# Modular Robotics: Molucube

Cole Butler
Ali Helmi
Andrew Hunt
Carlos Torres

# FUNCTIONAL SYSTEM REQUIREMENTS

REVISION – Draft
4 December 2022

# FUNCTIONAL SYSTEM REQUIREMENTS
FOR
# Modular Robotics: Molucube

PREPARED BY:

Author Date

APPROVED BY:

Project Leader Date

John Lusher, P.E. Date

T/A Date

# Change Record

| Rev. | Date | Originator | Approvals | Description |
|---|---|---|---|---|
| **-** | 10/03/2022 | Andrew Hunt | | Draft Release |
| 1 | 12/04/2022 | | | 403 Final Report |

# Table of Contents

# List of Tables

# List of Figures

# 1. Introduction

## 1.1. Purpose and Scope

This project aims to present a modularized robot design that is reconfigurable and capable of performing specific tasks it was built for, while maintaining simplicity in design and cost effectiveness. This robot represents the possibility of future applications where multitasking robot is needed in limited resources situations, such as: rescue missions, space explorations, and warehousing. In order to achieve a robot capable of multitasking and reconfigurability, we are designing job-specific individual modules which are going to be connected and assembled to realize the bigger robot. Each module's purpose and scope are explained briefly in the following sections.

### 1.1.1. Camera Module

The purpose of the camera module is to identify and track relevant objects and navigate the path for the robot. Through a programmed python script developed using OpenCV and YOLO libraries, the module knows what to look for in real-time, as the robot and the program are running the camera module should continue to scan all it sees.

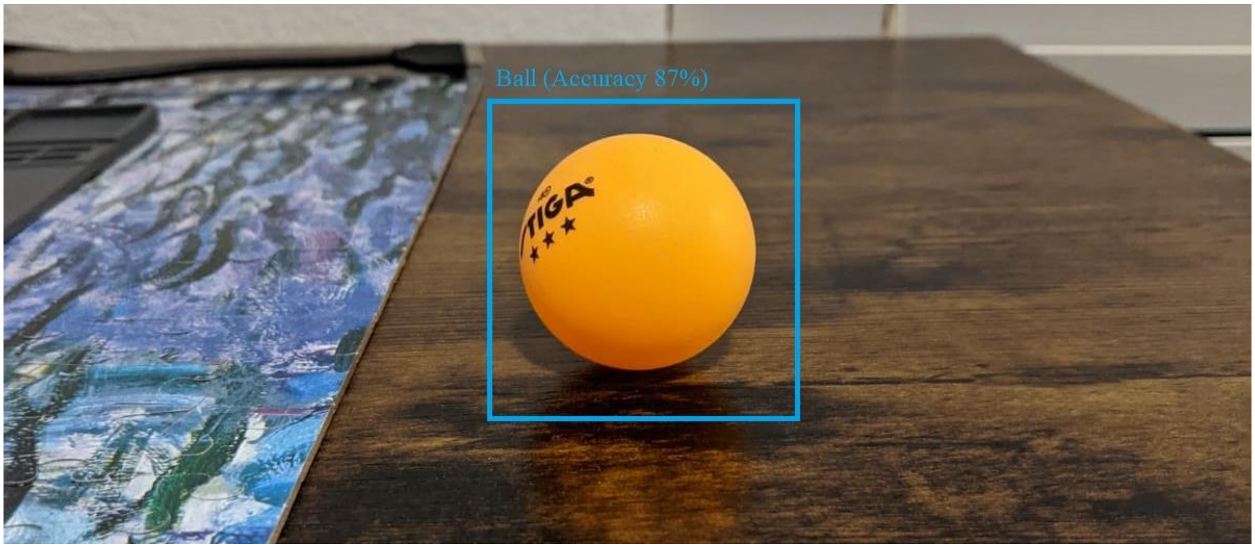


**Figure 1 Example of object identification through the camera module lens.**

### 1.1.2. Gripper Arm Module

The purpose of the gripper arm module is that its arm extends over the robot and reaches for the object identified by the camera and signaled by the controller. The gripper arm bends and reaches over the robot and picks up objects using servo motors mounted on the arm and controlled by the controller module.

