IUPUI- ECE59500 (PROJECT)
Embedded Autonomous Systems

SPEED CONTROL & COLLISION DETECTION OF AN ELECTRIC VEHICLE

Submitted By: (Group No:)

Annu James
Prasham Shah
Mugdha Nehete

Under the Guidance of
Dr. M. El-Sharkawy

Table of Contents

Abstract:

In this paper, we demonstrate collision detection and speed control of an autonomous electric vehicle. We are using LIDAR/Ultrasonic sensor to detect any obstacle and once detected send a warning to the driver and stop the vehicle to avoid any collision. The LIDAR/ultrasonic sensor is used to determine the distance of the obstacle which is then fed to the sensor microcontroller. Based on the distance a command is sent to the motor controller using UART protocol and the MCU reduces the speed of the autonomous electric vehicle. The motor used to demonstrate this feature in this study is a BLDC motor. For demonstration purpose we only use one motor to emulate one wheel of the autonomous electric vehicle. The same command from the sensor controller can be sent to the individual motor controllers to stop all the four wheels of the car for a four-wheel drive. In case of two-wheel drive, the commands are only sent to the two wheels in the front of the car.

Introduction:

There are lot of advancements happening in automotive industry and one of them is autonomous driving vehicle. This field is becoming very interesting and demanding in the recent years.
Autonomous driving vehicles bring enormous advantages over the conventional vehicle. To list few of them are as follows:

1. Autonomous vehicles can have interconnectivity on the road such that each car knows the position of other car in its surrounding hence minimizes the danger of collision.
2. Advanced features like, Adaptive cruise control, lane detection, obstacle detection etc. can add additional safety to the driving.
3. With intelligent software and hardware combination, the possibility of human error while driving can be reduced.

This list is just few of the examples and it keeps growing daily based on the intelligence that has been added to today's vehicles. Based on the degree of automation, the autonomous vehicles can be classified into five groups which are as follows

1. Level 0 : No Automation

   This group contains all the conventional cars without any automation features available. For the vehicles in this group while driving, driver is in full control.
2. Level 1 : Function Specific Automation

   This group contains all the conventional cars with some automation features available. For the vehicle in this group, one or more specific control function is automatic while driving.
3. Level 2: Combined Function Automation

   This group contains all the conventional cars with two or more automation features available. For the vehicle in this group, at least two primary control function is automatic while driving.
4. Level 3: Limited Self-Driving Automation:

   This group contains all the conventional cars with limited number of automation features available. For the vehicle in this group, control function of all safety-critical functions under certain traffic or environment condition are automatic while driving.
5. Level 4: Full Self-Driving Automation:

This group contains all the conventional cars with all automation features available. For the vehicles in this group while driving, car is in full control. This group contains all the conventional cars designed to perform all safety-critical driving functions and monitor roadway conditions for an entire trip.

