**ARMON: ARDUINO BASED BLUETOOTH CONTROLLED ROBOT ARM**

Requirement Specification Document

**22.11.2017**

**Revision 2.0**

**By: Onur Doğan ATAÇ**

**Revision History**

|  |  |  |
| --- | --- | --- |
| **Revision** | **Date** | **Explanation** |
| 1.0 | 02.10.2017 | Initial requirements |
| 2.0 | 21.12.2017 | Requirements Model |

İZMİR, 2017

**Contents**

**Revision History…………………………………………………………………...2**

**Contents…………..........................................................................................................3**

1. **INTRODUCTION**

## Functional Description**……………………………………………………….5**

## Robot Arm Details and Summary**……………………………………………5**

## Problem Statement and Proposed Solution………………….……………….**5**

# **REQUIREMENTS**

## Software Requirements**……………………………………………………….6**

## Dependencies on other systems**………………………………………………6**

## Trajectory planning requirements**…………………………………………….6**

## Other requirements on the software **……………………………………….....6**

## Robot requirements**……………………………………………………………7**

## Economy**……………………………………………………………………...7**

## Requirement List**……………………………………………………………..8**

1. **ACTORS AND USE CASE**

## Block Diagram**………………………………………………………………10**

## Sequence Diagram**…………………………………………………………..11**

## Use case diagram**……………………………………………………………11**

# **Glossary………………………………………………………...12**

# **References……………………………………………………………………..13**

# **INTRODUCTION**

The main structure of this project is to design a robot arm that can be rotated as four degrees of freedom while controlled by android mobile phone and direction of the robot can be show as LED indicators. The motion of the robot arms are done by using mobile app that has Bluetooth as smart phone, so the control of the robot is done by wirelessly through smart phone.

Android is has many great option for exchanging data by using Wi-Fi, Bluetooth, and Wireless such as cellular connection that named as GPRS, EDGE, 3G. Today millions of people have been using Android because of the android carry a big size of range of useful libraries and even progressive development of the applications are easy. Now a day Bluetooth is more common for the smart phone, work as short length connection between two data like short radio frequency that commit wirelessly. The controller of robot is Microcontroller or in the other is Arduino that use Bluetooth as connection between robot and smart phone as input to the controller, and direct current motors are interface to the microcontroller [1].

Since 1954 Design a robot had become by George Deyol after that period of years the designer have been trying for design a better robot. For instance, the circuitry, degree of freedom (DOF), algorithm, program, attachments, equipment, accuracy and speed, completely depend on the designer’s tact. The big reasons why they are trying to make a great robot because they try to make a fantastic robot arm work as human’s hand actions and even do something humans hand cannot do it.

****

**Figure 1. Armon.**

## **Functional Description**

The workstation for each robot arm consists of the following components:

* PC with MatLab and Aria C
* Motor with Quanser Control System
* Linear Power Amplifier
* Robot arm with Gripper

- OWI-535 Rotary Servo Plant

* A robot arm will also include an OWI-535 Rotating Flexible Link to add another degree of freedom.
* The inverted robot arm comprises a rotary encoder and the level robot arm comprises a potentiometer for measuring the position.
* One robot arm will be configured vertically in a pendulum-like fashion to incorporate the effects of gravity on the arm. A closed-loop PID control system will be implemented in Aria C and will use Quanser software to allow real-time control of the robot arm through ARIA C.

**System Inputs:**

* Internal Commands (position)
* Potentiometer Position Feedback (2 DOF arm configuration)
* Rotary Encoder (inverted arm configuration)

**System Outputs:**

* Position

## **ROBOT ARM DETAILS AND SUMMARY**

Actually, the controller of the robot are designer that apply many thing on the device that can let the robot arm can be able to grab, pick up and leave it down and also play a robot as have a four motion of direction according to the weights and shape of the objects. Generally robot arm use 4 to 5 servo motors for the operating [2]. So the details and material methods are given down.

