**EOD Bot**

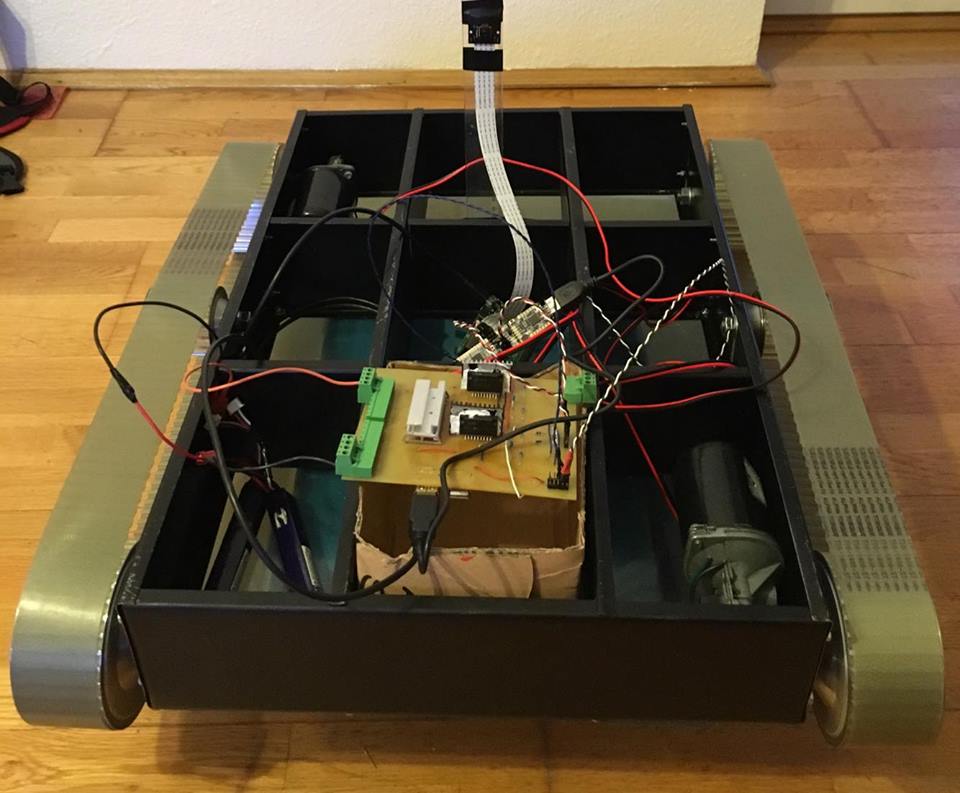


Table of content:

1. Purpose, goal 2
2. Requirements 3
3. The tests 4
4. Time plan 5
5. Buttom up developement 5
6. Function blocks 5
7. Software architecture 5
8. Test track 6
9. Part list 7
10. Frame 7
11. Belt gears 9
12. Shaft 10
13. Psoc program 10
14. GUI 11
15. PCB layout 11
16. The IC wiring diagram 13
17. Batteries 13
18. Software parts what we have done 14
19. Price list 15

Purpose, goal:

To built a well prepared reconnaissance robot which is able to explore dangerous or toxic places that can be life threathening, or even lethal for humans or places which are hardly approachable for us. It can help during counter terrorist operations or even in times of war, by exploring unstable objects or buildings while it is transmitting live video feed, or even can carry limited amount of medical equipments, or some resources, like food or fresh water.

Requirements and test against requirements:

**There will be three requirements in the first version which are:**

* The ability of movement
* The controllability
* The data transmission for the video feed

These tests what we are introducing now the explicit tests that it must be pass in order to consider it as success. Of course we will present further tests to be able to continously improve our product during our running project. We will introduce them during our development phase, but for now these are depends on unforseen difficulties or conditions, we can come back to this point after the first working prototype.

