Wi-Fi Controlled Car

Pre-analysis

This document contains information about the components and the decisions about hardware

Document title	Pre-analysis
Document ID	WCC-Pre-analysis
Authors	Ambroise Dhenain Szymon Klepacz Anatolii Shakhov Alvaro Garcia Pierre Le Texier
Supervisor	Torben Gregersen
N^{o}. pages	13
Date	30-08-2013

Document history

Date	Audit	Author	Description
30-11-2013	1.0	SzK	Information about hardware.
04-12-2013	1.1	AS	Updated information, titles and style.
06-12-2013	1.2	AS	Redesigned document style, added more information about communication.
08-12-2013	1.21	SzK	Added information about hardware, corrected spelling
10-12-2013	1.3	AS	Improved with conclusions after each part of analysis. Minor spelling fixes.
11-12-2013	1.4	SzK	Improved battery description, corrected spelling
11-12-2013	2.0	AD	Final version.

Approval

Author(s)	Ambroise Dhenain – 201301255 [AD] Szymon Klepacz – 201301204 [SzK] Anatolii Shakhov – 201301534 [AS] Alvaro Garcia – 201301220 [AG] Pierre Le Texier – 201301944 [PLT]
Project name	Wi-Fi controlled car
Document ID	WCC-Pre-analysis
No. pages	13

By signing this document both parties accept, that this is the requirements for the development of the desired system.

Place and date: 13-12-2013

Authors

Supervisor

Contents

Tables of figures

1. Description

1.1. Purpose

The purpose of this document is to explain the choices of hardware and software.

Here we will tell about the decisions made during the assembling part, testing and our group meetings. The detailed information about used parts and how they should be connected to each other is described in the document "WCC-Design".

1.2. Reading instruction

Chapter 2:	Contains information about communication methods
Chapter 3:	States the board choices and all the main components
Chapter 4:	Describes the power supply options
Chapter 5:	Overview of main components of the car
Chapter 6:	References for images and additional resources

