Application Sheet

In order to accomplish the tasks in the Home Mini Size category, our team managed to create a compact and efficient robot, meant to help the people around. We successfully integrated systems that performs the 3 basic tasks: robot guide test, follow me test and pick and place test. Because our team always comes up with solutions related to the current problems that people face, this time, we decided to help the sick elderly people. Our robot can interact with these people and help them take their medicines on a certain time. It can bring a specific drug depending on the day or time and bring a glass of water to help the elderly who cannot move easily and need treatment. A more detailed description and follow-up updates can be found at http://herotech.ro/wrs.html

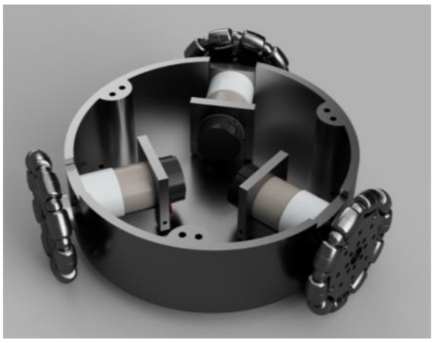
Overview of the robot

From the beginning we wanted to create a robot with an original design, so we made a lot of parts ourselves, using the 3D printing technique. Regarding the electrical part of our robot, we chose to have one single mainboard designed by us, fact that allowed us to implement our own components. Needless to say, creating all these parts on our own significantly reduced the cost of producing such a robot, so it could be affordable to any person if it will go into production. For the 3D printed components, we used Galaxy Black Prusament PETG from Prusa Research, because of its high impact strength and durability.

Motion of the robot

Our team embraced the design of a circular 3 omnidirectional wheels robot, even though the motion control system is far more complicate then the classic motion control used for 4 wheels robots or 2 wheels ones. We chose this option, because we wanted our robot to be able to move in any direction without changing its position. Also, we opted for this type of robot, because we had to use only 3 DC motors for the moving, so the power consumption is lower than in the 4 wheels robots.

On a circular 3D printed base (210 mm diameter) we mounted 3 Pololu 50:1 Metal Gearmotors (12 V) using 3 supports fixed on the base with screws.



The 3 motors used for movement are equipped with quadrature encoders, as a result, we know the exact position of the robot on the competition table.

Robot guide task

For this task, we used three distance sensors (Wenglor, P1KY001), of which two of them are mounted on servomotors to be able to detect objects in a larger area. The third sensor is fixed and it is used to detect objects behind the robot. These sensors are used to determine the distance between them and other objects by means of transit time measurement. These types of sensors have a great accuracy and a high switching frequency, using laser class 1 – safe for human eye. In order to rotate those two mobile sensors, we used two goBILDA servomotors controlled from our mainboard. To move the robot to a certain area on the table, we used the motor encoders, the gyroscope and the accelerometer (for rotations).

Follow me task

This task is accomplished using a Logitech C920 web camera, connected directly to the Raspberry PI 4. Using an algorithm (presented in software section), the robot follows the black cylinder. For the 90 degrees turns, our team used the sensor from the previous task to determine the direction of the followed robot.

Pick and place task

The web camera and the fixed sensor make possible the detection of a bottle. The robot stops in front of it at a distance lower than 100 mm. Using a 3D printed arm, the robot picks the bottle then transports it to the indicated place. The arm is an open source project from EEzyrobots, and it uses four micro servomotors. We adapted this arm to our needs, so we modified the base of the arm and we designed a different gripper that allows picking the bottle. Also, we used only three micro servomotors, because the purpose of the fourth one was to rotate the whole arm, and it was unnecessary in our case. Also, this arm is used to move the chair and we plan to adapt it to move the mystery object too.

Interacting with people

By using the Amazon Alexa Echo and building an Alexa skill, our robot can interact with people, especially with those who cannot move easily. It can bring them a certain medicine and a glass of water in order to help them take their treatment.

