ROBOTIC CANNON

Miquel Oller Oliveras

Tutor: Ticià Sala Mellado

Date: 14-12-12

Table of contents

1 Introduction

Since I was a child, weapons have fascinated me. I used to play a lot with guns when I was a child and I am a fan of action films, so my idea and desire had been to make something which would be able to shoot. On the other hand the robotic world was something new for me, but was something I wanted to learn about. During the project, I have been interested and fascinated with the robotic world which probably will be an important part of our future.

One of my main motivations has been to take the research project as a hobby. As the topic was something what I liked, it had been something pleasant and not heavy.

Another important thing is that I decided to write the research project in English because it is a way of learn two things at the same time. I have learned to program robots and electrical appliances and I have practised English too. Also, English is very important in the Technology world, the majority of the resources I found were in English, so it is something to take into account.

After thinking and investigating a lot (and after some previous types of projects) I decided to build a paint ball robot. That is, a wirelessly controlled fully moving robot with paint ball gun. There are a few competitions like fighting robot competitions but with paint balls. I saw some videos of these competitions and they were amazing. Without hesitation I started to search how to make one.

I have structured my project on three basics parts. The introduction where I introduce the project, the main part where I explain my project and all the things I have done and finally the conclusions.

I wrote my project in an easy format in which I have written the basic things. I have not written the definitions and the complex operation of everything to simplify it.

2 Acknowledgements

I would like to thank some people who have helped me and without them this project would not be possible.

First of all thanks to Ticià Sala, my project tutor who has helped me in many things relating to the project and if I have had a problem he has helped me.

Thanks to my parents who have supported me, and provided me with everything that I needed and have not got annoyed with all the space I have needed at home.

Thanks to the technology department who have let me use their tools and facilities and have helped me a lot.

Thanks to Marc Broderick who has helped me in the writing part and without him this project would be full of spelling mistakes.

Thanks to Jeremy Blum and his Arduino tutorial series which had been vey useful for my project.

Thanks to Bill Porter who developed a library to read a PS2 controller.

Thanks to Fusteria Pladevall who has given me the wooden pieces.

Thanks to Intactics Paintball.

Thanks to Cookinghacks, Pololu electronics and the Sentry project.

Thanks also to Jordi Gallifa who gave me the Play Station 2 controller .

3 Objectives

My Objectives, since I have started with the project have been to build a robot or a structure that is able to shoot. Another objective was to do something that is related to the robotic world and the amazing world of technology.

One of my main motivations has been to take the research project as a hobby. As the topic was something that I liked, it was easier for me to work on the project than if it was just an obligation.

I have tried to learn as many new things I can and try to use and apply the things I have learnt in class.

I decided to work alone because in my opinion, it is a way to learn more and you have to work more too. Another important point is that working alone has the advantage that you do not depend on another person to work, so it is easier to start working.

4 The evolution of the project:

4.1 My first project: A tennis ball robot

My first idea was to make a robot that could shoot tennis balls, like a tennis ball machine, but the difference between them would be that my robot would detect a sensor put somewhere and it would shoot there automatically. To detect the objective I decided to use some distance sensors which would send the distance between my robot and the objective to the micro controller, so I made an equation using the parabolic shooting equation to know the angle to hit the objective knowing the distance between them and the initial velocity of the ball (also the difference of height but I imagined that they would be at the same level) (You can find a table which calculates the angle of shoot on the Annex C). To read the distance between the objects with the Arduino there are basically two different types of sensors: the infra-red sensor (Picture 1)and the ultrasonic sensor (Picture 2). The infra-red sensors are cheaper than the ultrasonic but their range is quite short. So if I wanted to detect the objective in a wider range I would have to use an ultrasonic sensor.

Picture 2: Infra-red distance sensor

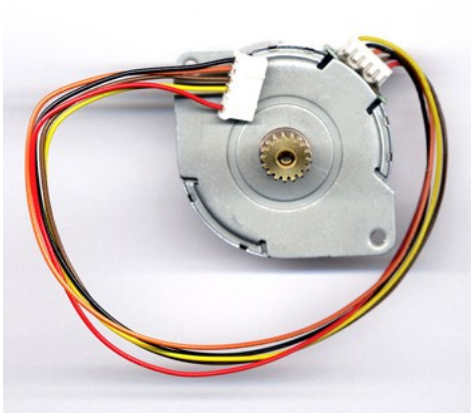
Picture 1: Ultrasonic sensor

To control the angle of the shot, Ticià recommended me to use a stepper motor (Picture) because are the most accurate motors and if I wanted to raise the objective

and not fail the shoot, I would have to use this kind of motor.

Another important thing to calculate was the direction of the shot, because I wanted the robot to detect and have a shooting range of 360º, so the robot had to be a full rotation robot and the sensor had to be able to detect the objective in the same range. Ticià proposed using an ultrasonic sensor that rotates like sonar. It was a good idea and it would have been cheaper but it wouldn't have been able to distinguish the objective from any other object in its range. Hopefully, I could find a sensor that worked in pairs and could detect the direction of the other sensor using infra-red signals (picture 3). This sensor is made for robots which use it to detect the direction of the other robot and move towards it. My idea was in order to detect the direction of the objective I would use the 360º ir beacon sensor and to know the distance between my robot and the objective I would use the ultrasonic sensor that would send the value to the micro controller Arduino that would calculate the angle of the shot and would send it to the steppers.

Picture 3: 360º infra-red sensor “ir beacon 2”



Picture 4: Stepper motor

The problems with this project were that if I wanted to detect the objective, I would have to combine both sensors and it could not differentiate the objective from other objects that were around, and if the objective was behind something, it could not detect it.

So, because of all these problems, I saw that if I wanted to do it, I would have to simplify it, so I changed some things

4.2 My second project: The evolution of the tennis ball robot

To To make it work, I saw that I had to change some things, basically the detection system. I decided that the best I could do was remove the detection part and transform

it into a tennis ball machine that could shoot tennis balls like a professional tennis ball machine (Picture 5). I started to look for information about how this type of machine works and how I could make one. My intention was to make it programmable, so I wrote a program with Arduino that could control the shot and use it to play against the robot.

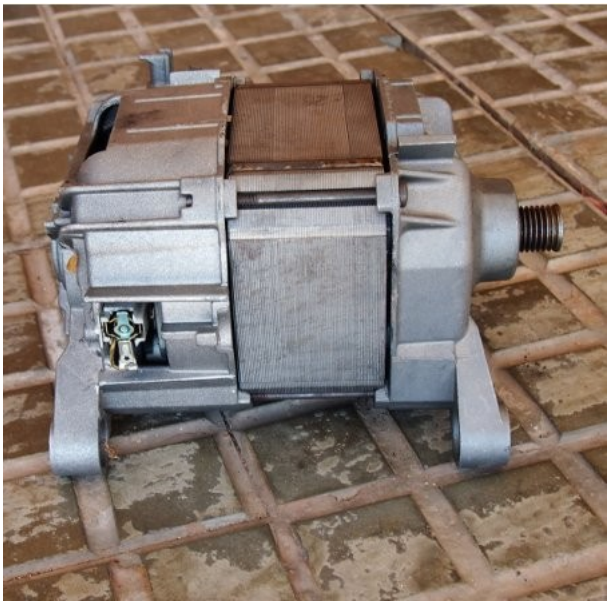
I found some interesting videos on YouTube in which somebody had build a homemade tennis ball machine. The majority of them used two wheels that rotated in opposite directions and when the ball passed between the wheels, it is shot.

I talked with Ticià and he recommended me to look for an electric motor to power the wheels in the scrap-yard, he told me that I had to look for some appliance like the motor of a washing machine or a dryer.

Picture 5: Tennis ball machine; Lobster elite 2

I went to the scrap-yard in Moià and I talked with the boss, he told me that if I wanted to pick out something from there, I had to talk with Moià's town council, so I went to the local Town Hall. There I talked with the secretary who told me that as it was a case related with the scrap-yard, I had to talk with the environmental manager, who only comes on Wednesdays. I phoned the following Wednesday and I talked with her, I explained my situation and she told me that in this kind of situation I had to make an official request. I took the request form and I wrote down my situation and requested if I could get a motor and some necessary pieces from the scrap-yard. I gave the request to the Town Hall and I waited a week until they phoned me. The environmental manager told me that to get the authorization, first of all Moià's town council had to meet up and deliberate about it and after that they had to deliberate it with Bages's council, so she told me that it was going to take at least a month. As I was not available to wait a month, I talked with a local technician who repairs electrical appliances and he

generously gave me a washer motor (Picture 6). I tried to make it work, but I could not read the reference code because it was not written clearly, I did not know the polarity, the function or the voltage that it needs. So it would be quite complicated to use it, and if I had used it, it would have needed a lot of work, some expensive parts and a lot of giant pieces to build the tennis machine.



