E1.06 – Robot Fetch

Product Specification

Andrew Casper, Project Manager

Kyerstynn Gonzalez

Britney Salinas

Dennise Trevino

Sponsored by:

Texas State University

601 University Dr.

San Marcos, TX 78666

Date: April 1st, 2020

TEXAS STATE

INGRAM SCHOOL OF ENGINEERING

Revision History			
Version	**Date**	**Description**	**Author**
0.1	04/01/2020	Most sections identified: authors named	Andrew Casper
0.5	04/06/2020	Majority of sections filled	Andrew Casper
1.0	04/08/2020	Product Spec Version 1.0 complete	Andrew Casper

Table of Contents

1 Introduction

The Introduction section was written by Britney Salinas.

Our product is an autonomous robot named ‘Spark’ that will retrieve a ball thrown into a hallway. The robot will identify the ball by color and bring it back to its starting point in a timely manner. On Senior Design Day, sponsors and employers are expected to attend. Spark’s intent is to demonstrate capabilities to sponsors and employers as well as give the team members experience in robotics and automation. To build the robot and aid in its function, parts will be repurposed from a hoverboard. Additions to the robot include, but are not limited to, an object collection arm and tail to retrieve the ball and help with stability. Figure 1 shows an overview of the systems with a block diagram.

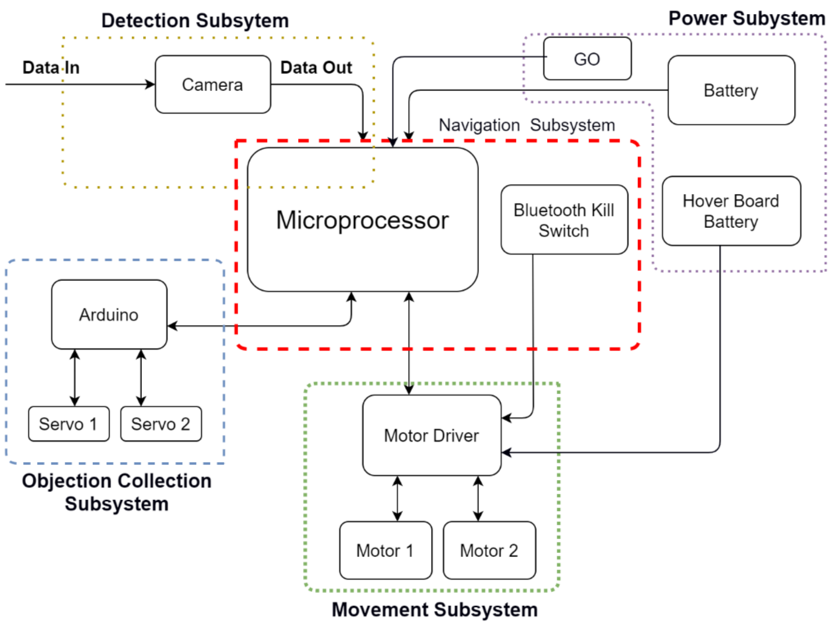

Figure 1: High-Level Block Diagram

1.1 Sponsor Requirements

The Sponsor Requirements was written by Britney Salinas.

The robot is required to have two wheels as well as a 3D printed tail dragger. The robot will retrieve a predetermined color ball after multiple balls are thrown and bring it back to its starting point. The autonomous robot will be expected to identify three different color low bounce balls. The ball must be thrown at least five meters away from the starting point and cannot land within two inches of any wall or object; if this happens, a rethrow is required. The robot's field will be approximately 12.19 X 3.05 meters down a hallway in Ingram Hall and will need to avoid damage to the walls. The time constraint on the robot for task completion will be five minutes.

1.2 Existing System

The Existing System was written by Britney Salinas.

This autonomous robot is the first time the Ingram School of Engineering has offered this senior design project. Therefore, there is not an existing system.

1.3 Terminology

The Terminology was written by all team members.