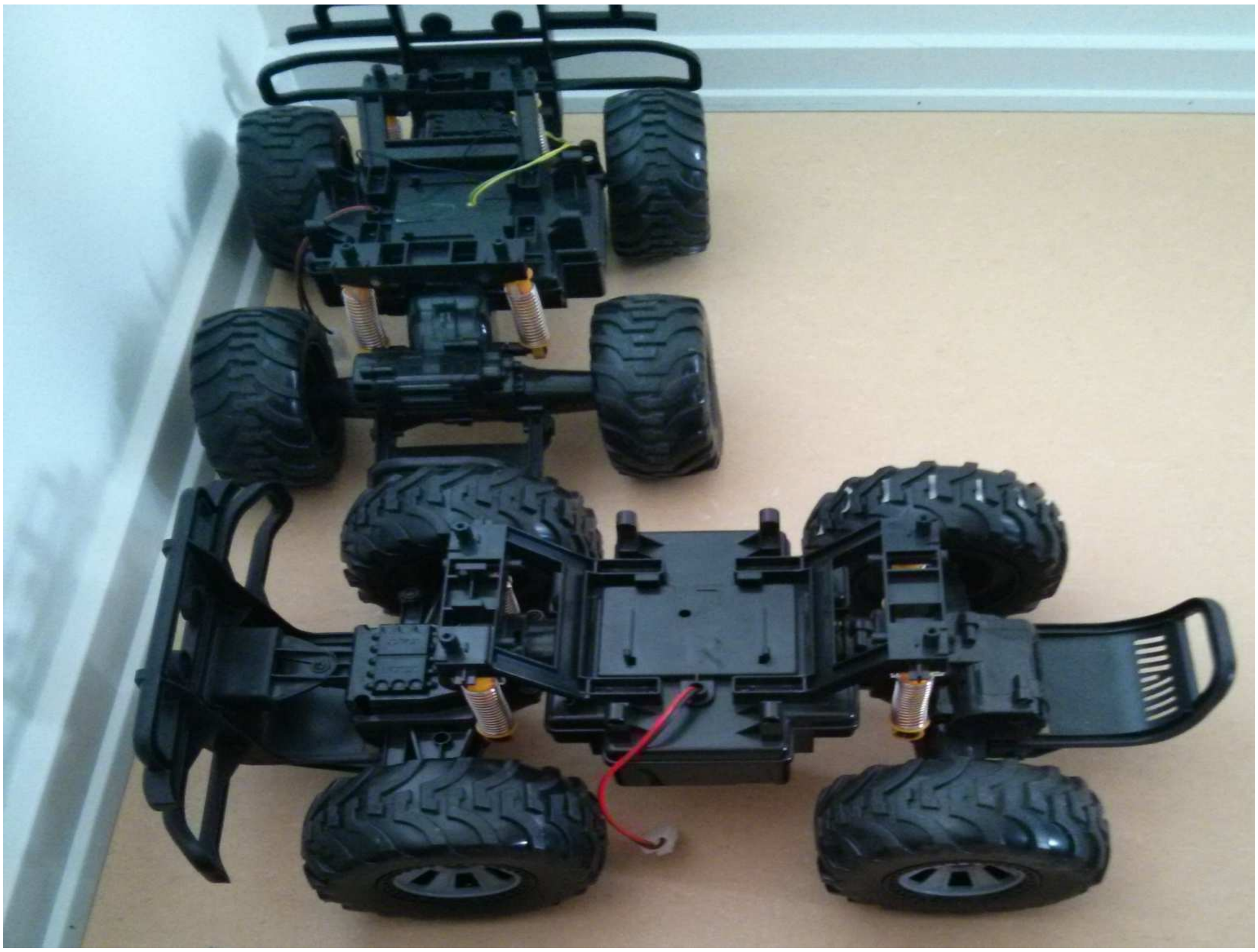

Figure 1 - Cars provided by the University

2. Communication

2.1. Communication

2.1.1. Wired

2.1.1.a. Wires

Wires are commonly used to establish a static connection. They provide a high-speed data transfer and reliable connection between the recipient and sender. In this project our main object (the car) is moving, so this type of connection would not fit all the needs. As well as movement restrictions, the car would rely on the length of the cable too much when it need mobility.

2.1.2. Wireless

2.1.2.a. Bluetooth

Bluetooth connection is an old technology but still popular nowadays. Its' popularity is as high as the Bluetooth module can be found on almost every modern device (mobile phones, smartphones, gps devices, etc.). It doesn't allow so high speed for transfer of data like wires but it would be enough to send commands to car. The worst drawback in this type of wireless connection is its' short range. Bluetooth usually operates at frequency rates lower than 2.4 MHz. We could use it, but we want stronger signal (Bluetooth concedes Wi-Fi technology in data transfer range) to let car moving further while connected to the phone.

2.1.2.b. Radio

Most wireless controlled toy cars are controlled by the radio remote controls. Those remote controls have a frequency range from 25 up to 75 MHz. This is why they can cover a long range and the user may control the device on a long distance. They can have a variety of antennas (depending on the budget), different designs and functions. One of the advantages of using the remote control is a variety of choice and a huge range to control the car. But it has disadvantages – the remote control may be an expensive choice and it could be really hard to organize. We had courses in programming for mobile devices but didn't have engineering subjects. The time frame for the project is not enough to create the remote control from scratch or use any existing one.

2.1.2.c. Wi-Fi

Our final decision was made for a Wi-Fi technology. This type of technology is very popular and every modern smartphone has a built-in Wireless module. It can be accessed from our board (it will be described later) and we can easily send commands from the phone to our car. The only drawback is a range, which is quite small compared to the Radio remote controls. But this is the price to build a working system using our knowledge gained from the courses.

2.1.3. Conclusion

Our final decision on connection technology was made for Wi-Fi. We decided to use it because it's very popular and every modern smartphone has it. Wi-Fi does not provide a huge cover range like Radio technology described, but it has a range which is enough to move the car for appropriate distances from user. Wi Fi technology is easy to use and easy to implement in our project.

3. Main Board

3.1. Main Board

3.1.1. Raspberry Pi

One of the smallest computers with Linux OS on board. The architecture of Raspberry Pi is based on an ARM processor. This tiny PC includes a lot of modern technologies built together to provide another level in hardware and software developing. The board has a high performance, in comparison to its' small size. One of the best things in Raspberry Pi is a great community support – hundreds and thousands people all around the Globe. In the bookstores and libraries we can already find a special literature based on this board and over the Internet a huge amount of tutorials, FAQs and forum topics available. The very low price comes as another great advantage of this system (approx. 40 euros for a board and 90 euros for a development kit). To use a Raspberry Pi we would need some additional components: SD Card, HDMI Cable, Wi-Fi modem, Power Supply and more. This powerful board would be even more than enough for our project, but into it the board would not use all of its' features and power. This board could be good for a project which is much bigger than ours. And it would also be better for a people programming in Python. During our studies here we learnt Android and Windows Phone systems, meaning that it would be better for us to use Java and C programming languages. We can apply the knowledge straight to the real project.