**The tests:**

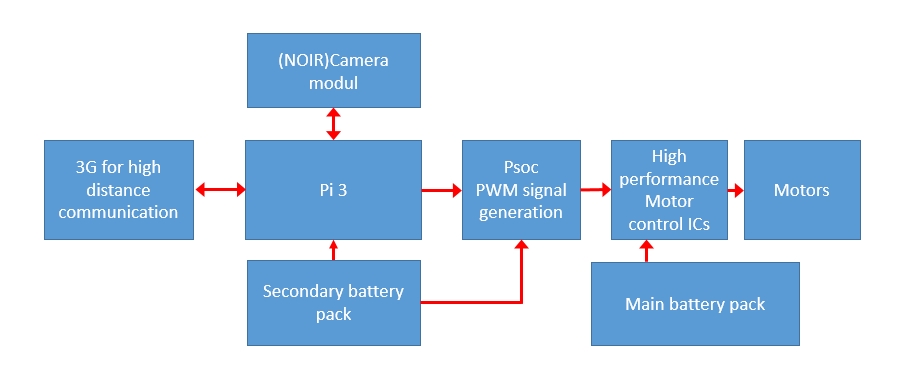
* For the ability of movement is the simpliest test, we can check that it can move back and forth, and it have the ability of turning in both direction without external help, of course the remote control not counted in this case.
* It’s controllability, we can test by a path that it must go through fluently, without any collision to the objects nearby. It is also depends on the person who is controls it. and its also a measure of how easy to handle this robot, which is very important in stressful situations for people with lack of experience. We attached the planned route that it must be acomplished without any difficulties, and can be seen after this paragraph.
* The data transmission is a critical point in case of the robot is behind an object for example a wall, or in a different room of a building, and we cannot see it directly. It must work in any case to stay controllable. It is maybe the easiest test of all of them, just simply check if the video arrives to our device or not.

|  |  |  |  |
| --- | --- | --- | --- |
| **Week** | **Levente Szövérffy** | **Bálint Molnár** | **Dates(2016)** |
| **1.week** | Planning, Collecting the needed information | | 02.29-03.06 |
| **2.week** | Ordering components | Time scheduling | 03.07-03.13 |
| **3.week** | Testing the components individually | Planning the base structure | 03.14-03.20 |
| **4. week** | Design the motor control (hardware) | Assembling the structure | 03.21-03.27 |
| **5. week** | Design the motor control  (software) | Estabilishing the connections | 03.28-04.03 |
| **6. week** | Code the camera software | Designing the camera software | 04.04-04.10 |
| **7. week** | Build up the GUI of the software | Get the needed information from the data received from the PC software and translate it to the control module | 04.11-04.17 |
| **8. week** | Testing, Bugfixing | | 04.18-04.24 |
| **9. week** | Bugfixing, Prepare the presentation documents, PPT | | 04.25-05.01 |
| **10. week** | **Presentation** | | 05.02-05.08 |

**Buttom up developement:**

During the project we used buttom up developement method. We designed manufactured and programmed all the small parts separately then we started to put them together and integrate them into one system.

**Function blocks:**

****

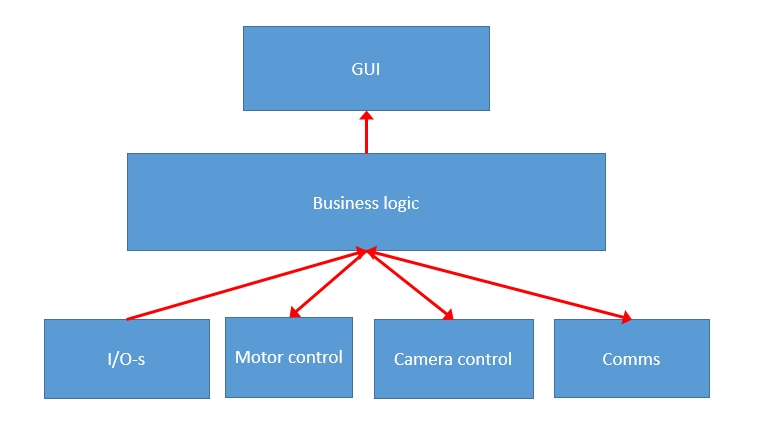
**Software architectrue:**

During the planning of the software we tried to remain simple with the architecture. The software is built up by three layers.