Term	*Description*
CAN	*Controller Area Network*
CPU	*Central Processing Unit*
IDE	*Integrated Development Environment*
GSB	*Gator Skin Ball*
MCU	*Microcontroller Unit*
PCB	*Protection Circuit Board*
WIMU	*Wireless Inertial Measurement Unit*

2 Functional Description

2.1 User Attributes and Use Cases

The User Attributes and Use Cases section was written by Andrew Casper.

This product will be designed to be used by the 1.06 senior design team only, with the exception of senior design attendees who will be able to throw the ball and use the voice input of the color of ball to fetch, with the assistance of the 1.06 senior design team. No special skills are needed since the robot will be autonomous.

There are three separate tests to be performed: 1)The robot fetches a stationary ball and returns to its starting position, 2) A user throws a ball and once it is stationary, the robot fetches the ball and returns to its starting position, 3) The user throws 3 balls of varying colors and tells the robot which to fetch, and the robot returns it to the starting position.

The product, once set in its designated starting position, would need to be powered on and then go into a waiting state where it would wait for a ball to be thrown and the color of the ball given as input. After the robot collects the ball and returns to the starting position, it will rotate 90 degrees and go back to waiting for a fetch command.

The user can power on the product with a power button. Once the product is in its wait state, the user can press the “go” button or use the voice input feature (stretch goal).

2.2 Administration Functions

The Administration Functions section was written by Dennise Trevino.

The product will not have administrative functions; due to the robot not having any sensitive information. Therefore, if a user is interested in modifying the code, they will have access as long as the robot is completely powered off. In order to accommodate with software update demands, a “Synaptic Package Manager” application will be installed onto Raspberry Pi. For the following semester, we plan to incorporate a voice command integration, which all users will have access to power on/off the robot.

2.3 Error Handling

The Error Handling portion was written by Dennise Trevino.

The autonomous robot will be able to acknowledge an error that might occur during performance. In order to aware the user of error, students will implement a LED Color Coded (Table 1) to help distinguish which function does the error belong to.

Error	LED Color
Low Battery	yellow
Motors Not Working	blue
Sensors Not Working	orange

Table 1: LED Color Coded

Low Battery

To signifying battery percentage is low, a yellow LED light will turn on. Moreover, to avoid having to constantly recharge the robot, a detachable battery will be available for swapping out.

Motor Not Functioning Properly

A Wireless Inertial Measurement Unit (WIMU) will be incorporated into the movement subsystem. For instance, the robot is commanded to redirect itself to a X distance at Y degrees of an angle, the WIMU will detect whether the robot has moved as stated command or not. If the robot didn’t move according to a given command, then a blue LED light will turn on, indicating there is an error with movement subsystem. Moreover, in the case that during performance the WIMU acknowledges the motors are not function properly, the robot will turn off the Raspberry Pi in order to save battery usage.

Sensors Not Functioning Properly

There are a few sensors integrated into the robot such as a camera and WIMU sensors. To help address the issue before the product performs any tasks, a self-test will be performed upon automatic start-up. The self-test will verify if the camera and WIMU sensors are working properly, as presented in Figure 2. In the instance of a malfunction, an orange LED light will turn on.

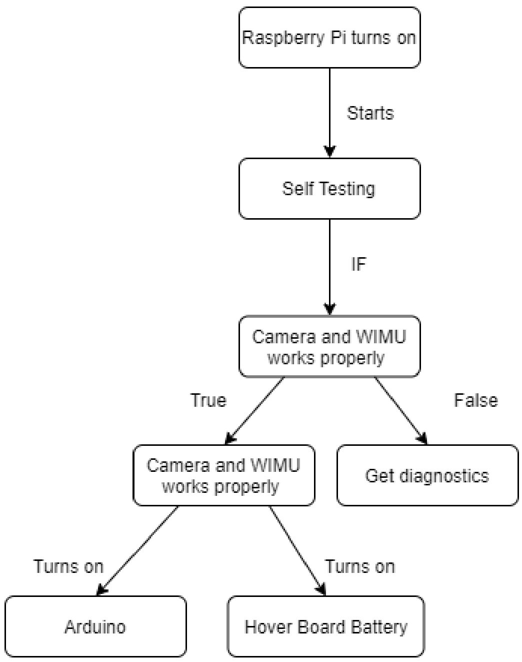

Figure 2: Power-On Cycle

2.4 Safety and Security

The Safety and Security was written by Kyerstynn Gonzalez

Speed and collisions will be a concern for Spark since a hoverboard is being used. Therefore, the top speed will be investigated and limited. The autonomous robot will have a kill switch or power button to shut down all power in case the autonomous system malfunctions.

