Line-Detecting Mini Monster

Watch this Car Detect Lines for Hours!

TEAM C1

Revision	Description	Date
A.1	(Thomas) Document created, added section 1 (Scope), 2 (Abbreviations), 4 (hardware), 5 (Power Analysis), and 6 (Test Process).	9/17/2024
A.2	(Jacob) added to section 2 (Abbreviations), added section 3 (Overview) and section 7(Software), edited section 6 (Test Process), and minor formatting correction.	9/18/2024
A.3	(Jacob) added footer, suggested alternate name, section 6.4, section 8 (flowchart), section 9 (Code), and section 10 (Conclusion). (Anish) added more information to section 4.4, 4.5, and 6 and updated section 2. Formatted section 9 (Thomas) Formatting corrections, changed Section 9 to screenshots	9/19/2024
A.4	(Daret) Added section 8 and section 9	9/17/2024
B.1	(Anish) Updated Section 1, 3.3 and 9 based on feedback	10/7/2024
B.2	(Jacob) added sections 3.4, 4.7, and 4.8. Updated sections 2, 3, 4, 4.1, 4.9, and	10/8/2024

	5 based on feedback. Adjusted Table of Contents.	
B.3	(Daret) updated section 6, 7 and 10	10/10/2024
B.4	(Thomas) Updated Device Name and Slogan. Added to and updated Sections 1, 2, 3, 4, 7, and 8 to reflect recent developments from LCD Screen to Toy Car. General edits across the document.	10/15/2024
C.1	(Jacob) added section 5. merged sections 4.8 and 4.9. added new section 4.9. updated sections 2, 3, 4, and 7.	10/24/2024
C.2	(Thomas) Updated Device Name and Slogan. Updated section 1, 2, 8 to reflect recent developments from Shape Making Toy Car to Line Detecting Toy Car. Updated section 4 for clarity within the IR detector system.	10/27/2024
C.3	(Daret) Updated Section 6 and 10 - conclusion	10/27/2024

Team Members:
Thomas Tien Van
Jacob Steven Storer
Anish Nagulavancha
Daret Galeana

Please use this Google Document to work on the Project Write Up. Thomas Tien Van will copy and paste from this document to a

Word Document with the proper formatting and turn it in for the team.

Table of Contents

1. Scope

The product is a toy car that is controlled by an embedded system that can detect black lines made out of electrical tape. The toy car will run until it finds a line of tape. Once it finds a line, it will stop and reorient itself along the line.

The product is controlled by an MSP430FR2355 microcontroller processing board, which is interfacing with motors and wheels, as well as an LCD screen. This LED screen displays four lines of text that tells the user its current action. The entire system is powered by four AA batteries.

This product would be an excellent educational tool that will fascinate children, as it is fun to watch a toy car run until it finds a line of electrical tape.

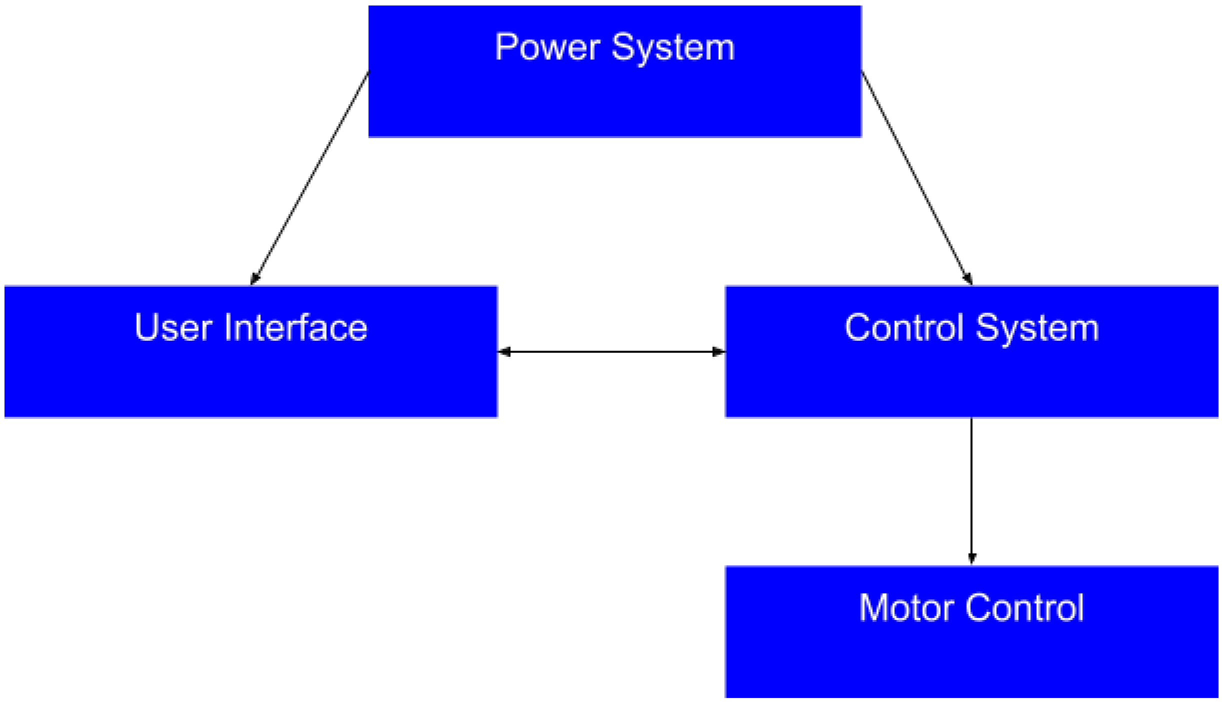
2. Abbreviations

Abbreviation	Definition
LCD – “Liquid-Crystal Display”	A flat display that modulates light to show a message.
FRAM – “Ferro-electric Random Access Memory”	A type of non-volatile memory that preserves essential code and data without power and provides fast writing times along with long data retention.
LED – “Light Emitting Diode”	A semiconductor device that emits light when current flows through it.
USB – “Universal Serial Bus Type”	A standard connection type for electronic devices. The MSP430 uses a micro-USB connection.
PCB – “Printed Circuit Board”	A circuit board used to electrically connect electronic components using pads and conductive tracks.
EZ-FET – “Flash Emulation Tool”	A component that debugs, configures, and delivers code to the central processing unit.
FET – “Field-Effect Transistor”	A circuitry component that utilizes an input voltage to control the current flow through a semiconductor channel.
pMOSFET –	A type of FET that is heavily doped with

