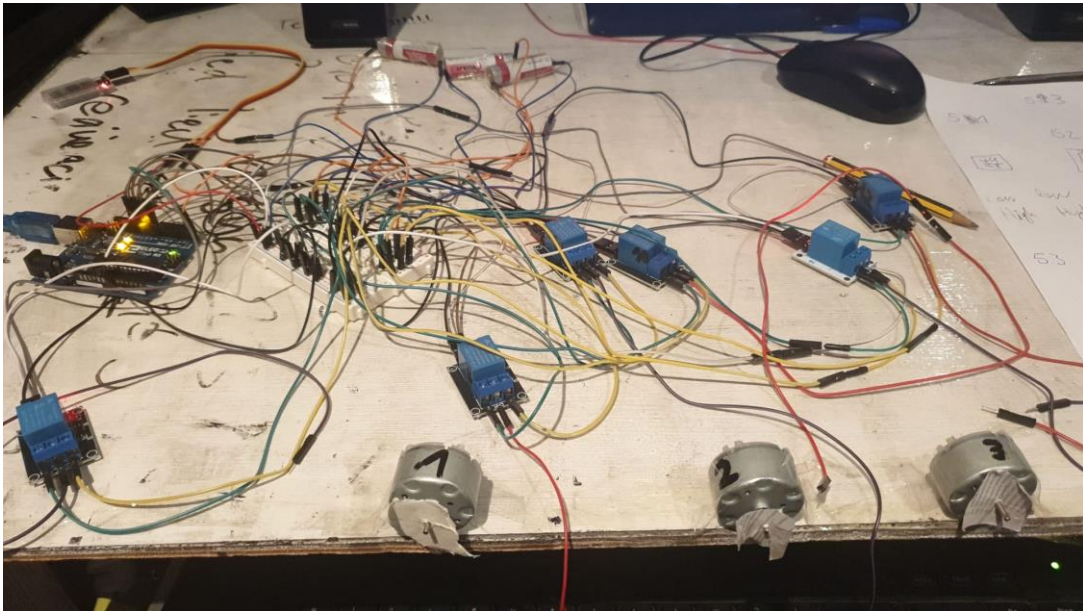
```

## IV. FOTOS DEL PROCÉS

### *Part electrònica*



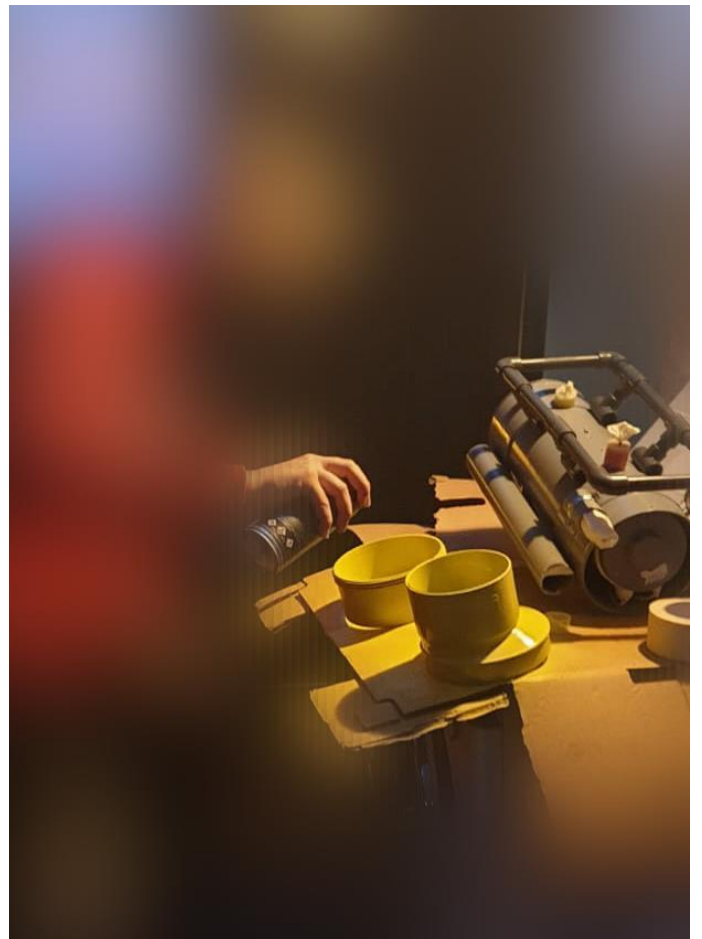






## ***Part mecànica***

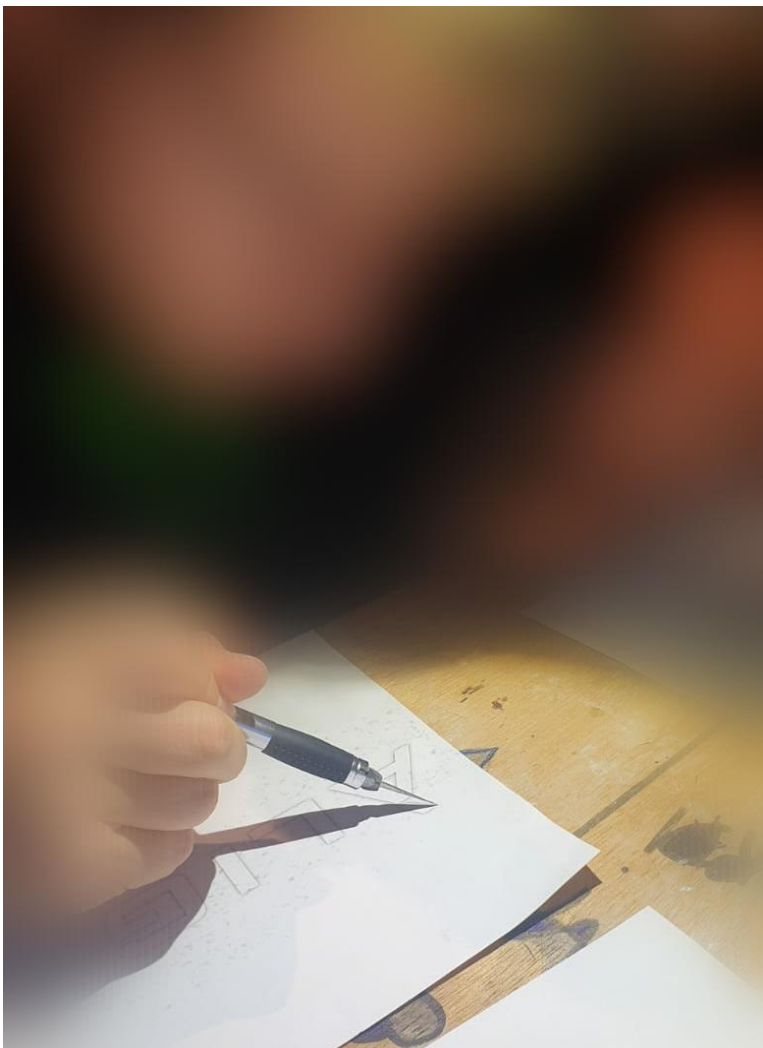
















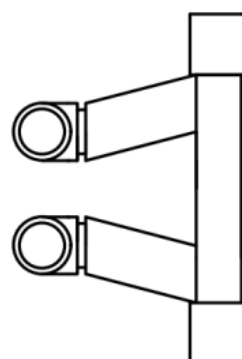
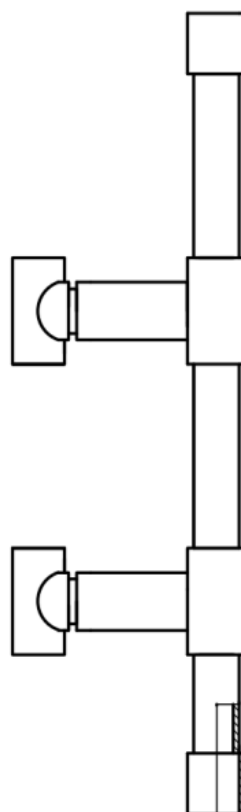
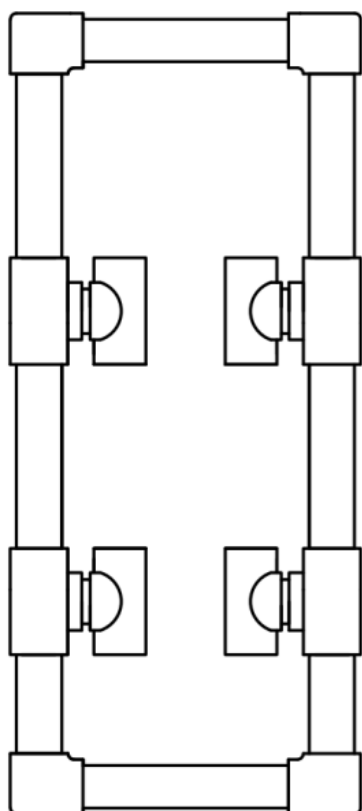


# PLÀNOLS

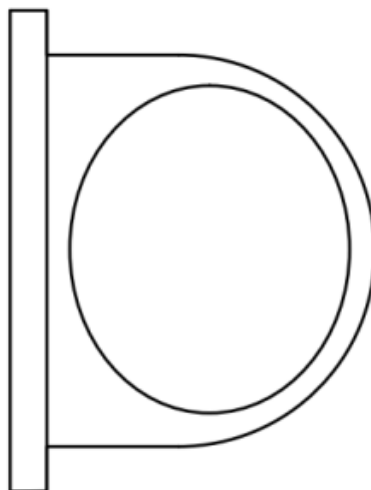