Every team member was in charge of a specific part of the project as follows:

- Daniel – electronics
- Codrin – programming
- Alex – 3D design

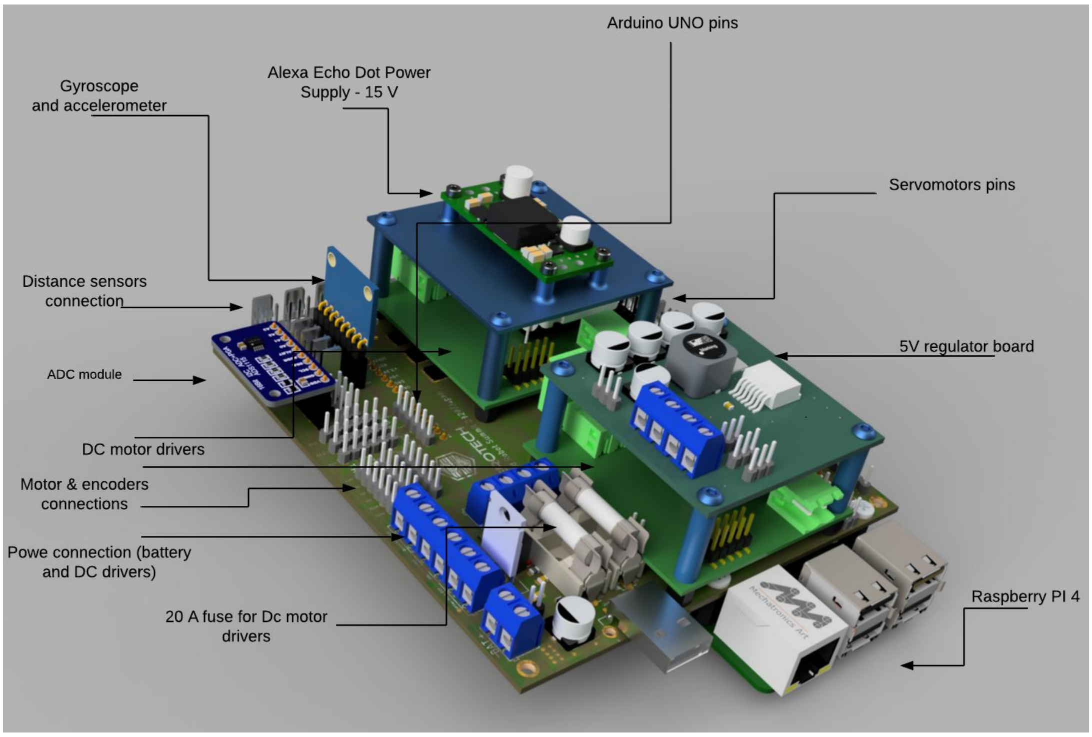
The electronics part:

The brain of our robot is a Raspberry PI 4 (4GB Ram), which is attached to our mainboard with pin headers. This controls 2 DC motor drivers (7A Dual DC Motor Driver from DFROBOT). The drivers use 2 digital pins for the direction control and one analogic pin (PWM) to control the speed of every motor. These drivers are powered from the main 12V battery and as a safety measure we added a 20 A fuse for each driver. The logic voltage for the drivers is 3.3 V. The Raspberry PI generates DMA timed PWM for a high accuracy.

Because we used all the available pins, on the mainboard, we integrated an Arduino UNO in case we want to add other components. In this case, the communication between the Arduino and the Raspberry PI will be serial.

Because we need a precise rotation of our robot, we use gyroscope and an accelerometer, which is connected to the Raspberry using I2C protocol. Initially, we designed the mainboard with a special analog to digital converter to read the values from SHARP distance sensors, but we gave up this idea, because the output of these sensors was non-linear and it was very hard to interpret it.

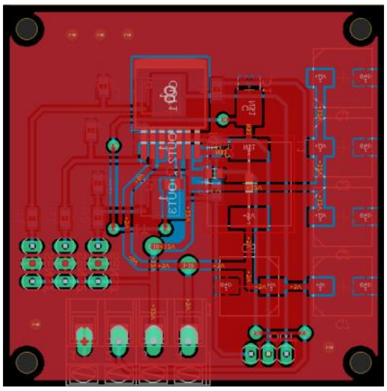
On the mainboard, we added 8 servomotors ports with 3 pins: GND, +5V and PWM (software PWM generated by Raspberry PI).