Picture 6: The electric motor which I received.

4.3 My final project: A paintball robot

Taking into account these new problems, I saw that it was not possible to make the tennis machine as I wanted to. It would have been too big and too expensive, so I started to look for a solution to my problems. Surfing the net for looking alternatives for my project, I decided that my robot had to be able to move, which would make it more fascinating and interesting but also more complicated because now I had to control more things such as the direction, the forward and backward movement and it could not be plugged into a socket as it could not move freely.

I started to think about it and I decided that the best thing would be take out the tennis machine motor and structure of shooter and look for an alternative that does not needed

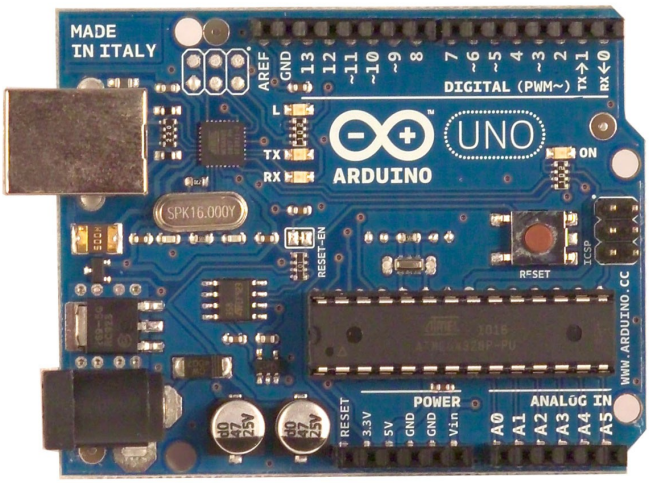
to be plugged. Some alternatives came to my mind, but the best and the one which I found the most exciting was for shooting, to use a paintball marker. Using the paintball marker I do not have to plug it in for shooting because the only energy that it needs is something that presses the trigger and a bottle of compressed air, which is totally portable.

I looked for other paintball robots on the internet to see how they worked and to get some ideas; I found some interesting pages and a company that makes these robots, but I got scared when I saw the price, they cost around two thousand dollars. (you can find the webs links on the annex A)

Picture 7: A paintball robot

5 The Arduino work

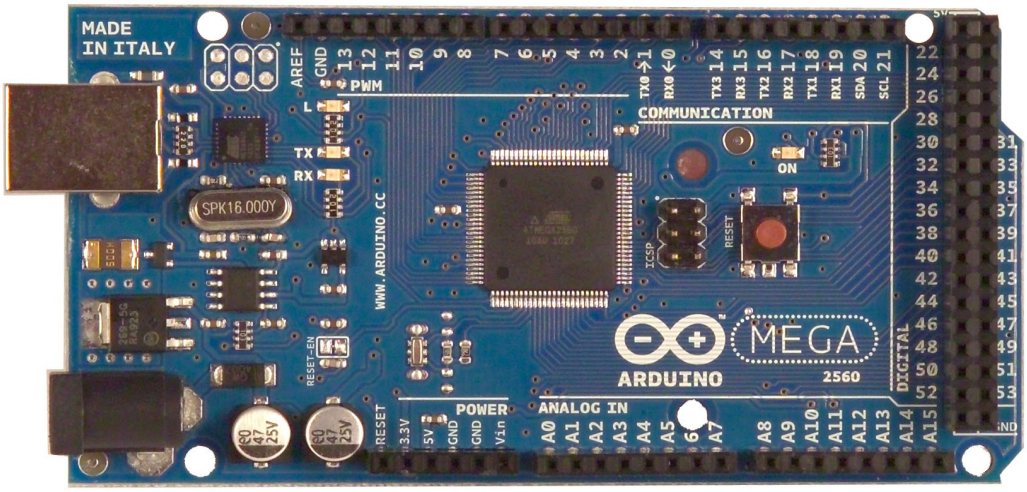
To make the robot work I had to use a micro-controller that would work as the brain of the robot. The best option was to use an Arduino. Arduino is an open source that is used around the world to control prototypes, electronic circuits, robots... and its main characteristic is that as it is an open source, all the contents and programs are public domain, so you can find a lot of examples and sketches on the internet. The most common Arduino board is the “Arduino UNO” (picture 8).



Picture 8: The "Arduino UNO board"

One of the first things that I did was to buy some initiation electronic pieces and learn the electronics basics things. I looked up some internet pages of Spanish electronic shops such as Cooking Hacks, BricoGeek and Barcelona Cybernetics. I bought an Arduino board, especially the “Arduino MEGA 2560” (picture 9) which is bigger than the “Arduino UNO”, it has more pins and it has double PWM (Pulse-width modulation) pins. I decided to buy the Arduino MEGA because as I wanted to make a robotic cannon I needed a lot of pins (for the servos, the drivers, the communication system...), so as I

didn't want to run out of pins I decided to get the MEGA.



Picture 9: Arduino MEGA 2560

Also I bough some basic gear to get start like some jumpers, a breadboard, some colour LEDs, a servomotor, some potentiometers, some button switches, a power supply to power the Arduino board... When it arrived (it arrived three weeks after the order because they had to backorder some products) I started work and I create some simple examples like blink a LED or control a servo using a potentiometer. To learn how to program and how it works I looked up the internet for some tutorials and examples. I found a very useful tutorial series on YouTube made by Jeremy Blum where in 15 tutorials he explains in a very understandable way how the Arduino program works and some tricks (you can find the web links on the bibliography and on the annex A). The Arduino program is based on the C++ language which makes the programming process easier.

I spent so much time working with Arduino and I made a lot of different programs, I found it fun and awesome so I took it as a hobby. (You can find all the testing programs on the annex E)

When I decided to change the way of the shot, I decided to control the robot wirelessly so I had to find a way to control the Arduino wirelessly. After searching a bit, I found the solution, the Xbee modules (Pictures 10 and 11). Xbee is a module for the Arduino

which allows it to communicate via serial wirelessly. But for the Arduino to communicate with the Xbee module you need two Xbee modules and two Arduino boards (or an Arduino and a USB adaptor if you want to control the Arduino by the computer), so I had to buy at least two Xbee modules. Another problem was that the Xbee modules and the Arduino board have different pin separation size, so to connect them together I found an adapter shield made specifically to connect them.

Picture 10: An Xbee module with the arduino adapter shield.

Picture 11: Xbee module.

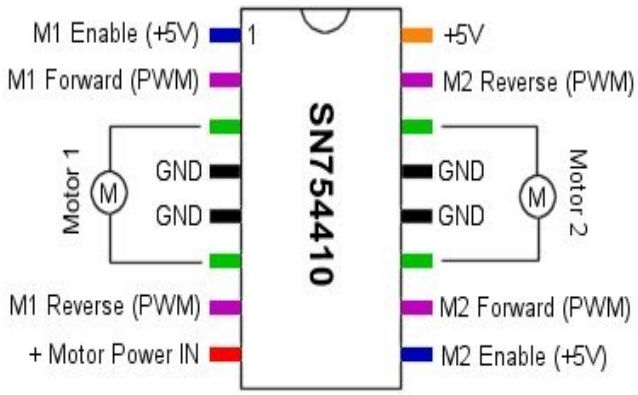
There are three different types of Xbee modules which are classified in three different grups: Series 1 (also known as 802.15.4), Series 2 (also known as ZigBee) and DigiMesh 900. The first two are the most used, the only difference between them is that they have a different way of communication. Series 1 uses a point to point communication, but Series 2 can also use a Mesh communication. Series 2 is more difficult to program because you have to assign the modules as a coordinator, as a router or as an end device which makes the programming process a bit more complicated. I decided to buy the Xbee Series 2 because both of them are the same price and it is more functional.

6 The construction process

6.1 My firsts prototypes:

As with almost all projects I started the construction with a prototype. What I did was take a radio-control car, which I used to play with when I was a child, and I disassembled it piece by piece. I made a simple program with the Arduino that it controlled four micro motors which I bought in cooking hacks and I controlled them using an H-bridge driver from Texas Instruments, the SN754410NE (Picture 12). This driver is an easy and very cheap way to control two or more DC motors, what I did was to put them in a breadboard to use it as a driver board. The SN754410NE driver has sixteen pins and you can see in Picture 13 their function. Pins 4, 5, 12 and 13 are connected to Ground, pins 1 and 16 are the power supply for the motors, pins 3 and 6 are the connections to the motor 1 and pins 11 and 14 are the connections to the motor 2. Pins 8 and 9 work as a power supply for the driver.

Picture 12: Driver SN754410NE from Texas Instruments.