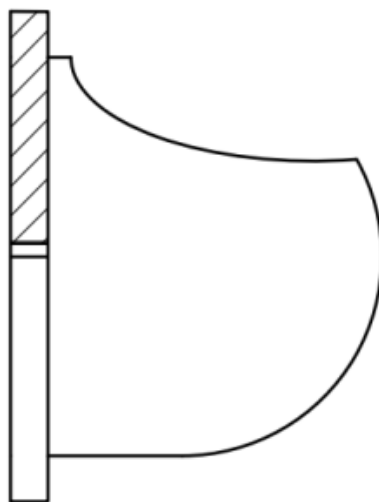
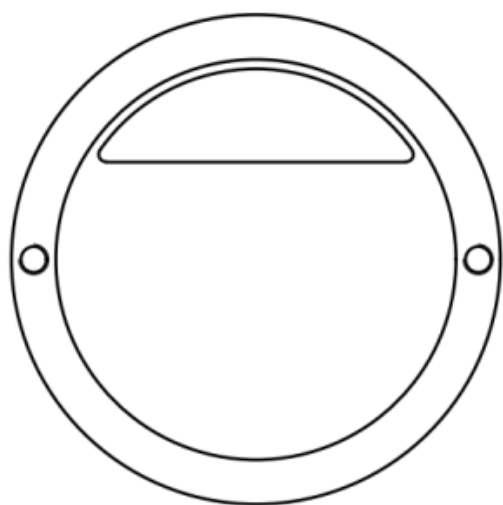


## **ÍNDEX**

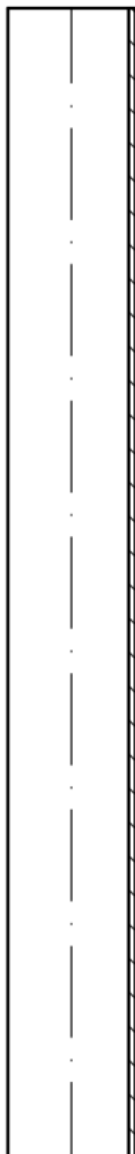
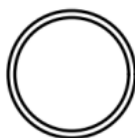
- 1. SUPORT**
- 2. CAPÇAL DE PROA**
- 3. TUB LATERAL**
- 4. CÀPSULA ESTANCA**
- 5. TAPA**
- 6. ENTRADA D'AIGUA POSTERIOR**
- 7. ELEVACIÓ D'ENTRADA POSTERIOR**
- 8. PROPULSOR**
- 9. PROTECTOR DEL PROPULSOR**
- 10. REGULADOR ENTRADA D'AIGUA**
- 11. ABRAÇADORA**
