ITAndroids Small Size League Team Description Paper for RoboCup 2023

André Gonçalves, Andrei Freiberger, Andryus Martins, Antônio Penna, Bismarck Moreira, Bruno Yamamoto, Eduardo Simplício, Emanoel Morais, Gustavo Pironi, Gustavo da Hora, Leticia Mendes, Lucas Neves, Lucca Zoppi, Marcos Maximo, Mariana Martins, Mateus Egger, Murilo Pedroso, Nando Farias, Valerio Barros, Victor Wakugawa, and Yves Queiroz

Autonomous Computational Systems Lab (LAB-SCA)
Aeronautics Institute of Technology (ITA)
São José dos Campos, São Paulo, Brazil
Contact: `itandroids-small@googlegroups.com`
Website: `http://www.itandroids.com.br`

Abstract. ITAndroids is a robotics competition group associated with the Autonomous Computational Systems Lab (LAB-SCA) at the Aeronautics Institute of Technology (ITA). ITAndroids is a strong team in Latin America. Our Small Size League (SSL) team started its activities in early 2017, after ITAEx, an alumni association, provided financial aid and support to acquire the necessary material to build a Small Size League team. The pandemic years brought many difficulties to our team, as expected, but in 2022 we reassembled the team, made our first participation in RoboCup and six years later of the beginning, this paper describes some of our features, our recent and brief future works and our efforts on the road to becoming a well established SSL team in the Brazilian and worldwide scenarios.

1 Introduction

ITAndroids is a robotics research group associated to the Autonomous Computational Systems Laboratory (LAB-SCA) at Aeronautics Institute of Technology. As required by a complete endeavor in robotics, the group is multidisciplinary and contains about 70 students from different undergraduate and graduate engineering courses, and about 20 of these are in the SSL category currently.

Unhappily, our team suffered a lot with the pandemic, and most of the time since the return of live activities we had to arrange and relearn our previous works, since the team underwent a really strong restructuring. None of the pre-pandemic members are currently on the team, except for the professor and advisor Marcos Maximo.

Our team suffered a lot with the lack of documentation by previous members, and for this reason we started to deeply focus on documenting our projects. Before 2022, there was little documentation, so we had a lot of rework from our own projects. As we expect to keep this rework as low as possible, extensive documentation is a high priority for us.

So this paper presents our efforts in developing a Small Size team to compete in Robocup 2023 with focus to works since our Team Description Paper for RoboCup 2022 [10]. The rest of the paper is organized as follows. Section 2 explains our electronics projects and the most recent developments. Section 3 tells our works in the mechanical projects. Section 4 shows the development of our Graphical User Interface and its desired functionalities. Section 5 shows our artificial intelligence main work and most recent refinement. Section 6 shows our developments in the control area. Section 7 concludes the paper.

2 Electronics

Currently, ITAndroids SSL has been working with two projects in parallel, the 1$^{\text{st}}$ and 2$^{\text{nd}}$ generations. Both are comprised of two PCBs: the mainboard and the kickerboard. The 1$^{\text{st}}$ one is based on the open-source project of the RoboFEI team [13]. In this version, the mainboard used is the original RoboFEI design with minimal changes, and the kickerboard, whose original project also belongs to RoboFEI, has undergone changes during the last three years. We praise the RoboFEI's boards and appreciate the efforts of RoboFEI members to help us in starting our journey. However, to continue our evolution in electronics, we decided to design brand new versions of these boards for the 2$^{\text{nd}}$ generation of robots.

So our recent works on electronic design were focused on two different areas: adapting a few features of the 1$^{\text{st}}$ generation, such as connections between the mainboard and the rest of the robot, and, most importantly, finishing the next generation. We had complications in the last two years that impeded us of finalizing the new mainboard, but we managed to release its first version in the end of 2022, and for the early 2023 we will begin to develop the firmware for it.

2.1 Last steps for the first generation

As our team's plan is to replace all the robots from the 1$^{\text{st}}$ model in the near future, we intended to only fix the most critical issues of the electronics of this model.

Connections to mainboard The most impactful issue was the dribbler motor and radio connections to the mainboard. It was impossible to connect the motor because the connectors we had were different. Also, we had to connect the nRF24L01 radio to the board through jumpers, for the same reason. Then we designed two small PCBs as adapters, so we can connect the radio and the dribbler motor without problems. It fixed the poor contact issues and made robot maintenance faster.