The upper layer is the GUI where the user can controll the robot and also it gives the user all the required information about the status of the robot.

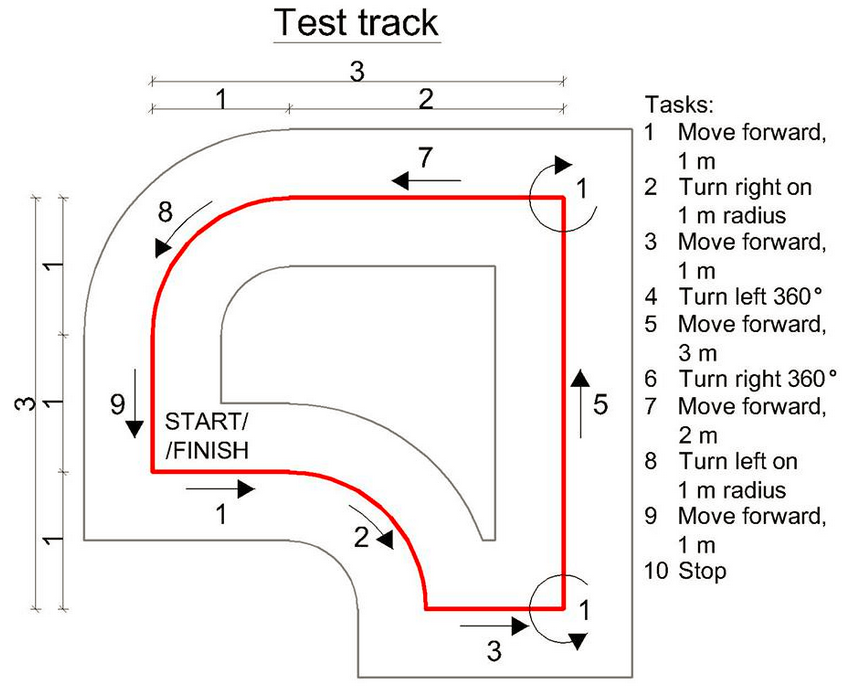
The middle layer is the business logic. In this layer we connected all the periferies. This interface layer collects all the information from the GUI and from the periferies and also these interfaces transfares the instructions to the other two software layers.

The bottom layer is the perifery layer. The layes contains all the periferies, the GPIO-s, the motor controll, camera controll and the communication too.

****

**The test track for the second requirement:**

The test track was set to cover the whole requirement. That’s why it contains turning in both directions in one place and during movement too. If the robot can go through the track we can say that it’s able to do movements in every direction.



**Part list:**

* Rapsberry Pi 3+8gb Micro SD card
* NoIR camera modul for rapsberry
* Controll PCB
* Frame
* Shafts:
* -2pcs for the motors
* -2pcs for the non motor driven shafts
* Belts
* Belt gears
* Psoc 5
* Battery

**The frame:**

We had rusty sheet metal. First step was to cut out the parts for the required sizes.

Then we did the angle bending. This step was essential for us to give our sheets some rigidity.

Removing the dust. To remove the dust from the sheets we used wire brushes.



Drilling the holes for screws, shafts, and motor mounts.



Cut out some lines to be able to put together the bent sheets in right angle.



Paint the steel parts. We spray painted the sheets to prevent them from rust. Also the paint job gave the parts some quality look. For painting we used simple spray cans in matt black color.



Frame srewed together with the drivetrain and motors. Belt fasteners are adjusted.



**Belt gears:**

The belt gears originally when we got them were quitey heavy(1550g each). We needed to reduce their weight.After the weight reduction each gears weighs 950g. With the weights reduction we could save more than 2 kg. To remove the wieght we took all the gears to a miling expert. He did all the miling work and also he manufactured the shafts for us.