- 12. ESQUELET CENTRAL**
- 13. CONJUNT**



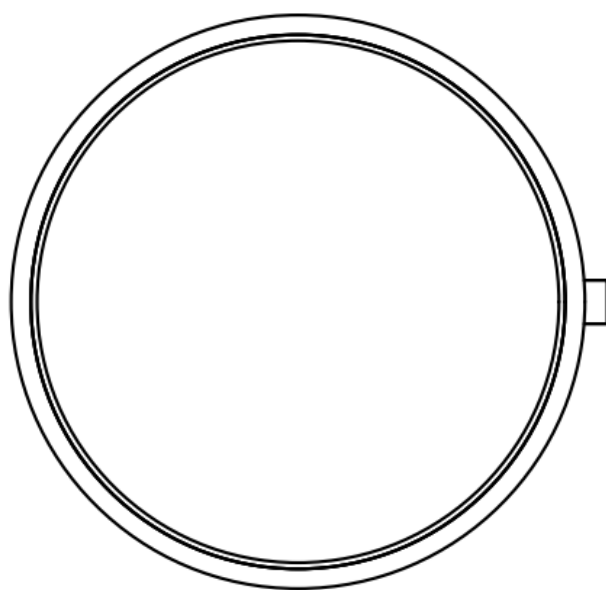
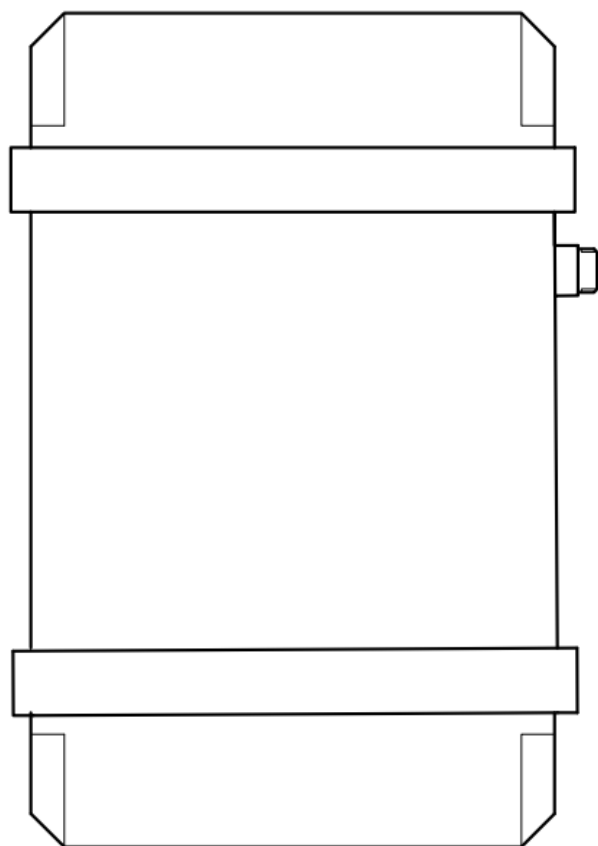
	<i>Dibuixat</i>		
	<i>Data</i>	2/07/2020	
<i>Escala</i> 1:4	SUPORT		Treball de recerca



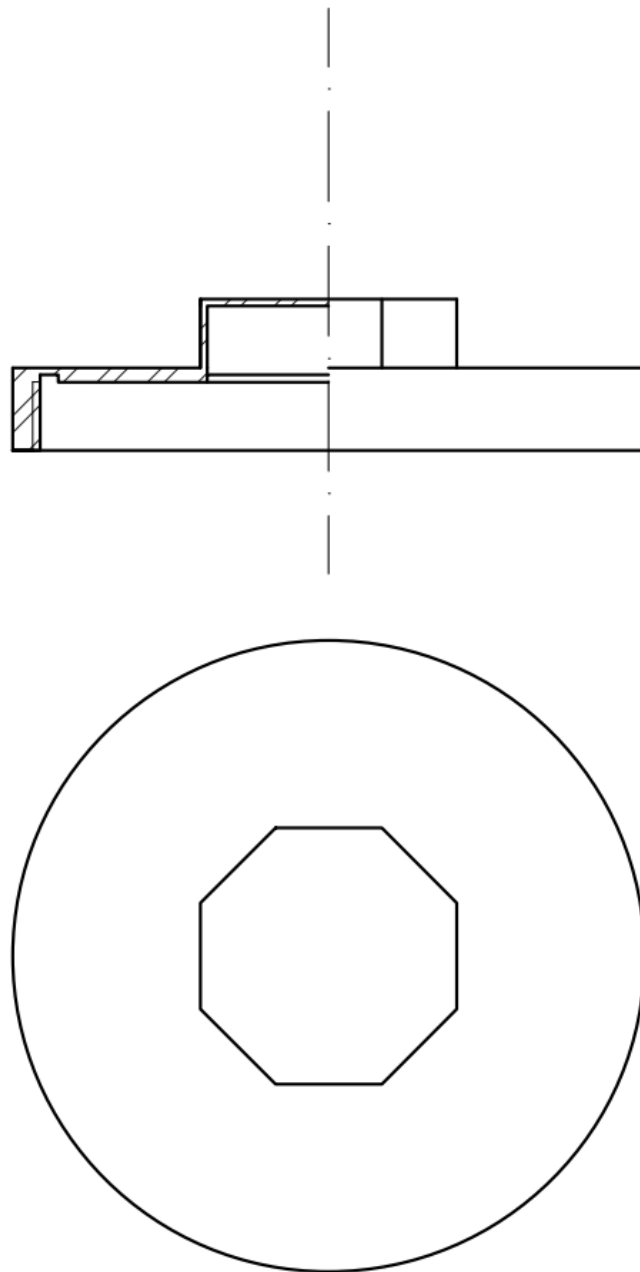
	<i>Dibuixat</i>		
	<i>Data</i>	12/07/2020	
<i>Escala</i> 1:2	CAPÇAL DE PROA		Treball de recerca



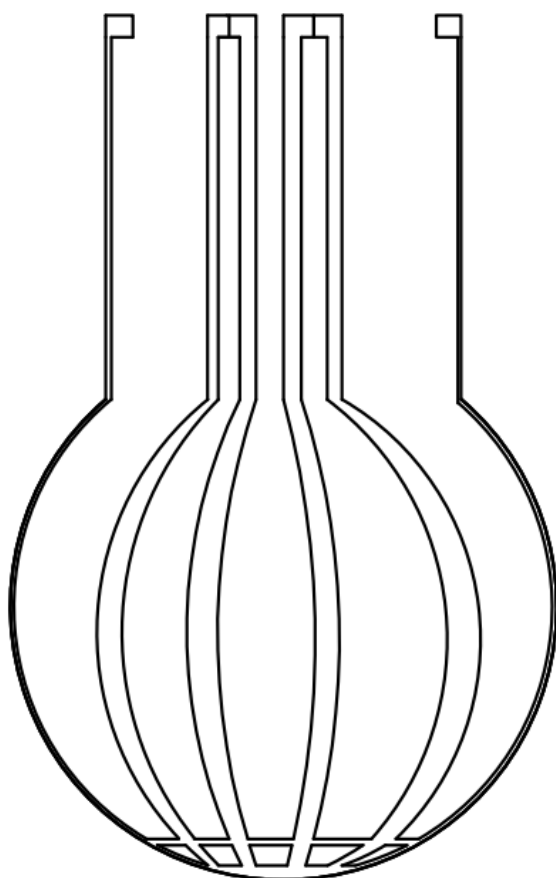