Due to the layout of the field, the damaging of walls and other obstacles is a major concern. Therefore, the product may have a sensor attached in order to notify if any obstacles/objects are in close range (which isn't the predetermined ball). When the robot approaches an object, it will slow down and try to maneuverer from it, with intentions of avoiding collision.

Commented [GA1]: need to specify sensor??

Commented [CA2R1]: just say the product may have* if it is determined to be needed

The lithium ion battery will include a battery management system. The PCB management system, which is the chip located inside each pack of batteries, will ensure that the batteries are not discharging.

The robot does not have any significant security concerns. Since the camera will not be recording or storing any valuable data into the microprocessor, it will not need any protection to access interfaces.

Commented [GA3]: @Casper, Andrew do we want to record video from robot while its retreiving ball?

Commented [CA4R3]: @Gonzalez, Kyerstynn A good question, I dont think so but im not 100% sure. i think what you wrote is good but can you verify with stevens report?

2.5 Help and User Documentation

The Help and User Documentation was written by Dennise Trevino.

The product is an autonomous robot and is required to compete against another team during Senior Design Day; thus, there will not be any "Help & User" documentation. On Senior Design Day, the teams official rule document will be presented along with a one pager of all operations dealing with the robot; for instance, explanations of all LED colors. The idea is to hand out copies of the pager as a friendly guideline handout. The robot's software, we will be posted onto GitHub after the Senior Design Presentation in hopes for being used as guidance for future teams. This will allow them to download and modify as deemed fit.

2.6 Interfaces

2.6.1 User

The User Interface section was written by Andrew Casper.

The product's user interface is limited by it being autonomous, however there will be a "go button" or voice input (stretch goal) for the robot to know when to start and what color of ball to find and a power on/off button for the robots main functions.

The power button is a multi-level button to tell the user when it is on, all the way down is "on" and all the way up is "off". Since the button is repurposed from another device it is still TBD if a second button will be needed for the microcontroller.

There will be a LED used to signal the user when the battery is low.

The user will be able to throw three GSB's of different colors and once the ball has come to a stop, use a voice command to tell the robot which color they would like to be fetched.

2.6.2 Software

The Software section was written by Dennise Trevino

The autonomous robot will be using Raspberry Pi 4B which consist of

- Microprocessor: ARM Cortex-A53
- OS: Raspbian Buster and Linux
- Libraries: Python 3.x and OpenCV 4.1.2
- Applications: Arduino IDE (to compute code onto Arduino via Raspberry Pi) and Synaptic Package Manager (allows to manage software updates easier)
- Compiler: AVR 4.8.1

2.6.3 Hardware

The hardware section was written by Kyerstynn Gonzalez

- The hardware of the hoverboard purchased will be dismantled and the platform motors and motor drivers will be repurposed for Spark. The motors used for Spark will be two brushless motors.
- There will be an additional lithium ion battery pack connected to the Arduino and Raspberry Pi. For the vision system a Raspberry Pi will be used and connected to a Pi cam.
- The ball collection system will be the Arduino servo motor or motors and a 3D printed gears and claw mechanism.

2.6.4 Mechanical

The Mechanical section was written by Britney Salinas.

A hoverboard is being used to limit the mechanical aspect of the project so that it is not the primary focus.

- The chassis and wheels of the robot are repurposed from the hoverboard, so the chassis and the wheels will already be put together.
- The chassis will be modified to mount the camera in a position that will allow it to accurately scope out the field.
- The grabber, which will be 3D printed, is placed at the front of the chassis, allowing it to properly open and close in order to enclose the ball.
- For the grabber to function, it uses a servo motor, allowing it to open and close at the desired diameter.
- Lastly, the tail will be 3D printed and mounted on the opposite side of the camera and grabber for stability.