Kickerboard During the last years, the kickerboard has undergone several changes in order to extend its lifespan. Among the modifications made, components that suffer a lot of thermal stress, such as MOSFETs and diodes, were resized. In addition, the plate area and width of most power tracks have been increased.

2.2 First steps for the second generation

The main goal of our team since RoboCup 2022 is to manufacture two robots of the 2nd version until RoboCup 2023. Our most difficult task was finishing our brand new mainboard, as we had a lot of difficulties in PCB design until 2022. Moreover, after RoboCup 2022, we also decided to change our kickerboard. Despite the changes made to guarantee more robustness to our first model, overheating was still a problem, causing the loss of many boards, then the development of a second version was justified.

Mainboard v2 In the second half of 2019, we started designing a new mainboard. Among the main changes are the FPGA replacement by the microcontroller STM32H753BI, with 480 MHz and 2 MB of Flash memory [16]. This change is due to greater familiarity in working with STM32 microcontrollers than with the FPGAs in our team [8] and the high performance of this microcontroller. Beyond that, another major change involves switching the 30 W motors to Maxon EC-45 50 W [12]. For this change, we needed to redesign the switched-mode power supply system to support a new 4S LiPo battery as well as substituting the motor drivers by the A3930 MOSFET Driver. This board also have an Inertial Measurement Unit (IMU), whose information provided to the microcontroller and the computer could further improve the robot position and speed estimation and further improve the robot motion control loop in the future.

The use of a STMicroelectronics microcontroller in the second generation Mainboard allows for a number of improvements in the robots embedded software. The new firmware is based on the last one, but it is being adapted to use Hardware Abstraction Layer (HAL) drivers and an open-source Real-Time Operating System (RTOS).

The HAL enables more agile code development at a high level and in the future it will facilitate code migration to future board versions. Besides this improvement, the implementation of a Real-Time Operating System in the robots through the open source FreeRTOS kernel is the most important upgrade from the last mainboard.

The use of FreeRTOS makes it possible for the system to meet non-trivial time requirements by establishing priorities for tasks, deadlines for their execution, and using a scheduling algorithm to choose the best task to be executed at the moment, also managing the memory being used. This results in a better temporal predictability of tasks, allowing critical tasks, such as executing controller calculations and kicking, to be guaranteed to run within a restricted time, allowing for an improvement in the code's determinism, a smoother, faster, and more stable response from the controller, greater assurance of robot and system integrity during a match, among other advantages. The focus here is not to perform tasks quickly, but rather to meet all the temporal requirements of the specification.

