Final Project:
Flashlight-Steered Vehicle
With Object Detection

ECE 110

Lab Section AB4

Josh Rabinowitz

12/5/2014

Table Of Contents

I. Introduction

Problem Description:

The goal for this project is to build a car that utilizes sensor feedback to navigate a course of our choosing. The first constraint is the design must use a non-random algorithm to navigate the course. The second constraint is that only parts available from the ECE 110 lab kit may be used; the only exception to this is the Ultrasonic Ranging Module HC-SR04, which did not come in each student's Sparkfun Electronics kit, but was instead handed out during the lab section Lastly, as stated above, the car must use at least one type of sensor in the design. These are the only rules and constraints that were explicitly given.

Design Concept:

While utilizing sensor feedback makes the car more useful, a car with very limited capabilities is no fun to own. Therefore, this design challenge calls for a vehicle with several added capabilities. The three features of the car will be flashlight steering, obstacle detection and the ability to drive backwards (as well as forwards).

The concept behind the flashlight steering is simple- the car must be able to drive straight, as well as turn left or right, in order to be fully mobile. The back of the car will be divided into the left side, middle and right side; shining a flashlight on each section drives the car in THAT direction.

As for the obstacle detection, the car should be designed so as to not bump into walls (and basically self-destruct). Therefore, the car must have a system for object detection mounted on the front. This system must be able to stop the motors from spinning in the case of a blocked path- even if the 'driver' is attempting to steer the car forward.

While the second feature helps to protect the car, it doesn't allow the car to continue driving once an obstacle is found, because power to the wheels has been cut. As a result, the driver must keep picking up the car and moving it manually in the event of an obstacle. To deal with this, the car must be given the capability to drive backwards. With this capability, the car can drive away from any obstacles and allow the driver to continue steering.

Control Scheme:

State (assuming car is powered)	Action
Object in front.	Drive backwards
No object; light on middle sensor.	Drive forwards
No object; light on left sensor.	Turn Left
No object; light on right sensor.	Turn Right
No object; no light.	Off

Parts Needed:

Item	Value (if applicable)	Quantity
Sparkfun Vehicle Chassis Kit		1
Dc Motor (Sparkfun Kit)		2
Protoboard		1
Breadboard		1
Redboard		2
Resistor	330-ohm	3
Photoresistor #GL5528		3
H-Bridge IC #754410NE		2
Ultrasonic Ranging Module #HC-SR04		1
AA Battery	1.5V	4
9V Alkaline Battery*	9V	1

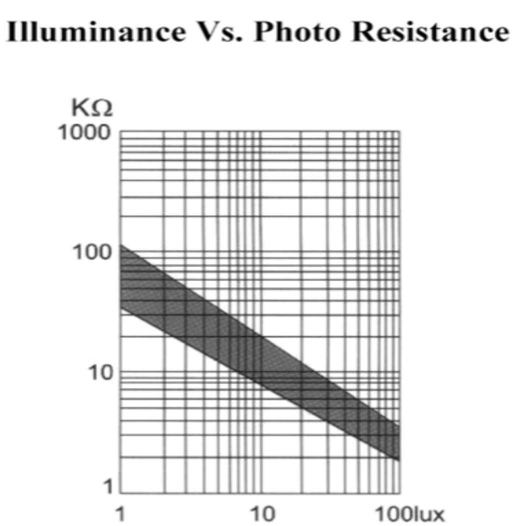
*May be replaced with 7.2V Rechargeable Ni-MH Battery for lab demonstration!

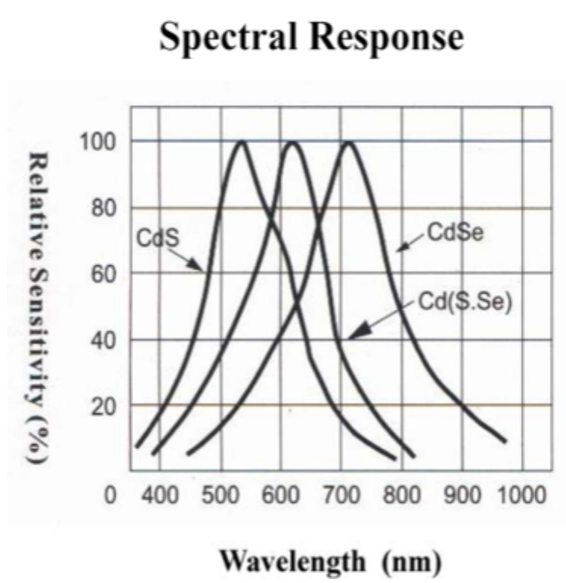
II. Component Analysis

<u>Characterizing the Sensors:</u>

- **Photoresistors**

The photoresistors used for this design are CdS Photoconductive Cells (part #GL5528). When not illuminated they have a Dark Resistance of at least 1M-ohm (at 0 Lux). When illuminated they have a Light Resistance of 8—20k-ohms. At room temperature, each photoresistor dissipates about 100mW. To test each photoresistor, a simple code was needed to read the analog input from the middle node of a voltage divider circuit. The voltage divider circuit included the photoresistor and a 330-ohm resistor in parallel, with the other end of the photoresistor at 5V and the other end of the resistor grounded. On the serial monitor, the input when a flashlight was pointed at the photocell rose well over 900, while without the flashlight the number remained in the double digits or slightly over 100. Therefore, in the final code, the car is 'told' to drive when the analog input from any of the photoresistor pins reaches 900. Graphs illustrating the photo-resistance (based on "illuminance") and the spectral response of the photoresistors at different wavelengths are included.