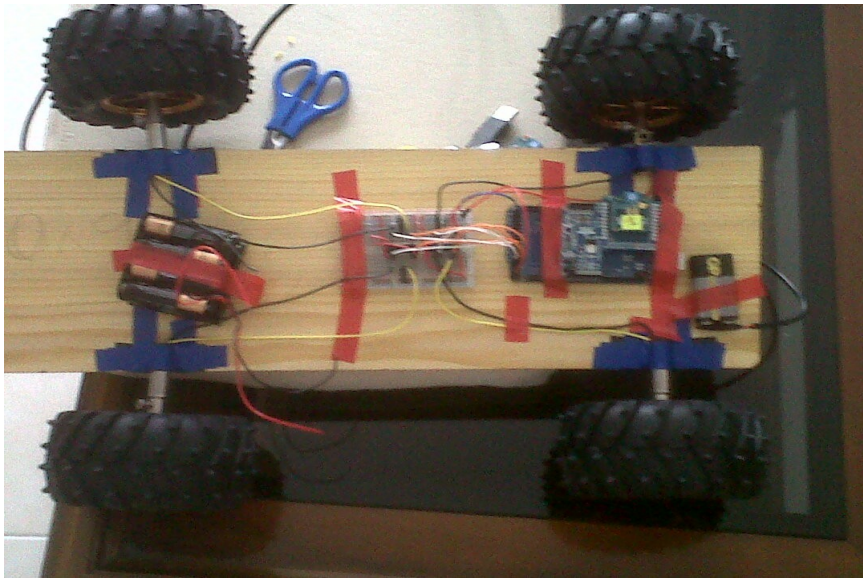


Picture 13 SN754410NE connections scheme.

The problem of my first prototype was that I could only attach my motors with tape and it was not the best way to build it. Another problem was that the motors which I bought (Picture 14) did not have enough power so it was a completely failure.

Picture 14: DC gear motor 6v 60rpm Dustproof

I built another prototype (picture 15) sticking the motors on a piece of wood. It could move slowly forward and backward but it could not turn around.



Picture 15: My second prototype

6.2 Building the chassis

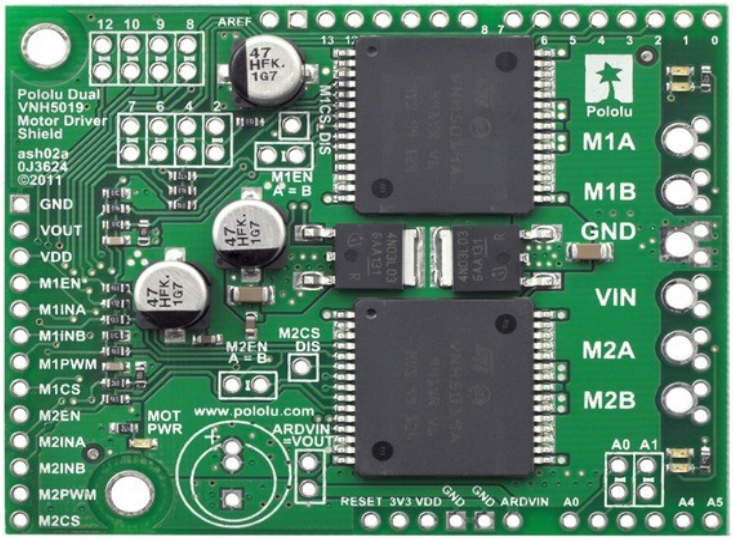
I saw that that motors did not have enough power to make the robot move, so I looked for more powerful ones. I found a mounted chassis called Wild Thumper (Picture 16) which is a 6-wheel chassis with very powerful motors and fully articulated wheels. I saw some videos (you can find them on the annex) about this chassis and I was impressed. The problem was that it is quite expensive (it costs around 250€), so I decided to get only the motors because it has the same wheels as the ones I had bought, and make a less expensive version..

Picture 16: Wild Thumper

I looked for the motors and I found that the nearest place that sells them was Pololu Electronics which is in the United States. I had another problem, which was that these motors (Picture 17) need a high intensity current. They work on 6v and a very high intensity of 6A, but they are very powerful.

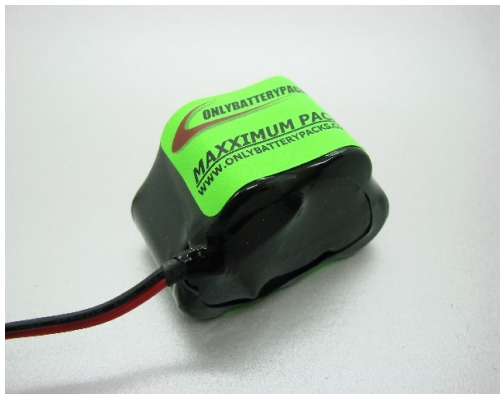
Picture 17: Pololu Gearmotor 6v

So I had another problem because the drivers I had were not able to withstand intensity higher than 1A. Luckily I found a driver that fitted my project perfectly on the same Pololu website. Its name is Pololu dual VNH5019 motor driver shield (Picture 18), it is completely compatible with the Arduino and it is able to stand a maximum intensity of 24A, just what I needed.



Picture 18: Pololu dual VNH5019 motor driver shield

I also bought a high torque servomotor, another Arduino board to connect the controller and some motor brackets to fix the motors. Unfortunately Spanish customs agents retained my items and I had to pay around 60€ of taxes. At the same time I ordered a battery pack to power the motors. I found another American company called Onlybatterypacks.com which sells worldwide all types of battery packs for radio-control toys. I ordered a NiCad 6v battery which has a capacity of 5Ah and a discharge range of 20A.

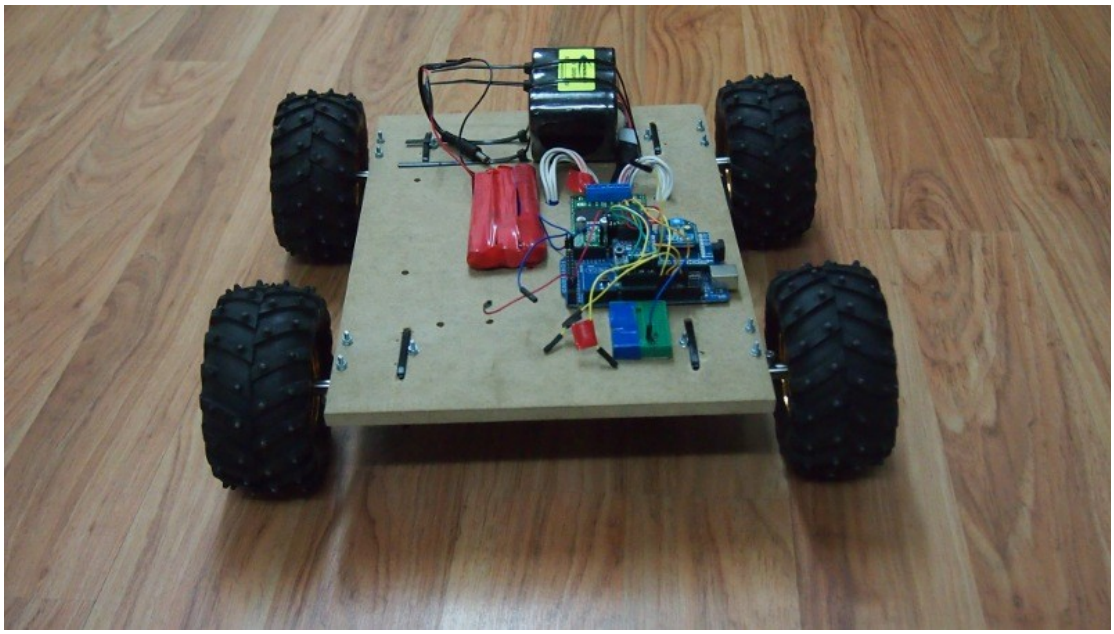


Picture 19: My 6v battery pack

When I had all what I needed I started to build the chassis. I got some pieces of wood from a local carpentry workshop and some screws. I bore some holes in the wood and I screwed the motors and the basic things there.