“Positive Metal-Oxide-Semiconductor Field-Effect Transistor”	boron in the p-region, and uses n-type material for the substrate, which allows voltage through when a negative voltage is applied to the gate.
nMOSFET – “Negative Metal-Oxide-Semiconductor Field-Effect Transistor”	A type of FET that is heavily doped with boron in the n-region, and uses p-type material for the substrate, which allows voltage through when a positive voltage is applied to the gate.
IDE – “Integrated Development Environment”	A type of software used for writing and compiling code. The main IDE used for this product is Code Composer.
MHz – “MegaHertz”	The unit for frequency. This determines how fast the clock of the MSP430FR2355 runs at. The product operates at a clock speed of 8 MHz.
MCLK – “Main Clock”	One of the MSP430FR2355’s clocks.
SMCLK – “Sub-Main Clock”	One of the MSP430FR2355’s clocks. The Sub-Main clock is based off of the Main Clock.
ADC – “Analog-to-Digital Converter”	A peripheral in the MSP430FR2355. It reads voltages from hardware and converts them into digital values that can be processed by the system.
IR – “Infrared”	Electromagnetic waves with a wavelength above that of visible light

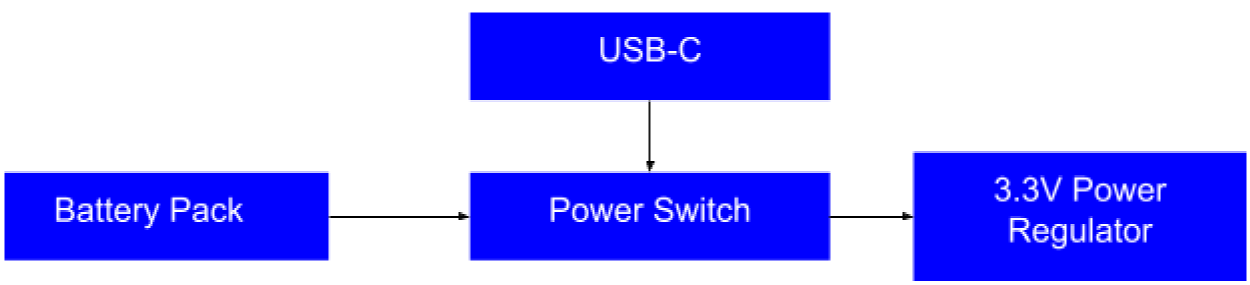
3. Overview

The system consists of a central FRAM wired to a PCB and a LCD screen. The power system provides a steady 3.3V to systems. The FRAM manages the behavior of the LCD screen and the motors. The LCD screen uses instructions from the FRAM to display messages according to certain user instructions, while the motors receive signals to turn on and off.



3.1 Power System

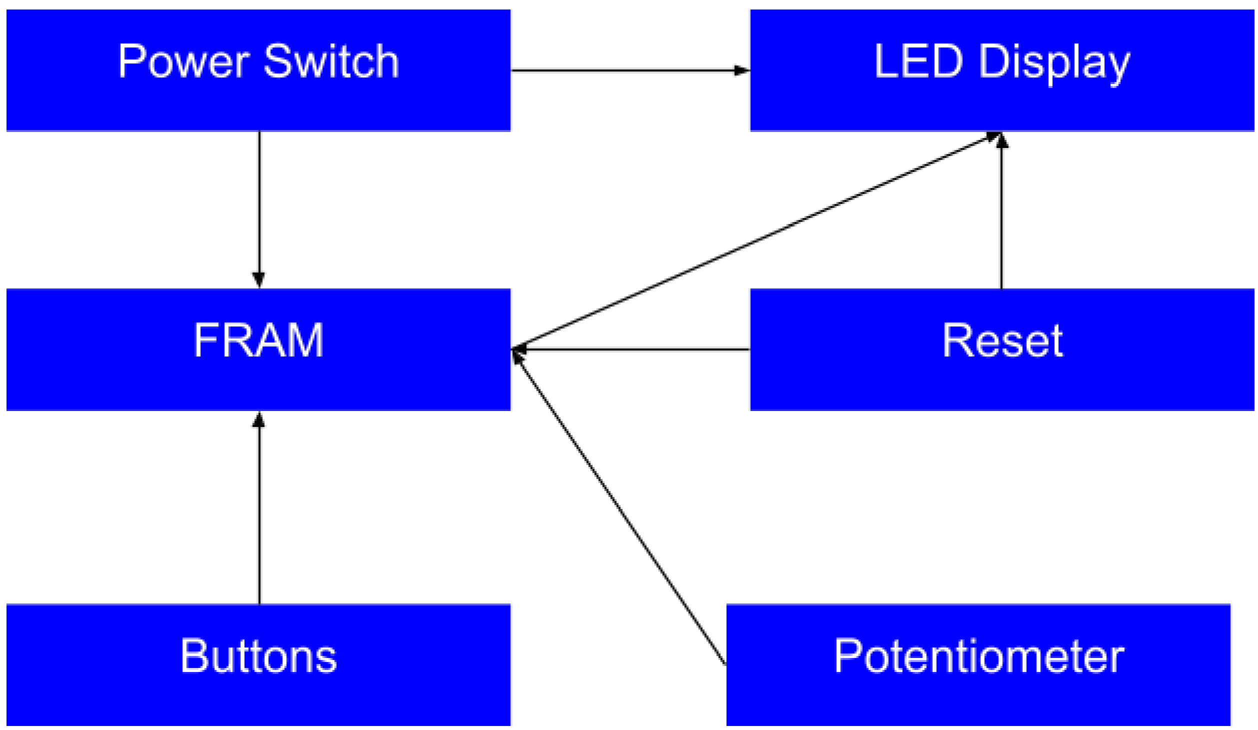
The power system is a 4 1.5V AA series battery pack. A mechanical switch, and a 3.3V regulator. The battery pack connects to the circuit through the LCD's circuit board and is bridged using the switch. When the switch is in the 'on' position, power flows through the 3.3V regulator on the FRAM which distributes power through the rest of the board.