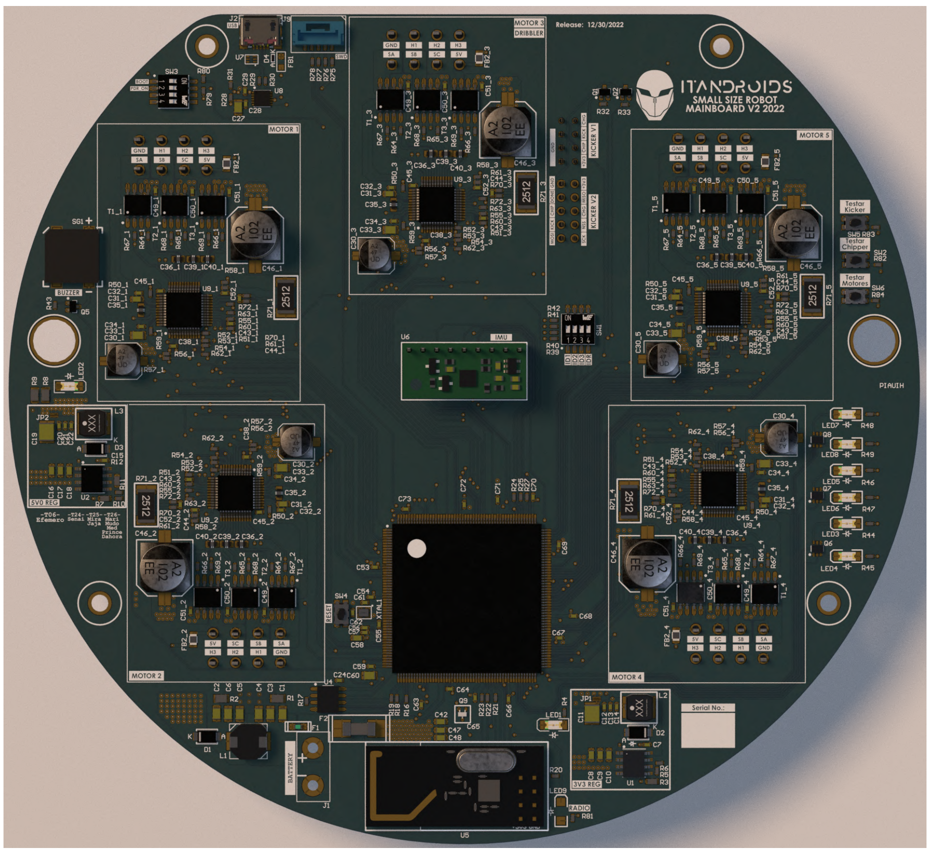


Fig. 1: A rendered image of Mainboard v2 from Altium Design.

In the end of 2022, we finished the mainboard PCB design, as shown in Fig. 1, and in the beginning of 2023 we will start to use it for first tests with firmware implementation. We expect to have a fully functional mainboard, with firmware concluded, in the middle of the first half of this year.

Kickerboard v2 For the second version of Kickerboard, as the major problem from the first one was overheating, we considerably increased its size, from 8.5 cm x 3.5 cm to 9.3 cm x 9 cm, with the purpose to better dissipate high currents and power, to avoid high temperatures on the board. The output will charge two 2200 μF capacitors in parallel to 160 V.

Also, we changed the power electronics circuit topology from a simple boost to a flyback one. For this, we started to use the LT3751 from Analog Devices [6], a high voltage capacitor charger controller with regulation, well known by SSL teams [15, 4], because it is easier to design and handle by new PCB creators. Therewith, we simulated in LTspice [5], software developed by the same Analog Devices, around 20 different circuits configurations, to obtain current, voltage and charging time values, important results to a power circuit. The most important result was the charging time of less than two seconds, with 164 V as voltage output and an average current through the battery of 2.704 A.

Another major change is MOSFET substitution for IGBT in the activation circuit, because its response to high currents results in much lower power dissipation and temperatures, mitigating even more the overheating issue from the first version. Thereby, we started to use the IGBT AIKB40 from Infineon, due to its high maximum ratings (650 V breakdown voltage and 175 °C maximum junction temperature) and high efficiency in hard switching tolopogies [7].

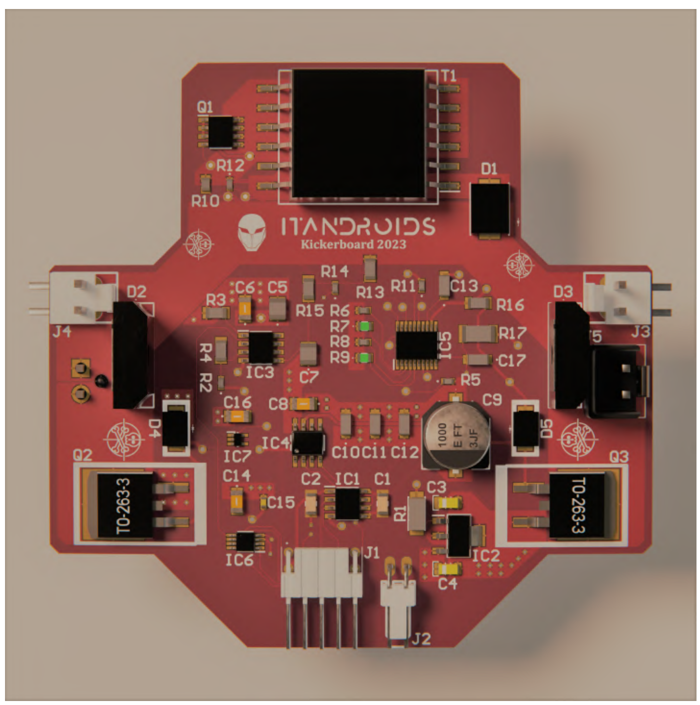


Fig. 2: A rendered image of Kickerboard v2 from Altium Design.