Picture 20: Making holes

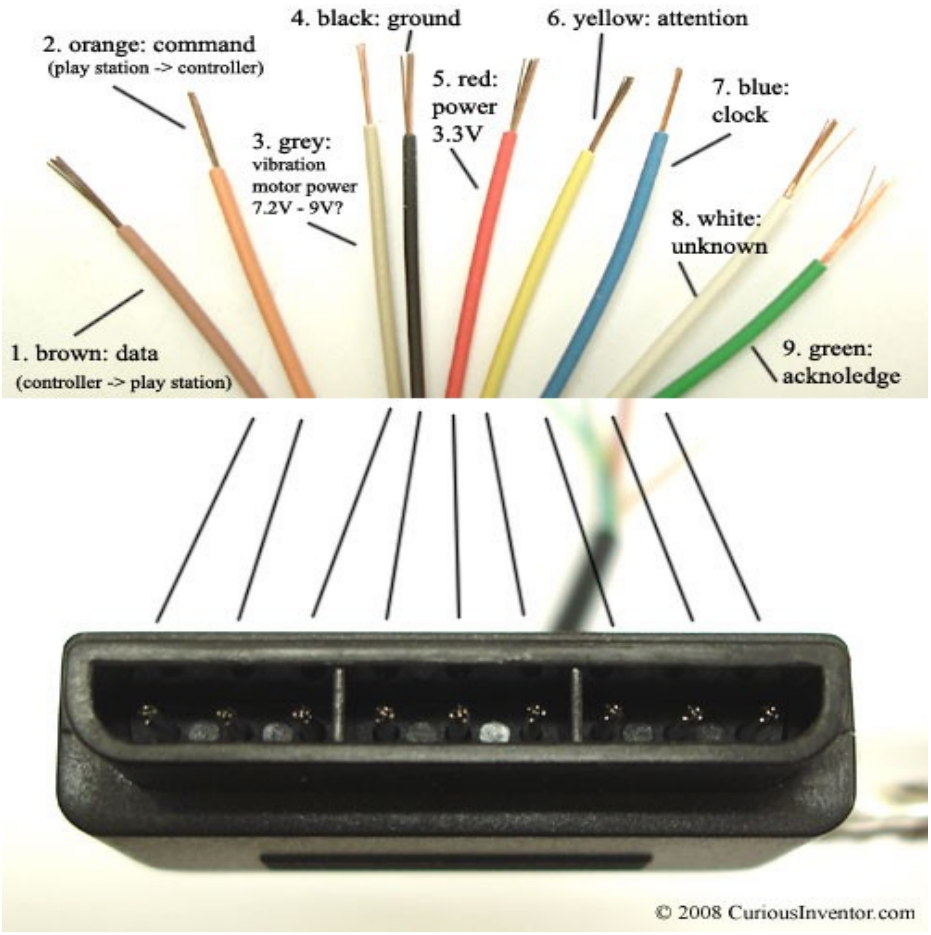
The result was a completely success (Picture 21). It could move very fast and it was powerful. The only problem was that the battery was quite heavy.



Picture 21: the robot base

6.3 The remote controller

At the same time I worked on the remote controller. I decided to use a Play Station 2 controller to control the robot, but as you know, PS2 controllers have a wired connection. I found a useful web page made by Bill Porter where he explains how to connect a PS2 controller to the Arduino. I cut the wire and I soldered the wires inside to another wire pluggable to the Arduino. I downloaded an Arduino library which allows and makes it easier to connect the PS2 controller to the Arduino (you can see I Picture 22 the wire connection). The brown, orange, yellow and blue wires are connected to Arduino digital pins, the red wire is connected to the 3.3v – 5v power output and the black wire is connected to ground. Green, grey and white wires had other functions that I do not use. (you can see the final result in Picture 23)

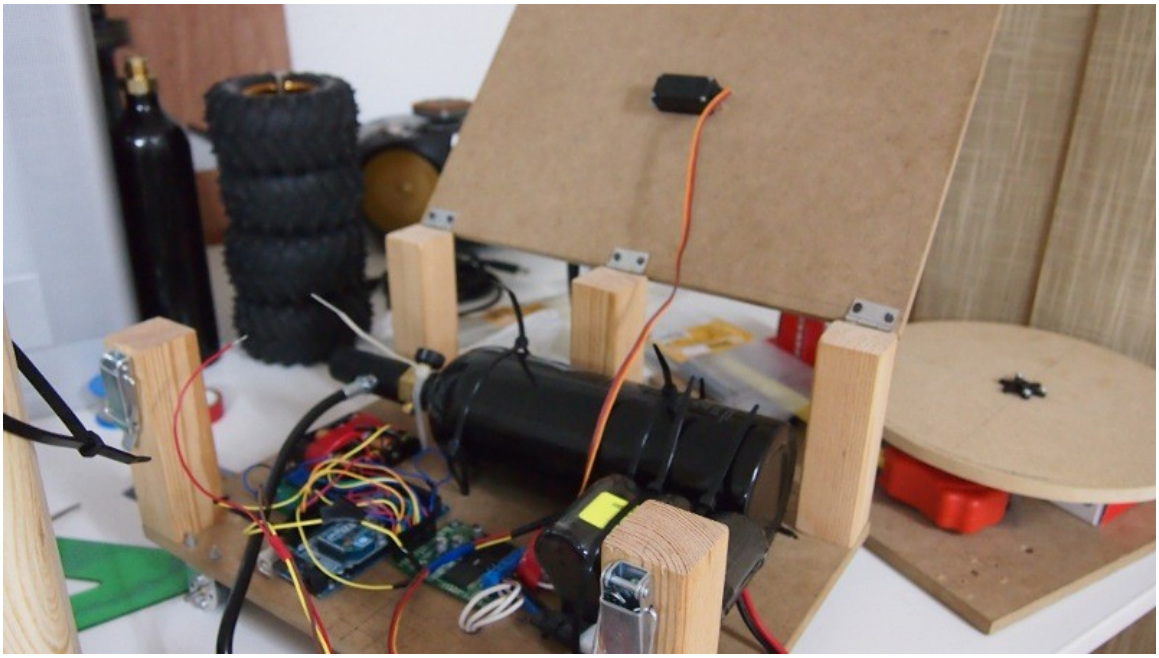


Picture 22: PS2 wired connection

Picture 23: The PS2 controller connected to the Arduino

6.4 Building the turret

To build the turret, first of all I mounted a platform on the base using some wooden pillars and using hinges I opened it like a box. After that, I made a hole with a saw in the middle of that base and I attached a servomotor. The servomotor controls the rotation of the turret.



Picture 24: The basic structure opened

I screwed a circular base at the top of the servo. This base rotates with the servo and on top of it will be the paintball gun. To hold better the gun first of all I sticked some little wooden pieces between the top base and the circular bases, but the problem was that it had to much friction and it could not move properly. To solve it I search in internet for a solution and I found some transporting balls (Picture 25). The company which makes them is from Barcelona so I went there and I bough some. This balls helps the servo to rotate and holds an important part of the turret's weigh.

Picture 25: Carring balls

To hold the gun I needed some strong supports because of its weight. I decided to hold the paintball gun with two wooden columns to equilibrate it. To control the inclination of the gun I decided to use another servomotor which connects directly to the gun.
I cut two equal pieces of wood and I made some holes and with some big screws I held the gun between them.

