# Project Proposal for ROBOCUP

## (RoboCup Small Size League 2016)

TEAM **SPAREPARTS** Lead by Mr. **Waqas ur Rehman**

Center for Advanced Studies in Engineering, Islamabad
Department of Electrical and Computer Engineering

This proposal explains the mechanical, electrical and software designs of our project. The performance of well-founded wireless communication is being focused in this project. It also includes accurate motion planning and locomotion procedures for our fleet of six (6) robots. We have implanted encoders to improve the movement in terms of precision and accuracy. In order to handle the runtime movement of the robot different types of algorithms will be implemented.

## 1) Robots Specifications:

| | |
|---|---|
| Robot Diameter | 175 mm |
| Robot Height | 150 mm |
| Ball Coverage | 18 % |
| Max. Linear Velocity | 3.2 m/s |
| Weight | 2.0 kg |
| Maximum kick speed | 7 m/s |
| Limited kick speed | 7.5 m/s |
| Maximum chip kick distance | 7.0 m |
| Maximum ball speed catching | 6.0 m/s |

Table 1: Table of specifications

## 2) Electrical System:

Following is the detailed description of our electrical design.

### Velocity Control:

We used a closed loop system for our velocity control. The block diagram of implemented design is given below.

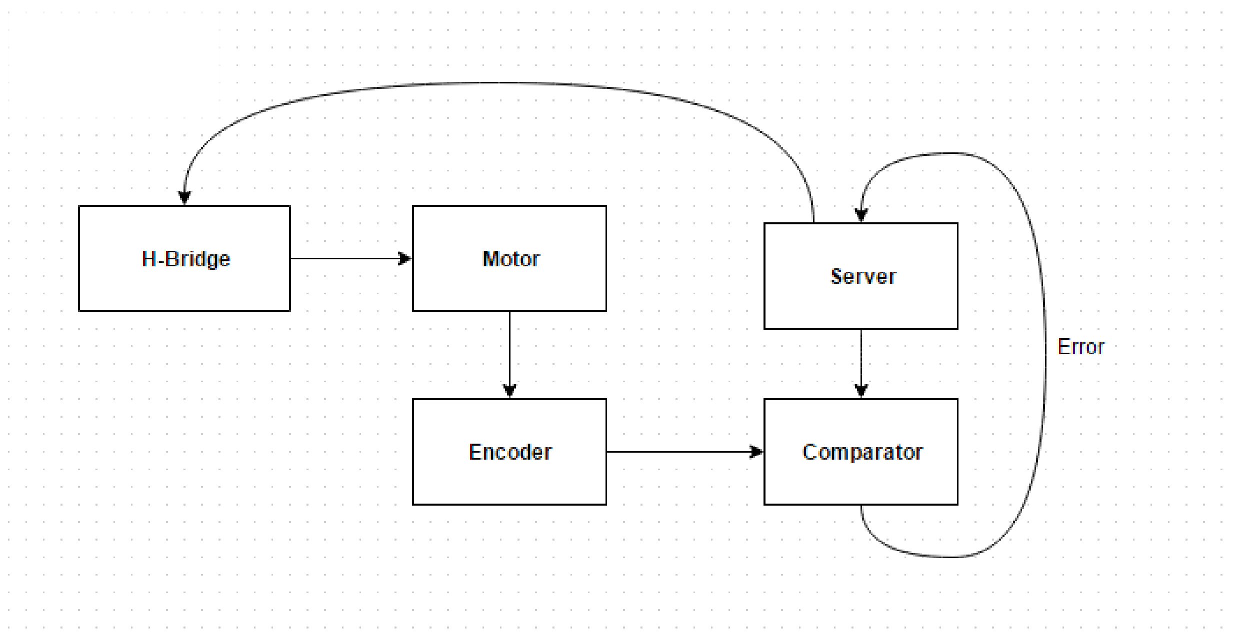


Figure 1: Block diagram.

### Motors:

For driving our robots we are using 12V 3A DC motors on each wheel. Each robot consists of 4 drive motors while an additional motor for the dribbler.

### Motor Drives:

LM298 is used as our motor driver. The driver is small in size and is economically effective and also has an advantage of good performance.