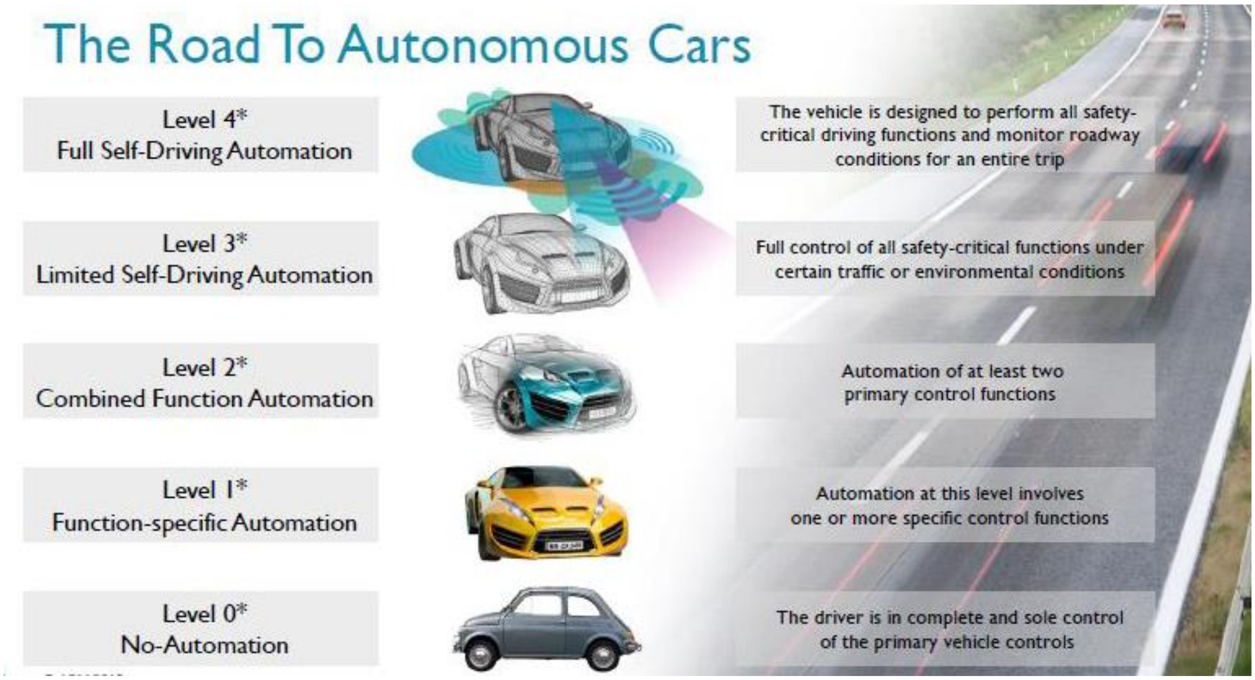

Figure 1: Transformation from Conventional vehicle to Full Self-Driving Automation

In this project we are demonstrating the Obstacle detection feature, which is part of adaptive cruise control and can be used in the vehicles which belongs to level 1 till level 4.

Project Objective:

The objective of this project is to develop an obstacle detection system which when detects an obstacle in front of a vehicle in motion reduces its speed and provide collision detection warning to the driver.

To demonstrate this feature, we have used LIDAR/ultrasonic sensor for sensing the distance of the obstacle. This distance is sent to the sensor controller which is FRDM- K64F. Based on this distance a command is sent over UART to the Motor controller which is S32K144. Based on the distance the speed of the electric vehicle motor is then reduced at different rates eventually making vehicle to stop. The motor control algorithm is a rule-based algorithm which is as follows:

1. Motor is running at the constant speed till the obstacle is detected.
2. When the obstacle is less than or equal to 2 meters the motor speed will reduce as per a defined rate and will in the end turns off avoiding the collision with the obstacle.

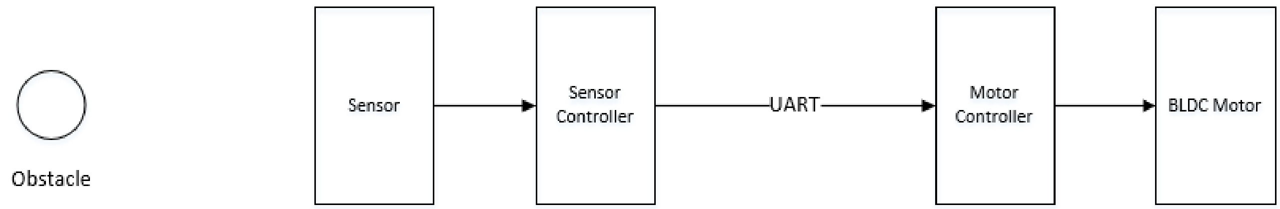

Figure 2: Overview of the project

Hardware description:

Garmin LidarliteV3:

This device measures the distance by calculating the time delay between the transmission of a Near-Infrared laser signal and its reception after reflecting off a target. This translates into distance using known speed of light. This device has a 2-wire, I2C-compatible serial interface. It is connect as an I2C slave device, under the control of an I2C master device(in this case FRDMK64F).

Item	Description	Notes
1	680µF electrolytic capacitor	You must observe the correct polarity when installing the capacitor.
2	Power ground (-) connection	Black wire
3	I2C SDA connection	Blue wire
4	I2C SCA connection	Green wire
5	5 Vdc power (+) connection	Red wire The sensor operates at 4.75 through 5.5 Vdc, with a max. of 6 Vdc.