Picture 26: The turret structure

6.5 The shoting system:

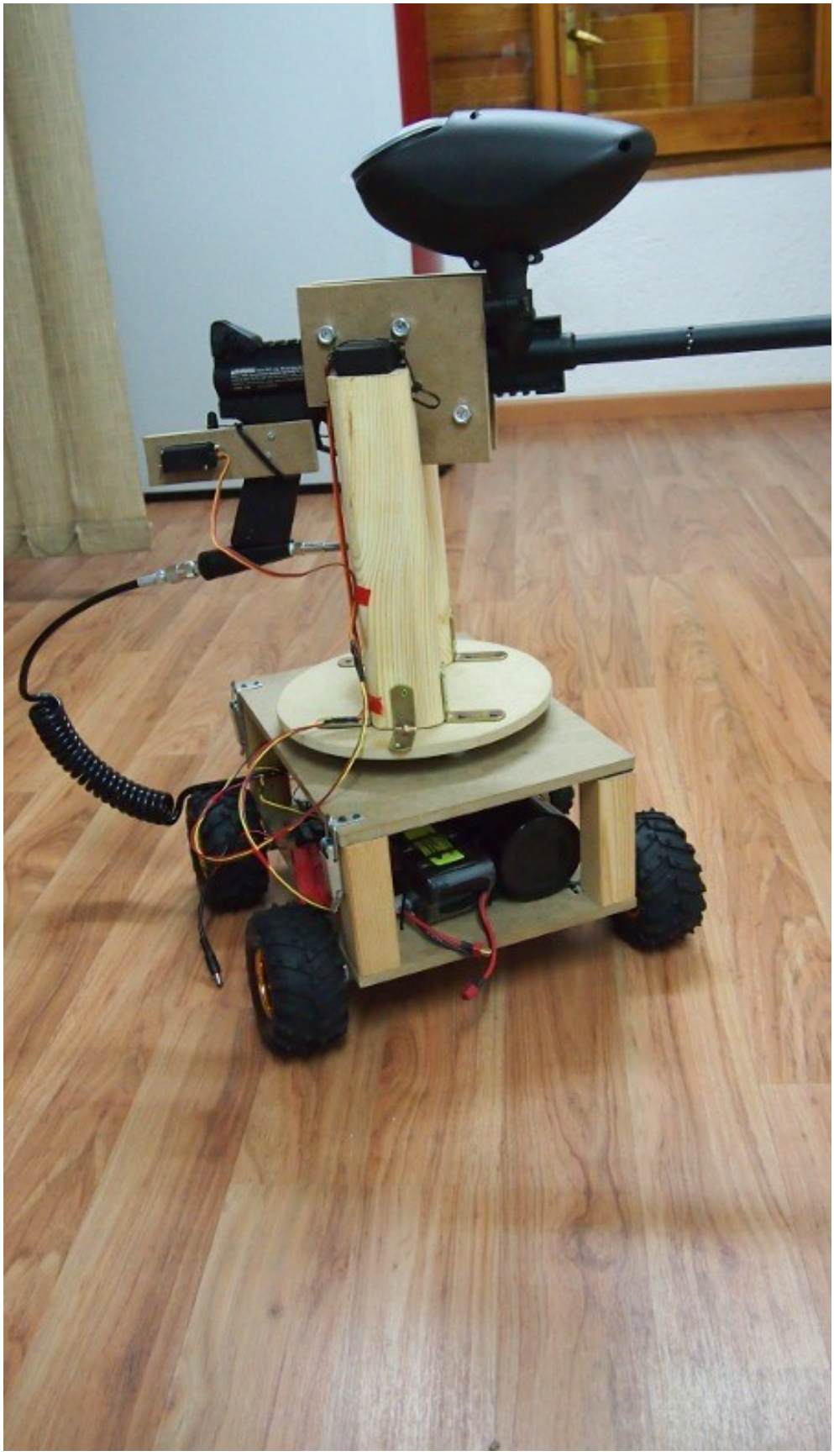
With another little piece of wood I made a simple shooting system. It is based on a servo motor which is attached to the trigger with a piece of metal. When the servo is working, the piece of metal squeezes the trigger making it shoot.

Picture 27: The shooting system. The paintball gun with the servo.

To shoot, the robot needs two thing: Paintball balls as ammunition and O2 or CO2 compressed. The compressed gas is stored on a tank which is in the base structure. The gas goes to the gun through a spiral conduct which is dirctly connected to the gun.

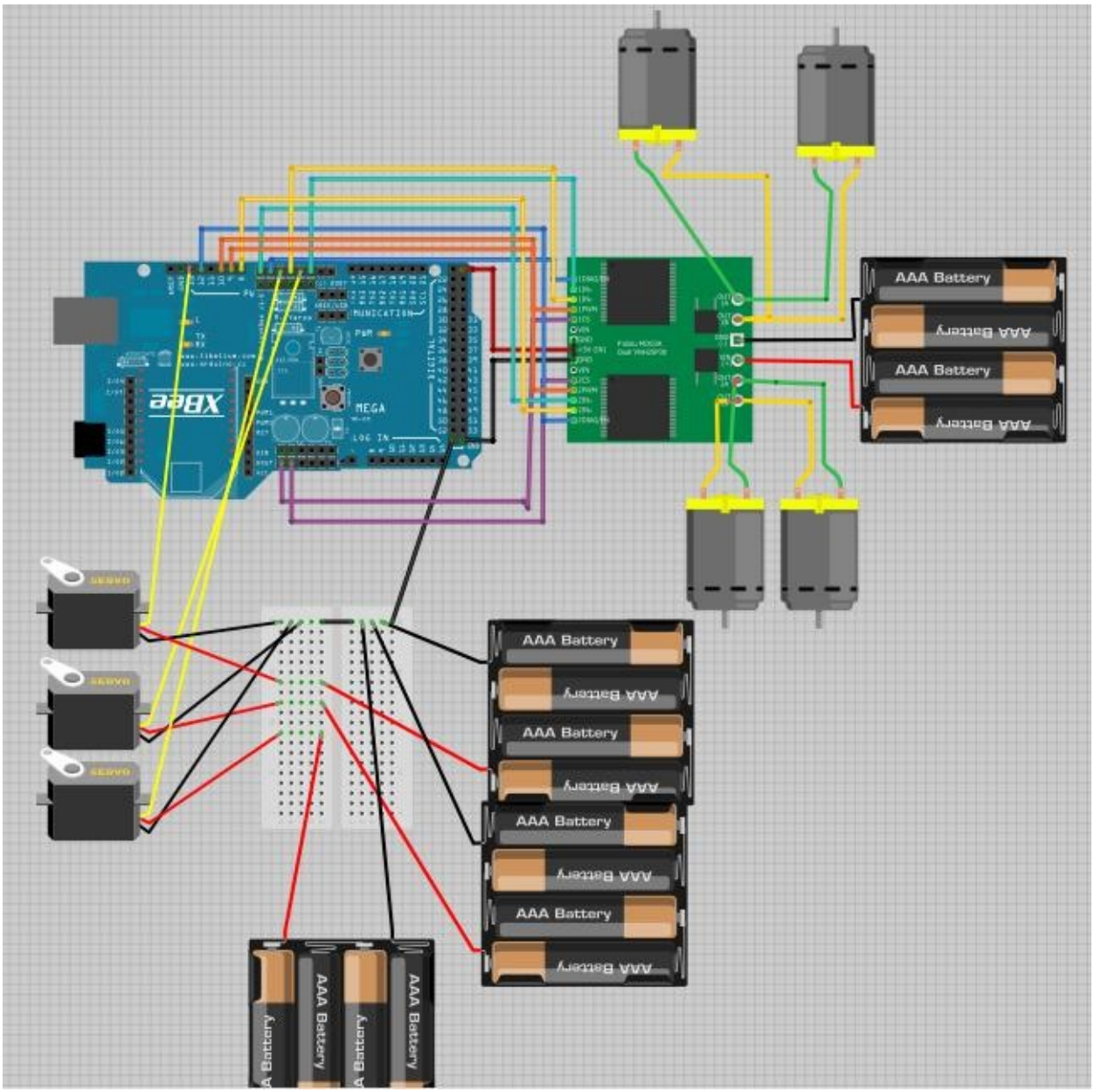
6.6 The project ended:

After all the previous steps, I put all the things together and I the robot was ended (picture 28). You can find more photographs on the annex D.



Picture 28: The robot ended.

6.7 The electronic scheme:



Picture 29: The electronic scheme.

7 Program sketch:

7.1 How does it work?

My program sketch is divided in two, one for the Arduino connected to the controller (transmitter) and another one for the robot's Arduino (receiver).

In both of them I used the Arduino Serial communication which allows Arduinos or an Arduino talk to a computer. In my case, I used an Xbee module to transmit data wirelessly between the Arduinos.

The transmitter module reads the information which comes from the PS2 controller through the four wires (attention, command, clock and data) and depending on the button pressed it sends a letter or another. The problem with the Serial communication is that you can only send single letters through it, so I had to encrypt (assign a letter to each one) the communication.

The receiver reads the information sent for the other module and depending on it, makes one thing or another. I made the movement of the robot be controlled through the joysticks. It has four different types of speed. To control the gun I controlled three servos through the pad and the button R1 (to shoot is the same as all war games).