3.2 Control System

The control unit is an MSP430FR2355 microcontroller and its accessory components, a micro USB port and the eZ-FET. The micro USB uploads code from an external source and delivers it to the eZ-FET to be handled. The eZ-FET checks incoming code for errors and configures for use by the microcontroller. The controller stores code to be carried out and instructs other components of the system how to behave. The IR camera consists of an IR LED and two IR detectors mounted to its own circuit board. The camera allows the MSP430 microcontroller to act without user input.

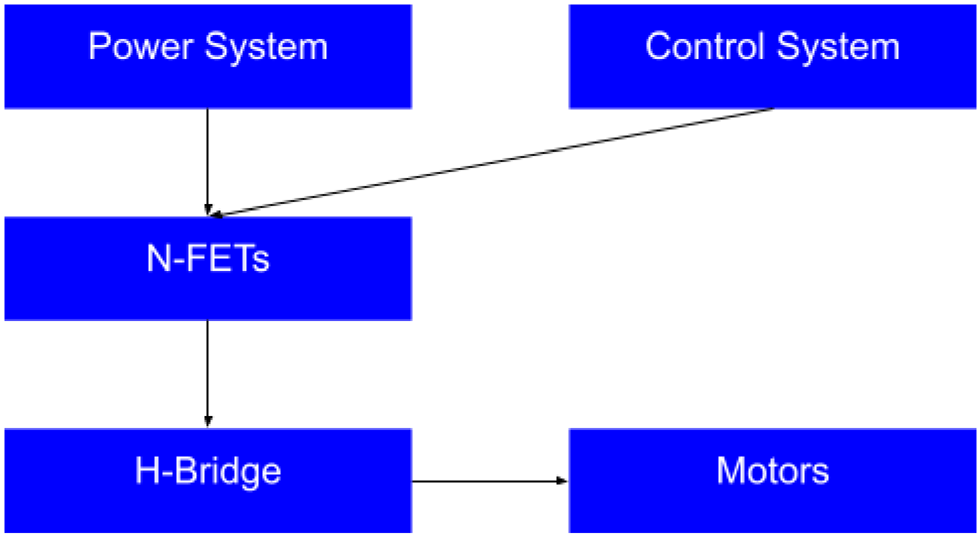
3.3 User Interface



The system's user interface has a power switch, a SSD1803 LCD display screen, two buttons, and a potentiometer. The power switch allows power to flow from the battery pack to the rest of the system. The buttons send instructions to the FRAM to set the machine's current action and changes the LCD's display to let the user know what the action is.

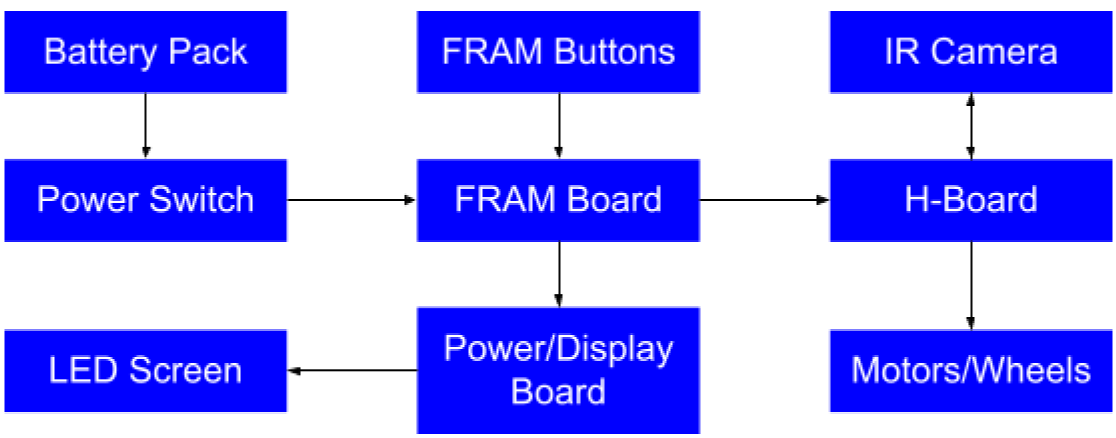
By default, the LCD displays "   NCSU   ". When the buttons are pressed, the machine counts the button presses and displays its current action on the LCD, i.e. "Waiting", "intercept", and "Get White". Lastly, there is also a reset switch allowing for the screen to reset to the default screen after performing a task.

3.4 Motor Control



The H-Board holds 2 N-Fets and mountings for the motors. The Control System manipulates the N-FETs by providing voltage to the board. When voltage is applied to the board, one output of the H-Board becomes positive and the other is set to ground, supplying power to the motor. Each motor can be powered independently.

4. Hardware



4.1 FRAM Board

The FRAM Board is a Texas Instruments board with a processor. This processor controls a multitude of LEDs on the board. The processor can also send signals to other boards, such as the Power/Display Board using the pins on the FRAM Board. The FRAM board also features two buttons that can be programmed to do various functions, as well as a reset button to reset the unit.

4.2 FRAM Buttons

The buttons on the FRAM board are used to control or provide inputs to the machine outside of the coding environment. Currently, the buttons are programmed to alter a state machine within the code, beginning movement sequences.

4.3 Power/Display Board

The power circuit and display board components are on the same board. This power circuit receives 6V from the battery pack and outputs 3.3V to the rest of the boards, including the display components, such as the LCD Display.

4.4 LCD Display

A liquid-crystal display that presents phrases that are programmed onto the FRAM Board. A backlight is also attached to the display to make it easier to read.

4.5 Battery Pack

The battery pack holds four AA batteries. It is attached to a connector that plugs into the power/display board. The wires are twisted to improve the strain relief; if the wires are tugged hard, it will spread the strain across both wires leading to wire breakage.

4.6 Power Switch

The power switch is soldered onto the power/display board and controls the power circuit. It is used to turn the power from the battery to the boards on and off. To further test whether power from the battery is flowing through the necessary areas of the power/display board, a voltmeter/multimeter is used to check between two terminals to see if the required voltage is present allowing for smooth functioning of the system. Additionally, this allows for successful power circuit testing and verifying connections.

4.7 FET board/H-Board

The FET board is a circuit board containing two H-Bridge circuits. Each H-Bridge is a network of N-FETs that has two outputs. One output is to be positive and connected to the power supply, while the other is connected to ground. Which side is which is determined by supplying power to the N-FETs. This board helps the main processing board interface with the rest of the device, including the motors.