## **PROBLEM STATEMENTS AND PROPOSED SOLUTION**

The main problem as we can during the projects we can say that finding the true materials as needed is difficult and after we founded that materials crease the four degrees of freedom are more critic because control of the DC motor are difficult and even much of the code used for Arduino IDE are wrong, so if we got a problem with DC motor rotation probable come for the motor driver L293D because the correct connection of the of motor driver is difficult and if we make any mistake with connection the motor driver may burn. So final solution of the projects we have to be careful and we have to be sure for everything before used [3].

# **REQUIREMENTS**

## **Software Requirements**

This section specifies the requirements on the software and is divided into four subsections: Target requirements, Trajectory planning requirements, Navigation and localization requirements and other requirements on the software.

## **Dependencies on other systems**

The software is specifically built for the wheeled robot. In the event that the

software will be used on a different robot or if sensors are replaced, software modifications will most likely be necessary.

## **Trajectory planning requirements**

This section lists requirements regarding the trajectory planning

* Whenever a new fixed obstacle is detected, the software shall be able replant the route if necessary
* To make the robot avoid moving obstacles, the software shall be able to re-plan the route.
* The software shall be able to find and plan the shortest, for the robot possible, route. The software shall be able to handle right-hand traffic when planning the routes.

## **Other requirements on the software**

This section lists other requirements on the software.

* The software shall be written in IDE or MATLAB.
* The software shall be written in IDE.
* There shall be a possibility to maneuver the robot using an external computer.
* There shall be a GUI displaying the estimated map and current robot pose.
* Via the GUI the user shall be able to place an end point on the displayed map-making the software re-plan the current route to this point.
* There shall be a possibility to use wireless communication.
* The external computer shall be able to communicate with the onboard laptop.

## **Robot requirements**

* This section lists requirements regarding the robots physical movement.

Given the initial position on the route, the robot shall be able to follow the route and avoid moving obstacles by slowing down when closer than 60cm from the center of the robot, and stopping when closer than 30cm.

* If the end position of a route is a location, the robot shall be able to stop within 2 decimeters from it. The error of the robot heading is insignificant.
* If the end position is arbitrary, the robot shall be able to stop within 4 decimeters from it. The error of the robot heading is insignificant.

The robot shall not be able to cross-restricted areas in the map.

* If the end position is arbitrary, the robot shall be able to stop within 2 decimeters from it. The error of the robot heading shall be less than 50 degrees.
* If the end position is arbitrary, the robot shall be able to stop within 1 decimeter from it. The error of the estimated robot heading shall be less than 25 degrees.

## **Economy**

This section lists requirements regarding economic limitations.

* The project must not take more than 1920 hours. This includes time for lectures, education, meetings, document writing, media work and development of the product.
* One member of the group shall spend equally much time on the project.