You can find both programs for the Arduino platform on the annex B

7.2 The remote control sketch:

```
/************ PROGRAMA COMANDAMENT CANÓ ROBÒTIC TDR ************\
              MIQUEL OLLER OLIVERAS
              -----EMISOR-----
 */

//CABLES MANDO:
   // attention (groc) [10]
   // command (taronja) [9]
   // clock (blau) [13]
```

```
   // data (marro) [12]
   // 5V (vermell)
   // GND (negre)

//LLIBRARIES
#include <PS2X_lib.h>

PS2X ps2x; // crea el mando de la play 2

 // PINS:

 //VARIABLES:
 int error = 0;
byte type = 0;
byte vibrate = 0;

void setup()
{
 // inici de la comunicació serial:
 Serial.begin(57600);

 // CONFIGURACIÓ DE PINS:

  error = ps2x.config_gamepad(13,9,10,12, true, true);   //inici dels pins del comandament:  GamePad(clock, command, attention, data, Pressures?, Rumble?) comprovar l' error
}

void loop()
{
  ps2x.read_gamepad(false, vibrate);
  //PAD
    if(ps2x.Button(PSB_PAD_UP)) {
     Serial.write('U');
    }
    if(ps2x.Button(PSB_PAD_RIGHT)){

     Serial.write('M');
    }
    if(ps2x.Button(PSB_PAD_LEFT)){

      Serial.write('N');
```

```
     }
     if(ps2x.Button(PSB_PAD_DOWN)){

      Serial.write('D');
     }

   //JOYSTICKS:
     //endevant:
    if(ps2x.Analog(PSS_LY)==0 && ps2x.Analog(PSS_RY)==0)
    {
     Serial.write('F');
    }
              if(ps2x.Analog(PSS_LY)>0 && ps2x.Analog(PSS_LY)<=40 && ps2x.Analog(PSS_RY)>0 &&
ps2x.Analog(PSS_RY)<=40)
    {
     Serial.write('c');
    }
    if(ps2x.Analog(PSS_LY)==0 && ps2x.Analog(PSS_RY)>0 && ps2x.Analog(PSS_RY)<=40)
    {
     Serial.write('c');
    }
    if(ps2x.Analog(PSS_LY)>0 && ps2x.Analog(PSS_LY)<=40 && ps2x.Analog(PSS_RY)==0 )
    {
     Serial.write('c');
    }
            if(ps2x.Analog(PSS_LY)>40 && ps2x.Analog(PSS_LY)<=80 && ps2x.Analog(PSS_RY)>40 &&
ps2x.Analog(PSS_RY)<=80)
    {
     Serial.write('b');
    }
     if(ps2x.Analog(PSS_LY)==0 && ps2x.Analog(PSS_RY)>40 && ps2x.Analog(PSS_RY)<=80)
    {
     Serial.write('b');
    }
     if(ps2x.Analog(PSS_LY)>40 && ps2x.Analog(PSS_LY)<=80 && ps2x.Analog(PSS_RY)==0)
    {
     Serial.write('b');
    }
           if(ps2x.Analog(PSS_LY)>80 && ps2x.Analog(PSS_LY)<=127 && ps2x.Analog(PSS_RY)>80 &&
ps2x.Analog(PSS_RY)<=127)
    {
```

```
 Serial.write('a');
}
if(ps2x.Analog(PSS_LY)==0 && ps2x.Analog(PSS_RY)>80 && ps2x.Analog(PSS_RY)<=127)
{
 Serial.write('a');
}
if(ps2x.Analog(PSS_LY)>80 && ps2x.Analog(PSS_LY)<=127 && ps2x.Analog(PSS_RY)==0)
{
 Serial.write('a');
}

 //enderrera:
if(ps2x.Analog(PSS_LY)==255 && ps2x.Analog(PSS_RY)==255)
{
 Serial.write('B');
}
      if(ps2x.Analog(PSS_LY)<255 && ps2x.Analog(PSS_LY)>=215 && ps2x.Analog(PSS_RY)<255 &&
ps2x.Analog(PSS_RY)>=215)
{
 Serial.write('f');
}
if(ps2x.Analog(PSS_LY)==255 && ps2x.Analog(PSS_RY)<255 && ps2x.Analog(PSS_RY)>=215)
{
 Serial.write('f');
}
if(ps2x.Analog(PSS_LY)<255 && ps2x.Analog(PSS_LY)>=215 && ps2x.Analog(PSS_RY)==255)
{
 Serial.write('f');
}
      if(ps2x.Analog(PSS_LY)<215 && ps2x.Analog(PSS_LY)>=175 && ps2x.Analog(PSS_RY)<215 &&
ps2x.Analog(PSS_RY)>=175)
{
 Serial.write('e');
}
if(ps2x.Analog(PSS_LY)==255 && ps2x.Analog(PSS_RY)<215 && ps2x.Analog(PSS_RY)>=175)
{
 Serial.write('e');
}
if(ps2x.Analog(PSS_LY)<215 && ps2x.Analog(PSS_LY)>=175 && ps2x.Analog(PSS_RY)==225)
{
 Serial.write('e');
```

```
    }
        if(ps2x.Analog(PSS_LY)<175 && ps2x.Analog(PSS_LY)>=129 && ps2x.Analog(PSS_RY)<175 && ps2x.Analog(PSS_RY)>=129)
    {
     Serial.write('d');
    }
    if(ps2x.Analog(PSS_LY)==255 && ps2x.Analog(PSS_RY)<175 && ps2x.Analog(PSS_RY)>=129)
    {
     Serial.write('d');
    }
    if(ps2x.Analog(PSS_LY)<175 && ps2x.Analog(PSS_LY)>=129 && ps2x.Analog(PSS_RY)==255)
    {
     Serial.write('d');
    }

     //dreta:
    if(ps2x.Analog(PSS_LY)==0 && ps2x.Analog(PSS_RY)==255)
    {
     Serial.write('R');
    }
          if(ps2x.Analog(PSS_LY)>0 && ps2x.Analog(PSS_LY)<=40 && ps2x.Analog(PSS_RY)<255 && ps2x.Analog(PSS_RY)>=215)
    {
     Serial.write('i');
    }
    if(ps2x.Analog(PSS_LY)==0 && ps2x.Analog(PSS_RY)<255 && ps2x.Analog(PSS_RY)>=215)
    {
     Serial.write('i');
    }
    if(ps2x.Analog(PSS_LY)>0 && ps2x.Analog(PSS_LY)<=40 && ps2x.Analog(PSS_RY)==255 )
    {
     Serial.write('i');
    }
          if(ps2x.Analog(PSS_LY)>40 && ps2x.Analog(PSS_LY)<=80 && ps2x.Analog(PSS_RY)<215 && ps2x.Analog(PSS_RY)>=175)
    {
     Serial.write('h');
    }
    if(ps2x.Analog(PSS_LY)==0  && ps2x.Analog(PSS_RY)<215 && ps2x.Analog(PSS_RY)>=175)
    {
     Serial.write('h');
```

```
    }
    if(ps2x.Analog(PSS_LY)>40 && ps2x.Analog(PSS_LY)<=80 && ps2x.Analog(PSS_RY)==255)
    {
     Serial.write('h');
    }
          if(ps2x.Analog(PSS_LY)>80 && ps2x.Analog(PSS_LY)<=127 && ps2x.Analog(PSS_RY)<175 &&
ps2x.Analog(PSS_RY)>=129)
    {
     Serial.write('g');
    }
    if(ps2x.Analog(PSS_LY)==0 && ps2x.Analog(PSS_RY)<175 && ps2x.Analog(PSS_RY)>=129)
    {
     Serial.write('g');
    }
    if(ps2x.Analog(PSS_LY)>80 && ps2x.Analog(PSS_LY)<=127 && ps2x.Analog(PSS_RY)==255)
    {
     Serial.write('g');
    }

     //esquerra:
    if(ps2x.Analog(PSS_LY)==255 && ps2x.Analog(PSS_RY)==0)
    {
     Serial.write('L');
    }
          if(ps2x.Analog(PSS_LY)<255 && ps2x.Analog(PSS_LY)>=215 && ps2x.Analog(PSS_RY)>0 &&
ps2x.Analog(PSS_RY)<=40)
    {
     Serial.write('l');
    }
    if(ps2x.Analog(PSS_LY)==255 && ps2x.Analog(PSS_RY)>0 && ps2x.Analog(PSS_RY)<=40)
    {
     Serial.write('l');
    }
    if(ps2x.Analog(PSS_LY)<255 && ps2x.Analog(PSS_LY)>=215 && ps2x.Analog(PSS_RY)==0)
    {
     Serial.write('l');
    }
          if(ps2x.Analog(PSS_LY)<215 && ps2x.Analog(PSS_LY)>=175 && ps2x.Analog(PSS_RY)>40 &&
ps2x.Analog(PSS_RY)<=80)
    {
     Serial.write('k');
```

```
    }
    if(ps2x.Analog(PSS_LY)==255 && ps2x.Analog(PSS_RY)>40 && ps2x.Analog(PSS_RY)<=80)
    {
     Serial.write('k');
    }
    if(ps2x.Analog(PSS_LY)<215 && ps2x.Analog(PSS_LY)>=175 && ps2x.Analog(PSS_RY)==0)
    {
     Serial.write('k');
    }
          if(ps2x.Analog(PSS_LY)<175 && ps2x.Analog(PSS_LY)>=129 && ps2x.Analog(PSS_RY)>80 &&
ps2x.Analog(PSS_RY)<=127)
    {
     Serial.write('j');
    }
    if(ps2x.Analog(PSS_LY)==255 && ps2x.Analog(PSS_RY)>80 && ps2x.Analog(PSS_RY)<=127)
    {
     Serial.write('j');
    }
    if(ps2x.Analog(PSS_LY)<175 && ps2x.Analog(PSS_LY)>=129 && ps2x.Analog(PSS_RY)==0)
    {
     Serial.write('j');
    }

     //STOP:
     if(ps2x.Analog(PSS_LY)==128 && ps2x.Analog(PSS_RY)==128)
    {
     Serial.write('S');
    }

    //R1
    if(ps2x.ButtonPressed(PSB_R1))
    {
     Serial.write('A');
    }
    if(ps2x.ButtonReleased(PSB_R1))
    {
     Serial.write('Z');
    }
    //R2 i L2

    if(ps2x.Button(PSB_L2))
```

```
    {
     Serial.write('F');
    }
    if(ps2x.Button(PSB_R2))
    {
     Serial.write('B');
    }



   delay(50);
}
```