Before miling:



After miling:

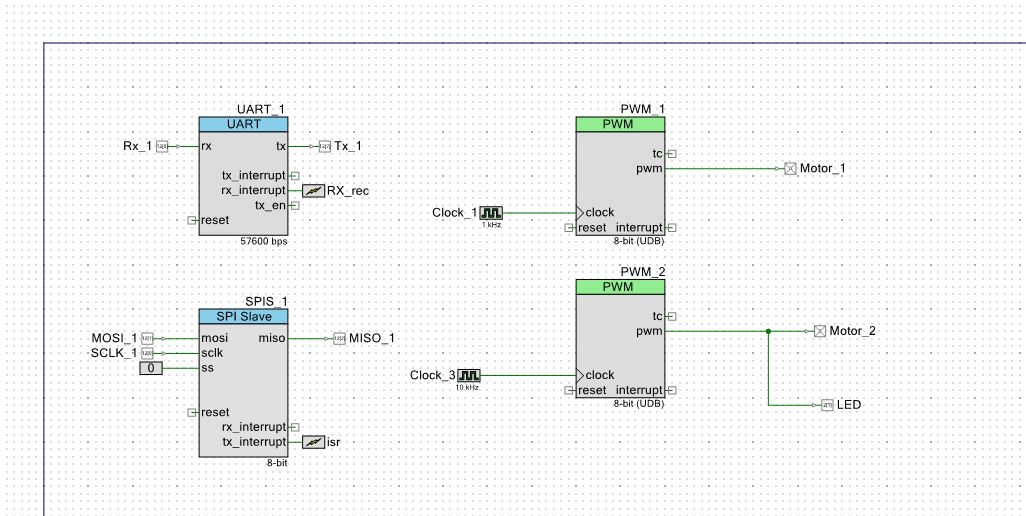


**Shafts:**

We need two shafts for the motors and two other ones which can rotate freely on the shafts. To make easier the rotation we applied some grease.

**Psoc program:**

The Psoc’s programm contains two PWM signal generator and two types of communication. We created UART and SPI communication blocks in the Psoc’s program to be able to change the communication method during the run of the software. The Psoc’s programme is written in C language. For test purposes we set 10kHz PWM signals with 50% duty cicle it gave us the appropirate torque and speed.

****

**Gui:**

The GUI contains all the controll buttons. (we can controll the robot with a keyboard too).

Print screen: saves the actaul picture of the camera

Rec: start recording the camera’s picture

Stop rec: stops recording the camera’s picture

Lamp: turns up the LED lights on the robot

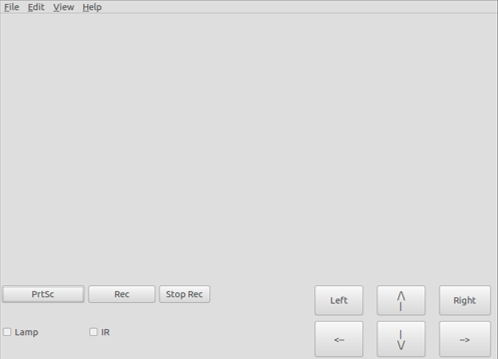
IR: turns up the IR LEDs

Left: Turns the robot left in one place

Right: Turns the robot right in one place

Arrow buttons: Moves the robot in the shown direction

The GUI has that full grey area above the buttons to be able to stream the video there.