- **H-Bridge**

The type of IC used for each H-Bridge is a SN754410 Quadruple Half-H Driver. It provides bidirectional drive currents up to 1A at voltages from 4.5V to 36V, and is designed to drive inductive loads such as relays, solenoids, motors and other high-current/high-voltage loads in positive supply applications such as this. To test each H-Bridge, the Vcc2 pin was connected to the motor battery and the other voltage pins were connected to 5V through the Redboard. A testing program sent the proper digital signals to first drive a motor forward, then backward to ensure bidirectional functionality. Each H-Bridge provides the same current amplification as a transistor based motor sub-circuit, since transistors are included in the IC.

- **Sonar Sensor**

The sonar sensor used is the Ultrasonic Ranging Module HC-SR04, which provides a 2cm-4m non-contact measurement function with a ranging accuracy of about 3mm. The module includes ultrasonic transmitters, a receiver and control unit, and runs on 5V. It works by sending eight 40kHz 'trigger' pulses and detecting an 'echo' pulse back, all after a 10 microsecond digital high-level signal. If the signal is reflected back, the distance from the detected object is calculated using the following formula:

- Test distance = (high level time * velocity of sound (340M/S)/2)

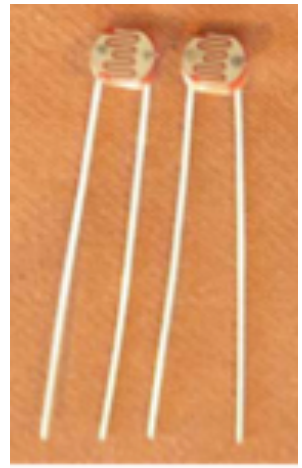
For the purposes of this design, the formula above is meant to calculate the desired distance from the object at which the car stops. The car is programmed to stop about 4cm in front of an object. Therefore, the formula was implemented in the final code as follows:

- ***..." distance = (duration/2)/29.1;***
  if (distance < 4)".....[Drive backwards]...

This part of the code causes the car to detect an object placed 4cm in front of the car and drive in reverse, succeeded by a delay of about half a second (or 500ms). The number '29.1' causes the distance to be in centimeters. Before the sonar sensor was mounted on the car, it was tested by connecting to the Redboard and output signals to a small circuit consisting of a red and a yellow LED, each connected in series with a current limiting 330-ohm resistor. If no object were detected within the desired range, the yellow LED would receive a high digital signal from the Redboard. If an object came within range, such as a hand, the red LED received a high digital signal instead. In this manner, the sensitivity of the module could be tested visually- just as with the photoresistors- and therefore **no** actual measurements were needed to characterize these sensors!

Design Considerations:

- **Photoresistors**



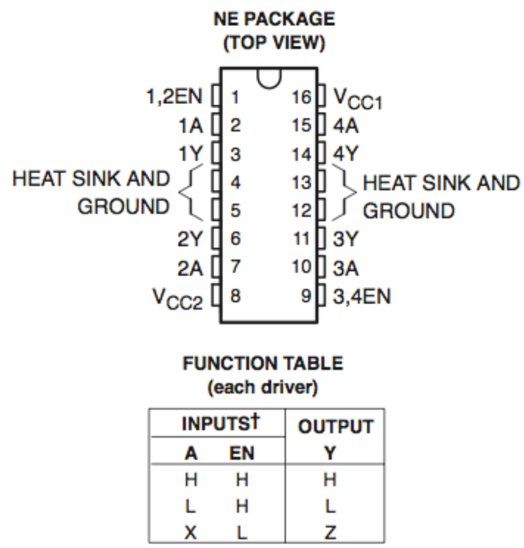
The first sensor chosen for this design was the photoresistor, which is used to implement the flashlight steering effect. This type of sensor was chosen because it was designed for use with bright lights. Initially, the design revolved only around three photoresistors, and used only hardware as opposed to software as well. Two motor sub-circuits were built using transistors (the same as in lab 8), and the biasing resistor for each transistor was 'preceded' by a photoresistor on the corresponding side. In the middle, a third photoresistor fed current to both sub-circuits through diodes. Therefore, shining a flashlight on each side of the back of the car drove the car in the corresponding direction (left, right or straight) by decreasing the resistance used to bias the correct transistor(s) and allowing power to reach the appropriate motor- or both. While this design could be steered perfectly, it could not detect obstacles or go backwards and the user would have to take care not to drive the car into anything- hence the need for more sensors, an H-Bridge and a program for the Redboard (which initially just provided power).

- **Optical Sensors**

The next step involved adding sensors on the front to stop the vehicle in the event of a blocked path. The most obvious choice for this sensor was the optical sensor, which is described on the ECE 110 site as useful for object detection and black to white transitions (at this point the sonar sensor hadn't been offered to the students in lab, and the optical sensors came with the lab kit). Specifically, this is the QRD1113/1114 Reflective Object Sensor included in the Sparkfun Kit. At first, one optical sensor was mounted on the front of the car using tape, and then powered and connected to the Redboard as indicated by the datasheet. The analog-read program written to test the sensor, by writing its output to the serial monitor, showed that the value being output to the monitor was inconsistent when the sensor faced different directions or when the lighting in the room varied by even a small degree. However, the number never fell below about 900 unless an object was placed directly in front of the sensor. Therefore, the sensor functioned using this value as a 'detection threshold' and was used to stop the motors even if the user pointed a flashlight at one of the photoresistors. The problem with this setup was the narrow width of the sensor's range, so the next modification was adding a second sensor of the same kind, and having each sensor on one side of the front of the car. Unfortunately, the two optical sensors had slightly different readings, as one proved to be more

sensitive than the other, and the car essentially used only one of the two sensors at a time. An even bigger problem was the extremely short range of each optical sensor. Ultimately, for the car to effectively avoid objects, these sensors had to be replaced with the sonar sensor (this was done after implementing the H-Bridge).