4.8 Motors and Wheels

A pair of 120:1 electric plastic motors from Pololu. They have a functional voltage of 3V-6V and amperage 130mA-1250mA with a nominal torque of 25 oz*in^3. Each motor is connected to a rubber wheel and powered by an H-Bridge.

The wheels are attached to the motors using a D-type socket connector. The wheels themselves can be various sizes; the larger the wheels, the faster the machine is. Wheels range from 60 millimeters to 32 millimeters in diameter.

4.9 IR Board

The IR board consists of one IR emitter and two IR phototransistors. Using the emitter, the IR board projects a narrow beam of infrared light downwards to the surface the car is currently on top of. The infrared light is absorbed by matte surfaces and reflected by glossy surfaces. If the phototransistors detect infrared light being bounced back, the board sends a signal to the FRAM, essentially acting as an IR camera. The IR board is connected to the FET board.

4.10 IR Emitter

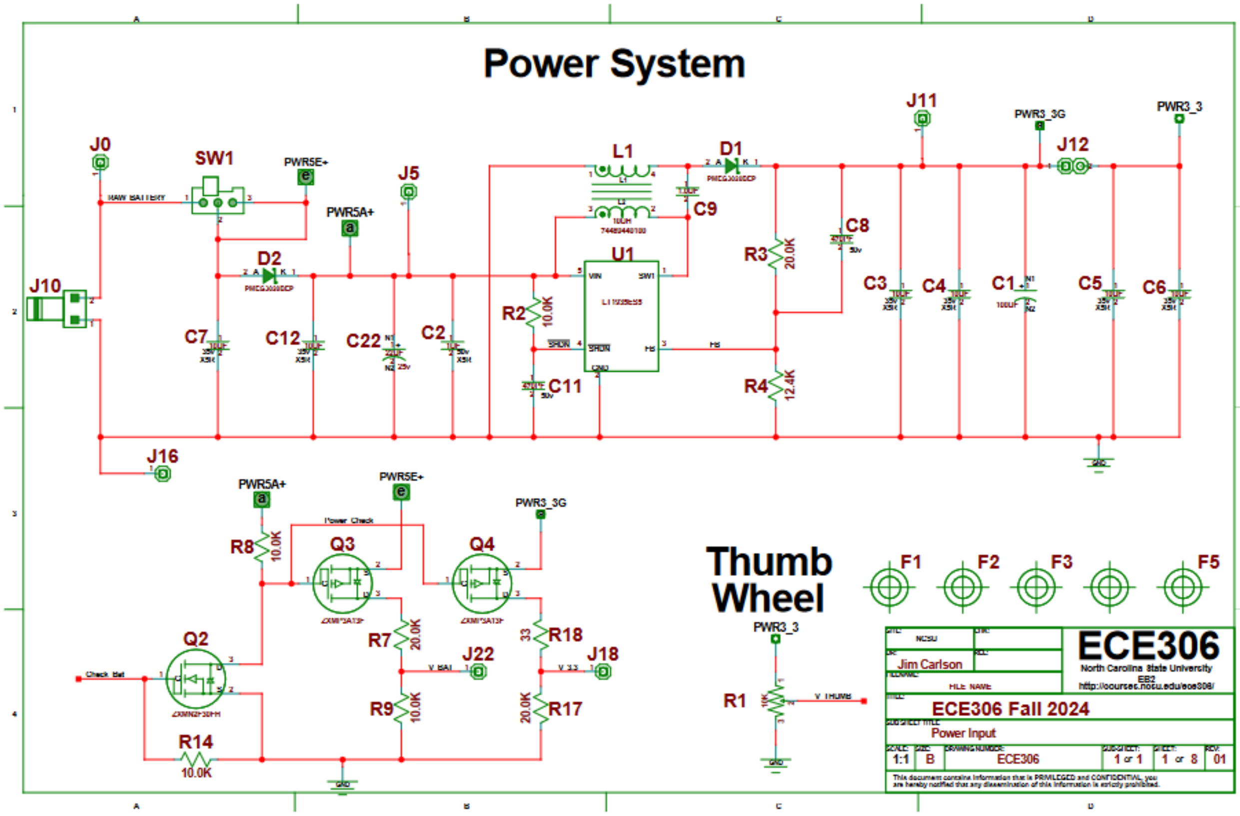
There is one IR emitter on the IR board. The IR emitter projects a narrow infrared wavelength downwards to the surface the car is on. It is placed between the two IR phototransistors. As the emissivity of electrical tape is high, the infrared given off by the emitter will bounce off of electrical tape.

4.11 IR Detectors

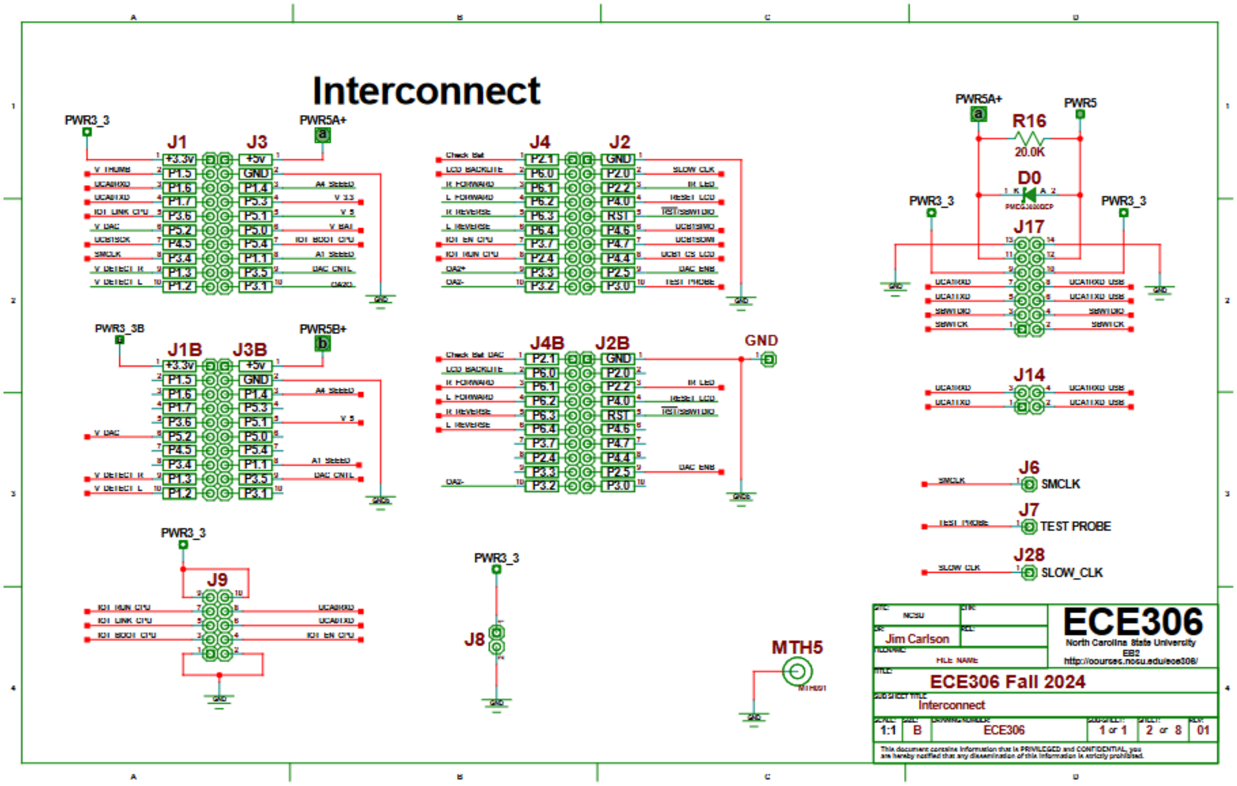
There are two IR phototransistors on the IR board. These phototransistors capture IR wavelengths, essentially acting as infrared detectors. The two IR phototransistors are placed on the left and right sides of the IR emitter, which helps discern if the black line is oriented to the left or right of the car.