In the start of 2023, we concluded its design, as can be seen in Fig. 2. We expect to have the first physical prototype in the next months and play with it in RoboCup 2023.

3 Mechanics

Our recent works on mechanics are divided in two areas, like in electronics. The first one was manufacturing the parts of the second generation and some new of the first generation. We had it done by sponsors, but a few of the manufactured parts were not working as expected, specially from the second generation, so we had to fix them.

The second area is working on a new version of the second generation. We had some big issues from the first one that were only noted after completing it, which are chipper solenoid sizing, chipper mechanism and, less importantly, a few cable management problems, so we are working now on a new version of the next generation. We started to work with it recently, so unfortunately there is not a result to show yet.

4 Graphical User Interface

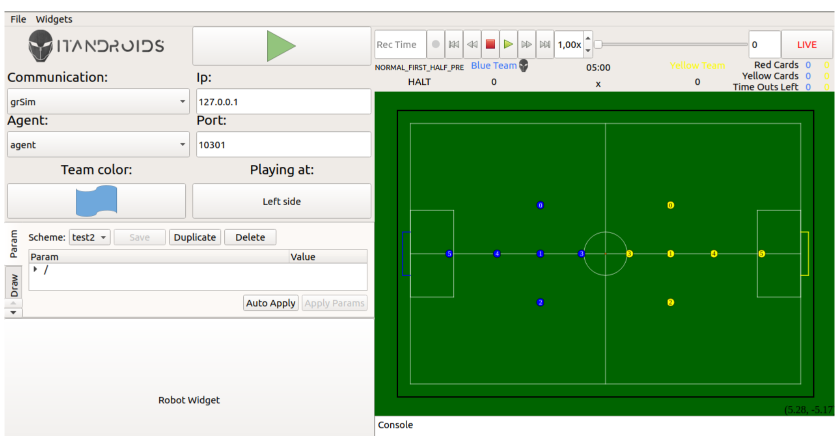


Fig. 3: Graphical User Interface

With the goal of improving testing and debugging, we continue the development of the Graphical User Interface (GUI), which contains several features for those purposes. Over the past year, the team has focused in the basic needs of it, such as real time updates for vision and game controller, communication port management, and others. The current state of our GUI may be seen in Fig.

3. We also hope to draw illustrations to enable a better understanding of each situation, helping the debugging process.

Furthermore, to enable the future analysis of our matches, the team developed a tool for storing information to be drawn during the match. However, the usage of this data is not yet fully implemented, some features such as replays features or playback functionality are still missing.

5 Artificial Intelligence

Our decision making is based on a Behavior Tree implementation, as explained in [9].

Each behavior intends to execute a specific task and implements the necessary logic for it. Behaviors have access to the world model given by the filters, as described in [9], that receive the parsed visual information and use it to decide what should the assigned player do in a medium level of abstraction, such as stay where it is, go to a given position A or aim to the opponent goal. These decisions are turned into requests that are sent to the Control layer, which implements lower-level decisions such as the wheels' velocities sent to the robots. We also implemented a Blackboard for communication between behaviors to create more complex logic.

5.1 Behaviors

The players' behavior tree, whether goalkeepers, defenders, or attackers, is made up of some basic behaviors. Among them, there are behaviors such as shooting to the goal, advancing the ball, and executing a simple pass. In some more complex ones, there is the Pass Planning, Threat Based Defense, and Delaunay Triangulation.