#### 1.1.3. Controller Module

The purpose of the controller module is to control the entire robot. Equipped with a Raspberry Pi, it sends commands or receives data with the microcontrollers of each module using serial communication.

#### 1.1.4. Sensor Module

The purpose of the sensor module is to provide the robot with additional information about the environment. It uses a ultrasonic sensor to send distance data to the controller module.

#### 1.1.5. Wheel Module

The purpose of the wheel module is to give the robot locomotion. It does this using a DC motor with a wheel attached. This is action is done by the usage of an H-bridge and a microcontroller to provide the logic needed for the mosfets to allow the motor to go backwards and forwards.

#### 1.1.6. Power Module

The purpose of the power module is to supply each module of a robot with power. It contains LiPo batteries, a battery management system and power converters. This is all so that each module is provided with the correct voltage.

## *1.2. Responsibility and Change Authority*

Team members are responsible for the completion of their respective subsystems. Requirement changes can be made only by the project sponsor, Swarnabha Roy.

**Table 1 Division of Responsibilities**

| Subsystem | Responsibility |
|---|---|
| Power Delivery & Motor Driver | Carlos Torres |
| Raspberry Pi & Microcontrollers | Andrew Hunt |
| Computer Vision & Arm | Ali Helmi |
| Housing & PCBs | Cole Butler |

# 2. Applicable and Reference Documents

## 2.1. Applicable Documents

The following documents, of the exact issue and revision shown, form a part of this specification to the extent specified herein:

| Document Number | Revision/Release Date | Document Title |
|---|---|---|
| 14867857 | 2014 | Multiple object detection using OpenCV on an embedded platform |
| 18357982 | 2018 | Real Time Object Detection and Tracking Using Deep Learning and OpenCV |
| 384 | 2011 | New object detection features in the OpenCV library |

## 2.2. Reference Documents

The following documents are reference documents utilized in the development of this specification. These documents do not form a part of this specification and are not controlled by their reference herein.

| Document Number | Revision/Release Date | Document Title |
|---|---|---|
| Doxygen HTML 4.6.0 | 4.6.0-dev Jun 2022 | OpenCV Documentation |

### *2.3. Order of Precedence*

In the event of a conflict between the text of this specification and an applicable document cited herein, the text of this specification takes precedence without any exceptions.

All specifications, standards, exhibits, drawings or other documents that are invoked as “applicable” in this specification are incorporated as cited. All documents that are referred to within an applicable report are considered to be for guidance and information only, except ICDs that have their relevant documents considered to be incorporated as cited.

# 3. Requirements

## 3.1. System Definition

This robot serves as a more affordable and accessible solution for users requiring a modular robot. It is manually reconfigurable which allows users to change its functionality to suit various applications. It consists of the following interconnectable modules: Controller, Power, Wheel, Camera, Gripper, and Sensor.