- **H-Bridge**



While the car could now stop in front of an obstacle, it would have to be picked up and moved by hand if this was the case. To solve this problem, the car had to be given the capability to go backwards, and this involved incorporating an H-Bridge into the circuit in place of the transistor-powered motor sub-circuits. Given the desired control scheme, however, one H-Bridge couldn't provide the needed functionality. Since both switches on one side of the chip can't be open at the same time, the chip can't turn off only one motor while it is powered- each motor can only go backwards or forwards, or the entire chip must be disabled given only two of the sixteen switch combinations are valid (according to the module on the web site). The two motors cannot use separate power supplies, because one power pin powers the logic of the chip and the other powers BOTH outputs. Although each side has an enable pin, this pin only enforces the internal logic of the IC and disabling that pin doesn't necessarily turn off the corresponding motor (this was tested). The desired control scheme involved turning off one wheel to turn the car either direction. Therefore, the design called for a second H-Bridge, with each H-Bridge running one of the motors. Both enable pins are kept on high digital outputs, the pins that power the motors are connected to Vin on the Redboard, and the chips' power pins are controlled separately using digital outputs. To turn the car, one of the chips is turned off while the other remains powered, and to drive straight both chips are powered.

- **Sonar Sensor**

Once the car could successfully drive backwards as well as forwards, the design considerations returned to the object detection system. The optical sensors, being too sensitive to the ambient light in the room, required calibration before use in each new setting. This calibration involved changing the threshold of luminosity that will indicate the presence of a detected object, and the threshold varied between the two sensors. The biggest problem was the extremely short range of the sensors (about 1-2cm). The car must be able to detect a wall BEFORE hitting it, as well as quickly stop and change direction. Once the sonar sensor was introduced during the lab section, it became apparent from observing other projects that the sonar sensor has a significantly longer range (about 4m). Having noticed this, the two optical sensors were removed and replaced with one sonar sensor, which was the final addition to the car and proved much easier to implement than the optical sensors. The optical sensors required more hardware and took up more space on the breadboard, while the sonar sensor requires only programming and needs to connect only to the Redboard. In addition to the longer range, the sonar system doesn't depend on the lighting conditions of the room, has a wider detection range and allows the user to choose the exact distance the car can be from an object.

While the optical sensor and the sonar sensor use very different methods to determine whether or not the car is approaching an obstacle, the sonar sensor has a much greater range of detection while operating at the same voltage. The data below compares some of the electrical parameters of the two types of sensors.

OPTICAL SENSOR:

PARAMETER	TEST CONDITIONS	SYMBOL	MIN	TYP	MAX	UNITS
EMITTER Forward Voltage	I_F = 20 mA	V_F	–	–	1.7	V
Reverse Current	V_R = 5 V	I_R	–	–	100	μA
Peak Emission Wavelength	I_F = 20 mA	λ_{PE}	–	940	–	nm
SENSOR Collector-Emitter Breakdown	I_C = 1 mA	BV_{CEO}	30	–	–	V
Emitter-Collector Breakdown	I_E = 0.1 mA	BV_{ECO}	5	–	–	V
Dark Current	V_{CE} = 10 V, I_F = 0 mA	I_D	–	–	100	nA
COUPLED QRD1113 Collector Current	I_F = 20 mA, V_{CE} = 5 V D = .050" (6,8)	$I_{C(ON)}$	0.300	–	–	mA
QRD1114 Collector Current	I_F = 20 mA, V_{CE} = 5 V D = .050" (6,8)	$I_{C(ON)}$	1	–	–	mA
Collector Emitter Saturation Voltage	I_F = 40 mA, I_C = 100 μA D = .050" (6,8)	$V_{CE\ (SAT)}$	–	–	0.4	V
Cross Talk	I_F = 20 mA, V_{CE} = 5 V, E_E = 0 (7)	I_{CX}	–	.200	10	μA
Rise Time	V_{CE} = 5 V, R_L = 100 Ω	t_r	–	10	–	μs
Fall Time	$I_{C(ON)}$ = 5 mA	t_f	–	50	–	μs

SONAR SENSOR:

Working Voltage	**DC 5 V**
Working Current	**15mA**
Working Frequency	**40Hz**
Max Range	**4m**
Min Range	**2cm**
MeasuringAngle	**15 degree**
Trigger Input Signal	**10uS TTL pulse**
Echo Output Signal	**Input TTL lever signal and the range in proportion**
Dimension	**45*20*15mm**

- **Code**

Since this design includes a software side to complement the hardware side, an additional design consideration was how to program the sensor. The first algorithm used with the sonar system essentially reversed the direction of the motors in the event of an obstacle. However once the car backs up and the obstacle is out of range, the car no longer 'senses' the need to go backwards. This resulted in the car not backing far enough from walls to be steered in a different direction, and it would become stuck like before. To fix this, a small delay was added in the program's loop just after the car is 'told' to go in reverse. This causes the car to back up slightly even after the wall is out of range, giving it room to turn left or right and continue being steered.

- **Power**

With the set of components established, only one problem remained: it became evident once the final prototype was designed, built and tested that between the sensors, IC's and motors, too much power was being dissipated overall. Therefore, a second battery was needed which, in turn, called for a second Redboard. The car uses one Redboard to power the sensors and send/receive signals, while the second Redboard is used only to power the motors via the H-Bridges.

III. Design Description

Construction:

- **Photoresistors**

  The photoresistors are placed where it is necessary to point the flashlight and steer the car; each photoresistor module connects to an analog input pin.