1. Pass planing: based on the work of the CMDragons SSL team [3], our algorithm attempts to maximize the probability of the pass being successful when searching for the best ally to pass. For this purpose, each ally receives a probability and the pass occurs to the player who has the highest chance of intercepting the ball. This probability depends on multiple factors and expecting to compute it exactly, in a dynamic domain as SSL, is unrealistic. However, we can approximate the result based on a defined set of important aspects. Currently, our algorithm works based on three main factors: the distance between the two allies involved, the distance to the closest enemy, measuring his ability to intercept the pass, and the opening angle.
2. Threat based defense: the threat-based defense is responsible for positioning the defenders and the goalkeeper against possible threats, which is done by analyzing the position of the enemy robots, the position of the ball, and its velocity. Our current algorithm is based on the work of the CMDragons SSL team [3]. The first step is to choose a primary level threat, which is the opponent with the highest risk of receiving the ball, by calculating the probability

of receiving the ball for each opponent robot. This factor depends on the velocity of the ball, the distance between the ball and the opponent robot, and the angle between these two vectors, besides a changeable constant, which was defined by previous tests. Based on the location of the enemy robot with the most risk of receiving the ball, our algorithm defines the positions of the defenders to block any open angles that could be used for a shot to our goal.

3. Delaunay Triangulation: there is the positioning behavior based on the Delaunay Triangulation, a common approach used in the Soccer 2D [2]. The idea behind it is to decide the robots' positions based on the positioning defined for some ball positions. Delaunay Triangulation can maximize the smallest angle of all triangles in the triangulation to avoid triangles with very small internal angles, and hence get a better interpolation [1].

5.2 Recent strategy changes

Based on the team last competitions, some strategy adjustments were made.

1. Support Attacker: Following the necessity to give our attacker a pass option in an attack situation, we implemented a new role on our team, the support attacker. This behavior follows the logic of helping both the attacker, to break through the defense, and the defenders, in a counter-attack situation. The behavioral change depends on the current situation of the game and the robots' positions.
2. Goalkeeper: The goalkeeper positioning and responsiveness was the main focus of the improvements made, even though we build up very basic functionalities. Within them, the main ones were the position before the enemy team shoots, which position the keeper in a way to reduce the opponent attacker angle to shoot, and the removal of the ball from the ally goal, to avoid fouls.

6 Control

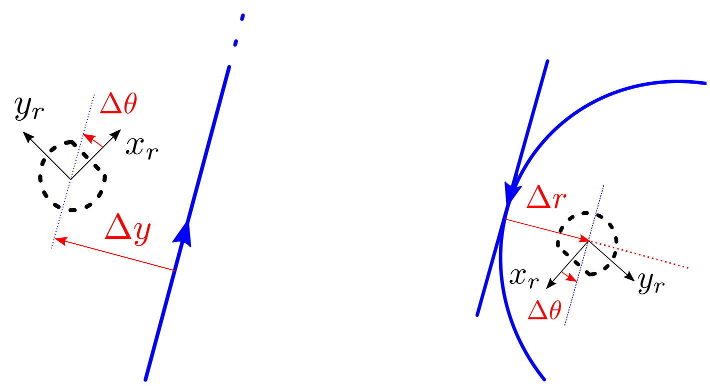
On account of the progress in our path planning algorithms, the team has been searching for better high-level control, implementing a tracking controller that uses feedback and feedforward for circular and linear paths. This model was designed to be used with recent developments in our path planning algorithm based on visibility graphs [9]. It is also imposed a limit for accelerations and speed, respecting the robot's parameters, such as the motor's saturation.

Furthermore, we are currently trying to improve our control design, implementing a simulation model in Simulink. Thus, we expect to improve the choice of gains for the robot's movement.

6.1 Tracking controller

The path planner is designed to obtain a set of linear and circular tracks, connected tangentially, to obtain a smooth transition. Furthermore, when the planner constructs a single path connecting the robot to the last goal point, it is assumed to be desired to reach this position with a zero speed.

Therefore, there are three main situations desired for the controller: two of them, linear and circular controllers, have similar approaches, with commands for feedforward (which is proposed initially considering a zero tracking error) and feedback (which commands the robot to the idealized trajectory in feedforward, as errors arise in the real movement) [11]. The third one is a waypoint controller, designed to reach the endpoint with zero speed.



(a) Line tracking. (b) Circle tracking.

Fig. 4: Possible pathways for tracking controller.