4.12 Circuits

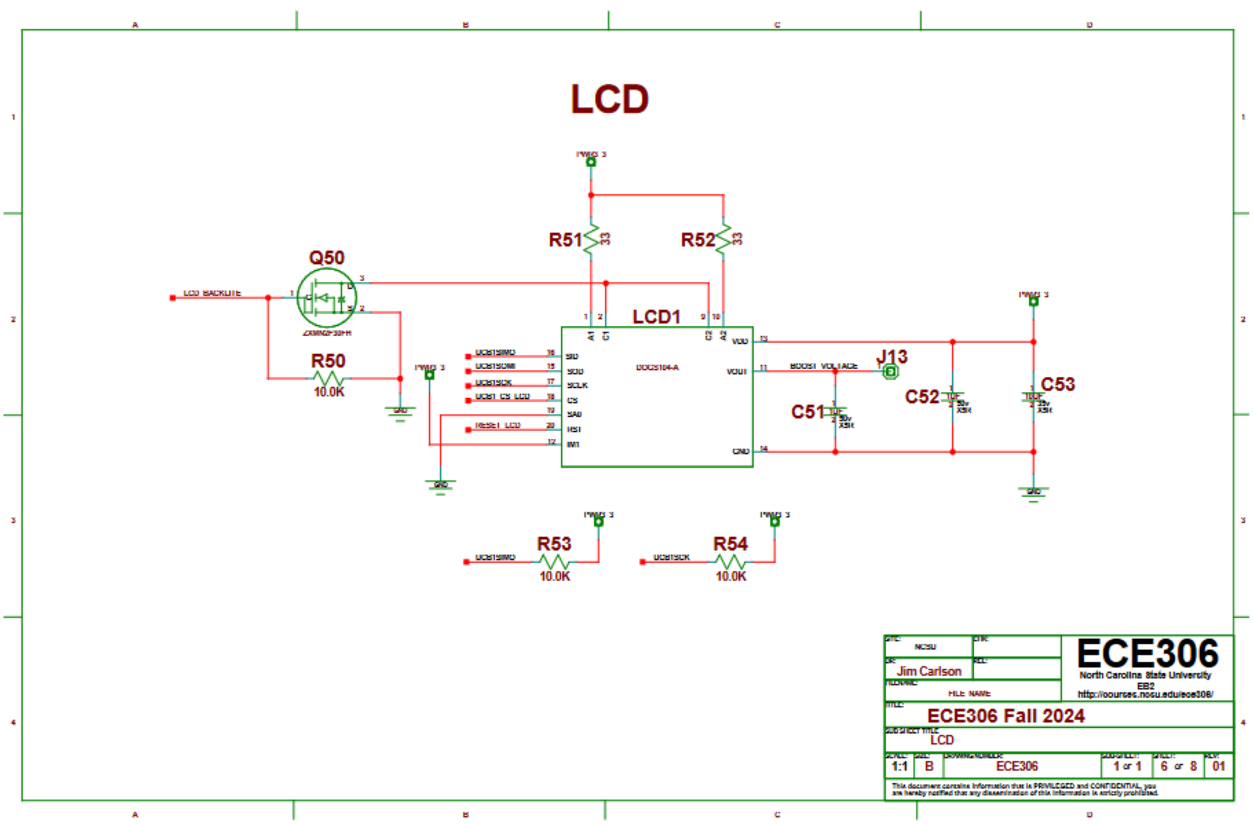
4.12.1 power system circuit



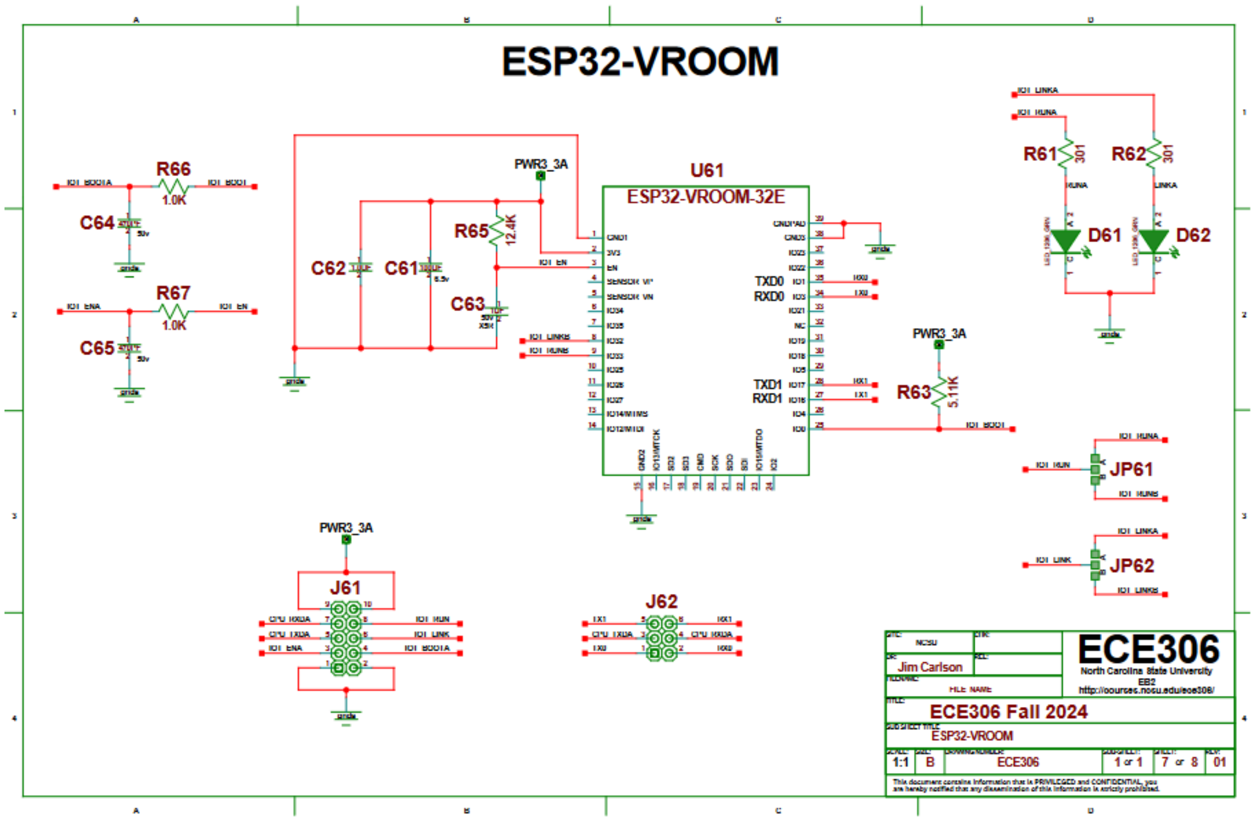
4.12.2 interconnect circuit



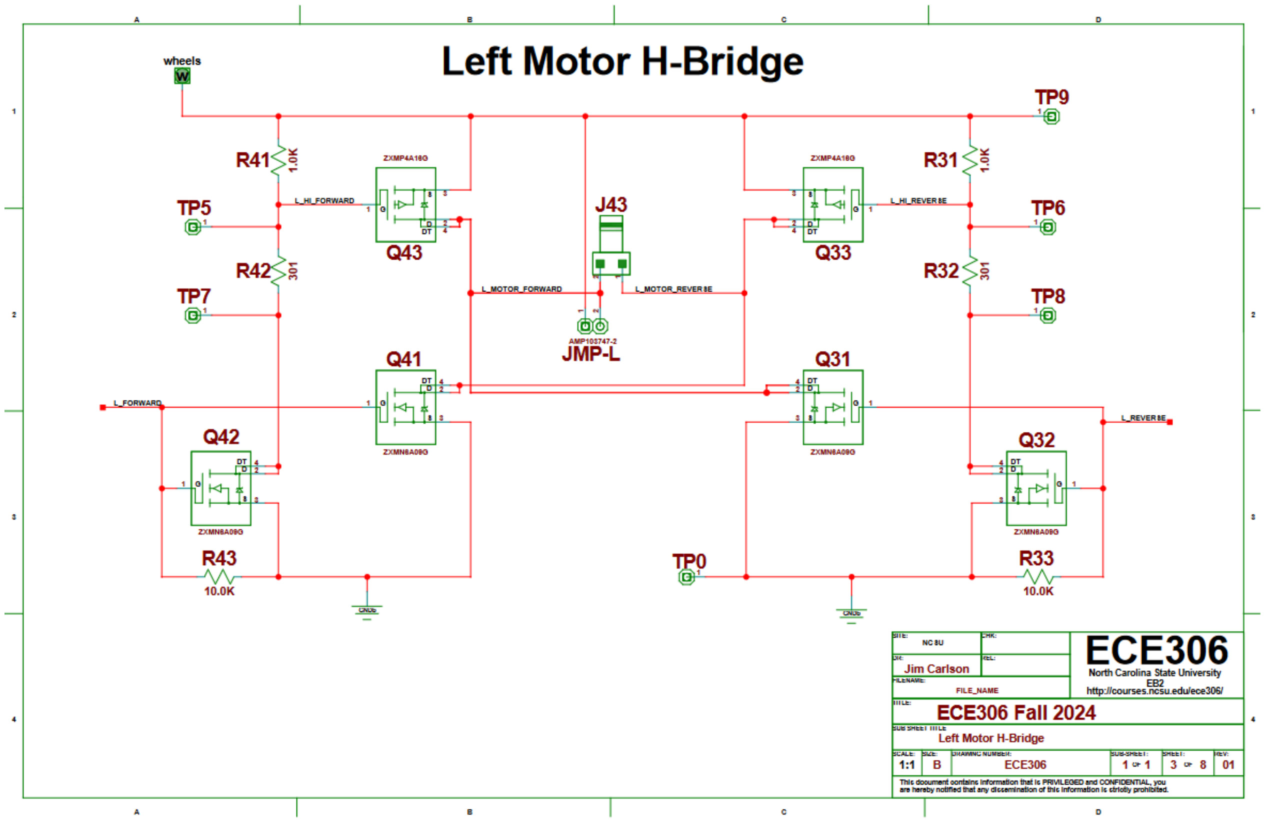
4.12.3 LCD screen circuit



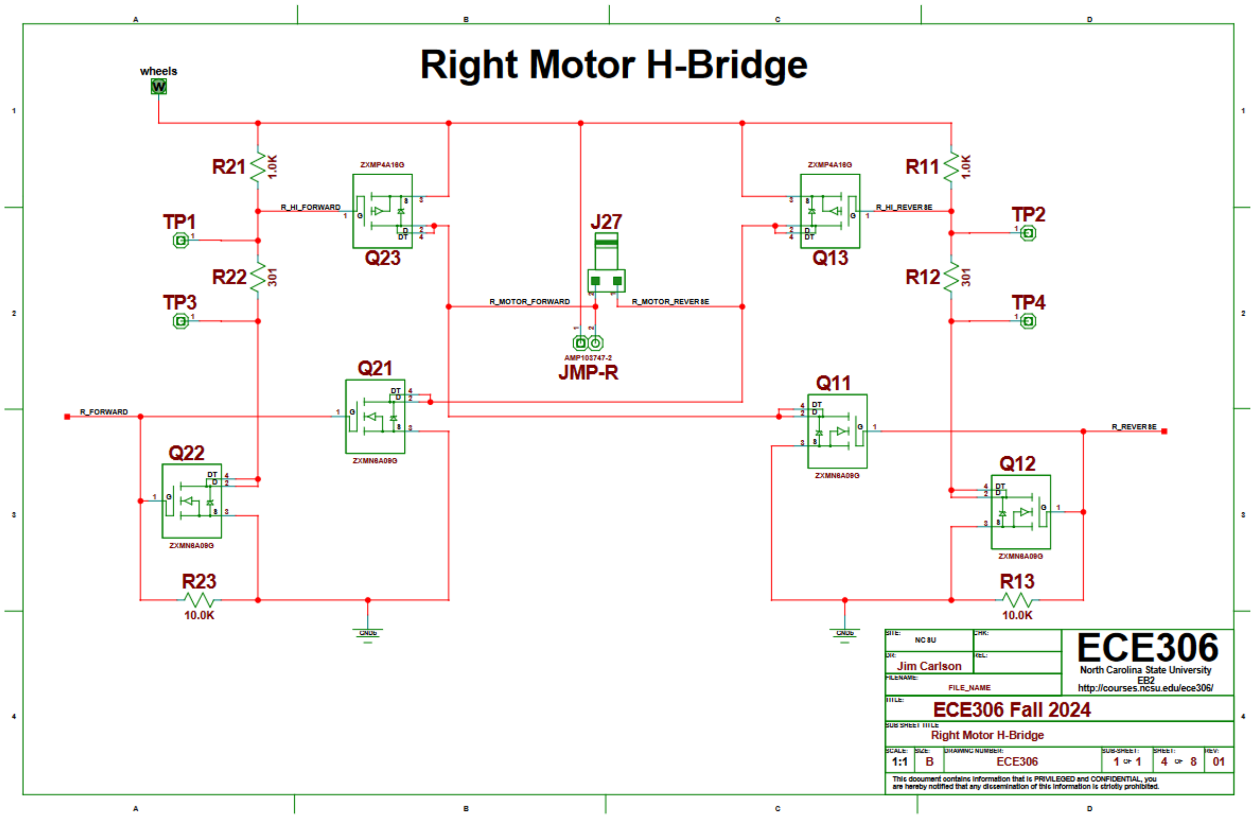
4.12.4 ESP32-VROOM circuit



4.12.5 Left Motor H-Bridge

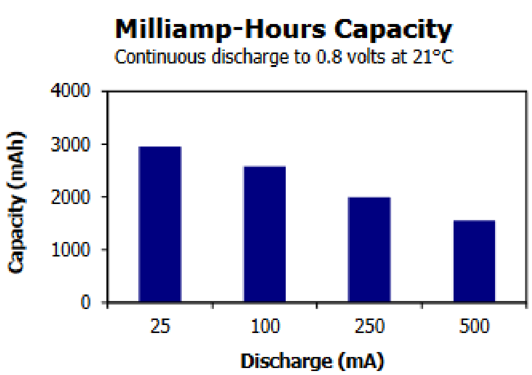


4.12.6 Right Motor H-Bridge