3.1.2. ARM

High performance microcontrollers. Used in Raspberry PI, phones and professional projects. Deficient tutorials and documentation compared to AVR microcontrollers. Most of IDEs are paid. Need special programmer. Not suitable for our project because of high additional costs and low support for beginners.

3.1.3. AVR

Small, cheap and very popular version of microcontrollers. May be programmed in Assembler or C language. We started our project from the second part of the semester – we didn't know much about the language and programming style when made a decision on the hardware. The AVR family may cause some inadmissible problems for us: Internet connection, communication by sockets, taking pictures etc.

3.1.4. Netduino

The board we used on the Windows Phone architecture course. Rather new and growing community but still hard to find different tutorials on the connection between different components. Also it requires additional WiFi shield to establish connection between the board and the phone (we decided on the wireless connection). Based on STMicro 32-bit microcontroller. The platform uses NET Micro Framework.

3.1.5. Arduino

Arduino has been placed on the list of the biggest invention in electronics of 21st century. Friendly IDE based on C/C++ programming language. Big community, thousands tutorials, dozens of board choices and one of the most important – it is OPEN SOURCE.

We chose the newest one – Arduino YUN. The board has ATmega32u4 microcontroller and AR9331 processor. Microcontroller controls Arduino board and processor to have access to Wi-Fi and USB host. Devices communicate by bridge. It provides ability to run shell scripts, communicate with network interface and receive information from AR9331.

Another option was to buy Arduino Leonardo and Wi-Fi shield. But this options cost more. Also communication is not as simple as by YUN. As an extra – YUN has a built-in USB port.

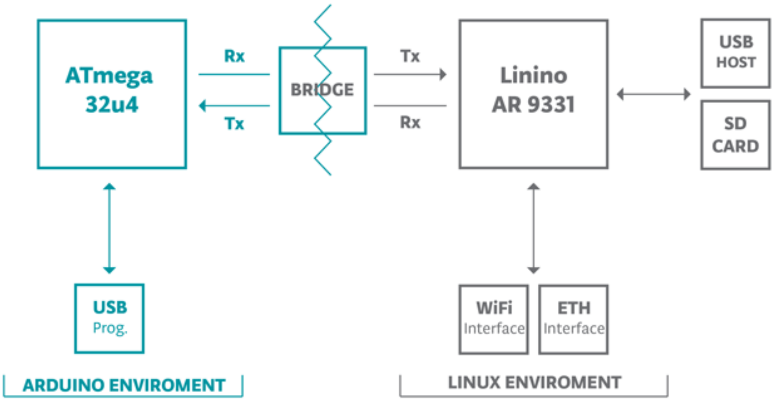

Figure 2 - Arduino board schematics

3.1.6. Conclusion

We decided to choose Arduino because this board is already created (we don't need to create our board), this board has a great support from different users (huge community), it has a built-in Wi-Fi (the reason why we decided to take this particular model of Arduino) and it has all the functionalities we need in the project. Other boards would be more expensive or just useless for us.

3.2. Shields

3.2.1. Arduino Motor shield R3

Allowed us to control only 2 motors. Based on L293 driver.

3.2.2. Adafruit Motor shield v2

Allows us to control max 4 DC-motors, 2 servo motors or 4 steppers. Use I2C protocol to communications, what allow to use 32 shields. Instead of L293D driver (Used in Arduino Motor shield R3) has TB6612 MOSFET driver. Cheaper than Arduino shield and more powerful.

3.2.3. Build our own

Each team member had at least basic knowledge in programming but very poor In hardware building. Thus we decided to choose an Adafruit device instead of creating out own from the scratch.

3.2.4. Conclusion

We decided to choose the Adafruit motor shield because it is a cheap and perfect option available for us. It is compatible with Arduino, can support more motors than Arduino motor shield and has more power than another one. Building an own shield would be too complicated for this project and we decided to use the existing shield.

