# Freescale CUP 2015

# POLIedro team

Report for Freescale CUP competition

Team members:

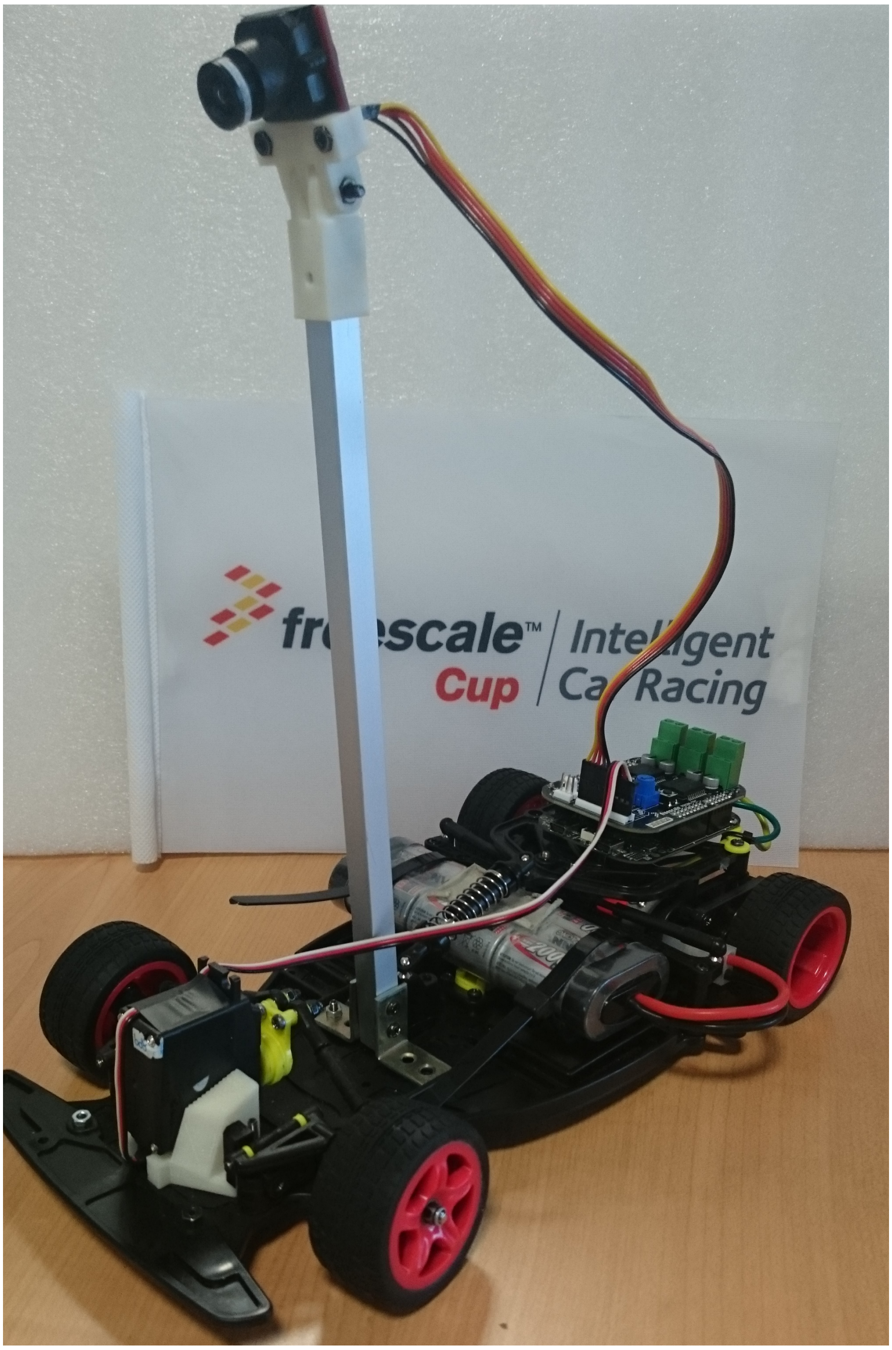
Edoardo Cavallotto, Emanuele Gianoglio, Luca Spinella.

# Index

## 1. Introduction

This report describes the features of the car developed by POLIedro team from Politecnico di Torino for the Freescale Cup competition. The car is an autonomous vehicle that drive in real time following the two black lines placed on the edges of the white track.

The entire project is done by using Freescale products in conformity to the EMEA 2015 rules.



# 2. Mechanical design of the vehicle

## 2.1 Chassis

The chassis of the car is the one given by Freescale.