Arduino UNO pins
Alexa Echo Dot Power Supply - 15 V
Gyroscope and accelerometer
Servomotors pins
Distance sensors connection
5V regulator board
ADC module
DC motor drivers
Motor & encoders connections
Powe connection (battery and DC drivers)
Raspberry PI 4
20 A fuse for Dc motor drivers

The electrical part assembly

The robot is powered from a 12 V battery and has an emergency switch in case if it is needed to turn it off immediately. The Raspberry PI is powered using a 12 V to 5 V and 2.5 A converter and for Alexa Echo we used a Pololu Adjustable 9-30V Step-Up/Step-Down Voltage Regulator, which is able to provide 15 V continuously (Amazon Alexa requires 15 V in order to operate at normal parameters).

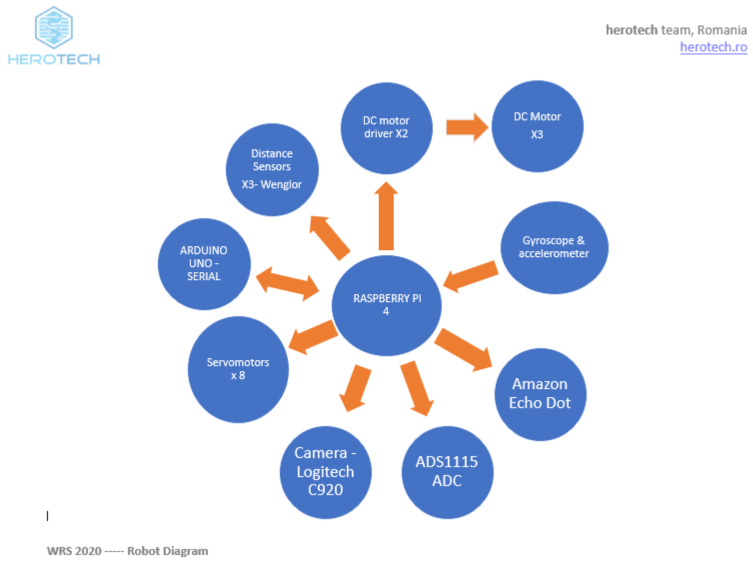
Because the servomotors have a power consumption that can not be provided by the Raspberry intern 5V regulator, we design our own regulator which can provide 5V and 5A. This uses the LM2678 regulator, a 20 uH coil and some electrolytic capacitors for a filtering. This circuit is mounted on the mainboard as a shield, using some spacers.



The layout of the 5V regulator – Autodesk Eagle

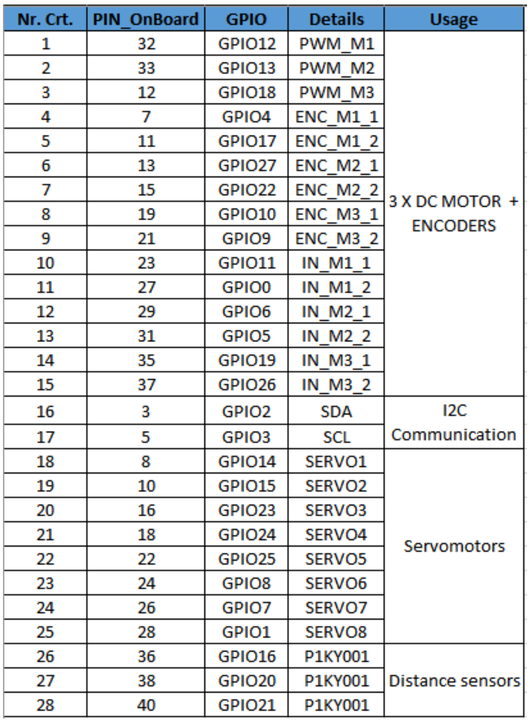
How did we do the electronics part?