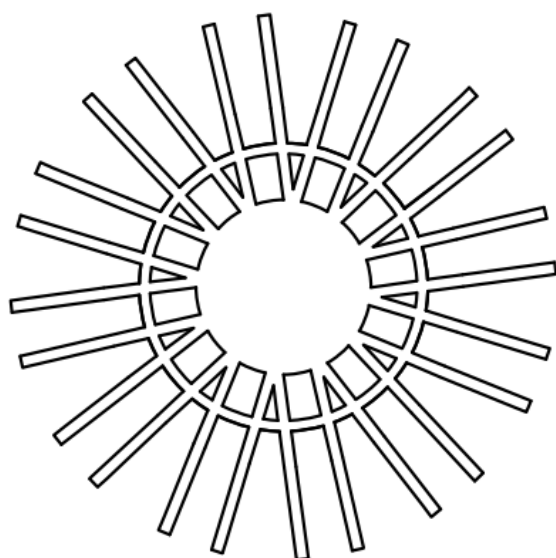
	<i>Dibuixat</i>		
	<i>Data</i>	16/07/2020	
<i>Escala</i> 1:4	TUB LATERAL		Treball de recerca



	<i>Dibuixat</i>		
	<i>Data</i>	18/07/2020	
<i>Escala</i> 2:3	CÀPSULA ESTANCA		Treball de recerca

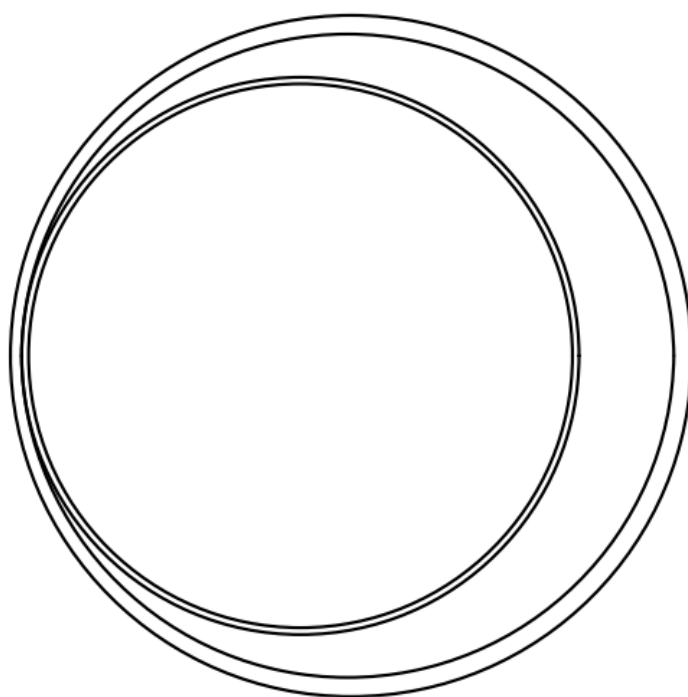
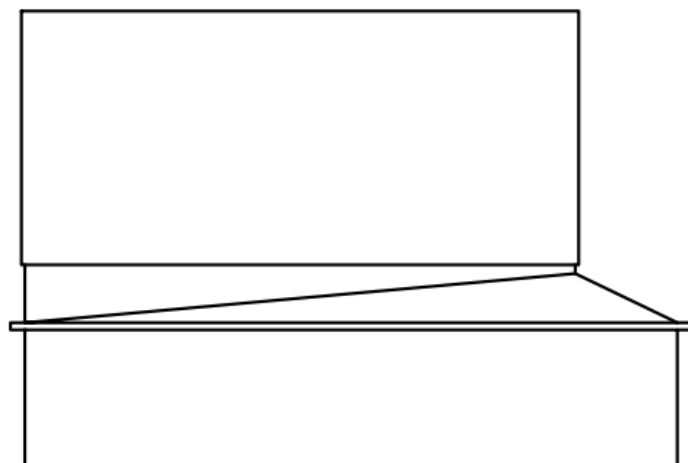


	<i>Dibuixat</i>		
	<i>Data</i>	18/07/2020	
<i>Escala</i> 1:1	TAPA		Treball de recerca

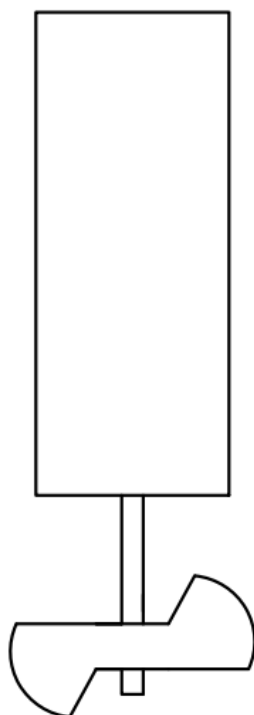