In the rear part of the car there are the two DC motors. In the center of the vehicle there is an alluminium pipe mounted perpendicular to the chassis in order to place the line scan camera through a special PLA support designed and printed. In the front of the car there is the servo motor, used for the steering control, placed on a special PLA support designed and printed in order to put it vertically instead of horizontally.

Between the pipe and the motors is placed the NiMh battery.

## 2.2 Weight and dimensions

- Length: 28 cm
- Width: 17 cm
- Height: 29,5 cm
- Weight: 0,97 Kg

# 3. Electronics

## 3.1 Boards

### 3.1.1 Main Board

The main board of the system is the Freedom KL25z produced by Freescale.

This board is equipped with a KL25 family 32bit microcontroller that is an ARM Cortex M0+ running at 48 MHz.

### 3.1.2 Shield Board

On the top of the main board there is the shield board provided by Freescale, used to drive the DC motors, the servo motor and the line scan camera.

On the main board we have done some modifications in order to improve the available ports of the peripherals. For example we have reached the outputs of certain timers originally wired to the RGB LED mounted on the board.

## 3.2 Linear Sensor array

The basis of the driving capability of the car is the line scan camera provided by Freescale. The camera consists in a CMOS linear sensor array of 128 pixels (TAOS TSL1401CL) and a 65° optical lens.

The sensor works with the following three signals:

- SI (Serial Input) - single pulse which defines the start of the data-out sequence.
- CK (Clock) - timing signal that control the internal charge transfer and the pixels output.
- AO (Analog Output) - analog signal that rapresent the pixel output from the sensor.

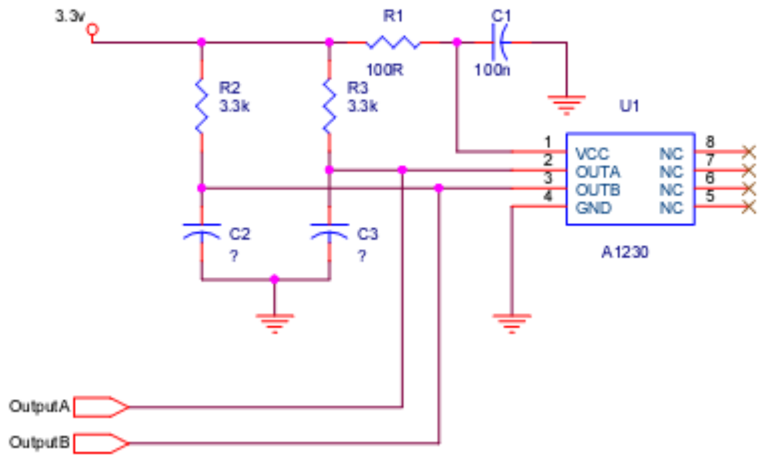
The choosen integration time for the optical sensor is 1,5 ms.

The sensor power supply is provided by the microcontroller (0 V - 3,3 V).

### 3.3 Speed sensor

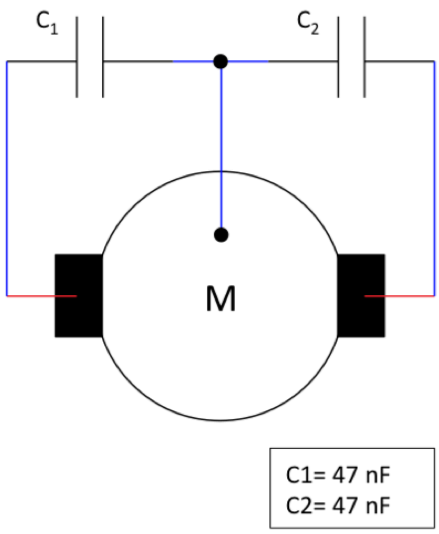
In order to control the speed of the car we designed a speed sensor system based on Hall effect sensor (Allegro MicroSystem A1230) and 48 neodymium magnets put around the wheel in a PLA printed support. The sensor is mounted on a dedicated board and fixed on the chassis near one of the rear wheels.

Here below are shown the rendering of the PLA ring and the schematic of the system:



### 3.4 Noise filtering

In order to reduce the risk of having disturbs on the signals, we decided to limit the noise produced by the two DC motors, driven at higher values of current with respect to the rest of the system. So we decided to fix the issue by mounting two capacitors of 47nF on the two poles and case of each motor as shown here below.




Moreover, we put 3 filtering capacitors (2x 10nF on the outputs, 1x 100nF on the power supply) to prevent noise on the signals provided by the speed sensor in order to limit the error while reading the speed.