- **H-Bridge Module**

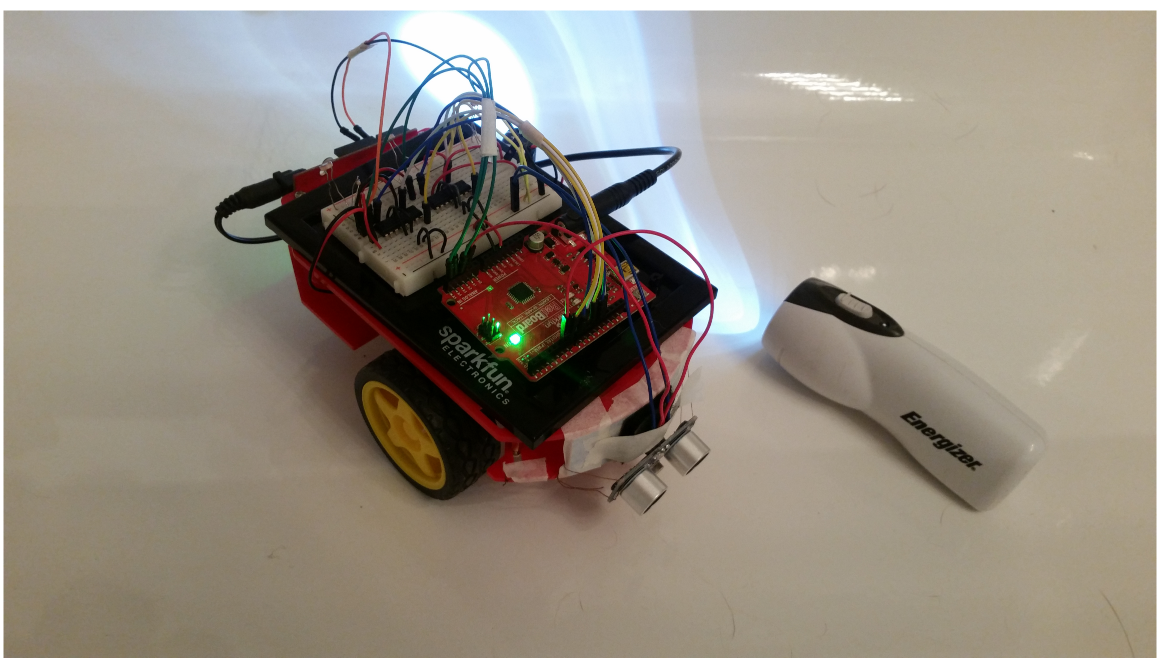
  Each H-Bridge connects to a motor, the Vin pin of the 'power Redboard', and three digital outputs on the 'signal Redboard'; two of these control the logic switches and change the direction in which the motor spins, while the third enables/disables the chip's power- allowing each motor to turn on/off regardless of the other motor

- **Sonar Module**

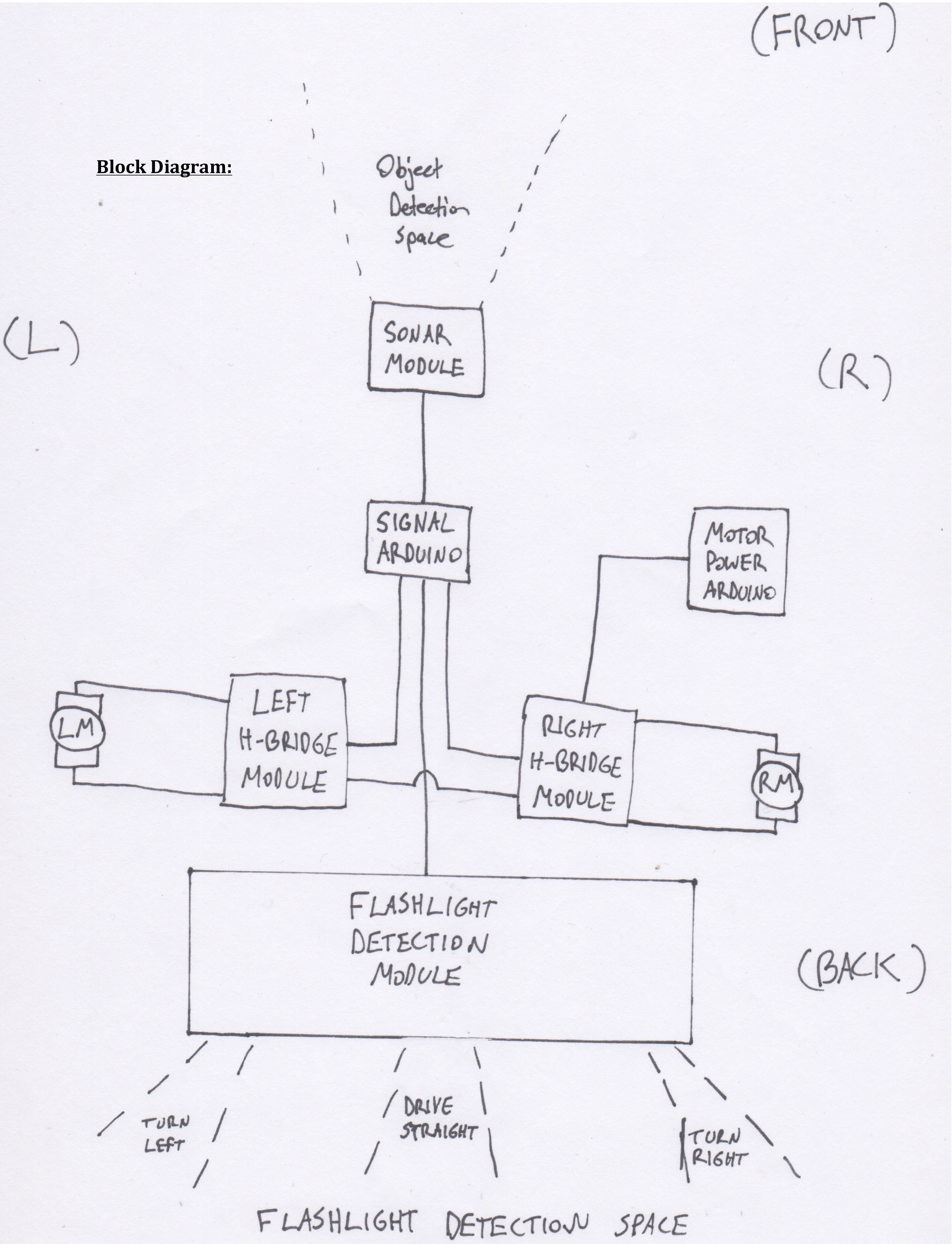
  The sonar module connects to the signal Redboard via the echo/trigger pins, as well as to 5V/GND on the breadboard