Figure 3: GARMIN LIDARLITEV3

HC SR04:

The HC SR04 ultrasonic sensor uses SONAR to determine the distance to an object. It provides 2cm to 400cm non-contact measurement functions the ranging accuracy can reach to 3mm. The modules include ultrasonic transmitters, receiver and control circuit. The basic principle of work:

- Using IO trigger for at least 10us high level signal, the Module automatically sends eight 40 kHz and detect whether there is a pulse signal back.
- If the signal is back, through high level, time of high output IO duration is the time from sending ultrasonic to returning.

$$Test\ Distance = \frac{high\ level\ time\ (in\ sec) * velocity\ of\ sound\ i.e.\ 340\frac{m}{s}}{2}$$

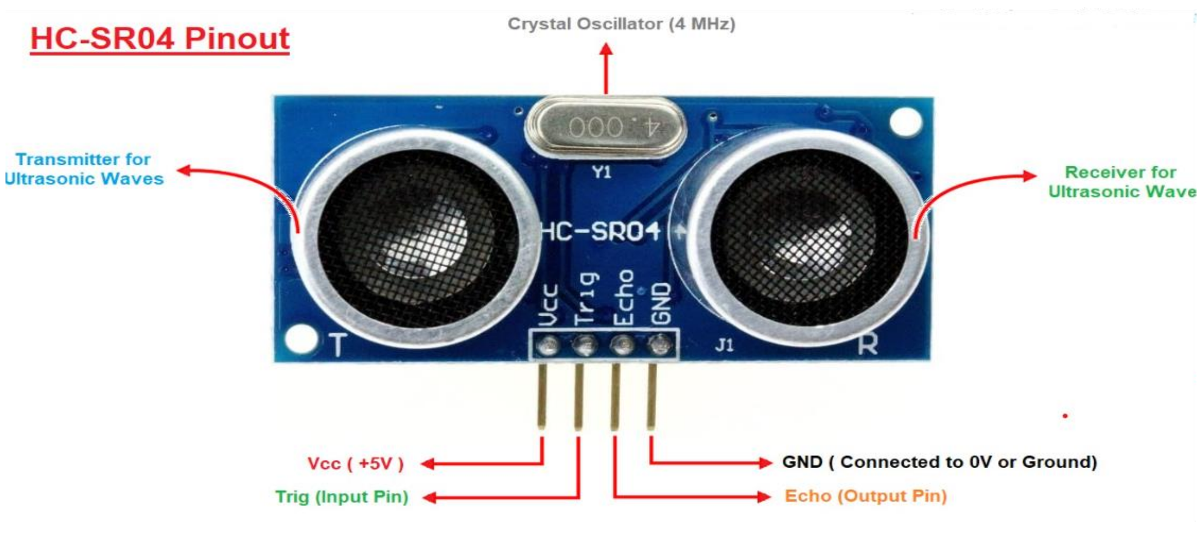

Figure 4: HC-SR04 Pinout [1]

Electric Parameter:

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**

FRDMK64F:

The Freescale Freedom K64 hardware is a simple, yet sophisticated design featuring a Kinetics K series microcontroller, built on the ARM® Cortex®-M4 core. It features the MK64FN1M0VLL12 MCU, which boast the maximum operation frequency of 120 MHz, 1 MB of flash, 256 KB RAM, a full-speed USB controller, Ethernet controller, secure digital host controller, and analog and digital peripherals. The onboard interface includes a six-axis digital accelerometer & magnetometer, RGB LED, SDHC, add-on Bluetooth module, add-on RF module, and Ethernet. The FRDM-K64F platform features OpenSDAv2, the Freescale open-source hardware embedded serial and debug adapter running an open-source bootloader. This circuit offers several options for serial communication, flash programming, and run-control debugging. OpenSDAv2 is an mbed™ HDK-compatible debug interface preloaded with the open-source CMSIS-DAP Interface firmware (mbed interface) for rapid prototyping and product development, with a focus on connected Internet of Things devices.

FRDM-K64F hardware overview

The features of the FRDM-K64F hardware are as follows:

- MK64FN1M0VLL12 MCU (120 MHz, 1 MB flash memory, 256 KB RAM, low-power, crystalless USB, and 100 LQFP)
- Dual role USB interface with micro-B USB connector
- RGB LED
- FXOS8700CQ – accelerometer and magnetometer
- Two user push buttons
- Flexible power supply option – OpenSDAv2 USB, K64 USB, and external source
- Easy access to MCU input/output through Arduino R3TM compatible I/O connectors
- Programmable OpenSDAv2 debug circuit supporting the CMSIS-DAP Interface software that provides:
    - Mass storage device (MSD) flash programming interface
    - CMSIS-DAP debug interface over a driver-less USB HID connection providing run control debugging and compatibility with IDE tools

- Virtual serial port interface o Open-source CMSIS-DAP software project: github.com/mbed micro/CMSIS-DAP.