# 4. Control Software Design

## 4.1 Processing image

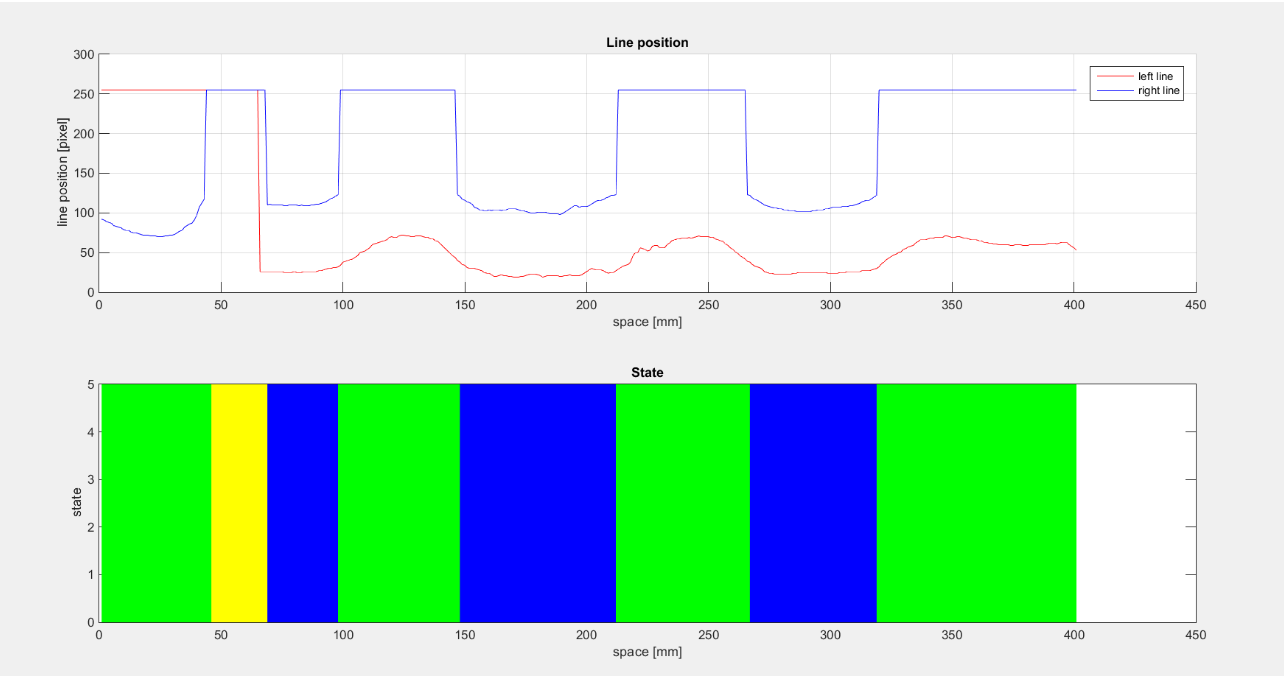
The output signal (AO) of the camera is acquired through the ADC and processed by the microcontroller through a derivative based algorithm in order to determine the position of the two black lines delimiting the track area.

This information is used by the designed software to determine if the car is moving in a straight segment, in a curve, a cross or a hill. Depending on these conditions the software determines the desired speed and the steering angle of the front wheels.

The software analyzes the driving condition every 6mm according to the information taken from the speed sensor and schedules the new driving condition state thanks to a state machine algorithm.

Here below are shown two plots rapresenting an example of a section of the track. The first, on the top, shows the two lines positions in terms of pixels (left line in red, right line in blue). The second on the bottom shows the corresponding state scheduled by the algorithm:

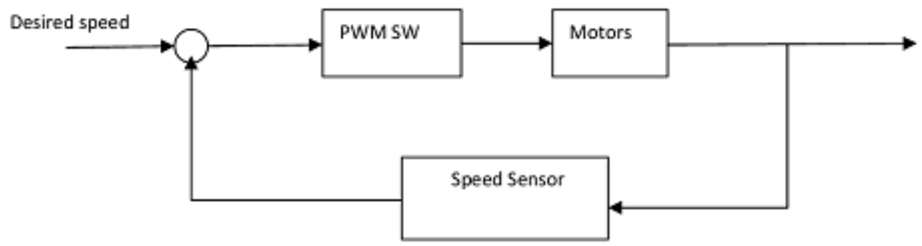
- Straight -> blue area
- Curve -> green area
- Cross -> yellow area



## 4.2 Speed control system

Speed sensor is used to measure the speed and actuate a software speed control mechanism, optimized for having the fastest possible response.

Here a block diagram showing how the system works.




## 4.3 Steering

The software calculates the center reference depending on the lines position and processes this information using a steering algorithm. The algorithm is PD based (proportional/derivative).

Here a diagram showing how the algorithm works: