ROBOT – PROGRAMMING, ASSEMBLING AND IMPLENTATION OF IDEAS

Mechanical Engineering

Sarah Breit, Barbara Wiedermann, Andjela Arnautovic and Julia Pöschl
TGM Information Technology
Vienna, Austria

***Abstract* — This paper gives an overview of our robot, our ideas and how we implemented them. The first section describes our initial ideas and how we developed these with regard to our Demo-Bot. The second section depicts how we assembled both of our robots, how we programmed them and why we made the decisions to use the various effectors.**

Keywords— implementation, Demo-Bot, two robots, effectors

I. INTRODUCTION

As students studying in the first year at the TGM in the Department of Information Technology, we were asked if we wanted to take part in the Botball tournament by our head of Department. Of course we said yes, because we were interested in Robotics and it sounded like a cool extra-curricular activity. Without having any real idea of what we were letting ourselves in for, we attended the initial workshop. There we were suddenly confronted with a wealth of information and were sitting side by side with older students who all seemed to know a lot more than we did. The information we received was rather overwhelming, and the fact that we were going to compete with these older, more knowledgeable students was also a bit intimidating. Although our enthusiasm was and is great, our knowledge of programming in general was and is very limited. Being in the first year we had only programmed real Java for a few days, having previously started learning Java hamster a couple of months before. None of us had any experience with robotics. Our technical knowledge was limited to Design and Technology lessons at school and 3 introductory months of workshop practice at the TGM.

II. WORKSHOP

In February we attended the Botball-Workshop and there we came into contact with robots for the first time. We were suddenly made aware of their huge impact on society and how these can change real human life. At the workshop we had to build a Demo-Bot in order to understand how we can build a new one in practice and program it to fulfill certain tasks.

At the robotics workshop we had to work with C, a new computer language for us, which was very confusing at the beginning. We were able to adjust to it relatively quickly, which surprised us all, and so we made gradual progress and we were able to follow the information given.

We found the workshop very well done but there were some things that in our opinion could be improved on.

For us as first year students it was very good that everything was explained from the beginning and in such detail. However we found the explanations for the software updates concerning the Wallaby rather confusing, and struggled here somewhat. Another problem was that everybody wanted to do the updates at the same time, but this was not possible, because there were too many Wifis in one room. We then had to update the Wallaby in the lunch break and so we couldn't charge the batteries in that time. But in the end we had the update and we were then able to get started, even though we were a bit behind the rest.

At the workshop there were always people there who could help teams with any problems that arose, a big plus for us. It was also good that the programming was explained from the beginning and in detail. But the older students were bored while the Pria team explained everything and were doing other things, which was a bit distracting. So we would like to suggest that it would be a good idea for students who have no experience to get a basic introduction a day before. After that day the older /experienced students could join the workshop and all new things, the rules and any specific information for this year's games, could be explained.

We particularly enjoyed the fact that theory was alternated with practice. This contrast between explanation and trying things out, for instance following the black line at the end of the workshop, made it easier to digest the theoretical information. A problem which had nothing to do with the workshop itself is that laptops with Windows 10 don't recognize the robot if it is connected with the cable rather than via Wifi. This could be a very big problem at the ECER if a team only has Windows 10 laptops and there are too many Wifis in the room.

To summarize, during the two days we received a great deal of information in a very dense short period, and afterwards we were a bit apprehensive as to whether we would be able to meet the challenge, particularly as we weren`t able to process everything and use the new knowledge immediately after the workshop, as afterwards the rest of life went on with tests and catching up on missed school work. Nevertheless, after a while we were able to find time as a team to accumulate our newly won knowledge and form ideas on how we could shape our robot.

III. Documentation: Parts

After the workshop we were not able to meet up as a team until a week later, so by the time we actually did, we had forgotten a great deal. Therefore it was very good that we were able to look at the slides which explained the things we had forgotten. (For example which sensor is for what and which port is necessary for the things we need)

The slides were very well structured but it would be better if there would be an index slide at the beginning, because the slides are very extensive and it takes a lot of time to find things when you are looking for something specific. The slides are in English but very easy for everybody to understand.

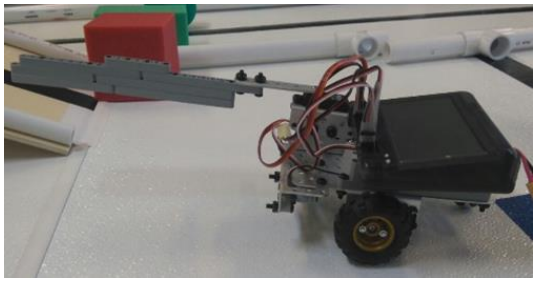
We would like to suggest that more long screws be supplied in the set, because each team we talked to ran out of long screws and therefore had to find another solution for things because they didn't have enough of them.

What also was a problem was that the holes from the metal pieces are not superimposable on the Lego parts, so it was therefore more difficult to fix the Lego parts with the metal pieces.

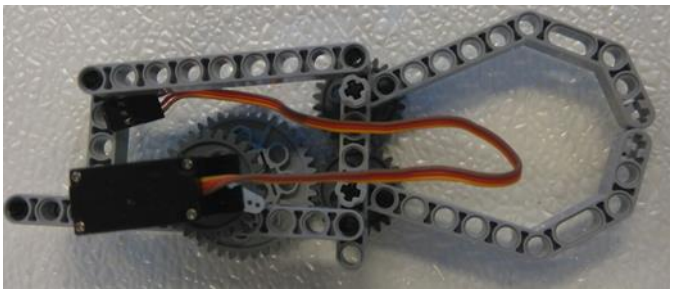
IV. The Demo-Bot and our ideas

Our robots are developments of our demo-bots. After the workshop we decided to implement an effector. For the first robot we made a "wiper", which is constructed only out of Lego-blocks and connected with a servo. This type of effector occurred to us as soon as we saw the Game-Table, because our first robot goes up to the crater rim, and with this construction it is able to throw the red balls off of the solar panels.

At first our robot had a shaver on the front side because we thought that it would be easier to place the solar panels out of our way. But then we had to acknowledge that with this construction we would not be able to get up to the crater rim. So we chose a different construction which then worked.



The second robot was not as simple to construct as the first one. With both of the robots we tried many other types of effectors, but we were not able to convert most of our ideas into reality because their implementation was too complex and complicated for us as "beginner mechanical engineers". So we decided to build only simple effectors. On the second robot we used gears to construct an effector. This effector takes the red Botnout, draws the Lego-blocks together which were connected with the gears and pushes the red Botnout to a certain point.



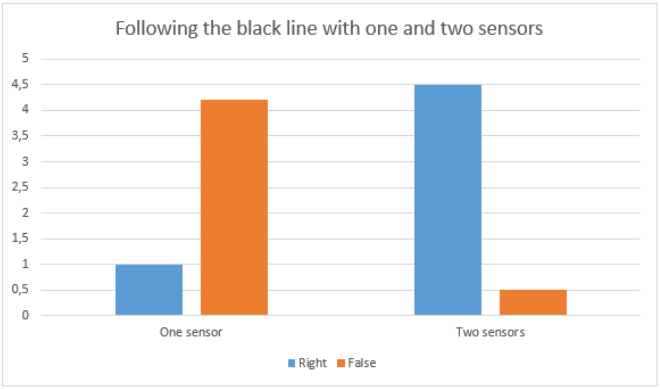
At first our second robot had a different crew arm. This crew arm was made out of metal bits and one servo. But this construction was too long and we couldn't get up to the crater rim. Together with the other robot it didn't fit into the starter box, so we had to abandon this idea.

V. The robots so far

On the first robot we used the bigger wheels because they fitted better onto our construction. The wheels are connected with the motors, which are also directly connected to our Wallaby. We decided to use this type of drive train because if we would have used the smaller wheels, the metal pieces would have touched the ground underneath, so the wheels would have spun in the air without any ground contact.

The effector is attached to the front of the robot. Our effector consists of thin, long Lego blocks, which are attached with gears. We are using just one servo. The servo is immediately connected with the wallaby. In our opinion, it is better to make a smaller and not as complex craw arm. Most of the effectors are complicated to build, like forks, because forks mostly don't need any servo or motor to move, but they have a complicated construction and so we decided on using this type of grabbing claw arm, because of its simplicity and its lightness in terms of weight, which could have otherwise presented a problem.

The robot`s task is to follow the black line, so at first we mounted one sensor, which was programmed to follow the black line. This didn`t work as it should have, because the robot often deviated off of the given track, so we decided to incorporate another sensor. Two light sensors worked much better (see diagram below) and so we stuck to this construction method.