3.3. CAMERA

3.3.1. uCam tll

This camera is really low level, we have it in a documentation. But in order to make it work we needed to control it using only Arduino. It could have disturbed the functionality of the car. Moreover it was impossible to have a proper stream.

3.3.2. Microsoft LifeCam HD-300

USB camera. Can be connected to a USB port of Arduino and controlled by a Linino OS on the board. The Microsoft camera was a later decision – first of all the idea was to use any web camera, but we didn't know if it will work with Arduino. After some unsuccessful tries with connecting an old camera and the absence of choice we searched through Arduino forums. In one of the topics some developer posted an answer that he was able to use the camera together with the video stream and making snapshots. So our final decision on camera was made just a few weeks before the project hand in and now it works perfectly.

3.3.3. Conclusion

We decided to use a Microsoft camera – it is easier to set up and it works with the processor of Arduino – does not loses memory on the board and we can send video stream (not just static pictures) on our smartphone.

3.4. Car

3.4.1. Build our own

Too hard to implement, takes too much time and nobody in the crew has knowledge about built a robot or a car from scratch.

3.4.2. University car

It is already disassembled and tested by previous students. Two working motors and a box to store the battery. Wires from motors which can be easily connected to our motor shield.

3.4.3. Conclusion

It was a better option to use the university car – saving money and time to work on a real project.

3.5. Phone

3.5.1. Windows Phone

Good platform for developing Applications but may be a small trouble because the platform still lacks of some functionalities. Not as versatile as an Android. Can be problematically to program on Windows phone without the smartphone on this OS. Only one member in our team had a Windows Phone device.

3.5.2. IPhone

Powerful and popular. Unfortunately nobody in our group have IPhone and never programmed for this phone. To program an IPhone we would need apple computer. Also, it is not an open source system so it would be a hard way to do the program for it.

3.5.3. OS Firefox

New system for Phone with a lot of bugs. Not popular. Programming based on HTML5. Every app can be opened in browser.

3.5.4. Android

The most popular OS in Europe, million Phones and tablets based on Android. Open source. Learnt in the second part of semester. User friendly, a lot of tutorials and a huge community. Many free samples and fast support. Best choice for our project and this part of it can be used in the ITSMAP course.

3.5.5. Conclusion

We decided on Android because it has a lot of opportunities and we have this course during the project. We can apply our knowledge on practice and make the program we like. Android has a great support communities so if we had any troubles – it would be easy to find an appropriate answer.

4. Power supply

4.1. Battery

We didn't find any data sheet for the car and found out that it supports up to 9.7V – we used a 9.7V battery with a charger for it.

4.1.1. Alkaline

The most popular, energy efficient and have almost this same voltage all the time when it works. But they cannot be recharged. The car engines and an Arduino board with motor shield consume a lot of energy – using non-rechargeable alkaline batteries would be too expensive and non-productive.

4.1.2. Ni-MH (nickel metal hydride)

Rechargeable batteries. They have less capability than a simple battery used in remote controls, watches etc. They lose some energy even If not in use. This battery has only one great disadvantage: when it is fully charged – it has a higher voltage. Full 9V battery can have even 11,2V. Because battery life is not very important for our project, we decided to use them for test. They were available at University. For the contest instead of Ni-MH we would use Li-P.

4.1.3. Li-Po (lithium polymer)

Rechargeable batteries. They are lightweight, powerful and save capability. Disadvantage is high price and the need in special charger.

4.1.4. Conclusion

We decided to use Ni-MH type of batteries because they provide the functionality we need for the board.

4.2. Build 5volt regulator

We built a voltage regulator from components available at University. The most important element is voltage regulator. We are using TS7805 substitute of LM7805. Popular and cheap component. Output voltage 5.00±0.25 V. Input voltage between 7 - 25 V. This regulator is very important to supply an Arduino board. When we connect a board to the shield – we use our 9V battery to supply the shield. Also we will need to supply an Arduino board (otherwise the first tests were done with Arduino connected through wire).

