Thumper the Bumper

Conceptual Design Review (5/23/2023)

Sydney Fernandez [Hopper Mechanism Engineer]
Alyssa Maldonado [Block Code Engineer]
Carrot Engineer
Adell Somuerfan [Project Manager]

What is Thumper the Bumper?

Thumper the Bumper is a physical reproduction of the Google Doodle game celebrating 50 years of Kids Coding!. For our replication, we will be incorporating MIT Inventor to create the app that lets the team to move the robot. The robot's goal is to collect carrots along the path. Inspired by the Google Doodle game, the robot doesn't have to worry about falling off the mesa, so another goal of our replication is focused on the collect fall of the path. Since we are running the 3D robot on top of a 2D path. For the robot, we are using the Pololu Turnaround for Robots 30x7 Paperclip Chassis Kit and for the levels, we are using dimensions from a maze path given to us by Arduino.

- Features:
  - 3DoT robot controlled via MIT Inventor app
  - Robot is controllable from an Android phone
  - Carrot color cards
  - Levels inspired by Google Doodle game and maze by Arduino

Start over

Job Responsibilities

The Hopper Mechanism Engineer (Sydney Fernandez) is in charge of designing the 3D model of the body and the modeling for the moving parts of the robot and its motors. They are responsible for the mechanical aspect of the rabbit hopping and moving.

The Block Code Engineer (Alyssa Maldonado) is responsible for developing an app capable of relaying directions to Thumper the Bumper. The goal is to control Thumper the Bumper utilizing block coding commands chosen by the Google Doodle game's style.

The Carrot Engineer (Jamie Epstein) is in charge of designing and coding the hardware and software needed to pick up carrots and detect how far they are. This includes the brush motor and servo motors for the respective mechanical specifications. Once the Hopper Mechanism Engineer and Hopper Software Engineer complete all the hardware and software, the Carrot Engineer will keep on working to combine the two.

The Project Manager (Adell Somuerfan) is responsible for keeping a track of different employees to duties, deadlines, and meeting requirements to ensure that it is a success. To do so, the Engineer will keep an updated the schedule during the project.

Hopper Mechanism Engineer

Requirements:

L2.1 To hop forward, the robot will use a servo motor to propel itself onto the next block or on the same block.

L2.2 Servo motors attached to a spring shall create a force that will propel Thumper upwards as derived from the L2.1 requirement.

L2.3 Using a brush motor, the robot will turn right, left, and turn around

L2.4 The robot shall have plastic feet with a foam cushion underneath for stability and shock-absorption.

Conceptual Design Solution

- Servo Motor: Continuous Rotation Servo - FeeTech FS5103R
  - Have one on both sides of the robot
  - Servo will wind up a string that is attached the the top of the spring mechanism, pulling it down to stretch the spring
  - Once stall torque speed is met, the string is let go, launching the robot
- To turn, the robot will use the wheels from the 3DoT Paperclip kit
- Have large rabbit feet with foam tape at the bottom of robot for shock absorption

Hopper Mechanism Engineer

Requirements:

L2.1 To hop forward, the robot will use a servo motor to propel itself onto the next block or on the same block.

L2.2 Servo motors attached to a spring shall create a force that will propel Thumper upwards as derived from the L2.1 requirement.

Continuous Rotation Micro Servo - FS90R -> Continuous Rotation Servo - LD-Hope 3 gears can be controlled with an Arduino. Both uses any servo code, library, or hardware to control it.

Requirements	FS90R	FS5103R	Requirements Met?
Speed (sec/60°)	Not given - only given no load speed: 100RPM (4.8V)	Average speed - 0.18sec/60°	FS90R - no FS5103R - yes
Stall Torque	18.08 oz-in (1.3kg)	41.74 oz-in (3.0kg)	FS90R - no FS5103R - yes
Voltage	4.8V - 9V	4.8V - 6V	FS90R - yes FS5103R - yes
Dimensions (must be less than 2.5 x 2.5 x 1in)	1.9" x 1.2" x 0.9"	1.5" x 2.1" x 0.8"	FS90R - yes FS5103R - yes

Hopper Mechanism Engineer

Requirements:

L2.4 The robot shall have plastic feet with a foam cushion underneath for stability and shock-absorption.

L2.5 The robot shall have a spring mechanism that helps the servo motors launch the robot as derived from the L2.3 requirement.

L2.6 Springs shall be extension springs that are less than 2" stretched and about 1½" - ¾" compressed. This requirement is based on L2.5.