7.3 The robot sketch:

```
/************** PROGRAMA CANÓ ROBÒTIC TDR **************\
                MIQUEL OLLER OLIVERAS
                -----RECEPTOR-----
 */



// LLIBRARIES:
#include <Servo.h>      //Afegim una llibreria pel control dels servos
#include "DualVNH5019MotorShield.h"   //Afegim una llibreria creada expressament per controlar els motors mitjançant el driver DualVBH5019

DualVNH5019MotorShield md;   //assignem el driver en nom de md (motordriver)i prenem els valors dels pins PER DEFECTE!:
/*   INA1 = pin 2, INB1 = pin 4, EN1DIAG1 = pin 6, CS1 = pin A0,
     INA2 = pin 7, INB2 = pin 8, EN2DIAG2 = pin 12, CS2 = pin A1,
     M1PWM = pin 9, M2PWM = pin 10
*/
//Si un cas hem de canviar els pins, escriure: [on unsigned char = pin ]
/*DualVNH5019MotorShield(unsigned char INA1, unsigned char INB1, unsigned char EN1DIAG1, unsigned char CS1,
```

```
unsigned char INA2, unsigned char INB2, unsigned char EN2DIAG2, unsigned char CS2) */
// M1PWM= pin9 M2PWM= pin10   // aquests no es poden canviar

// VARIABLES:
int serial; //variable anomenada serial per la lectura de la comunicació serial
int i1=90; //variable servo1      */// utilitzem aquestes variables per recordar la posició del servo. Aquests són els valors inicials.
int i2=90; //variable servo2
int i3=90; //variable servo3
int x = 180; //Posició servo 1(SERVO DE DISPARAR ) inicial
int y = 90;//Posició servo 2(SERVO CONTROL GIR BASE) inicial
int z = 90;//Posició servo 3(SERVO CONTROL ANGLE) inicial

Servo servo1; //afegim un servo anomenat servo1 -> servo del gatell
Servo servo2; //afegim un servo anomenat servo2 -> servo control de gir
Servo servo3; //afegim un servo anomenat servo3 -> servo control de l'angle

//PROGRAMARI

//_____________________________________ SET UP
void setup()
{

  // inici de la comunicació serial:
  Serial.begin(57600);
  //configuració dels pins:

 //inici del Driver:
 md.init();
 //Pins SERVO:
servo1.attach(3); // servo1(servo del gatell) al pin 3
servo2.attach(5); // servo2(servo control de gir) al pin 5
servo3.attach(13); // servo3(servo control de l'angle) al pin 13
}

//_____________________________________ LOOP
void loop()
{

  if (Serial.available() > 0) {   //comprovem si hi ha comunicació Serial
  serial = Serial.read();  //anomanem seral als valors que rebem per la comunicació Serial
```

```
      //servo 1:
    if(serial == 'A') x=0;      // si el valor del serial és = a A [que equival al botó R1 del comandament] el servo es mou a la posició 0º
    if(serial == 'Z') x=180;    // si no és igual a A, posició = 180º            <<< x = posició del servo1  >>>
    //servo 2:
    if(serial == 'M') y = i2 - 5; // si els valor del serial és = a M [que equival al botó del PAD DRET] el servo gira 5º a la dreta      inicialment està a la posició intermitja = 90º
    if(serial == 'N') y = i2 + 5; // si els valor del serial és = a N [que equival al botó del PAD ESQUERRA] el servo gira 5º a l'esquerra    <<< y = posició del servo2 >>> <<< i2 = posició anterior del servo >>>
    //servo 3:
    if(serial == 'U') z = i3 - 5; // si els valor del serial és = a U [que equival al botó del PAD AMUNT] el servo de l'angle puja 5º    inicialment està a la posició intermitja = 90º
    if(serial == 'D') z = i3 + 5; // si els valor del serial és = a D [que equival al botó del PAD AVALL] el servo de l'angle baixa  5º a l'esquerra    <<< z = posició del servo3 >>> <<< i3 = posició anterior del servo >>>
    //S2:
      if(y>=180) y=180;  // ens assegurem que tant y com z siguin valors compresos entre [0º, 180º]
      if(y<=0) y=0;
    //S3:
      if(z>=180) z= 180;
      if(z<=0) z= 0;

      //MOVIMENT:--------              M1 = motors del costat esquerra              M2 = motors del costat dret velocitat de 0 a 400
      //endevant:
   if(serial == 'a'){       //controlem la velocitat de gir dels motors: a = 100 endevant / b = 200 endevant / c = 300 endevant i F = 400 endevant
    md.setM1Speed(-100);
    md.setM2Speed(100);
    }
    if(serial == 'b'){
    md.setM1Speed(-200);
    md.setM2Speed(200);
    }
    if(serial == 'c'){
    md.setM1Speed(-300);
    md.setM2Speed(300);
    }
    if(serial == 'F'){
    md.setM1Speed(-400);
    md.setM2Speed(400);
    }
```

```
    //endarrera:
   if(serial == 'd'){       //controlem la velocitat de gir dels motors: d = 100 endarrera / e = 200 endarrera / f = 300 endarrera i B = 400 endarrera
    md.setM1Speed(100);
    md.setM2Speed(-100);
   }
   if(serial == 'e'){
    md.setM1Speed(200);
    md.setM2Speed(-200);
   }
   if(serial == 'f'){
    md.setM1Speed(300);
    md.setM2Speed(-300);
   }
   if(serial == 'B'){
    md.setM1Speed(400);
    md.setM2Speed(-400);
   }
   //dreta:
   if(serial == 'g'){          //controlem la velocitat de gir dels motors, per girar a la dreta els motors del costat dret giren endarrera i els de l'esquerra endevant: g = 100 gir dreta / h = 200 gir dreta / i = 300 gir dreta i R = 400 gir dreta
     md.setM1Speed(-100);
     md.setM2Speed(-100);
   }
    if(serial == 'h'){
     md.setM1Speed(-200);
     md.setM2Speed(-200);
   }
    if(serial == 'i'){
     md.setM1Speed(-300);
     md.setM2Speed(-300);
   }
    if(serial == 'R'){
     md.setM1Speed(-400);
     md.setM2Speed(-400);
   }
   //esquerra:
    if(serial == 'j'){           //controlem la velocitat de gir dels motors, per girar a l'esquerra els motors del costat dret giren endevant i els de l'esquerra endarrera: j = 100 gir esquerra / k = 200 gir esquerra / l = 300 gir dreta i L = 400 girdreta
```

```
    md.setM1Speed(100);
    md.setM2Speed(100);
  }
  if(serial == 'k'){
    md.setM1Speed(200);
    md.setM2Speed(200);
  }
  if(serial == 'l'){
    md.setM1Speed(300);
    md.setM2Speed(300);
  }
  if(serial == 'L'){
    md.setM1Speed(400);
    md.setM2Speed(400);
  }
  //STOP:
  if (serial == 'S'){      // si el serial és igual a S (Stop) (joystics en la posició de repòs) els motors paren -> motor brake = 400 = màxim
     md.setM1Brake(400);
     md.setM2Brake(400);
  }

     //Servo 1:              // escribim la posició en graus en els servos
     servo1.write(x);
     //Servo 2:
     servo2.write(y);
     //Servo 3:
      servo3.write(z);

      i1 = x;                 //per recordar la posició igualem la posició actual a la i del motor que es farà servir per ampliar o reduir el valor de l'angle
      i2 = y;
      i3 = z;
 }

     delay(50);               // fem una petita pausa per que els motors es puguin posar a la posició
}
```

8 Conclusions

8.1 The basic conclusions:

To sum up, the conclusions of my project are that I have been able to work alone and build something by myself. Another conclusion is that I have learned many things about robotics and digital programming, such as how to program and write in Arduino code, I have learned to use C++ language and to use Auto-cad to make a 3D preview of my project. I have compiled a lot of information I have found on the net and I have been able to select the most important parts. Another satisfactory conclusion is that I have discovered a new technology dimension which is the robotic world.

I think that this project has been very useful for me, I have learned a lot of new things and I have built a robot. Even though it has been a bit stressing at the end I think that I have spent a good time on it.

