# TABLE OF CONTENTS

2. MECHANICAL DESIGN 4
   2.1 GENERAL 4
   2.2 CHASSIS 5
   2.3 STEERING SYSTEM 5
   2.6 SPEED SENSORS 7

3. ELECTRONIC DESIGN 8
   3.1 BLOCK DIAGRAM 8
   3.2 SENSORS 8
   3.3 COMMUNICATION DEVICE 9
   3.4 MAINBOARD 9
   3.4.1 MODULES DESCRIPTION 9
      1. μC : ARM Cortex-M4 9
      2. H-Bridges : MC33887APVWR2 9
      3. Led Driver : IS31LT3360 10
      4. WiFi Module : RS9110-N-11-2X 10
   3.5 SCHEMATICS 11
      3.5.1 SPEED SENSORS 11
      3.5.2 WIFI MODULE 11
      3.5.3 LINE SCAN CAMERA 12
      3.5.4 MAINBOARD 13
   3.6 PRINTED CIRCUIT BOARD 16
      3.6.1 SPEED SENSORS 16
      3.6.2 WIFI MODULE 16
      3.6.3 LINE SCAN CAMERA 17
      3.6.4 MAINBOARD 17

4. CONTROL SOFTWARE DESIGN 18
   4.1 SOFTWARE ARCHITECTURE 18
   4.2 USER INTERFACE 20
   4.3 VISION SYSTEM 21

5. CHARACTERISTICS OF THE CAR 22
   5.1 DIMENSIONS AND WEIGHT 22
   5.2 POWER CONSUMPTION 22
**LEGEND TABLE**

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 1</td>
<td>Top view of the car</td>
<td>4</td>
</tr>
<tr>
<td>Figure 2</td>
<td>Side view of the car</td>
<td>4</td>
</tr>
<tr>
<td>Figure 3</td>
<td>Bottom view of the chassis</td>
<td>5</td>
</tr>
<tr>
<td>Figure 4</td>
<td>Bottom view of the steering mechanism</td>
<td>5</td>
</tr>
<tr>
<td>Figure 5</td>
<td>Lighting system /w camera</td>
<td>6</td>
</tr>
<tr>
<td>Figure 6</td>
<td>Top view of the mainboard</td>
<td>6</td>
</tr>
<tr>
<td>Figure 7</td>
<td>Speed Sensor</td>
<td>7</td>
</tr>
<tr>
<td>Figure 8</td>
<td>Block Diagram of the car</td>
<td>8</td>
</tr>
<tr>
<td>Figure 9</td>
<td>CPU Specifications</td>
<td>9</td>
</tr>
<tr>
<td>Figure 10</td>
<td>Top view of the mainboard /w Description</td>
<td>10</td>
</tr>
<tr>
<td>Figure 11</td>
<td>Speed sensor Schematic</td>
<td>11</td>
</tr>
<tr>
<td>Figure 12</td>
<td>Wi-Fi module Schematic</td>
<td>11</td>
</tr>
<tr>
<td>Figure 13</td>
<td>Line scan camera Schematic</td>
<td>12</td>
</tr>
<tr>
<td>Figure 14</td>
<td>Mainboard Schematic (1 of 3)</td>
<td>13</td>
</tr>
<tr>
<td>Figure 15</td>
<td>Mainboard Schematic (2 of 3)</td>
<td>14</td>
</tr>
<tr>
<td>Figure 16</td>
<td>Mainboard Schematic (3 of 3)</td>
<td>15</td>
</tr>
<tr>
<td>Figure 17</td>
<td>Speed sensors PCB</td>
<td>16</td>
</tr>
<tr>
<td>Figure 18</td>
<td>Wi-fi module PCB</td>
<td>16</td>
</tr>
<tr>
<td>Figure 19</td>
<td>Line scan camera PCB</td>
<td>17</td>
</tr>
<tr>
<td>Figure 20</td>
<td>Mainboard PCB</td>
<td>17</td>
</tr>
<tr>
<td>Figure 21</td>
<td>Software Architecture</td>
<td>19</td>
</tr>
<tr>
<td>Figure 22</td>
<td>State machine in gCompute.c</td>
<td>20</td>
</tr>
<tr>
<td>Figure 23</td>
<td>Switch use table</td>
<td>20</td>
</tr>
</tbody>
</table>
1. INTRODUCTION

The following technical report explains the mechanical and software specificities of our car used in the NXP Cup, which is an international competition organized by NXP. The competition is a race between little smart cars that have to stay on a track and complete a lap as fast as possible.

The competition implies embedded programming, PCB designing and system regulation. As the PCB and most of the hardware were provided, our team mainly had to do the software part. This competition and the work put into it stands for our semester project.

Our team “ARCar1” is composed by Aurélien eckerlen and Lionel Schaming, two students engaged in a trinational schooling in France, Germany and Switzerland. We study engineering in embedded systems and we are actually doing our third year in the “Haute-Ecole Arc”.

At the time this report is written, we already won (1st place) the internal qualifications of our school, in which three teams competed.

This project is very exciting and we have a lot of pleasure working on it. We hope that we can go as far as possible in the competition!