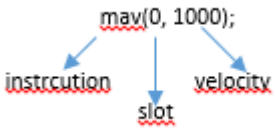
We chose this method of mounting the sensors because of its simplicity, and were therefore able to attach them easily and quickly.

V. PROGRAMMING

The Source:

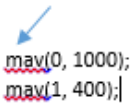
Motor: In order for the robot to drive forwards you have to give him the instruction “mav” In the semicolons you give him the port number and the speed.

e.g.:



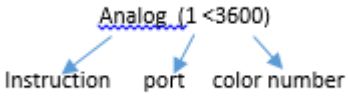
So that you can program the robot to drive a curve you have to let one motor drive faster than the other one.

This would look like this :



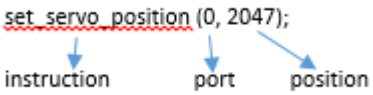
Sensor:

Since we also use sensors we need some instructions too. (We use them to follow the black line because it is more exact than time value). You have to write an instruction and the port number again but you don`t have to write the speed down, instead you write the color number. e.g.:



Servo:

We used the servo two times; one time to remove the “dirt” from the solar panels and one time to open and close the “grabber”. The Syntax of the Servo consists of the instruction, the port and the position.



VI. THE WAY OF OUR ROBOTS

We constructed two robots to increase the chance of getting more points. Our first or “main” robot has to fulfill the following functions:

1. Chuck the green ball off of its base (We decided to do this because we will get 3 points more than if we only collect the little balls (green, yellow))
2. Drive on our crater rim
(We decided to do this because we will get 10 points for it and we need it for the next point)
3. When the robot is on the rim, it should push the blue squares together and throw the red balls down in order to let the sun shine on the solar panel (for this we hope to get 20 points).

Our second robot or “substitute” robot has the more complex functions:
1. The robot has to bring the Botguy to our field.
(This function gains 10 points)
2. Drive on the crater rim of the opposing field.
(This function is the most important because we will get 30 points for it and we can shut off the road from the opponent)

VII. LIVE ON THE GAME TABLE

Having built our robots it was time to put them to the test. We were very excited and apprehensive whether they would work. When it came to it, we had a lot of fun at the game table but we also had a lot of problems.

A problem we didn’t even acknowledge at the beginning was that the game table halves are not exactly the same, and if the robot worked on one half of the table it didn’t necessarily work on the other half and vice versa. We solved this problem by deciding to use two different programs.

Another problem was that the light sensor sometimes identified the brown of the game table as black when it was in the shade. But by changing the data we were able to get the robot to work correctly.

While we were in the Pria Laboratory we were able to observe other teams for the first time and could look at their robots and compare them to ours. We were able to improve the construction of our crow arm by adapting an idea from a third year team`s robot. To make our work more efficient and productive, and because we were in different classes with different timetables we decided we split up into two groups with one group concentrating on the one robot and the second group on the other robot.

VIII. WHAT WE ARE PLANNING TO DO NEXT

Up until ECER we want to make our robots more exact and reliable. After Botball we want to go on with robotics and try out tasks we were not able to fulfill on time, like collecting all

balls from the table. By next year we want to get better at programming and constructing. Now we know how we have to organize ourselves and who is better in constructing and who is better in programming. We definitely want to take part in the Botball game next year because we had a lot of fun and learned a great deal.

IX. OUR EXPERIENCE

This is the first time that we have taken part in Botball, so we didn't have any previous experience with robots, or with this programming language. This is why we had problems at the beginning, but we haven't given up and have tried to solve the problems as they arose. We learned to organize ourselves, especially after missing one deadline because a team member didn`t pass on the dates. We found out how important it is to be able to rely on each other, which team members can be counted on and who needs to work on their time management! We learned that shared leadership works best, letting each team member work in the area they are best at. We also learned that documenting the code takes longer when you write it but saves a lot of time afterwards. (For example the other teammates know what the part of code is for) We were surprised that things which we thought were easy took ages and other more complex problems were solved relatively fast.

We are really learning by doing, especially when it comes to programming, and we are really working together as a team of beginners with no input from those "in the know". The competition has been a really enjoyable challenge, and although we could have asked older more experienced students for help, (as other teams have been seen to do!) we can honestly say this is our own work, and we are quite proud of our work so far!