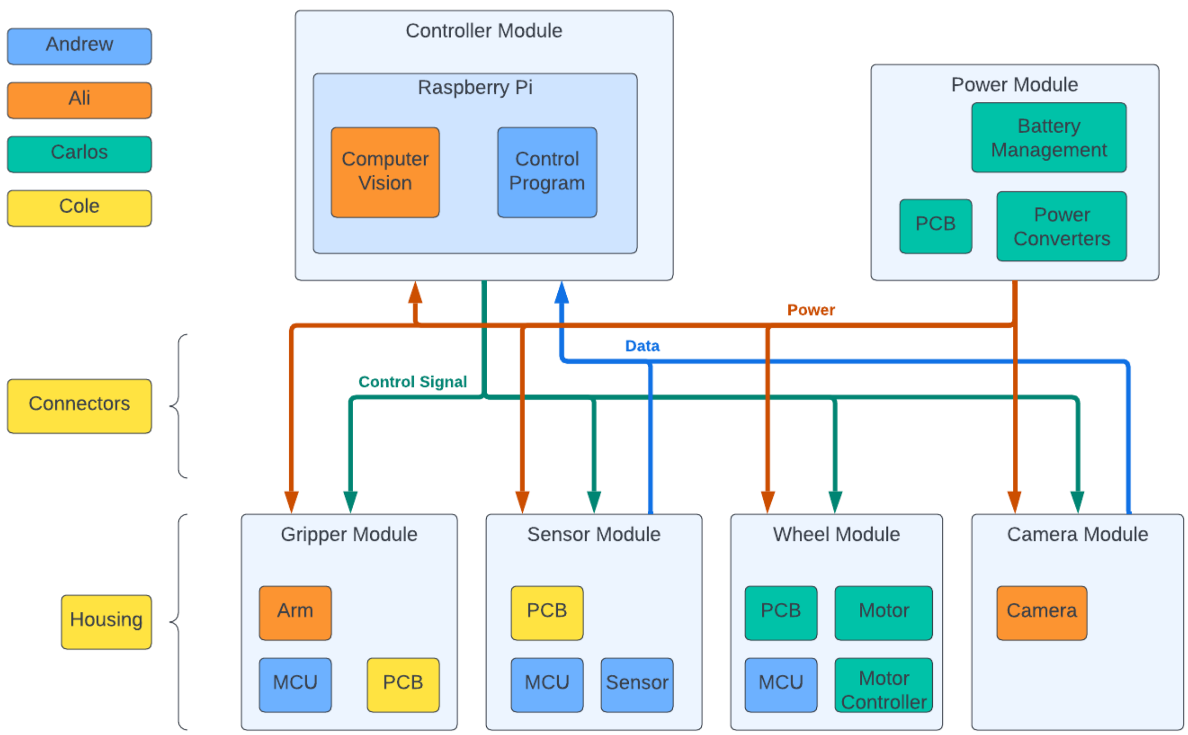


**Figure 2. Block Diagram of System**

The Controller Module contains a raspberry pi and is used to control the robot's functions including navigation, motion, and environmental detection. It uses serial communication to send commands to the microcontrollers in other modules to control them or receive data. It also uses USB to interact with the camera module.

The Wheel Module consists of a microcontroller, a motor controller, a motor, and a wheel. The microcontroller receives commands from the Controller Module to determine the direction and speed that the motor should run to move the robot where it needs to be to complete its task.

The Sensor Module contains a microcontroller as well as an ultrasonic sensor. The microcontroller collects distance data from the ultrasonic sensor and relays that information back to the Controller Module when requested.

The Gripper Module contains a microcontroller and a gripper arm. The Microcontroller receives commands from the Controller Module to move the gripper arm. The Gripper arm contains an actuator to raise and lower the arm, and a servo motor to move the gripper itself. The gripper arm is used to pick up objects.

The Camera module contains a camera that connects to the controller module via USB. The camera data is used by the controller module to locate and track objects; as well as gather information to aid in navigation.

The Power Module contains a battery management system as well as power converters to create the voltages needed to power the Raspberry Pi, Microcontrollers, and motors used in the robot.

## *3.2. Characteristics*

### 3.2.1. Functional / Performance Requirements

#### 3.2.1.1. Run Time

The robot needs to have decent run time. Something where it lasts a good amount of time before batteries need to be charged. So, 2 x 22.2V LiPo batteries will be placed in parallel.

*Rationale: Placing batteries in parallel would increase the duration of the batteries and it keeps us from having to purchase one big battery and have to fit that. As opposed to 2 smaller sized batteries.*

#### 3.2.1.2. Locomotion

The robot shall have the ability to move itself around its environment.

*Rationale: The robot needs to be able to move itself around the environment to complete the tasks required by the user*

#### 3.2.1.3. Computer Vision and Sensor Based Navigation

The robot shall use a camera and computer vision as well as ultrasonic sensors to determine what is in its environment to successfully navigate.

*Rationale: The robot needs to be autonomous, which will require a camera and sensor to navigate and find its tasks.*

#### 3.2.1.4. Customizable

The robot shall be capable of varied module combinations to produce different robots.

*Rationale: Dividing functions into separate modules allows for different combinations of functions. This increases the versatility, and potential use cases of a modular robot.*

### 3.2.2. Physical Characteristics

#### 3.2.2.1. Mass

The mass of the robot is TBD.

*Rationale: We do not have a requirement for the overall robot to be a certain mass.*

#### 3.2.2.2. Volume Envelope

The volume of each module can vary depending on its intended purpose, but the starting dimensions will be around 4”x4”x3” (LxWxH).