- **Mechanical Construction**

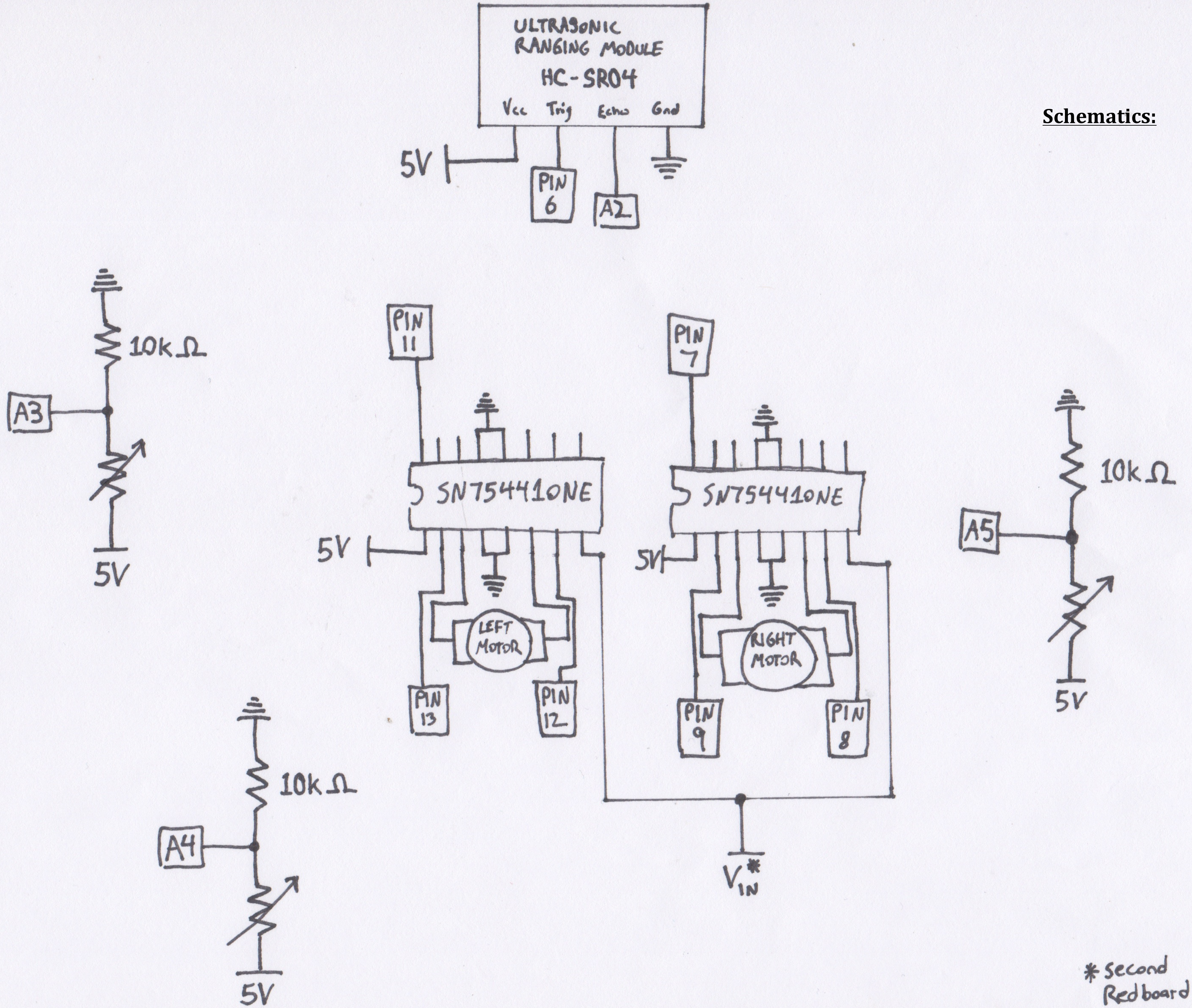
  As shown, the sonar module is mounted on the front of the car using a combination of tape, string and Velcro. While the signal Redboard mounted on the Protoboard connects to the AA batteries inside of the chassis, the power Redboard and the 9V battery to which it connects are fastened onto the car using tape.