- Springs: steel Extension Springs with machine or side hooks (¾" - 2")
  - Hooke's Law: k = F/x where k is how strong the spring is
  - Needs to satisfy the servo motor stall torque requirement
  - We do not have hard calculations yet, but we will
- Spring mechanism: 3D-Printed
- Feet: made of sturdy plastic

Block Code Engineer

Objective:

To develop an app capable of relaying directions to Thumper the Bumper. The goal is to control Thumper the Bumper using block coding commands chosen by the user, similar to the Google Doodle game.

Block Code Engineer

- Block coding based
- Available resources
- Frequently used in Arduino projects
- Android emulator

Block Code Engineer

L1.6 App will communicate with robot through bluetooth.

Block Code Engineer

Carrot Engineer

Main Objective:

- Design and code the sensors that indicate when a level starts, ends, and keep track of the carrots (points) that Thumper picks up
- Using a sensor sensor and an RFID sensor to "pick up" carrots and detect how Thumper is on the right path

Requirements:

- L1.1) Thumper will use an RFID sensor to detect the carrots to symbolize level completion and progression.
- L2.1) Utilize passive RFID tags to simulate the carrots (...)
- L1.2) Thumper shall use a color sensor to remain on the path in order to execute or stop moves.
- L2.2) A color sensor will be used for Thumper to remain on the correct path and not off course; the path will be made from colored RFID tags (...)

Carrot Engineer

- The RFID reader will be used to keep track of the level progression and completion based off of the number of carrots per level.
- The two compared sensors will be the MFRC522 Standard RFID Module and the PN532 NFC & RFID Module

MFRC522 Standard RFID Module

PN532 NFC & RFID Module

Carrot Engineer

Requirement	MFRC522 Standard RFID Module	PN532 NFC & RFID Module
Built-in Antenna	Yes	Yes
Open Source Documentation	Yes (Eagle layout from pasenda.com; datasheet from NXP)	Yes (Eagle layout from adafruit; datasheet from NXP)
I2C or SPI Compatible	I2C Compatible (address 0x3C); SPI Compatible	I2C Compatible (address 0x48); SPI Compatible
Supports 13.56MHz ISO14443A MIFARE cards	Yes	Yes
Range	50 mm depending on antenna size, tuning, and power supply	100 mm depending on antenna size, tuning, and power supply
Programmable Timers	Yes, accessible through the MFRC522's internal state machine	Yes, accessible through 80C51 Microcontroller and the CIU (Contactless Interface Unit)

Cross-talk

Carrot Engineer

- The Color Sensor will be used to ensure that Thumper is still on the correct path while executing the movement code. If the Green path is not found, then Thumper will interrupt the code and not move.
- The three compared sensors will be the TCS34725, AS7341, and ISL29125

TCS34725

AS7341

ISL29125

Carrot Engineer

The Color Sensor will be used to ensure that Thumper is still on the correct path while executing the movement code.

Requirement	TCS34725	AS7341	ISL29125
Open Source Documentation	Yes (from adafruit)	Yes (from adafruit)	Yes (from sparkfun)
I2C or SPI Compatible	Allows for I2C (Addressed at 0x29)	Allows for I2C (Addressed at 0x39)	Allows for I2C (Specific address couldn't be found on datasheet)
Suitable for underneath Thumper (might be dark)	Has an RGBIC engine with a gain	Has independently configurable timing and gain/automatic gain adjust	Designed to reject IR in light sources (Active IR Compensation)

Sensor shield

Carrot Engineer

Conceptual Design Solutions

- RF Sensor: MFRC522 Standard RFID
  - The MFRC522 allows for lower level instructions to control the interrupt flag registers and timeout registers to be faster.
  - We do not require NFC or the larger distance of the other module currently.
- Color Sensor: TCS3472
  - The TCS3472 allows for I2C connection which is necessary for interfacing with multiple sensors.
  - The ISL29125 was considered, but too advanced for our needs.

Thanks, I love carrots!

Project Manager

- Objective:
  - Keep the project running in a timely manner
  - Update with any necessary changes on Click Up
  - Keep meeting minutes
- Research
  - Assist team members
  - Keep track of expenses

Project Manager

Project Manager

Project Manager

Project Manager

Project Manager

Concluding Thought

- Missing Level 1 Requirements
  - We shall use a new color sensor for the robot to perform commands
  - We will use the wheels from the 3DoT Paperlet kit to turn
  - The app shall have a remote/blue button to remove unwanted commands
  - The app shall have a play button to execute commands
- New Level 2 Requirements
  - We shall use the color sensor to determine whether or not Thumper will start executing the commands
  - Using the 3DoT Paperlet Chassis wheels, we will have them turn 90 degrees if needed
  - Use the RFID tags to keep track of carrots consumption, level completion, and power on or off states.

Run-Time Program Flow

App

Arduino