1. Firstly, we created a flowchart of our robot, which includes all the required components.



herotech team, Romania
herotech.ro

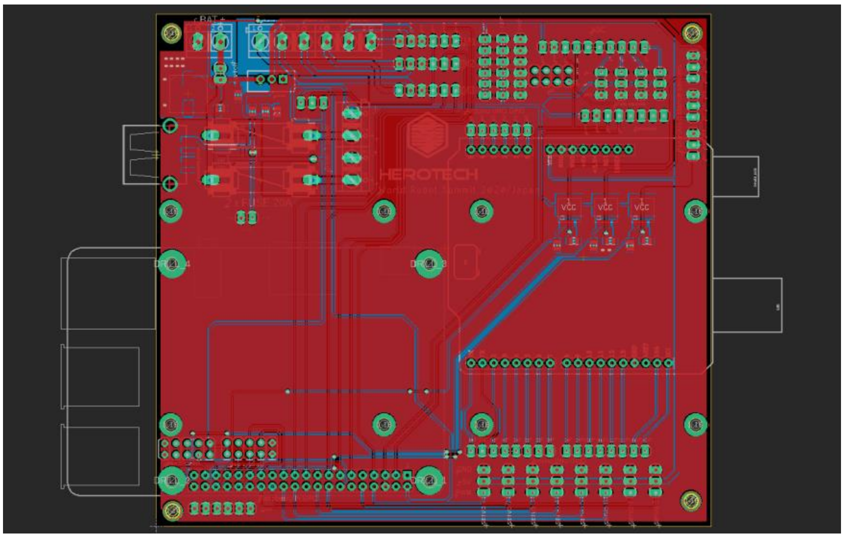
WRS 2020 ----- Robot Diagram



Nr. Crt.	PIN_OnBoard	GPIO	Details	Usage
1	32	GPIO12	PWM_M1	3 X DC MOTOR + ENCODERS
2	33	GPIO13	PWM_M2	
3	12	GPIO18	PWM_M3	
4	7	GPIO4	ENC_M1_1	
5	11	GPIO17	ENC_M1_2	
6	13	GPIO27	ENC_M2_1	
7	15	GPIO22	ENC_M2_2	
8	19	GPIO10	ENC_M3_1	
9	21	GPIO9	ENC_M3_2	
10	23	GPIO11	IN_M1_1	
11	27	GPIO0	IN_M1_2	
12	29	GPIO6	IN_M2_1	
13	31	GPIO5	IN_M2_2	
14	35	GPIO19	IN_M3_1	
15	37	GPIO26	IN_M3_2	
16	3	GPIO2	SDA	I2C Communication
17	5	GPIO3	SCL	
18	8	GPIO14	SERVO1	Servomotors
19	10	GPIO15	SERVO2	
20	16	GPIO23	SERVO3	
21	18	GPIO24	SERVO4	
22	22	GPIO25	SERVO5	
23	24	GPIO8	SERVO6	
24	26	GPIO7	SERVO7	
25	28	GPIO1	SERVO8	
26	36	GPIO16	P1KY001	Distance sensors
27	38	GPIO20	P1KY001	
28	40	GPIO21	P1KY001	

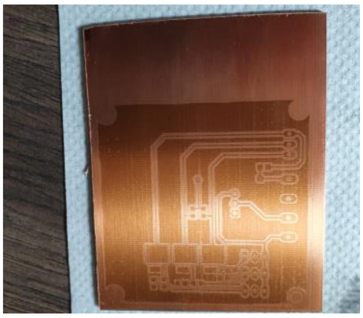
In order to keep things organized, we created a table which contains the connections for every component.

2. After we read the datasheet for every component, we designed the electrical schematics in Autodesk Eagle, then we created the layout of our printed circuit board.



The layout of the mainboard – Autodesk Eagle

3. We chosed to prototype our mainboard with a PCB specialized manufacturer, while the 5V regulator board was made by us using the ultraviolet method.