### Central Processing Control:

For the central processing, we are using “Microchip”. The Microcontroller that we used is “P33FJ64MC802”. We are using 4 modules of PWN of our microchip so each of the motor is controlled by each of the module. We are using UART module for communication between our server and the robot.

**Circuit board**

**-Zigbee (Telegesis):**

Zigbee is a wireless communication module. We decided to use this module to make communication between our server and robots. The reason to choose this communication module is because it’s easy to operate and debug.


Figure 2: Zigbee Wireless Module.

**-Kicker Circuit:**

Kicker circuit is a separate circuit which is also known as boost circuit. We used PIC12 microcontroller temporarily for the Boost circuit but for further improvements i.e. implementing the snubber circuit, we will choose PIC16. We are using this microcontroller for monitoring the capacitive boost circuit charging levels.

**Schematic Design**:

Following is the schematic of the kicker circuit.

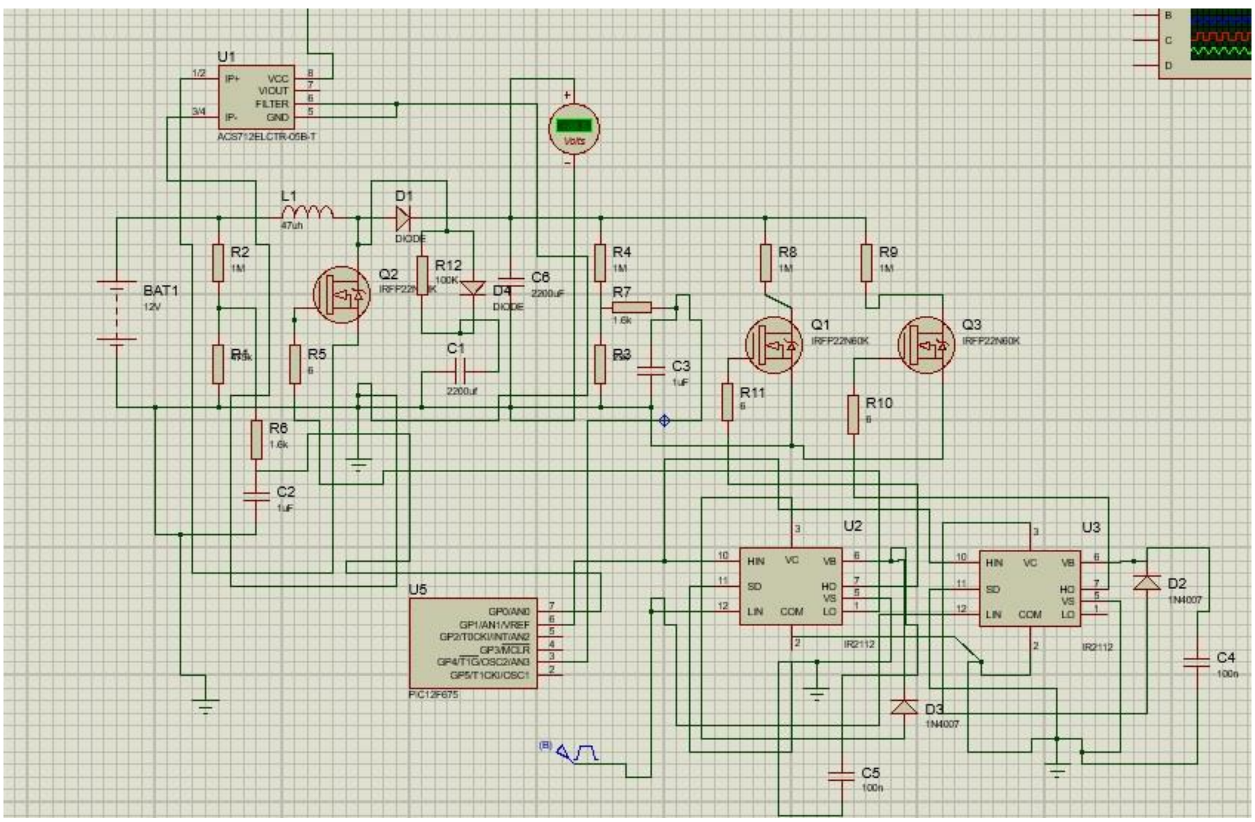


Figure 3: Kicker Simulation Schematic Diagram

**Mechanical System:**

Following figure shows the 3D Design of Wheel assembly and chasses.