Block Diagram:



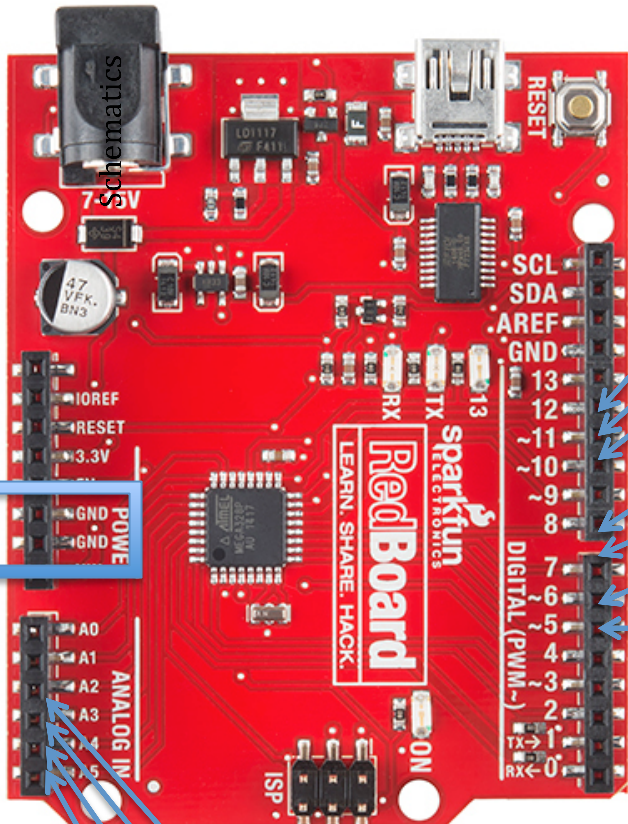
Schematics:



ULTRASONIC RANGING MODULE
HC-SR04
Vcc
Trig
Echo
Gnd
5V
PIN 6
A2
10k Ω
A3
5V
PIN 11
SN754410NE
5V
LEFT MOTOR
PIN 13
PIN 12
PIN 7
SN754410NE
5V
RIGHT MOTOR
PIN 9
PIN 8
V_{IN}^{*}
10k Ω
A5
5V
10k Ω
A4
5V
* Second Redboard

Redboard Connections:

- **Signal Redboard**



(+)
(-)

13: L H-Bridge (switch1)

12:L H-Bridge (switch2)

11:L H-Bridge (Vcc1)

9: R H-Bridge (switch1)

8: R H-Bridge (switch2)

7: R H-Bridge (Vcc1)

6: Sonar (Trig)

A2: Sonar (Echo)

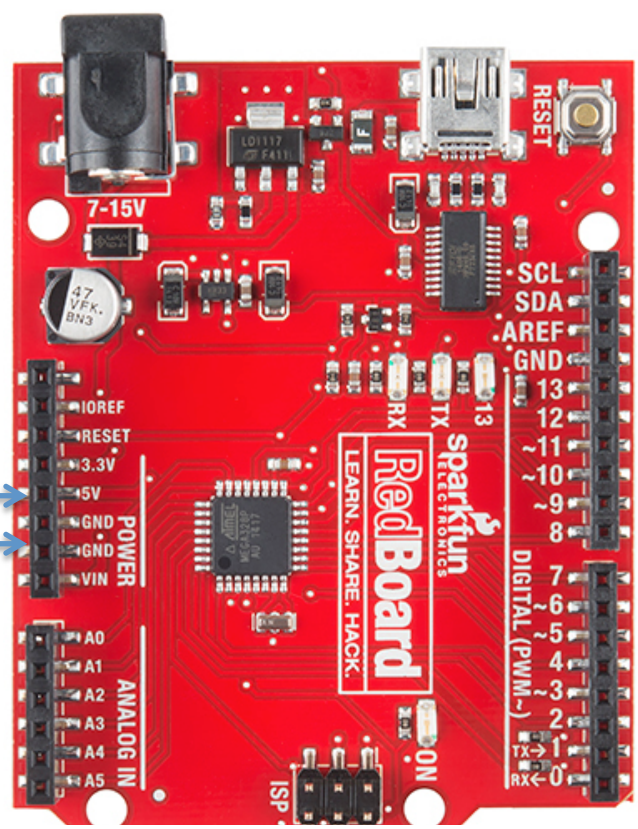
A3: L PR Module

A4: M PR Module

A5: R PR Module

**(+/-) indicates power bus of breadboard

- **(Motor) Power Redboard**



(-)