- Ethernet
- SDHC
- Add-on RF module: nRF24L01+ Nordic 2.4GHz Radio
- Add-on Bluetooth module: JY-MCU BT board V1.05 BT

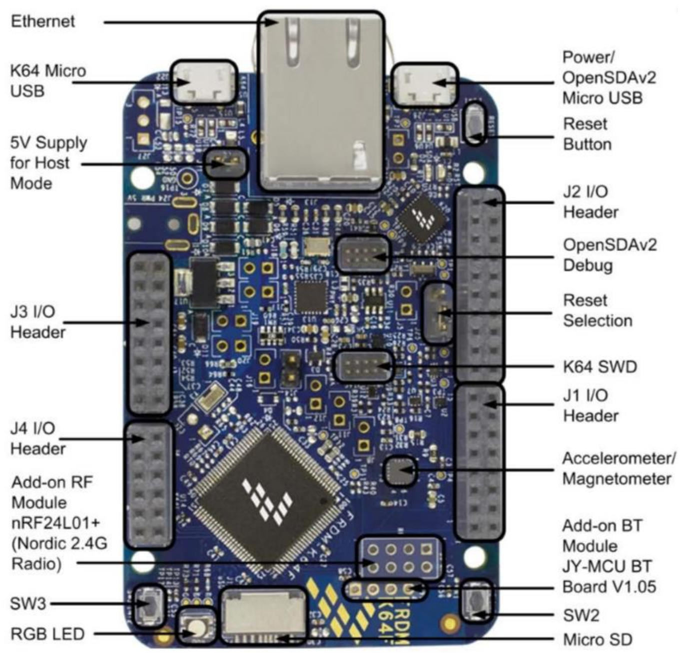

Figure 5: FRDM-K64F main components placement [2]

S32K144:

NXP's S32K144EVB evaluation board provides a low-cost platform tor prototyping and demonstration for the S32K144 AEC-Q100 qualified MCU family. The board allows flexible power supply options as well as easy header pin access to the ARM Cortex-M4F, 512 KB Flash MCUs. All family members are designed in accordance with the ISO 26262 standard, AEC-Q100 qualified at up to 125°C, and guaranteed by NXP’s product longevity program which assures supply for a minimum of 15 years.
Features:

- Small form factor
- On-board connectivity for CAN, LIN, UART / SCI
- Arduino™ UNO footprint-compatible expansion "shield" support
- Integrating an SBC (UJA1169) and LIN phy (TJA1027)
- Easy access to the MCU I/O header pins for prototyping
- Potentiometer, RGB LED, two touch pads, and 2x push buttons

- Integrated open-standard serial and debug adapter (OpenSDA) with support for several industry-standard debug interfaces
- Flexible power supply options (micro USB or external 12 V supply)
- High performance ARM Cortex-M4F up to 112 MHz, IEEE-754 floating point unit (FPU)
- Software friendly architecture
    - 512 KB Flash and 64 KB RAM with ECC on both, 4 KB cache, 4 KB EEPROM (Flex Memory)
    - Independent CPU and peripheral clocking
    - External 8 MHz to 40 MHz crystal oscillator or resonator
    - Up to DC- 60 MHz external square wave input clock
    - Internal clock references
- Connectivity and communications
    - 3x ISO 11898-7 compliant Flex CAN controllers. One controller with flexible data (FD) rate support at 2 Mb/s in normal mode and 8 Mb/s in programming mode with 64 bytes message payload. Suitable for dedicated CAN-FD networks or mixed CAN/CAN-FD networks
- Cryptographic services engine compact (CSEc) hardware security
    - Dedicated security co-processor
    - SHE spec. compliant
    - Secure key storage, AES-128 encryption and decryption, cipher-based message authentication code (CMAC) algorithms, secure boot functions
- Low-power consumption
    - Low leakage 90 nm TFS technology
    - Multiple low power operating modes and IRC combinations - RUN, very low Power RUN (VLPR), STOP, very low power stop (VLPS), current consumption from 29 µA at 25°C in VLPS with a recovery time of 5 µs. All register contents maintained in all power modes
- Analogue and timing
    - 2 x 16-ch 12-bit ADCs at 1 Msps, analogue comparator
    - 4 x 8-ch 16-bit flex timers - IC / OC / PWM modes, fault detection, dead-time insertion, quadrature decoder
    - Low power timer, real time clock, programmable delay block for analogue/timer module triggering
- Voltage range: 2.7 V to 5.5 V