Figure 4: 3D Design of Complete wheel assembly and Chasses.

**Main Parts**

The main mechanical parts of our robot design are as following.

1. Omni Directional Wheels
2. Main Mechanical Structure (Base of Robot)
3. Kicker Assembly
4. Chipper Assembly
5. Dribble

**Main circuit board connector:**

In our design, complete electronic circuitry is mounted just above the Drive Plate. This electronic circuitry is composed of the main controller board with onboard motor drivers. We have mounted this circuitry with the jerk absorbing rubber washers.

**Jerk absorbers for kicker jerks:**

When a kicker shot is missed, a lot is energy is to be absorbed by the chassis of the robot. We have also installed thick rubber jerk compensators for absorbing sudden jerks.

Figure 5: 3D Design of Kicker with Rubber Washers at both ends.

2) **Software**

Our software part consists of a Main Module which is further divided into following modules

1. AI Module
2. Communication Module

## AI Module

The AI Module consists of the entire major program. Our path planning algorithm, data filtering, data smoothening and decision making are done in this module.

**Path Planning (Ball Tracking):**

Previously in Robocup 2014 and Robocup 2015, our seniors were not using any particular technique for path planning. For robust path planning, we have implemented RRT (Recursive Root Tree Algorithm).To reduce the planning delays we have implemented many path custom smoothing techniques. After RRT, we have also implemented RRT * (RRT Star) which has helped us a lot to reduce the processing delay.

We are also preprocessing the whole field reducing and scaling our field instead of checking and processing on every valid point on the field. We have used “NODE SKIPPING” to smooth the path during obstacle avoidance; we are actually using this technique by populating a set of points from robot to its target. Then checking the direct path condition on every node, we keep on skipping the previous node if next node makes a direct link to the source robot. A small illustration of our concept is shown below.

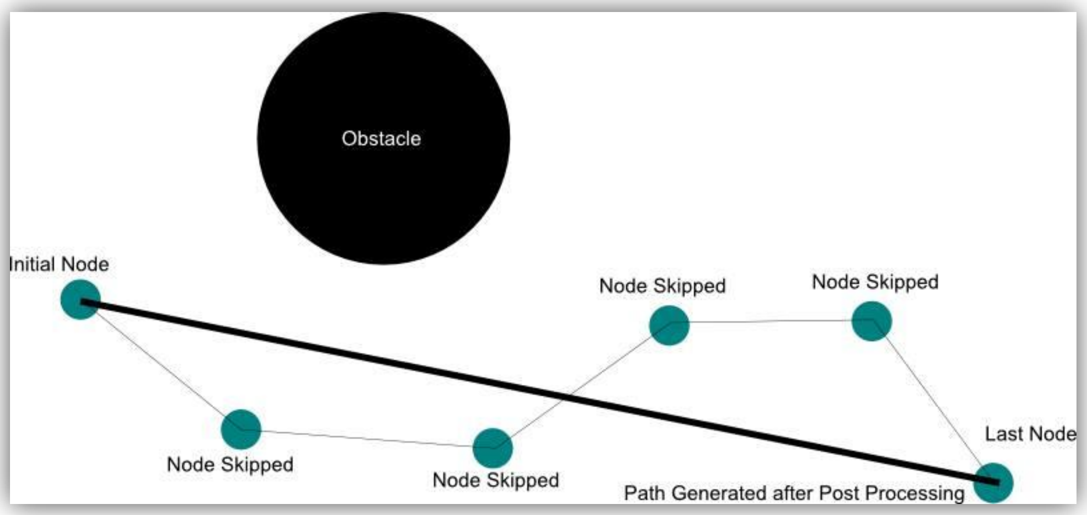


Figure 6: Node Skipping Methodology for path smoothing