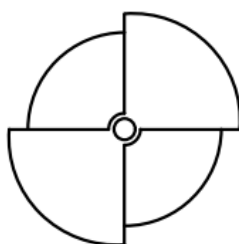


	<i>Dibuixat</i>		
	<i>Data</i>	24/07/2020	
<i>Escala</i> 2:3	ENTRADA D'AIGUA POSTERIOR		Treball de recerca

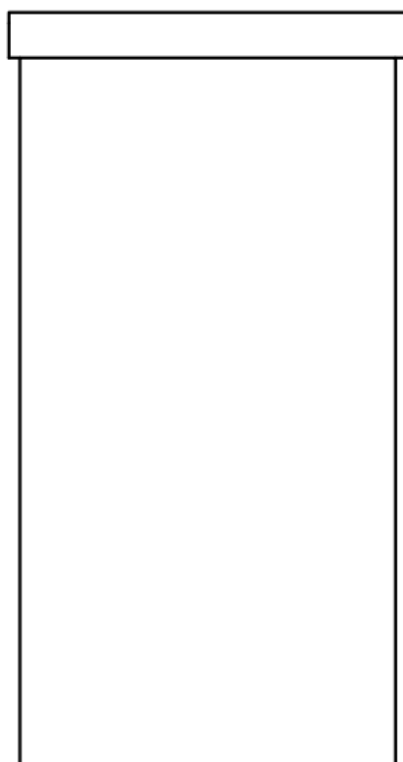
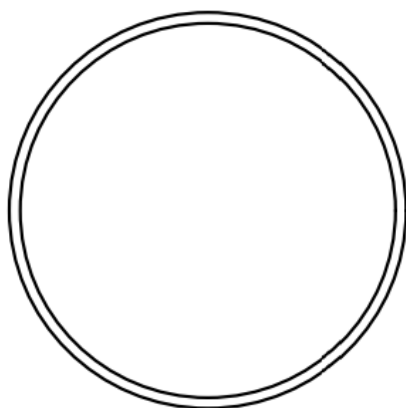




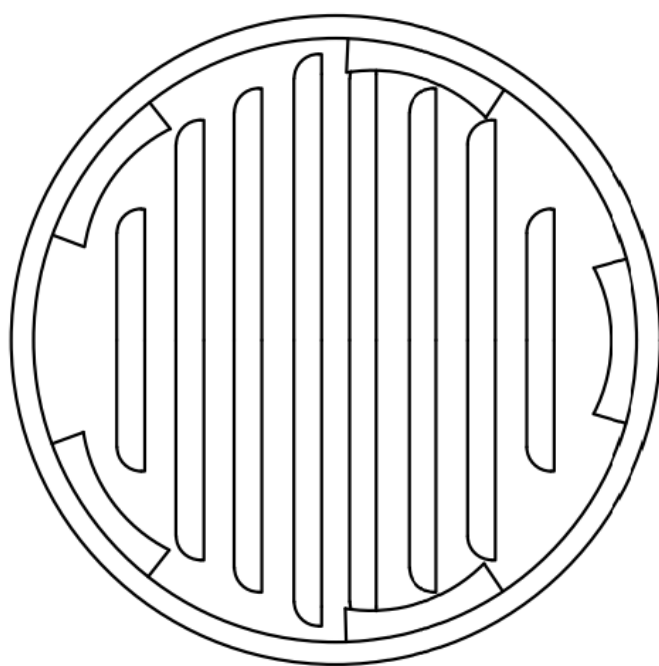
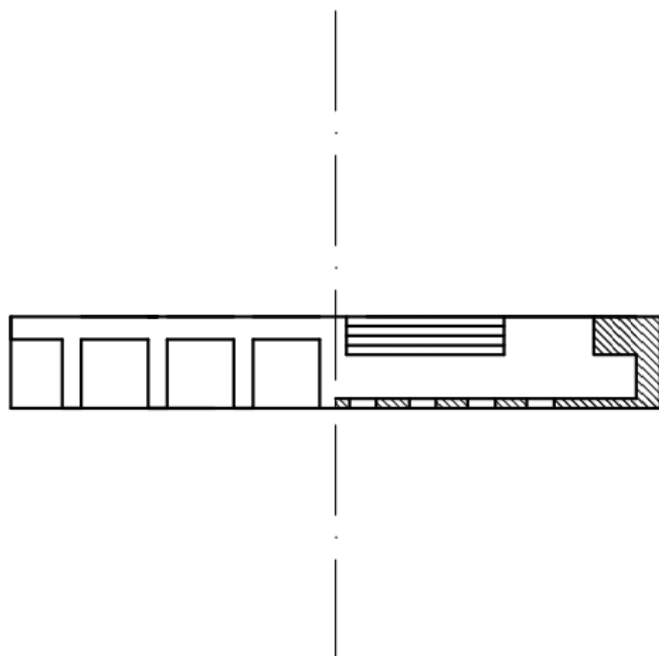
	<i>Dibuixat</i>		
	<i>Data</i>	31/07/2020	
<i>Escala</i> 2:3	ELEVACIÓ D'ENTRADA POSTERIOR		Treball de recerca