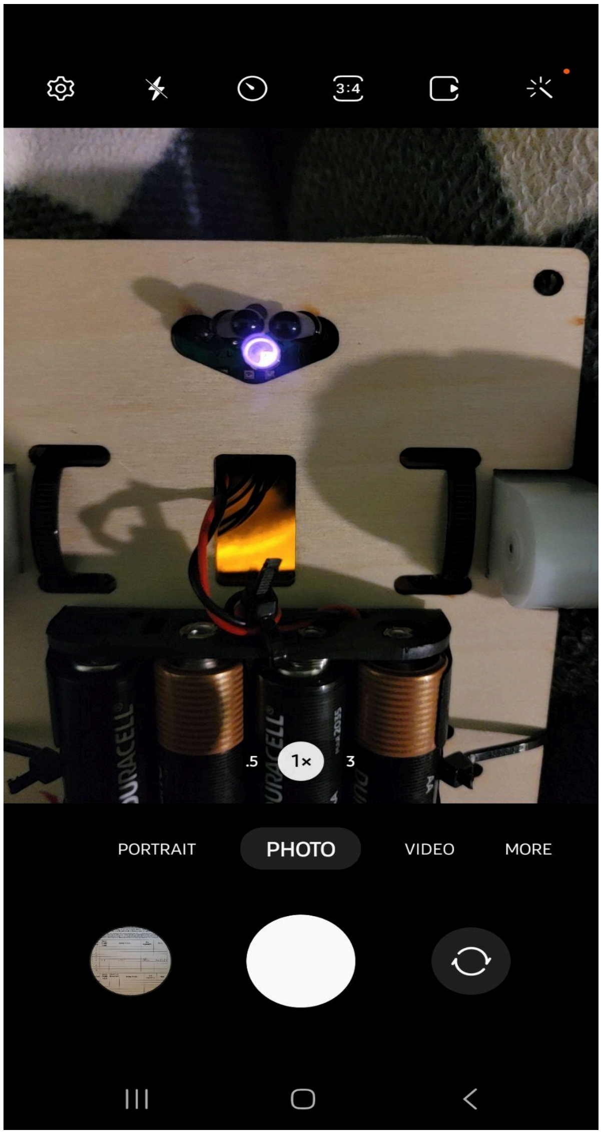


5. Power Analysis

The car operates at a supply voltage of 6V from 4 1.5V AA batteries regulated to 3.3V. When idle, the car draws a current of 0.08A and a total power drain of 0.48W. When in motion, the car has a current of 0.29A and a total power draw of 1.74W. Given the amp-hour capacity in the figure below, the car has an idling runtime of 36.25h and a moving runtime of 6.55h.



6. Test Process



To test the IR emitter and ensure it was installed correctly, we utilized the front-facing cameras of smartphones, which typically lack IR filters. This allowed us to visually confirm whether the IR light was functioning. By using our phone cameras, we could easily verify the operation of the IR emitters and ensure they were emitting the expected infrared signals.