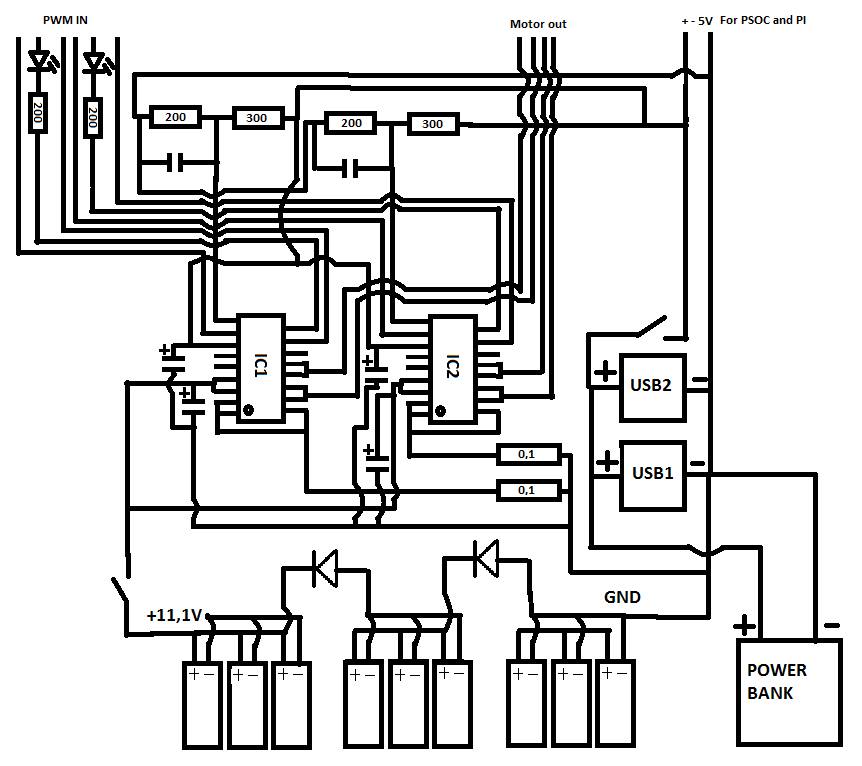
**PCB layout:**

We designed and made a pcb which estabilishes the connections between the parts of the robot.

PCB part list:

* 1x Kill switch
* 1x USB female connestor
* 1x double USB female connector
* 2x stk681 332-e IC
* 2x 100nF condensator
* 2x (másik fajta kondi)
* 2x (harmadik féle kondi)
* 4x 200 Ohm resistor
* 2x 300 Ohm resistor
* 4x 0,1 Ohm 5W resistor
* 2x double schottky diode 100V
* 2x 3V LED
* 1x Power bank???
* 9x LI-PO battery 5000mAh 3,7V???
* 2x 9 pin serial connector
* 14x male pcb pin connectors

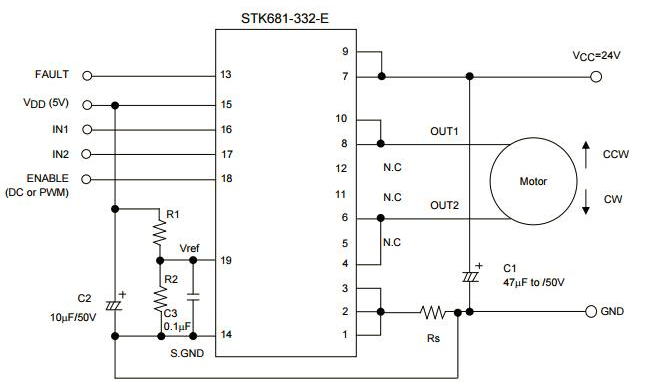
PCB design layout:

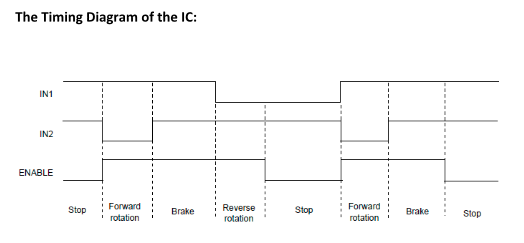


**The IC-s wiring diagramm:**

For the current limitation of the IC, we van use small resistors on the place of the Rs resistor on thewiring diagram below, and setting the Vref reference voltage by voltage division with the R1 and R2resistor. We used 0,1 Ohm resistor at first as Rs, and we could adjust the current limitation byconnecting more of them parallelly, we can calculate it according to this equation:Iomax=(Vref/4.9)/Rs In our case Vref was set to 2V, and the lesser the value of the Rs, the higher the

current limitation.