Control commands for the linear pathways are described by

$$\begin{aligned} v_t &= K_p \Delta y \sin(\Delta\theta) + v_f \cos(\Delta\theta), \\ v_n &= -K_p \Delta y \cos(\Delta\theta) + v_f \sin(\Delta\theta), \\ \omega &= K_a \Delta\theta, \end{aligned} \tag{1}$$

and for the circular pathways by

$$\begin{aligned} v_t &= K_p \Delta r \sin(\Delta\theta) + v_f \cos(\Delta\theta), \\ v_n &= -K_p \Delta r \cos(\Delta\theta) + v_f \sin(\Delta\theta), \\ \omega &= K_a \Delta\theta + v_f / R. \end{aligned} \tag{2}$$

Supposing a counterclockwise movement (if the chosen path is clockwise, an analogous control is commanded). In these equations, v_t, v_n, and ω are, respectively, the tangent, normal and angular speeds in the robot's referential. The values Δy and $\Delta\ \theta$ are feedback errors in linear and circular modes, K_p and K_a are linear and angular proportional control gains, respectively; v_f is the desired feedforward speed and R is the radius of the circular path. The representations of those situations are described in Fig. 4.

The robot has acceleration constraints due to the robot's friction and motor's limits. To determine the feedforward speed v_f, this must be taken into account. We are developing a tracking method considering these constraints, attempting to develop a path tracker with near minimum time behavior. The robot control commands will lie inside the acceleration constraints, therefore generating feasible motions [14].

6.2 High-fidelity Dynamic Simulation

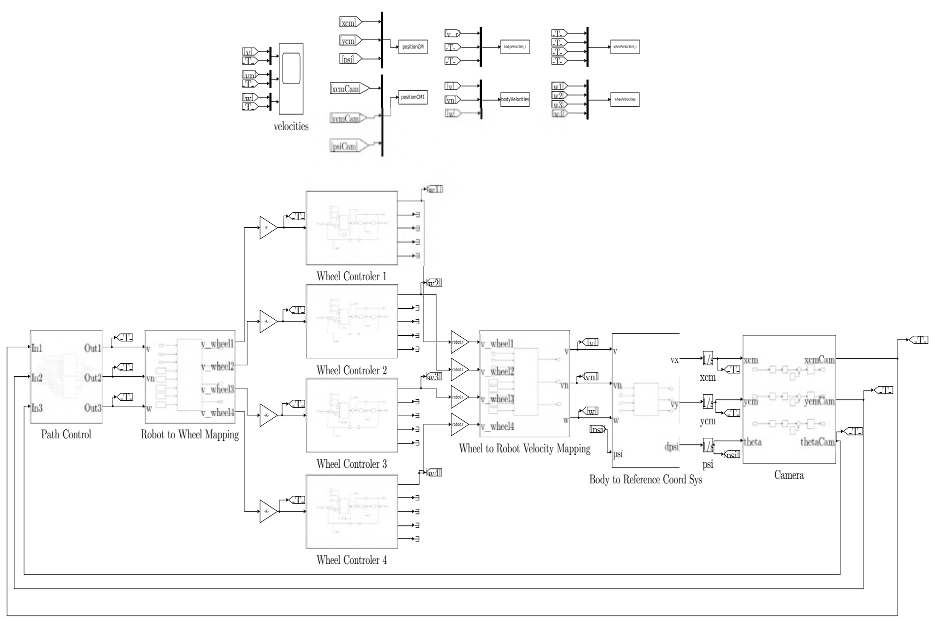


Fig. 5: Dynamic model in Simulink.

To design our control gains and general design, analyzing the effects of the real-world model, which presents extra challenges when compared to the grSim simulated model, we are designing a Simulink MATLAB model. The simulation so far is in Fig. 5, and will allow the project to be analyzed for each of our robots with simple modifications. The simulation consists of path control (for a more high-level movement) and wheel controllers (which simulate the motor dynamics and saturations, in a low-level control), besides some effects for the camera delay and noise.

7 Conclusion and Future Work

This paper describes some of our efforts during the year of 2022 and for next months until RoboCup 2023, providing some insight into how we used open-source projects to build a working team, how we start to develop our changes to fit our proposals and how we start to develop on our own. We expect to manufacture two new generation robots for RoboCup 2023, as we are in the final stages for hardware to accomplish it.