2.7 Boundary Conditions and Constraints

The Boundary Conditions and Constraints was written by Team Members.

Component	Constraint	Bound
Cost	Budget	Under $500

Size	Design	>20 cm. multisegmented 3D printed tail
Chassis	Design	Two wheel “tail dragger” chassis
Field Size	Ingram Hallway	12.19 x 3.05 meters
Run time	Competition	Complete tasks within 5 minutes
Completion	Working robot	D2 senior design day
Battery	12.6V Supply 12 A current draw	12.6V Supply 12 A constant current draw
Step down regulators	38V input to 5v output 5A output	5V 5A
Motor	Voltage and current	36 V maximum 4 A maximum
Motor Drivers	Drive two motors	36 V 4 A
Pi Cam V2	480p-1080p resolution. 30-60 fps. 64 MB of ram from Raspberry Pi 4B.	Resolution and FPS will be determined once testing requirements are met
Raspberry Pi 4B	Temperature, SD/SDIO Interface, Power.	0 to 50 degrees Celsius. 1.8V, DDR50 mode (at a peak bandwidth of 50 Megabytes / sec). 5V at 3A.

Table 2: Boundary Conditions and Constraints

2.8 Performance

The Performance section was written by Team Members.

Hardware Performance Parameters					
Parameter	***Test Conditions***	***Min***	***Max***	***Units***	***How Tested***
Battery Life	*50% Usage*	*6*	*NA*	*Hours*	*Actual performance retrieving ball*
Linearity	*Vin=12.6V*	*0*	*5*	*A*	*Measure input versus output voltage*
Ball collection	*Diameter of ball*	*0.5*	*2*	*feet*	*Check object collection can open to at least 0.5 foot in diameter*
Ball collection speed	*TBD*	*0*	*30*	*seconds*	*TBD*
Robot Speed	*Flat smooth floor*	*0*	*TBD*	*mph*	*A stopwatch, *determined after 5 min req. and safety concern*

Table 3: Hardware Performance

Commented [CA5]: @Gonzalez, Kyerstynn A this is pretty wide, might be overkill. These test are going to be looked at by the professors and they will say did you pass this test.

Commented [CA6R5]:

Software Performance Parameters		
Function	***Description***	***How Tested***
Object Detection	*Detects object in a timely manner and relays the object position to the navigation subsystem. The max distance the camera needs to detect is 15 feet and must be able to detect the ball within 30 seconds.*	*Test various colored GSB's at various distances and field of view angles.*
Navigation	*Robot will compute algorithms in order to calculate the shortest route in order to get to the location of predetermined ball.*	Testing code via Gazebo robot simulator, RViz, and sobot-rimulator.

Table 4: Software Performance

2.9 Software Platforms

The Software Platforms section was written by Andrew Casper.

The operating system for the Raspberry Pi 4B will be Raspbian Buster. The specific version is still TBD depending on the best version available to download with the NOOBS downloader built into the microprocessor's memory card.

The Pi Cam 2 will be used for processing visual data in conjunction with the Raspberry Pi 4B.

Python 3 with be the Raspberry Pi's primary coding language as well as the OpenCV version 4.1.2 library for the object detection subsystem.

An Arduino will be used for the ball collection subsystem to be implemented, it will run Arduino Mega and use the Arduino IDE.

2.10 Service, Support, & Maintenance

The Service, Support, & Maintenance was written by Kyerstynn Gonzalez.

- The on-board battery will be detachable to be easily removed and replaceable.
- The Microcontroller will not be removable and will be mounted on the robot. Easily accessible by USB for monitoring.
- The Pi cam and sensors will be mounted onto the robot. The camera will rotate on an axis in order to check for objects and obstacles around the robot.
- Operation and software maintenance will be done by team members. Once the final product is complete, the hardware and software will be made accessible to websites such as GitHub.