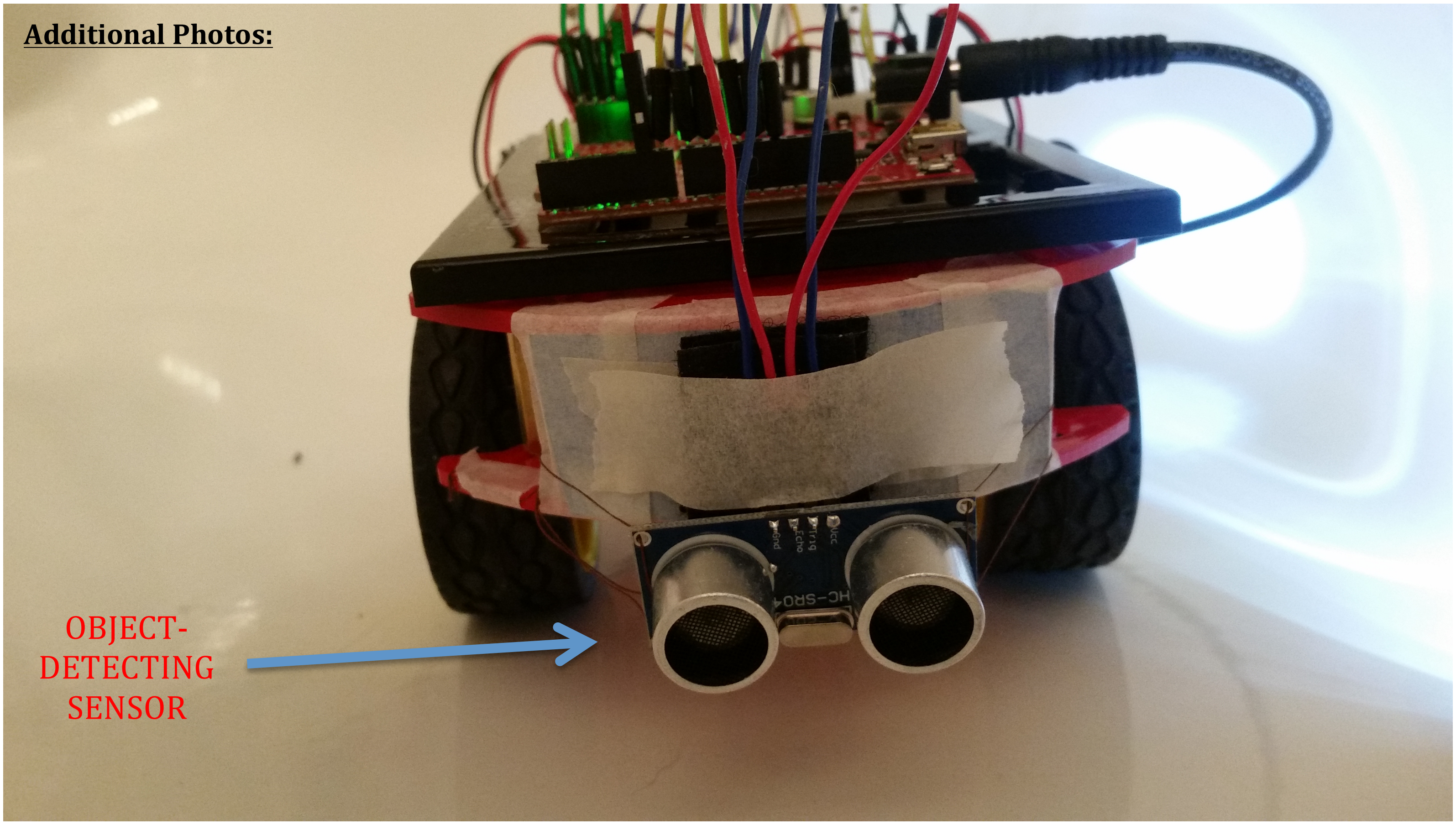
Vin: R H-Bridge (Vcc2)

Program:

```
//SET PINS
#define echo A2
#define left_photo A3
#define mid_photo A4
#define right_photo A5
#define trig 6
#define enable_RM 7
#define Rswitch2 8
#define Rswitch1 9
#define enable_LM 11
#define Lswitch2 12
#define Lswitch1 13
void setup()
{
 //INITIALIZE
 pinMode(echo, INPUT);
 pinMode(left_photo, INPUT);
 pinMode(mid_photo, INPUT);
 pinMode(right_photo, INPUT);
 pinMode(trig, OUTPUT);
 pinMode(enable_RM, OUTPUT);
 pinMode(Rswitch2, OUTPUT);
 pinMode(Rswitch1, OUTPUT);
 pinMode(enable_LM, OUTPUT);
 pinMode(Lswitch2, OUTPUT);
 pinMode(Lswitch1, OUTPUT);
}
void loop()
{
 //CHECK FOR ECHO
 long duration, distance;
 digitalWrite(trig, LOW);
 delayMicroseconds(2);
 digitalWrite(trig, HIGH);
 delayMicroseconds(10);    //SENDS TRIGGER PULSE
 digitalWrite(trig, LOW);
 duration = pulseIn(echo, HIGH);   //TIMES ECHO
 distance = (duration/2)/29.1;   //CALCULATE DISTANCE OF OBJECT IN CM
 if (distance < 4)
```

```
  {
   //BACKWARDS
   digitalWrite(Lswitch1, HIGH);
   digitalWrite(Lswitch2, LOW);
   digitalWrite(Rswitch1, HIGH);
   digitalWrite(Rswitch2, LOW);
   digitalWrite(enable_LM, HIGH);
   digitalWrite(enable_RM, HIGH);
   delay(200);   //KEEPS CAR IN REVERSE AS OBSTACLE LEAVES RANGE
  }
  else
  {
   //FORWARDS
   digitalWrite(Lswitch1, LOW);
   digitalWrite(Lswitch2, HIGH);
   digitalWrite(Rswitch1, LOW);
   digitalWrite(Rswitch2, HIGH);
   //CHECK FOR FLASHLIGHT
   int left = analogRead(left_photo);
   int mid = analogRead(mid_photo);
   int right = analogRead(right_photo);
   int flashlight = 900;
   //LEFT
   if (left > flashlight) {
     digitalWrite(enable_LM, HIGH);
     digitalWrite(enable_RM, LOW);}
   //STRAIGHT
   else if (mid > flashlight) {
     digitalWrite(enable_LM, HIGH);
     digitalWrite(enable_RM, HIGH);}
   //RIGHT
   else if (right > flashlight) {
     digitalWrite(enable_LM, LOW);
     digitalWrite(enable_RM, HIGH);}
   //OFF
   else {digitalWrite(enable_LM, LOW);
         digitalWrite(enable_RM, LOW);}
 }
 }
```