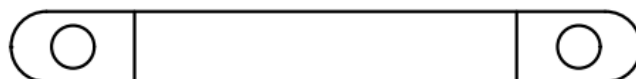
	<i>Dibuixat</i>		
	<i>Data</i>	5/08/2020	
<i>Escala</i> 1:1	PROPULSOR		Treball de recerca



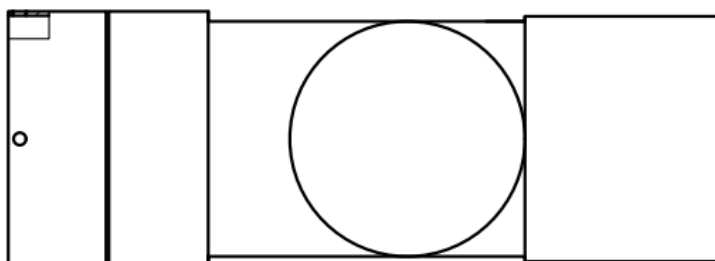
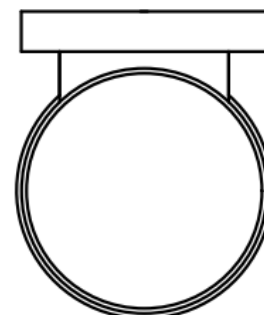
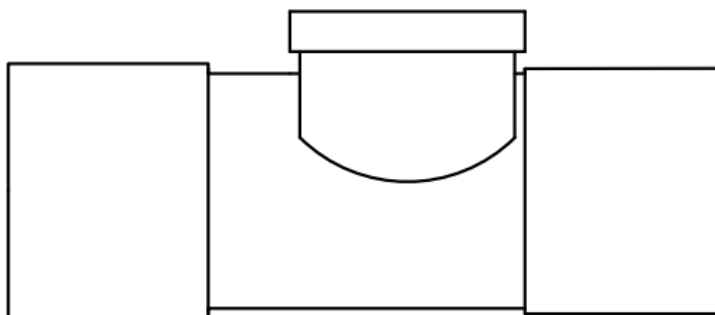
	<i>Dibuixat</i>		
	<i>Data</i>	13/08/2020	
<i>Escala</i> 2:1	PROTECTOR DEL PROPULSOR		Treball de recerca



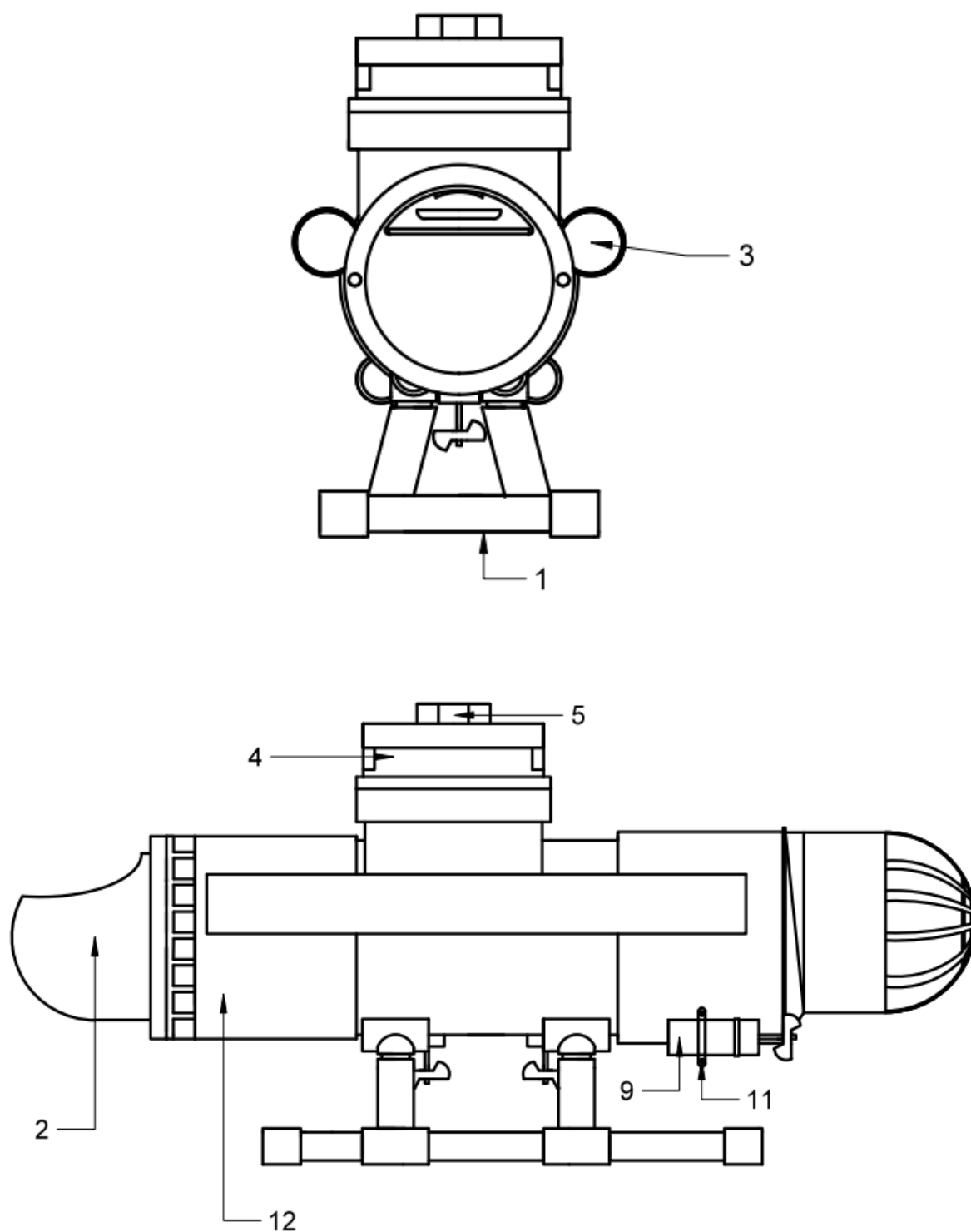