2.11 Expandability or Customization

The Expandability or Customization was written by Kyerstynn Gonzalez

The platform, object collection and object detection can all be made customizable and expanded on after the project is done. The platform will be made by dismantling the hoverboard and designing a customizable chassis. The object collection system will be used by connecting a servo motor or motors to an Arduino with a 3D printed gear and claw mechanism. When the ball is within range, the claw will close and roll the ball back to the desired start position. The object detection system on our robot will be composed of a Pi cam and ultra-sonic sensors. This could also be done by using Lidar sensors or infrared sensors. For each of these subsystems, standard components are being used for this robot and all design files will be made available to be modified.

3 Project Alignment Matrix

The Project Alignment Matrix section was written by Andrew Casper.

Course No.	**Core knowledge**	**Specific knowledge incorporated by team**
EE 3350 (Electronics I)	Design and analysis of active devices and equivalent circuits	Understanding the electronic pre-built systems in the robot hardware will require skills from this class
EE 3370 (Signals and Systems)	Frequency domain representation of signals and frequency response, transfer functions	The robot's motors will need to be PWM to ensure stability
EE 3420 (Microprocessors)	Principles of operation and applications of microprocessors	The skills from this class will be highly affiliated with this project. From understanding how to program a microcontroller and its peripherals to building a micro controlled robot.
EE 4352 (Introduction to VLSI Design)	Analysis and design of CMOS integrated circuits	
EE 4370 (Communications Systems)	Transmission of signals through linear systems, analog, and digital modulation, and noise	The voice input will need to pass a verbal signal to the robot through interference.

Table 5: Knowledge Alignment Matrix

ABET Criterion 3 (c): "an ability to design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability."

Constraint Type	**Specific Project Constraint**
Economic	Given a spending budget of $500.
Environmental	To make sure the batteries of the power system are safe for the environment.
Health and safety	To give the robot a kill switch and to protect Ingram halls from damage.
Social/Ethical	Need to be able to express the skills and ethics of the Engineering students of Texas State University
Applicable Standards	Needs to be able to do its job in under 5 minutes. Batteries must be cared for and handled safely.

Table 6: Constraint Alignment Matrix (and applicable standards)

4 References

PiCamera V2 data sheet
https://www.raspberrypi.org/documentation/hardware/camera/

Raspberry Pi 4B data sheet

https://www.raspberrypi.org/documentation/hardware/raspberrypi/bcm2711/rpi_DATA_2711_1p0_preliminary.pdf

5 Approvals

The signatures of the people below indicate an understanding in the purpose and content of this document by those signing it. By signing this document, you indicate that you approve of the proposed project outlined in this Functional Specification and that the next steps may be taken to proceed with the project.

Approver Name	Title	Signature	Date
Andrew Caspar	Project Manager		
Claude Garrett	D2 Project Manager		
Lee B. Hinkle	Faculty Sponsor		
Lee B. Hinkle	Sponsor		
Mark Welker	Instructor		

Section	Author	Word Count
1. Introduction	Britney Salinas	122
1.1 Sponsor Requirement	Britney Salinas	130
1.2 Existing Systems	Britney Salinas	25
1.3 Terminology	Team Members	24
2.1 User Attributes and Use Cases	Andrew Casper	233
2.2 Administration Functions	Dennise Trevino	80
2.3 Error Handling	Dennise Trevino	274
2.4 Safety and Security	Kyerstynn Gonzalez	176
2.5 Help and User Documentation	Dennise Trevino	110
2.6.1 User Interfaces	Andrew Casper	149
2.6.2 Software	Dennise Trevino	50
2.6.3 Hardware	Kyerstynn Gonzalez	83
2.6.4 Mechanical	Britney Salinas	134
2.7 Boundary Conditions and Constraints	Team Members	140
2.8 Performance	Team Members	185
1.9 Software Platforms	Andrew Casper	100
1.10 Service, Support, & Maintenance	Kyerstynn Gonzalez	104
1.11 Expandability or Customization	Kyerstynn Gonzalez	132

3. Project Alignment Matrix	Andrew Casper	208
4. References	Team Members	5