The whole project was supervised by four teachers, who have been of great help: Philippe Amez-Droz, Yves Meyer, Serge Monnerat and Patrice Mueller
2. MECHANICAL DESIGN

2.1 GENERAL

Figure 1: Top view of the car

Figure 2: Side view of the car
2.2 CHASSIS
We received a basic chassis with mounted wheels. The rules of the competition do not allow any modification of the chassis, therefore we didn’t change anything here.

Figure 3: Bottom view of the chassis

2.3 STEERING SYSTEM
The steering system is composed of controls arms connected to a Futaba S3010 (servomotor), which was imposed.

Figure 4: Bottom view of the steering mechanism
2.4 LIGHTING AND CAMERA

The track’s lines are scanned by a linear camera (1×143 pixels) coupled with a lighting system. The camera and the LED units used for the lighting are mounted at the top of a mast fixed at the front of the car. The camera and lighting’s tilt can be changed manually.

![Figure 5: Lighting system /w camera](image)

2.5 MAINBOARD AND POWER SUPPLY

The mainboard is placed between the two rear wheels, above the two DC motors. Its position allows an easy access to the speed sensors and the DC motors. The power supply comes right after the mainboard. As it is a heavy component, we found better to have it at the center of the car so that the weight of the car is well distributed.

![Figure 6: Top view of the mainboard](image)
2.6 SPEED SENSORS

Two speed sensors, one for each rear wheel, are located on each DC motor. A magnet is stucked on each motor’s axis right after the transmission system. The speed sensor stands between the wheel and the motor, with its two Hall Effect sensors around the magnet.

Figure 7: Speed Sensor
3. ELECTRONIC DESIGN

3.1 BLOCK DIAGRAM

![Block Diagram of the car](image)

Figure 8: Block Diagram of the car

3.2 SENSORS

1. Melexis MLX75306 – Linear Digital Camera

   The line scan camera is a linear sensor array of 143 pixels with 100µm height and 50µm pitch. The communication with the camera is done with an SPI interface. The focal can be adjusted manually by screwing the lens.

2. Honeywell SS443R – Speed Sensor

   The Honeywell SS443R digital Hall-effect sensor is operated by a permanent magnet. It responds to a magnetic South pole by setting up a signal. We use a speed sensor board for each rear wheel and two SS443R sensors for each board, so that we can get the direction of rotation of each wheel.

3. Xtrinsic FXOS8700CQ – 6-Axis Accelerometer and Magnetometer

   There is a FXOS8700CQ on the mainboard. Actually we don’t use these sensors because we couldn’t manage to get relevant informations.
3.3 COMMUNICATION DEVICE

1. Redspine Signals RS9110-N-11-22 – Wifi Module

   This module allows us to send and receive data on a wireless ad hoc network. The mainboard dialogues with this module using a RS232 communication. We use this module in test mode to send data in real-time from the car to the PC for monitoring purposes and to send start and stop signals from the PC to the car. This device is removed for the competition.

3.4 MAINBOARD

   The mainboard has been entirely designed inside of our school. Its MCU is an NXP MK64FN (the processor is an ARM Cortex-M4). An extension board has been developed specifically for the NXP Cup Worldwide Final to change the H-Bridge. Indeed, the original were two TLE6209R and this component is forbidden for the worldwide final, so we changed them with two NXP H-Bridges, the MC33887APVWR2.

3.4.1 MODULES DESCRIPTION

1. μC : ARM Cortex-M4

   The MCU is an NXP MK64FN (the processor is an ARM Cortex M4). The Cortex M4 processor is based on the ARMv7 Architecture. The CPU frequency is 120MHz with 1MB Flash and 260KB SRAM.

<table>
<thead>
<tr>
<th>Freescale part number</th>
<th>CPU frequency</th>
<th>Pin count</th>
<th>Package</th>
<th>Total Flash memory</th>
<th>Program flash</th>
<th>EEPROM</th>
<th>SRAM</th>
<th>GPIO</th>
<th>Tamper Detect</th>
<th>Program Flash Swap</th>
</tr>
</thead>
<tbody>
<tr>
<td>MK64FN1M0VLL12</td>
<td>120 MHz</td>
<td>100</td>
<td>LQFP</td>
<td>1 MB</td>
<td>1 MB</td>
<td>—</td>
<td>280 KB</td>
<td>66</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

   Figure 9: CPU Specifications

2. H-Bridges : MC33887APVWR2

   There are two H-Bridges on the board, each one individually controls a motor. The 33887 is a monolithic H-Bridge Power IC with a load current feedback feature making it ideal for closed-loop DC motor control. The IC incorporates internal control logic, charge pump, gate drive, and low RDS(ON) MOSFET output circuitry. The 33887 is able to control inductive loads with continuous DC load currents up to 5.0 A, and with peak current active limiting between 5.2 A and 7.8 A.
3. **Led Driver : IS31LT3360**

   In order to drive the red LED on our car we use an IS31LT3360 which is a step-down converter, designed for driving a single LED efficiently from a voltage source higher than the LED voltage. The chip operates from an input supply voltage between 6V and 40V and provides an output current of up to 1.2A.