## **Requirement List**

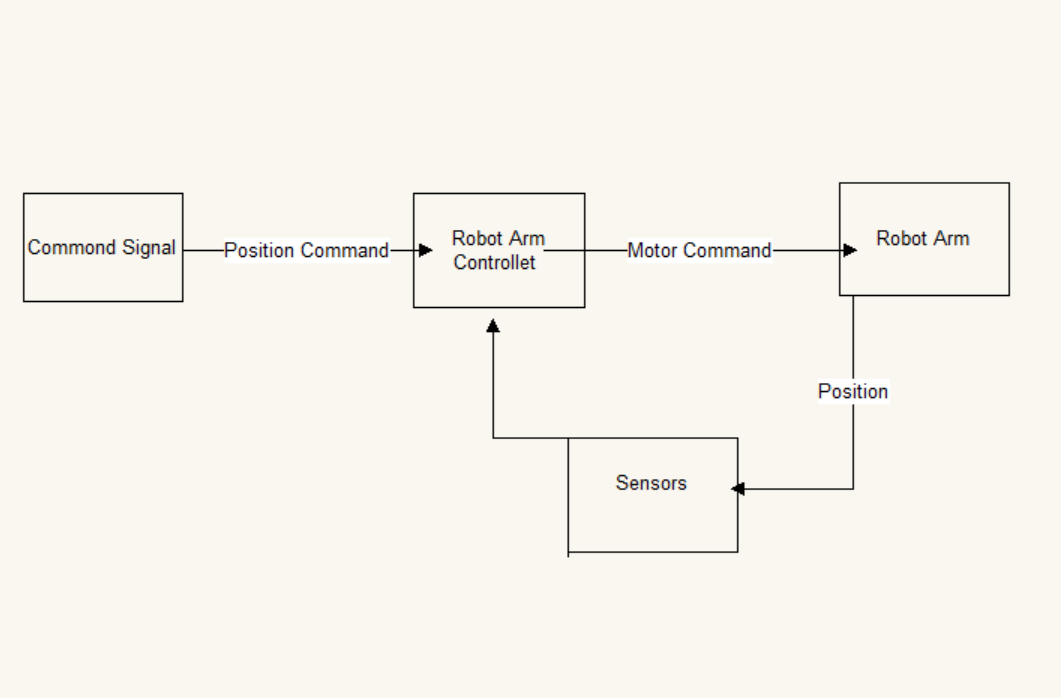
|  |  |  |
| --- | --- | --- |
| Requirements | No: | Short Statements of requirements |
| Trajectory planning requirements |  | Whenever a new fixed obstacle is detected, the software shall be able replant the route if necessary |
| Trajectory planning requirements | **1** | Whenever a new fixed obstacle is detected, the software shall be able replant the route if necessary |
| Trajectory planning requirements | **2** | To make the robot avoid moving obstacles, the software shall be able to re-plan the route. |
| Trajectory planning requirements | **3** | The software shall be able to find and plan the shortest, for the robot possible, route. The software shall be able to handle right-hand traffic when planning the routes. |
| Other requirements on the software | **4** | The software shall be written in IDE or MATLAB. |
| Other requirements on the software | **5** | The software shall be written in IDE. |
| Other requirements on the software | **6** | There shall be a possibility to maneuver the robot using an external computer. |
| Other requirements on the software | **7** | There shall be a GUI displaying the estimated map and current robot pose. |
| Other requirements on the software | **8** | Via the GUI the user shall be able to place an end point on the displayed map-making the software re-plan the current route to this point. |
| Other requirements on the software | **9** | There shall be a possibility to use wireless communication. |
| Other requirements on the software | **10** | The external computer shall be able to communicate with the onboard laptop. |
| Robot requirements | **11** | Given the initial position on the route, the robot shall be able to follow the route and avoid moving obstacles by slowing down when closer than 60cm from the center of the robot, and stopping when closer than 30cm. |
| Robot requirements | **12** | If the end position of a route is a location, the robot shall be able to stop within 2 decimeters from it. The error of the robot heading is insignificant. |
| Robot requirements | **13** | If the end position is arbitrary, the robot shall be able to stop within 4 decimeters from it. The error of the robot heading is insignificant. |
| Robot requirements | **14** | The robot shall not be able to cross-restricted areas in the map. |
| Robot requirements | **15** | If the end position is arbitrary, the robot shall be able to stop within 2 decimeters from it. The error of the robot heading shall be less than 50 degrees. |
| Robot requirements | **16** | If the end position is arbitrary, the robot shall be able to stop within 1 decimeter from it. The error of the estimated robot heading shall be less than 25 degrees. |
| Economy | **17** | The project must not take more than 1920 hours. This includes time for lectures, education, meetings, document writing, media work and development of the product. |
| Economy | **18** | One member of the group shall spend equally much time on the project. |