PCB before etching

Ultraviolet lamp used to print the traces on the copper board

4. Next step was the soldering process, which requires an increased attention, because most of the components were SMD-s.

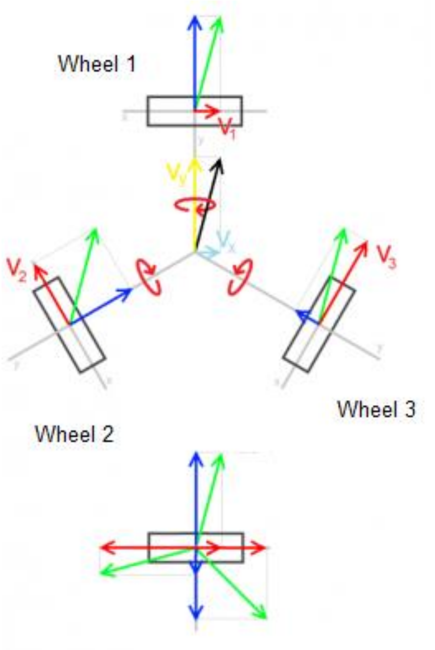
5. The last step regarding the electrical part was testing the functionality of every component. We faced some problems at the DC driver control, but after some measurements we adjusted the PWM signal, such that the speed of each motor is constant.

The programming part:

The code of the robot runs on the Raspberry PI. We opted on using python even if we weren't as proficient as with other languages because of the libraries found on the internet for different hardware parts.

Firstly, we wrote the code for rotating the dc motors. We input PWM for each motor which can be a value between 0 and 100 and represents the speed our motors rotate by. Next it was time to figure speed for all 3 motors based on the direction we want our robot to go.

In the following illustration, the desired speed of the robot is represented by the vector in the center of the platform (black arrow). It has the components v_x (light blue) and v_y (yellow), these are the two input values for the calculation of the three motor speeds v_1, v_2 and $v_{3.}$ Wheel 1 can only contribute to the movement of the platform in the x direction since it is perpendicular to the y axis of the coordinate system. Therefore, v_1 is equal to v_x.



Since the wheels are arranged in a circle at a 120° distance from each other, the coordinate systems of Wheel 2 and 3 can be made to coincide with that of Wheel 1 by rotating them by -120° or +120°.

The formulas for a rotation of a vector with the components x and y by the angle θ around the origin of a coordinate system are:

$$x' = x*\cos(\theta) - y*\sin(\theta)$$
$$y' = x*\sin(\theta) + y*\cos(\theta)$$

However, in our case we only care about the x component, so we apply the first formula for v_2 and v_3:

$$v_2 = v_x*\cos(120^\circ) - v_y*\sin(120^\circ)$$
$$v_3 = v_x*\cos(-120^\circ) - v_y*\sin(-120^\circ)$$

We also made a simulation in Unity to check if our math is right by visualizing the speed vectors for each motor. Sadly, we couldn't implement this idea on the hardware because the input PWM for each motor does not seem to be directly proportional to the rotation speed. It might be a hardware problem and we will look for a solution in the future. In our demos we limited ourselves by giving the motors equal power, since the rotation speed of the motors seemed equal in this case. For forward movement we powered 2 motors in opposite directions and for rotating around the robot origin we powered all 3 motors in the same direction.

For the follow me task we implemented two different approaches. One using 2 distance sensors and based on the output of them we turn the robot in the right or left direction.

Left sensor	**Right sensor**	**Action**
DETECTED	DETECTED	-
DETECTED	NOT DETECTED	TURN LEFT
NOT DETECTED	DETECTED	TURN RIGHT

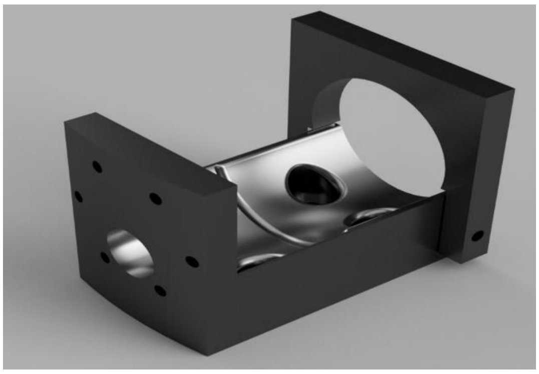
The other implementation is based on the output of a camera fixed on the robot. We use a tracker algorithm provided by opencv to determine the position of the object we need to follow. Also, by knowing it's dimensions, we calculate the distance between the camera and the object so we can stay in the 30 cm range.