   There are two outputs. A simple ON/OFF output and an output with a variable current.

4. **WiFi Module : RS9110-N-11-2X**

   On our main board we can optionally plug a Wi-Fi module. Wi-Fi communication is used for our monitoring via Labview for the test in real time. Naturally it will be removed from the board during the NXP cup official race.

   The WIFI module is referenced as “RS9110-N-11-2X”.

   We can communicate with this module either with SPI or UART. In our case, we use UART.

3.4.2 MAINBOARD’S PICTURE DESCRIBED

![Figure 10: Top view of the mainboard /w Description](image-url)
3.5 SCHEMATICS

3.5.1 SPEED SENSORS

![Speed Sensor Schematic]

Figure 11: Speed sensor Schematic

3.5.2 WIFI MODULE

![Wi-Fi Module Schematic]

Figure 12: Wi-Fi module Schematic
3.5.3 LINE SCAN CAMERA

Figure 13: Line scan camera Schematic
3.5.4 MAINBOARD

Figure 14: Mainboard Schematic (1 of 3)
Figure 15: Mainboard Schematic (2 of 3)
3.6 PRINTED CIRCUIT BOARD

3.6.1 SPEED SENSORS

Figure 17: Speed sensors PCB

3.6.2 WIFI MODULE

Figure 18: Wi-fi module PCB
3.6.3 LINE SCAN CAMERA

Figure 19: Line scan camera PCB

3.6.4 MAINBOARD

Figure 20: Mainboard PCB
4. CONTROL SOFTWARE DESIGN

4.1 SOFTWARE ARCHITECTURE

The software is written in language C and is based on a 3-layers model:

1. Interface Layer
   The lowest layer is the one which interfaces the hardware peripherals. The largest part of this layer was provided by NXP, like the microcontroller configurations, the control of the servomotor or the digital input and output. We added some features like SPI for accessing the camera and control the H bridge, UART for accessing the wifi module and digital-to-analog converter to control the LED driver.

2. Module Layer
   The middle layer makes the peripherals work, using the interface layer. It controls the lighting luminosity, the acquisition of the image from the camera, the control of the motors through the HBridge, the control of the servomotor, the measure of the speed of each motor and the control of the wifi module.

3. Manager Layer
   The highest layer, which is the application layer, controls the direction and the speed of the car, and uses the Wifi module for the monitoring of the car by a computer through a Java program. This layer is based on four managers. There are an input manager, which acquires the sensors values, a compute manager, which filters the input and computes the output values, an output manager, which updates the actuators with the computed values and send data through the wifi module to the Java program on the computer so we can view the state of the car.
The car’s speed depends of a state machine into gCompute.c. The next figure shows a simplified view of the state machine:
The speed is regulated differently depending on the actual state. During the curve we give a bigger order to the exterior (the turnSpeed value) motor than the interior one, so as not to skid. And during the straight line we give the StraightLineSpeed value to the two motors. See this two value in the array below.

### 4.2 USER INTERFACE
The mainboard has three switches that are used in our software to set some parameters.

<table>
<thead>
<tr>
<th>Switch</th>
<th>Name</th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>StraightLineSpeed</td>
<td>Disabled</td>
<td>Read potentiometer number 1</td>
</tr>
<tr>
<td>2</td>
<td>TurnSpeed</td>
<td>Disable</td>
<td>Read potentiometer number 2</td>
</tr>
<tr>
<td>4</td>
<td>StartSwitch</td>
<td>Disable</td>
<td>Enable</td>
</tr>
</tbody>
</table>

*Figure 23: Switch use table*
4.3 VISION SYSTEM

As indicated in the last chapters of this report, we use a linear digital camera that gives us a 143x1 pixels array. The value of those pixels may be between 0 (darker) and 255 (lighter). The only operations we do on the original image is a threshold.

Threshold

If the value of the pixel is greater than our value of threshold then, it would be interpreted as a white pixel (255). In the other cases, it would be a black pixel (0).

With the threshold, our state machine is easier, because we have just two values of light intensity per pixel and less disturbance with for example the shadow on the race circuit.
5. CHARACTERISTICS OF THE CAR

5.1 DIMENSIONS AND WEIGHT

Length: 293 mm
Height: 298 mm
Width: 165 mm
Weight with battery: 1080 grams

5.2 POWER CONSUMPTION

<table>
<thead>
<tr>
<th>Label</th>
<th>Current</th>
<th>Power</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mainboard idle</td>
<td>120 mA</td>
<td>0.9 W</td>
</tr>
<tr>
<td>LED</td>
<td>240 mA</td>
<td>1.7 W</td>
</tr>
<tr>
<td>Mainboard + LED + Motors at full power¹</td>
<td>6.3 A</td>
<td>45.4 W</td>
</tr>
</tbody>
</table>

¹ We made these measures when the car wasn’t on the track for practical reasons. In order to imitate the track’s friction (requiring more current than when the wheels are free), the wheels were held a little bit manually. When the wheels were not held, the current consumption was almost 1.1A.