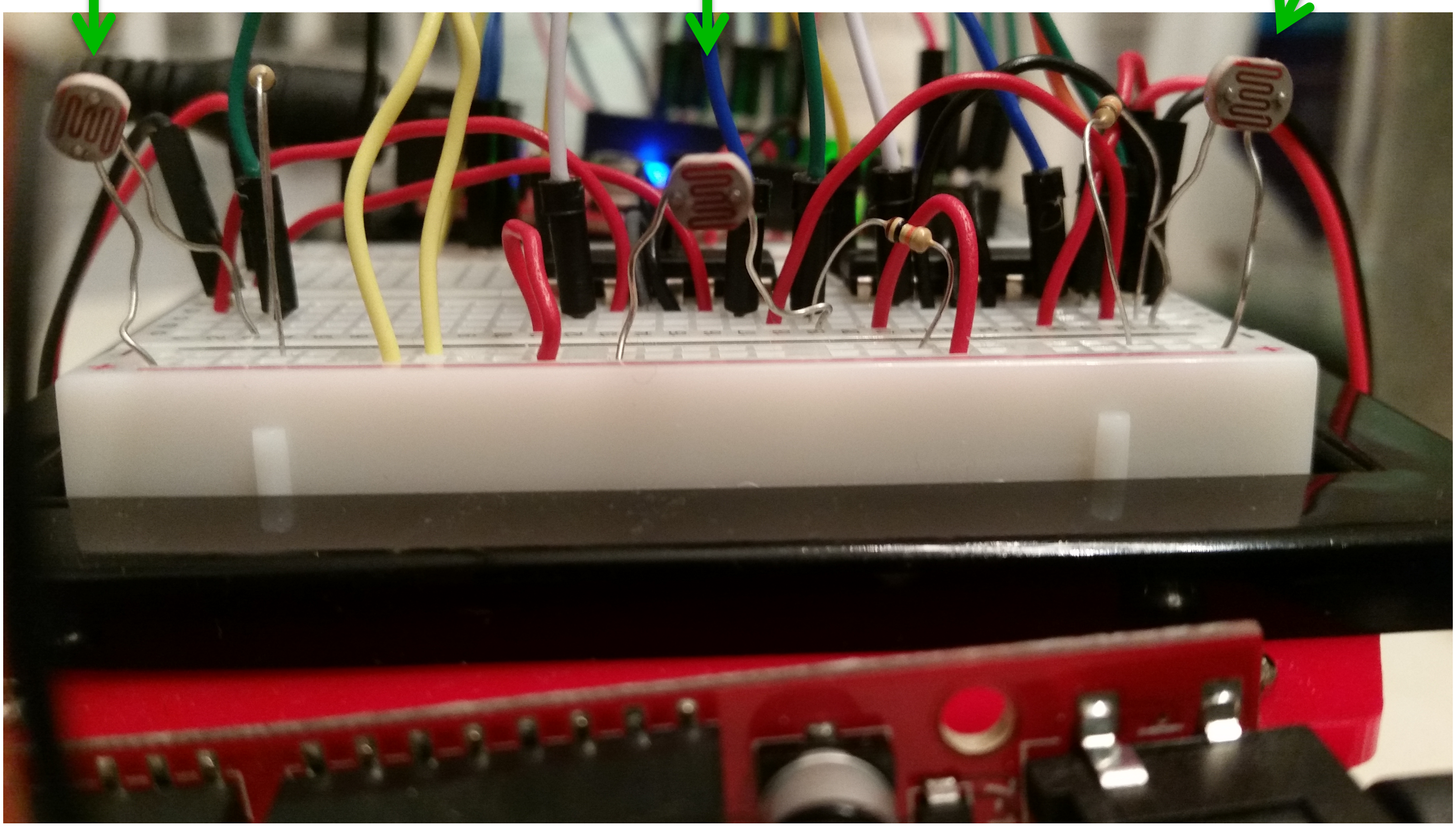
Additional Photos:



SHINE HERE TO TURN LEFT

SHINE HERE TO DRIVE STRAIGHT

SHINE HERE TO TURN RIGHT



IV. Conclusion

This design project was a good introduction to hands-on electrical engineering as well as a prime example of how sensor feedback can be used within a fairly simple design to implement any number of practical real world tasks. The problems that needed to be solved during the project are also typical parts of the engineer's design process. It is often the case, as it was with this project, that the initial design doesn't quite match the final prototype. What matters, however, is that all the available resources were used to ensure the finished product accomplishes what was intended from the beginning. In this case, the initial design concept involved some sort of sensor on the front that would detect obstacles and allow the vehicle not to destroy itself by driving into walls. While the optical sensors didn't optimize this function of the car as planned, the sonar sensor was incorporated just as easily, and it led to an essentially equivalent design- but with better performance. In addition, this project provided a good sample of what engineering entails beyond the initial creative idea. The real work often comes with testing, modifying and reiterating. Much of the time on this project was spent building different circuits just to test and 'calibrate' the various sensors (for instance, finding the analog input value that indicates a flashlight is FOR SURE pointing at a given photocell). Perhaps the obstacle in this design that was hardest to overcome was the H-Bridge problem. It took a bit of 'outside-of-the-box' thinking to determine that unless two switches could be open on one side, a second H-Bridge was needed to implement the bidirectional current while maintaining the ability to shut off only one wheel at a time. Treating the chip-powering pins as enables for our purposes allowed the two to be controlled separately with a few simple lines of code.

Overall, the design concept was implemented as intended and the car has the all the desired functionality- it can be driven on any 'course', because it can be steered by the user! It has the capability to not only respond to flashlight steering (which controls quite fluently) but to detect obstacles and drive backwards to distance itself from them. Additionally, the object avoidance system overrides the flashlight steering. The lesson learned from this project is that to design something successfully as an engineer, one must be willing to change things along the way and to admit when something simply does not work the way it should. It wasn't easy tearing off the optical sensors after spending so much time testing them and incorporating them into the full circuit. However, it was worth it in the end; it's much more fun to play with a toy car when you know you're not just the driver, but also the one who designed it and built it from nothing but an idea.