REMOTE CONTROLLED SMART ICE CREAM TRUCK (WITH MUSIC PLAYER):  
PROJECT 4 PROPOSAL

ARTHUR MAKUMBI

CS342 FALL 2018

horizontal line

# Pitch

## Abstract

My goal this project is to try and pull everything together. I want experiment and get comfortable with DC motor control, analog to digital conversions, pulse width modulation, and Infrared Remote control, while also configuring, triggering, and servicing interrupts. In this project, I will program my board to interact with digital and analog inputs (buttons, IR receiver, and IR distance sensor) and outputs (motors, piezo element,, and LEDs) to create a remote controlled smart ice cream truck. The resulting product will be remote controlled and able to move in any direction while also being able to come to a halt as it approaches an obstacle. It will also play some lovely jolly ice cream tunes as it travels. One will set the smart ice cream truck in motion by pressing a button either on the remote or on the board.

I have designed this project because I have always loved ice cream trucks, especially the music they play, since my childhood. Additionally I am excited about automation, especially the kind aided by data from sensors.

## 

## Rough sketch

## 

## Design Criteria

I will wire two DC motors (outputs) to my board by the help of the L293D H-bridge as shown above. I will also add Pulse Width Modulation (PWM) to both motors whereby Counter 0 drives the left motor and Counter 2 drives the right motor.

I will add and configure an IR distance sensor (input) to my board which will collect my data for my Analog to Digital Conversions (ADC). I will also add two LEDs (outputs) to my circuit to detect when the motors are working correctly. I will wire each of them in series with a 2.2Ω resistor to limit overall power consumption, and avoid short circuits.

I will also add an infrared sensor to enable my ice cream truck to be remote controlled. It will get its data from PD[2]

I will also wire a push button (input) to PD[3] by writing to the GPIO registers, and connect it in series with a 10 kΩ resistor, a “pull down resistor”. This is to ensure that when no high voltage is arriving on the input side of the button, this resistor “pulls” the button’s voltage down to 0 V, and does not let much current leak to GND in the case that there is a voltage arriving. This will be the start button to kick everything off.

Additionally, I will wire 1 piezo element to PB[1], in series with a resistor in the [2.2 Ω, 10 kΩ] range. I will wire 1 LED to the OC0A pin, in series with a resistor in the [500 Ω, 2.2 kΩ] range. This will limit the overall power consumption of my circuit, and avoid short circuits.

I will then add my board to a chassis to give it a smart ice cream truck look, and then will then write code in C++ to control my setup. This will involve for and while loops to manage flow

## Evaluation

I will know my project is working if my smart ice cream truck indeed is indeed responsive to my remote, that is, it moves in the instructed direction for example left when I press the left arrow, right when I press right, forward when I press the up arrow, and stops when I press the down arrow, starts playing jolly ice cream tunes when I press button 5, and stops when I press 8, and correctly comes to a stop when it approaches an obstacle.

# Resources

## Parts list

* Any computer with the Arduino IDE installed (x1)
* Metro Mini development board (x1)
* USB-microUSB data cable (x1)
* insulated copper wire (40 cm should do it)
* full-sized breadboard (x1)
* LEDs (x2)
* resistors in the   
  [500 Ω, 2.2 kΩ] range (x2)
* resistors in the [2.2 kΩ, 10 kΩ] range (x2)
* L293D H-bridge (x1)
* DC barrel jack (x1)
* DC motors (x2)
* TSOP382 IR receiver
* IR remote control
* piezo element (x1)
* Battery pack (x1) with 4 rechargeable AAs
* Push button (x1)

Chassis

* Wheels (x2)
* Cardboard rectangle (x1)
* Large binder clip (x1)
* Twist ties (x4) ( or spare wire)
* Double-stick tape (1-2 in2)
* Rubber band (x1)

## References

* ATmega datasheet
* IR sensor datasheet

# Timeline

|  |  |
| --- | --- |
| **Date** | **Goal** |
| 11/14 | Think about idea, and draft proposal |
| 11/16 | Write out proposal and submit it |
| 11/17 | Design circuit on the board, write up |
| 11/18 | Design circuit on the board, write up |
| 11/23 | Writing code to control motors, write up |
| 11/24 | Testing, debugging and write up |
| 11/26 | Writing code to incorporate IR receiver and ADC, write up |
| 11/27 | Testing, debugging and write up |
| 11/29 | Writing code to produce sound on piezo, write up |
| 11/30 | Testing, debugging and write up |
| 12/1 | Attach to chassis and demo |
| 12/2 | Demos debugging and write up |
| 12/3 | Demos debugging and write up |
| 12/4 | Submit project |

# 