# **ACTORS AND USE CASE**

## **Block Diagram**

The top level block diagram of the project is shown in Figure 3. Command signal

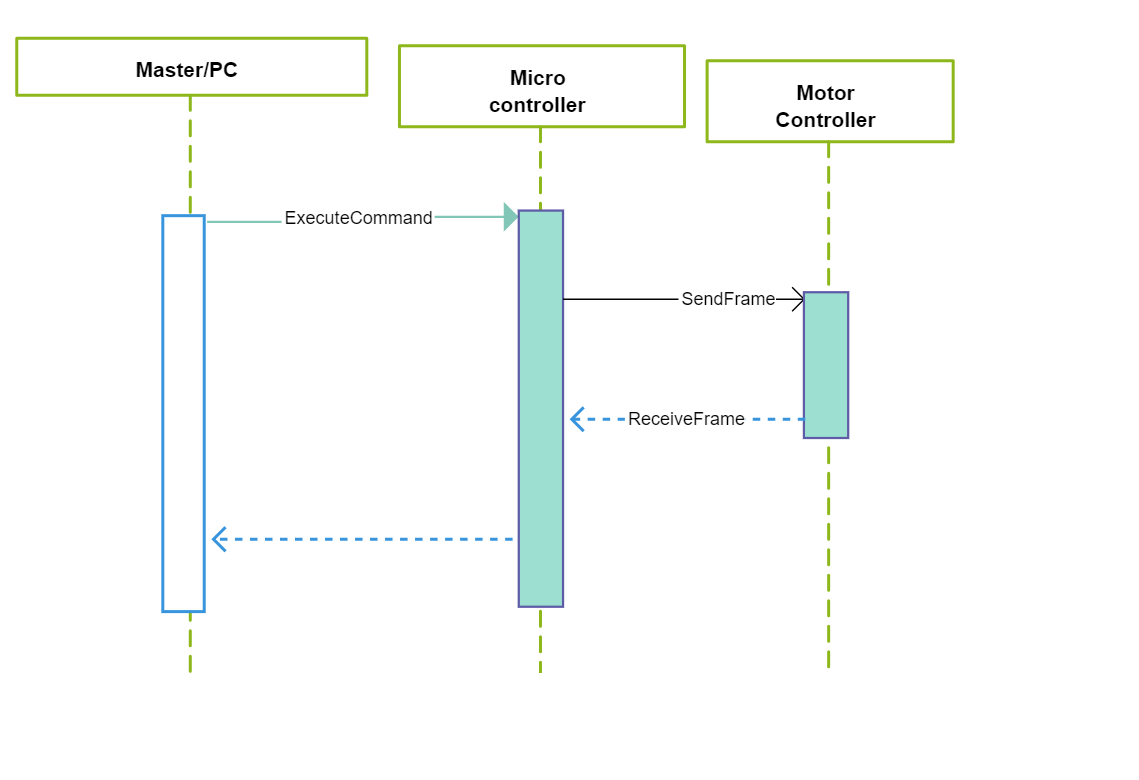
It is placed in Simulink and then a signal is sent to the applied arm controller; then sends the signal to the arm. The sensors connected to the arm will then control device that allows closed loop control. Power electronics related to robot arm control, robot arm and sensors, external malfunctions voltage source noise, load changes, including friction and quantization error.

**Figure 2. High-Level Block** **Diagram**

The command signal to the system, we have a value in Aria C! This value plus or minus 90 degrees. Position command, Controller over Simulink. The controller will generate a digital control signal. This is a D / A converter. This location the arm will be measured in two different ways on two platforms. 2-DOF platform use a potentiometer to measure the position. The analog position signal is