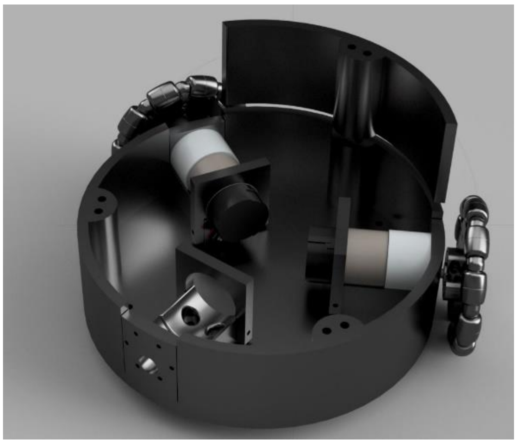
The 3d design part:

All the 3D printed components were designed by us in Autodesk Fusion 360 through multiple iterations to be strong and durable and to remove as many errors that as we could, so we wouldn't have to 3D print them and then find a problem. The parts were also made keeping manufacturing in mind, mostly trying to create them so we wouldn't have to use supports while printing them. The design of the robot was completed through multiple stages:

1. Motor level

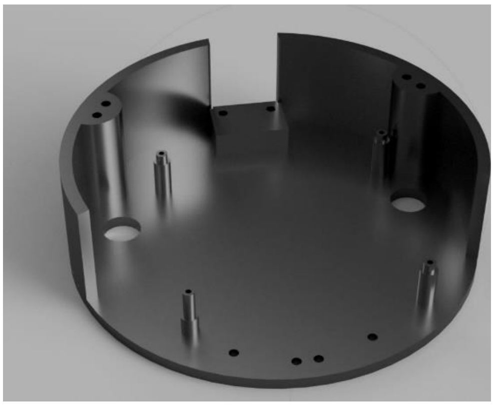
We firstly imported the CAD of the motors and based on it designed a mount that could be secured to the base with 4 screws and an extra bracket to secure the motor in place.



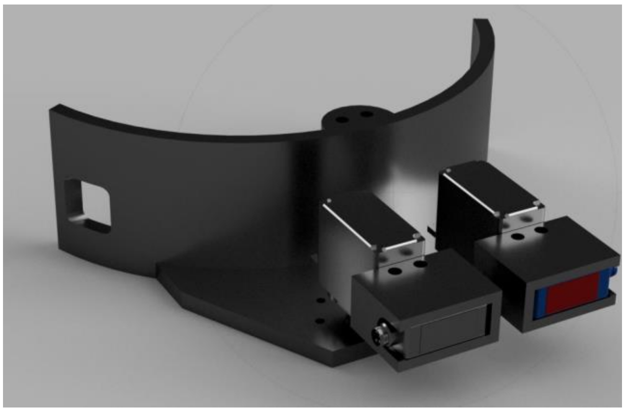


Next, we created the base of the robot with a radius of 105 mm which is the maximum size our 3D printer can print. To secure the future levels to this one we decided to use 6 long screws that run through the whole height of the robot.

2. Mainboard level



For the second level we started by creating the layer between level 1 and 2 which has 4 mounts for the mainboard and one for a distance sensor.



Here we also have the part that holds the two other distance sensors mounted on servomotors. This whole component is mounted on the previous part and has a space to access the mainboard.

3. Top level

This part is the final one and has multiple roles. It holds the battery mount that we designed, the camera, the Alexa and the robot arm. The component has double the number of holes it actually needed because we wanted to be able to change the position of the robot arm, which required all the other components to be moved to new sets of holes.