*Rationale: As this is a modular robot, a module could be nearly any size to fit its purpose, but controller, power, and blank modules are intended to be the above-mentioned size. This size was chosen as it fits around the Raspberry Pi and leaves room for wires to go to the side connectors*

#### 3.2.2.3. Module Connections

Modules shall be able to connect to each other and stay together during use. They must be able to come apart at the discretion of the user.

*Rationale: If modules were to come apart during use, then a robot in use could lose parts and functionality. If the user cannot easily disconnect a module, then maintenance and modularity of a robot would be difficult.*

### 3.2.3. Electrical Characteristics

#### 3.2.3.1. Inputs

a. Input levels for the robot would be 22.2v provided by 2 x 22.2v lipo batteries placed in parallel.

*Rationale: By design, should improve battery run time and series would only provide more voltage which we don't need.*

##### 3.2.3.1.1 Power Consumption

a. The highest power for the entire system shouldn't go over 200 Watts.

*Rationale: The heaviest load on this power delivery module are the motors and those need a max current (stall) of 9A so there should be any more power needed than this*

##### 3.2.3.1.2 Input Noise and Ripple

The input noise for the robot produced by the buck converters would be an issue to data collection relayed to the Raspberry Pi but this would be hammered out in the PCB design.

*Rationale: This would need to be situated in the PCB design by placing the converters and data traces far from each other to minimize noise*

##### 3.2.3.1.3 External Commands

The robot may receive external commands and/or configurations from the user via wireless communication.

*Rationale: Wireless configuration through the Raspberry Pi will make it easier to re-program or manually control the robot, rather than having to disassemble it to access the SD card.*

#### 3.2.3.2. Outputs

##### 3.2.3.2.1 Diagnostic Output

The robot may have a log file stored on the raspberry pi in order for the user to diagnose configuration and programming issues.

*Rationale: The robot is configured and programmed by the end user, so a file logging what the robot is doing will be helpful for troubleshooting issues*

##### 3.2.3.2.2 Raw Video Output

The real-time video footage captured by the camera module will be streamed to the controller module running the object identification program.

*Rationale: The program should be able to identify objects of interests and signal the robot to take actions.*

#### 3.2.3.3. Voltage Outputs

The Power Delivery module will be utilizing the buck converters to drop the 22.2v input voltage to the desired following output voltages: Raspberry Pi = 5v, Wheel Motors = 12v, Gripper arm linear actuator = 12v and Gripper servo = 5v.

*Rationale: These are the operating voltage requirements provided on the data sheets.*

#### 3.2.4. Environmental Requirements

The path we want the robot to travel through should be marked by tape or any kind of distinctive markers that can be recognized by the camera.

*Rationale: The robot is going to be able to avoid obstacles, but in order for it to not wonder aimlessly we need to guide its path and see how it navigates through.*

#### 3.2.5. Failure Propagation

The modular robot shall not allow propagation of errors outside the scope of the presented system interface.

##### 3.2.5.1. Failure Detection, Isolation, and Recovery (FDIR)

##### 3.2.5.1.1 Camera Failure Detection

The written program is made to throw errors in case of any failure occurrence, such as failing to connect with the webcam.

##### 3.2.5.1.2 Built In Test (BIT)

The controller module shall have scripts programmed on it which include failsafe programs that ensures the correctness of the robot’s performance.

##### 3.2.5.1.3 Communication error detection

The controller module and microcontrollers will have code to detect when a communication error may have occurred and will request confirmation of the last data frame.

# 4. Support Requirements

To program the robot, the user will need to use a computer with a terminal that can access the Raspberry Pi through either Bluetooth or Wifi. The user may also remove the SD card from the Raspberry Pi to add programs directly. All robot designs will require at least one controller module and one power module.

# Appendix A: Acronyms and Abbreviations

| | |
|---|---|
| BIT | Built in Test |
| LiPo | Lithium Polymer |
| PCB | Printed Circuit Board |
| TBD | To Be Determined |
| USB | Universal Serial Bus |

# Appendix B: Definition of Terms

Computer Vision Computer’s ability to understand and interpret reality through image or videos.