Acknowledgment

We would like to acknowledge the RoboCup community for sharing their developments and ideas. We especially acknowledge RoboFEI and Skuba for open sourcing their electronic and mechanic designs, respectively, since our 1st design is heavily based on their designs, and TIGERs for open sourcing their both electronic and mechanical designs, as our 2nd robot is inspired on them. Moreover, we would also like to thank members of CMDragons, RoboFEI and RoboIME for helping in various contexts. Finally, we thank our sponsors Cenic, FieldPRO, Intel, ITAEx, Metinjo, Micropress, Polimold, Rapid, STMicroelectronics, Wildlife Studios and Virtual.PYXIS. We also acknowledge Altium, JetBrains (CLion), Mathworks (MATLAB) and SolidWorks for providing free access to high-quality software.

References

1. Akiyama, H.: HELIOS2007 Team Description (2007)
2. Akiyama, H., Nakashima, T.: Helios base: An open source package for the robocup soccer 2d simulation. vol. 8371, pp. 528–535 (01 2014). https://doi.org/10.1007/978-3-662-44468-9_46
3. Biswas, J., Mendoza, J., Zhu, D., Choi, B., Klee, S., Veloso, M.: Opponent-Driven Planning and Execution for Pass, Attack, and Defense In a Multi-Robot Soccer Team **1**, 493–500 (01 2014)
4. Bos, L., Citgez, Y., Dorenbos, H., Eichler, A., van der Hulst, R., Nagelvoort, L.K., van der Kuil, T., Luong, J., Poort, L., Prinsenberg, F., de Regt, C., Uiterkamp, L.S., Schrijver, R., Steerneman, E., Vacariu, P., van Werven, J., van der Werff, S., Zwerver, S.: RoboTeam Twente Extended Team Description Paper 2020
5. Devices, A.: All things LTSpice. `https://www.analog.com/en/design-center/design-tools-and-calculators/ltspice-simulator.html`
6. Devices, A.: LT3751 - High Voltage Capacitor Charger Controller with Regulation. `https://www.analog.com/media/en/technical-documentation/data-sheets/LT3751.pdf`
7. Infineon: Datasheet AIKB40N65DH5. `https://www.infineon.com/dgdl/Infineon-AIKB40N65DH5-DataSheet-v02_01-EN.pdf?fileId=5546d462749a7c2d0174b69731a33216`
8. Justa, A., Azevedo, A., Herculano, D., Vacarini, D., Tanomaru, F., Silva, I., Filho, J., Steuernagel, L., Maximo, M., Ângelo, M., Levi, M., Fonte, P., Aki, R., Tonaco, T., Araujo, V.: ITAndroids Humanoid Team Description Paper for RoboCup 2019

9. Maranhão, A., Schuch, A., Azevedo, A., Rodrigues, A., Lima, E., ao Sarmento, J., Santos, M., Maximo, M., Sales, M., Dias, M., Lima, R.: ITAndroids Small Size Soccer Team Description Paper for RoboCup 2020
10. Maranhão, A., Schuch, A., Freiberger, A., Albani, A., Rodrigues, A., Simplício, E., Benicio, G., Haddad, L., Zoppi, L., Rezende, L., Maximo, M., Egger, M., Moreira, M., Silva, M., Celente, R., Lima, R., Augusto, V., Wakugawa, V.: ITAndroids Small Size Soccer Team Description Paper for RoboCup 2022
11. Maximo, M.: Model Predictive Controller for Trajectory Tracking by Differential Drive Robot With Actuation Constraints (2015)
12. Maxon: Maxon EC-45 Flat Brushless Motor 50W Datasheet. `https://www.maxongroup.com/medias/sys_master/root/8833813119006/19-EN-263.pdf`
13. Pucci, D., Jr., F.R., Schunk, C., Braga, C.J., Torres, V., Silva, T., Amaral, V., Silva, A.D.O.D., Jr., J.A.G., Bianchi, R.A.C., Tonidandel., F.: RoboFEI 2014 Team Description Paper
14. Purwin, O., D'Andrea, R.: Trajectory generation and control for four wheeled omnidirectional vehicles. Robotics and Autonomous Systems **54**(1), 13 – 22 (2006). https://doi.org/https://doi.org/10.1016/j.robot.2005.10.002
15. Ryll, A., Jut, S.: TIGERs Mannheim Extended Team Description for RoboCup 2020
16. ST: Stm32h753xi datasheet. `https://www.st.com/resource/en/datasheet/stm32h753bi.pdf`