5. Conclusion

5.1. Main components

The result of our pre-analysis is:

1. Arduino YUN board with built-in Wi-Fi module to provide connection between the board and the phone
2. Adafruit motor shield to establish a connection between an Arduino and two motors (responsible for turning left/right and moving forward/backward)
3. Microsoft LifeCam HD-3000 to connect to Arduino board using USB port
4. A car skeleton from University (including two engines, four wheels and a plastic frame with battery box)
5. An Android smartphone
6. Ni-MH rechargeable battery and a charger for car
7. A set of wires and tools for assembling components
8. Voltage regulator for Arduino YUN
9. Alkaline battery for Arduino YUN

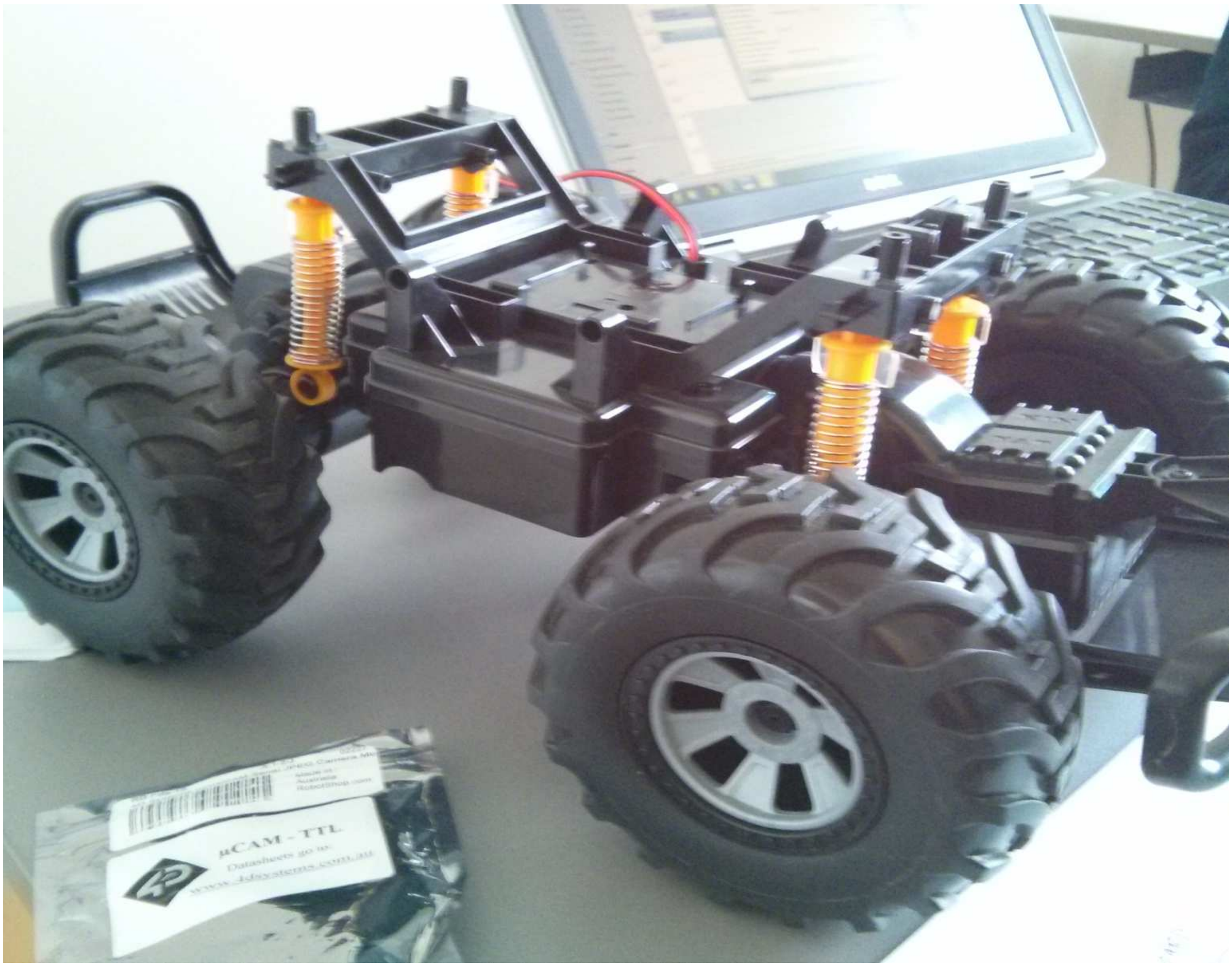

Figure 3 - The car we finally borrowed from the University

6. References

- **Arduino**:
  - http://arduino.cc/en/Main/ArduinoBoardYun?from=Main.ArduinoYUN