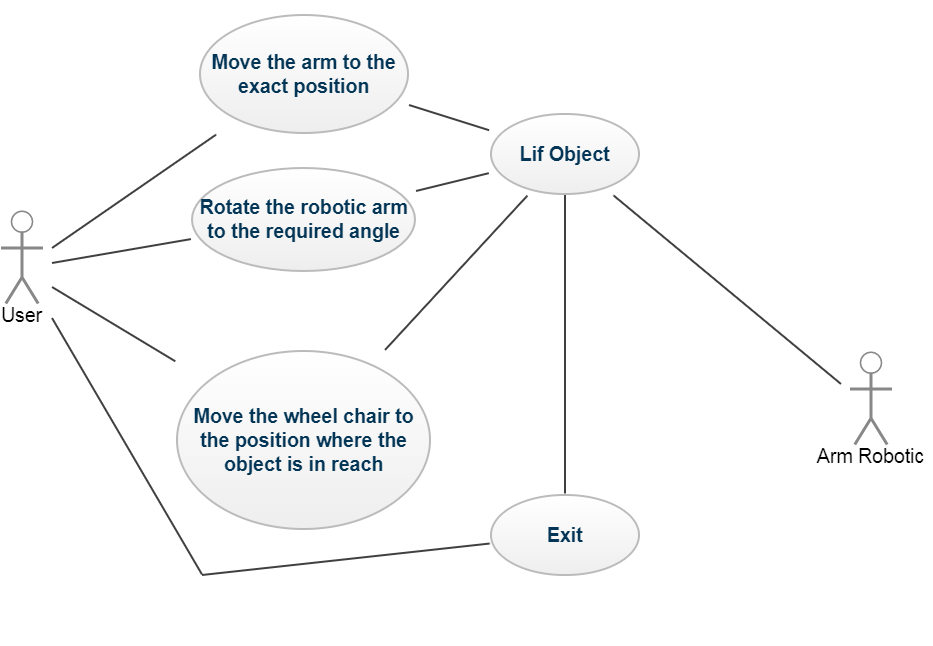
A / D converter. The digital signal is then compared to the reference signal. to produce. The pendulum platform will use a rotary encoder to measure position is fed to the computer quadrature encoder interface. The signal is then the reference signal is compared to generate an error signal to operate the control unit.

## **Sequence Diagram**



**Figure 3. Sequence Diagram**

## **Use case diagram**

****

**Figure 4. Use Case Diagram.**

# **GLOSSARY**

**Terms: Description:**

|  |  |
| --- | --- |
| Bluetooth | Bluetooth is a wireless technology standard for exchanging data over short wavelength |
| Wİ-Fİ | Wireless local area networking |
| Android | Android is a mobile operating system developed by Google, based on the Linux kernel and designed primarily for touchscreen mobile devices such as smartphones and tablets. |
| DC | Direct current motor |
| LED | Light-emitting diode |
| PDA | Digital Assistant or Personal Data assistants for Mobile Electronic Device. |
| PCI | Peripheral Component Interconnect |
| IC | Originally IC Corporation |
| SPP | Serial Port Protocol |
| EDR | Enhanced Data Rate |
| CSR | CSR Blue Core Bluetooth |
| CMOS | Complementary metal–oxide–semiconductor |
| AFH | Adaptive Frequency Hopping Feature |
| DoF | Degree of freedom |
| API | Android platform |
| PWM | Pulse-width modulation |

# **REFERENCES**

[1] Möckel, Rico, et al. "YaMoR and Bluemove—an autonomous modular robot with bluetooth interface for exploring adaptive locomotion." *Climbing and Walking Robots*. Springer, Berlin, Heidelberg, 2006. 685-692.

[2] Wang, N., Zhang, N., & Wang, M. (2006). Wireless sensors in agriculture and food industry—Recent development and future perspective. *Computers and electronics in agriculture*, *50*(1), 1-14.

[3] Moeckel, Rico, et al. "Exploring adaptive locomotion with YaMoR, a novel autonomous modular robot with Bluetooth interface." *Industrial Robot: An International Journal* 33.4 (2006): 285-290.

[4] Neto, Pedro, J. Norberto Pires, and A. Paulo Moreira. "Accelerometer-based control of an industrial robotic arm." In*Robot and Human Interactive Communication, 2009. RO-MAN 2009. The 18th IEEE International Symposium on*, pp. 1192-1197. IEEE, 2009.

[5] Selvam, M. (2014). Smart phone based robotic control for surveillance applications. *International Journal of Research in Engineering and Technology*, *3*(03).

[6] Hamori, A., Lengyel, J., & Resko, B. (2011, June). 3DOF drawing robot using LEGO-NXT. In *Intelligent Engineering Systems (INES), 2011 15th IEEE International Conference on*(pp. 293-295). IEEE.

[7] Badamasi YA. The working principle of an Arduino. InElectronics, Computer and Computation (ICECCO), 2014 11th International Conference on 2014 Sep 29 (pp. 1-4). IE