I have invested arround 400 hours in it, a third of them getting started, learning how to program and looking for information and useful products, another third in programming and writing the project and the other third in building the robot more or less.One thing that I am pleased about is that I satarted my project in March of 2012 so I had a lot of time to work and I could distribute my time properly.

Another thing that I have learned is that things usually tand to end different form what you thing or what you expect, so you have to planned carefuly before you start your project and the building part. In my project somethings have gone wrong but luckily I could solucionate them with patience and working hard.

8.2 The project expenses:

ITEM:	PRICE (IVA incuded):
Arduino Mega 2560	45,00€
Arduino Uno R3	23,00€
Leds, buttons, potentiometers, breadboards, jumpers...	40,00€
High-Torque Servomotors	60,00€
Xbee modules	80,00€
Off-road wheels	25,00€
Bridges, diodes, transistors, micro motors, wire, resistors...	100,00€
Paintball gear	220,00€
Battery and charger	85,00€
4 x High Power Metal Gearmotor	75,00€
Pololu motor driver shield	50,00€
Wood costs	20,00€
Screws, nuts, brackets, washers, tin...	25,00€
5 x Carring balls	10,00€
Motor state taxes (customs department)	50,00€
TOTAL	908,00€

Table 8.1: Table of expenses

8.3 Future plans:

In the future, I would like to continue working in the robotic world. One of my next steps would be to equip the robot with a camera and be able to control it in a safe place. Another step would be to improve the structure, make it in aluminium which would be lighter and stronger.

I would like to start a new project such as a fighter robot or an autonomous paintball turret improving the wireless communication and the structu*re.*

9 Bibliography:

Arduino tutorial series made by Jeremy Blum:

http://www.youtube.com/watch?v=fCxzA9_kg6s

http://www.youtube.com/watch?v=_LCCGFSMOr4&feature=fvwrel

http://www.youtube.com/watch?v=abWCy_aOSwY&feature=fvwrel

http://www.youtube.com/watch?v=js4TK0U848I&feature=fvwrel

http://www.youtube.com/watch?v=5bHPKU4ybHY&feature=fvwrel

http://www.youtube.com/watch?v=g0pSfyXOXj8&feature=fvwrel

http://www.youtube.com/watch?v=GJX0BRUagCg&feature=fvwrel

http://www.youtube.com/watch?v=1nO2SSExEnQ&feature=relmfu

http://www.youtube.com/watch?v=vKVNmA8C6m8&feature=fvwrel

http://www.youtube.com/watch?v=CRJUdf5TTQQ&feature=fvwrel

http://www.youtube.com/watch?v=5v5A3j7Rrco&feature=fvwrel

http://www.youtube.com/watch?v=gIlSLwcbeTU&feature=relmfu

http://www.youtube.com/watch?v=oIiDseJO4dM&feature=relmfu

http://www.youtube.com/watch?v=CoG_Czyr7z0&feature=relmfu

http://www.youtube.com/watch?v=TtZEZYQG0xk&feature=relmfu

http://www.youtube.com/watch?v=2QJ4CkZpu3A&feature=relmfu

http://www.youtube.com/user/sciguy14?feature=watch

Arduino page:

http://www.arduino.cc

http://arduino.cc/es/Main/Buy

http://arduino.cc/es/Main/Software

http://arduino.cc/es/Reference/Extended

http://arduino.cc/en/Guide/ArduinoXbeeShield

http://www.arduino.cc/cgi-bin/yabb2/YaBB.pl?num=1294631347

http://arduino.cc/es/Serial/Read

http://arduino.cc/es/Reference/String

http://arduino.cc/es/Main/ArduinoXbeeShield

http://www.arduino.cc/cgi-bin/yabb2/YaBB.pl?num=1203744779/30

http://www.arduino.cc/cgi-bin/yabb2/YaBB.pl?num=1282093275

http://www.arduino.cc/cgi-bin/yabb2/YaBB.pl?num=1232032570

http://www.arduino.cc/cgi-bin/yabb2/YaBB.pl?num=1203744779/30

http://www.arduino.cc/cgi-bin/yabb2/YaBB.pl?num=1282093275

http://www.arduino.cc/playground/Main/DCMotorControl

http://www.arduino.cc/cgi-bin/yabb2/YaBB.pl?num=1242422823

http://www.arduino.cc/cgi-bin/yabb2/YaBB.pl?num=1192247405

http://arduino.cc/es/Reference/Servo

http://arduino.cc/es/Tutorial/HomePage

http://www.arduino.cc/playground/ComponentLib/servo

Electronics and robotics shops:

http://www.pololu.com/

https://www.sparkfun.com/

http://www.bricogeek.com/shop/

http://www.cooking-hacks.com/index.php/

http://www.onlybatterypacks.com/main.asp

http://es.farnell.com/

http://www.seeedstudio.com/

http://www.bcncybernetics.com/

http://adafruit.com/

http://www.dfrobot.com/index.php

Paintball shops and fields:

http://www.tiendapaintball.com/

http://www.intactics.es/

Bill Porter's web: “The mind of Bill Porter”

http://www.billporter.info/

http://www.billporter.info/playstation-2-controller-arduino-library-v1-0/

http://www.billporter.info/arduino-playstation-2-controller-library-troubleshooting-guide/

https://github.com/madsci1016/Arduino-PS2X

http://www.billporter.info/category/projects/autonomous-rover/

http://www.billporter.info/category/tutorials/

Driver pololu Dual-VNH5019:

http://www.pololu.com/docs/0J49
https://github.com/pololu/Dual-VNH5019-Motor-Shield

Battery pack:

http://www.onlybatterypacks.com/

Pololu's motors:

http://www.pololu.com/catalog/product/1573

Driver texas instruments SN75441ONE

http://www.alldatasheet.com/view.jsp?Searchword=SN75441ONE

http://www.sparkfun.com/datasheets/IC/SN754410.pdf

Examples of paintball robots:

http://www.engadget.com/2011/02/28/mega-hurtz-paintball-robot-is-the-remote-controlled-armor-plate/

http://hacknmod.com/hack/absolutely-epic-paintball-gun-robot/

http://uncrate.com/stuff/paintball-robot/

http://www.neoteo.com/ayuda-a-construir-un-paintball-robot

http://www.societyofrobots.com/robotforum/index.php?topic=8025.0

Xbee modules:

http://arduino.cc/en/Guide/ArduinoXbeeShield

http://foro.elhacker.net/electronica/arduino_encender_y_apagar_led-t295171.0.html

http://arduino.cc/es/Serial/Read

http://www.arduino.cc/cgi-bin/yabb2/YaBB.pl?num=1294631347

http://arduino.cc/es/Reference/String

http://www.instructables.com/id/Make-Wired-Robotic-Arm-Edge-to-Wireless-with-DIY/

http://letsmakerobots.com/node/23869

http://www.instructables.com/id/Make-Wired-Robotic-Arm-Edge-to-Wireless-with-DIY/step14/Arduino-Sketch-Processing-code/

http://code.google.com/p/smduino/wiki/PlaystationControl

http://www.adafruit.com/blog/2012/04/23/weekend-project-arduino-xbee-ps2-controller-20-channel-universal-project-controller/

http://en.wikipedia.org/wiki/Binary_Synchronous_Communications

https://www.sparkfun.com/products/9501

http://www.sparkfun.com/tutorials/356

http://www.sparkfun.com/datasheets/Robotics/POP-BOT_e_small.pdf

http://www.billporter.info/easytransfer-arduino-library/#more-863

http://www.adrirobot.it/

http://www.multiwii.com/forum/viewtopic.php?f=7&t=2295

http://www.youtube.com/watch?v=Q0PPsqq9xDE

http://www.youtube.com/watch?v=3zzdjsxl5D0&feature=plcp

http://antipastohw.blogspot.com.es/2008/12/passing-large-values-between-arduino-to.html

http://nootropicdesign.com/projectlab/2010/03/04/wireless-robotics-platform/

http://www.electronics-lab.com/blog/?tag=xbee&paged=2

http://code.google.com/p/xbee-arduino/

http://www.cooking-hacks.com/index.php/documentation/tutorials/arduino-xbee-shield

http://ftp1.digi.com/support/images/Win7DriverInstall.pdf

http://blog.didierstevens.com/2009/06/15/quickpost-arduino-xbee-shield-series-2-configuration/

http://tutorial.cytron.com.my/2012/03/08/xbee-series-2-point-to-point-communication/

The sentry project:

http://www.paintballsentry.com/

http://www.youtube.com/watch?v=tfEpdVqOLXs&list=UUd9kWDeRbGF-L5WwHrVom4Q&index=5

10 Index of pictures and tables

Pictures

Tables