**Batteries:**

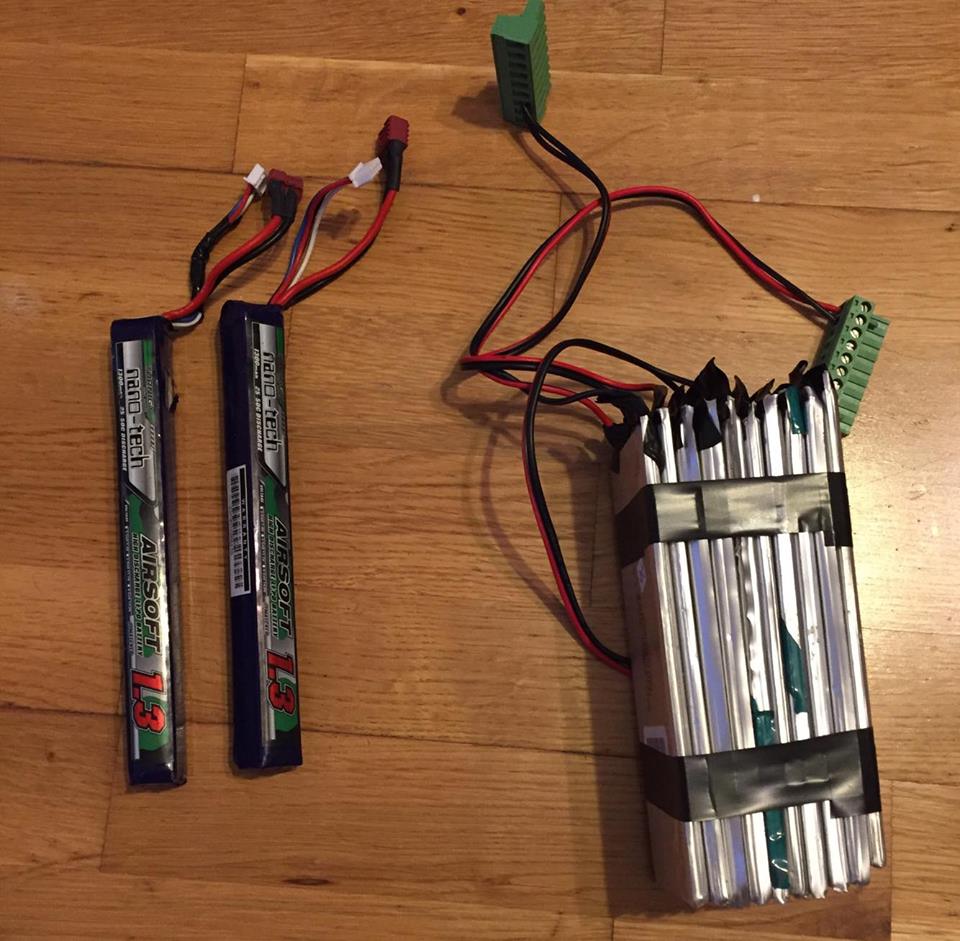
Fort he motors we use 9 lithium polymer cells. Each of them features 3.7V 5000mAh. In our case after we wired together. We got a 11.1V 15000mAh battery pack.

For the controll units (Pi 3 and Psoc) we use a power bank.

We planned to use two battery packs separately because we try to protect our controll units from any kind of short circuits or overloads this way.

For the test runs we used two 11.1V 1300mAh battery packs connected serialy. This gave us 10 V output on each motors. The planned 11.1V battery pack arrangement can’t give 10V output voltage on the motor outputs. We will redesign the battery arrangement in this way on our next PCB.

During the tests we used an external power supply for the 5V controll voltages because our power bank haven’t arrived yet.



**Software parts what we have so far:**

* Psoc programme
* Camera controll
* Tcp/IP communication
* GUI

**Price list:**

* PCB parts 10000 Ft
* Pi 3 16000Ft
* PsoC 5 5000 Ft
* Camera 9000 Ft
* Wires and misc 3000 Ft
* Miling 20000 Ft
* Paint 1000 Ft
* Frame parts 10000 Ft
* All other part we got for free