	<i>Dibuixat</i>		
	<i>Data</i>	14/08/2020	
<i>Escala</i> 2:3	REGULADOR D'ENTRADA D'AIGUA		Treball de recerca



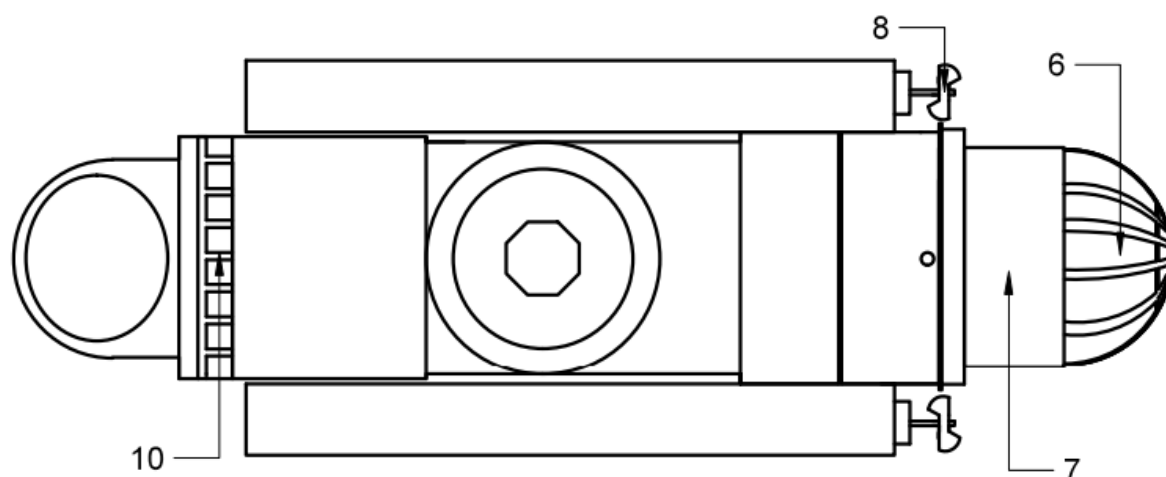
	<i>Dibuixat</i>	-	
	<i>Data</i>	19/08/2020	
<i>Escala</i> 2:1	ABRAÇADORA		Treball de recerca



	<i>Dibuixat</i>		
	<i>Data</i>	21/08/2020	
<i>Escala</i> 1:4	ESQUELET CENTRAL		Treball de recerca



	<i>Dibuixat</i>		
	<i>Data</i>	2/09/2020	
	CONJUNT		Treball de recerca



1	SUPORT		1
1	CAPÇAL DE PROA		2
2	TUB LATERAL		3
1	CÀPSULA ESTANCA		4
1	TAPA		5
1	ENTRADA D'AIGUA POSTERIOR		6
1	ELEVACIÓ D'ENTRADA POSTERIOR		7
4	PROPULSOR		8
2	PROTECTOR DEL PROPULSOR		9
1	REGULADOR D'ENTRADA D'AIGUA		10
4	ABRAÇADORA		11
1	ESQUELET CENTRAL		12
NOMBRE DE PECES		DENOMINACIÓ	MARCA
	Data	Nom	Cognom
Dibuixat	2/09/2020		
CONJUNT			Batxillerat Científicotecnològic
			Treball de recerca