7. Software

The product was programmed using TI Code Composer Studio IDE, version 21.6.1 LTS. Code composer runs in the C programming language. When testing and debugging the code, the program is connected to the eZ-FET, which checks for compatibility with the MSP430FR2355 and any critical errors in the code.

7.1 main.c

It is the beginning of the code. Calls initializations of all other files and functions. Uses state machines to control the behavior of the motors.

7.2 ports.c

Contains the functions to initialize the port pins to GPI/O or peripheral signals.

7.3 clocks.c

Contains the function for clock initialization for the program. It configures MCLK for 8MHz. It also contains timer-based interrupt functions and variable updates.

7.4 display.c

Contains the function that updates the display. This function checks for the update_display flag in order to update the display.

7.5 switches.c

Contains the functions for switches one and two. When a switch is pressed, it enables interrupt functions that control button-debouncing in switches.c and state machines in main.c. A second function determines the current action of the machine based on how many times the switch has been pressed.

7.6 wheels.c

Contains the functions that control the movement of the car, such as going straight, turning left, and turning right. The functions control the motors by modifying the ports pins. (Not used by all members)

7.7 state_machine.c

Contains the function for a state machine that is based off of the current action. The main states are wait, start, run, and end. (Not used by all members)

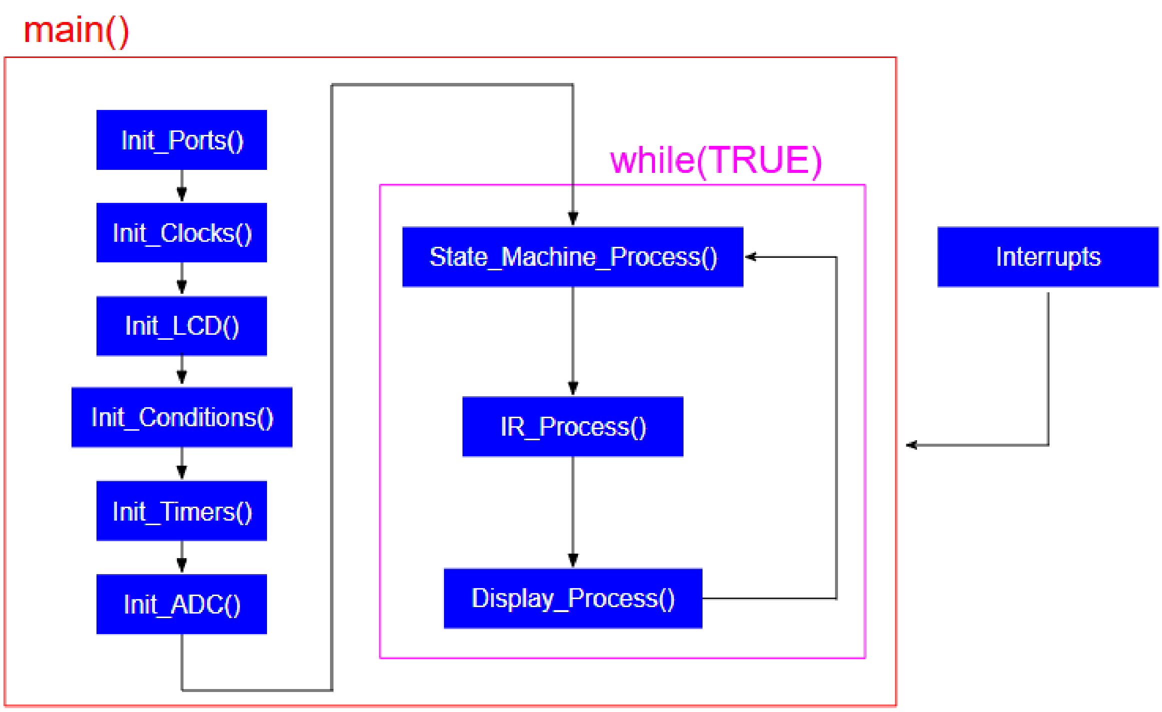
7.8 system.c

Enables interrupts across the program.

7.9 ADC.c

Enables the ADC interrupt vector and contains functions for converting the ADC to hexadecimal and for automatically inserting text into the LCD.

8. Code Flow



8.1 Main Block

8.1.1 Main()

The main block of the machine. First, it calls initialization functions, and then it runs an infinite “while” loop of system processes.

8.1.2 Init_Ports()

Initializes ports one through six, setting the peripherals needed for the machine.

8.1.3 Init_Clocks()

Initializes the clock system. This sets the main clock and sub-main clock to 8 MHz.

8.1.4 Init_LCD()

Initializes the LCD screen.

8.1.5 Init_Conditions()

Initializes variables for the LCD. Once these variables are initialized, they can be used to make the LCD display letters and numbers.

8.1.6 Init_Timers()

Initializes the variables for the clock. One of the variables initialized is Time_Sequence, which is used to time the machine’s actions.

8.1.7 State_Machine_Process()

The first function in the infinite “while” loop. This function checks for the current action of the machine. If the current action is to run, it will run until it finds a black line.

8.1.8 IR_Process()

The second function in the infinite “while” loop. This function utilizes the ADC in order to take readings, and changes the current state for State_Machine_Process().

8.1.9 Display_Process()

The third and last function in the infinite “while” loop. This function manages the LCD screen, making sure that it does not display garbage characters. After this function, the machine starts the loop over, running State_Machine_Process() again.

8.1.10 Interrupts

There are several interrupts used by the product's system. While most of the interrupts are timer-based, there are button interrupts as well. The timer interrupts are used to time ADC conversions and state machine functions, while the button interrupts are used to start the toy car's actions.

9. Software Listing

Blank for Project 6 Writeup

10. Conclusion

This product has evolved to include the integration of IR detectors and emitters, enhancing the car's ability to navigate its environment. By using an ADC to capture values and convert to hexadecimal format, the system can effectively distinguish between white and black surfaces.

The display now shows the ADC values alongside the detected colors, providing real-time feedback on the car's surroundings. When the vehicle detects a black line, it not only stops but also aligns itself with the line, demonstrating advanced control and precision in its movements.

Overall, this product enhances the educational experience by introducing young learners to concepts in robotics, programming, and sensor technology through hands-on interaction. By integrating IR detection with real-time feedback and responsive movement, this product provides an engaging way for children to grasp complex ideas such as line following and signal processing.