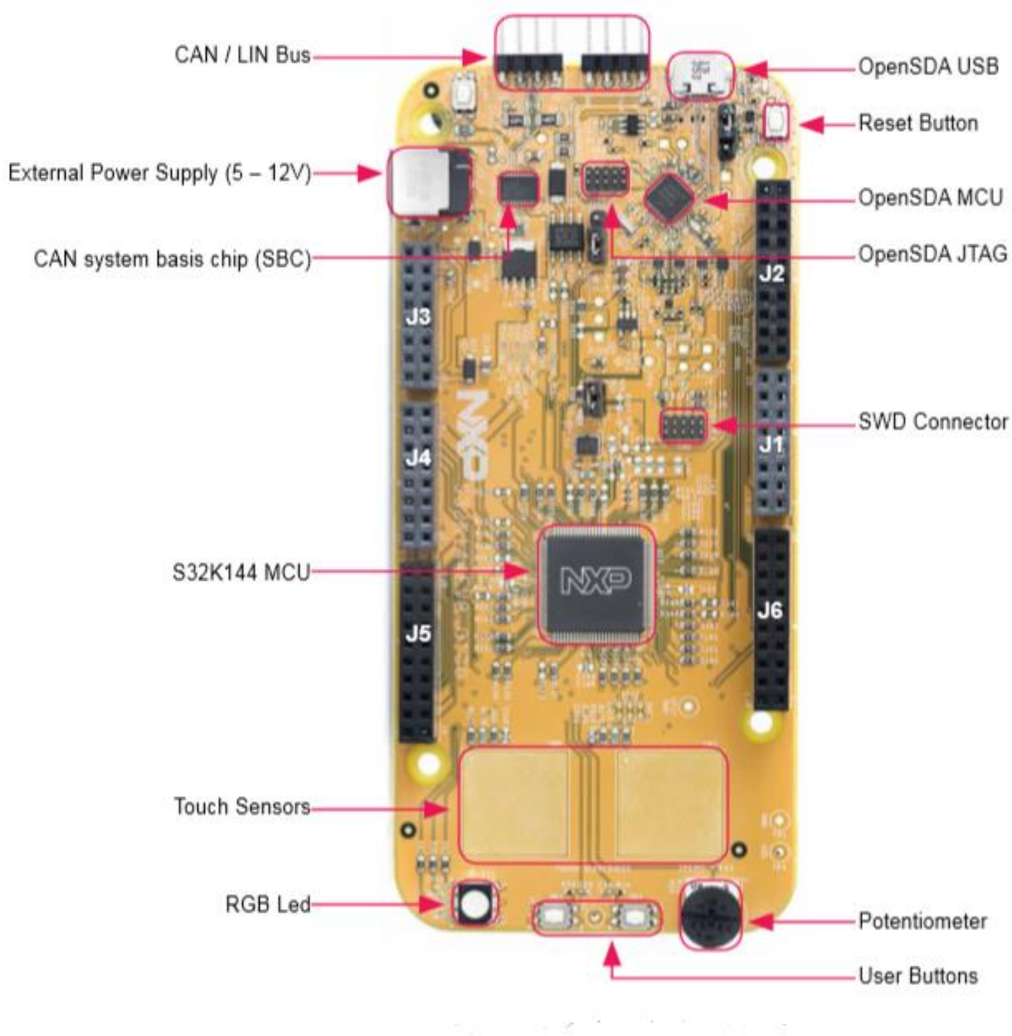

Figure 6: S32K144 [3]

DEVKIT-MOTORGD:

The DEVKIT-MOTORGD is an ultra-low-cost development platform motor control. Features include easy access to all base board I/O's and a standard based form factor compatible with the Arduino™ pin layout, providing a broad range of expansion board options. The shield can be powered by external supply from 10V to 18V.

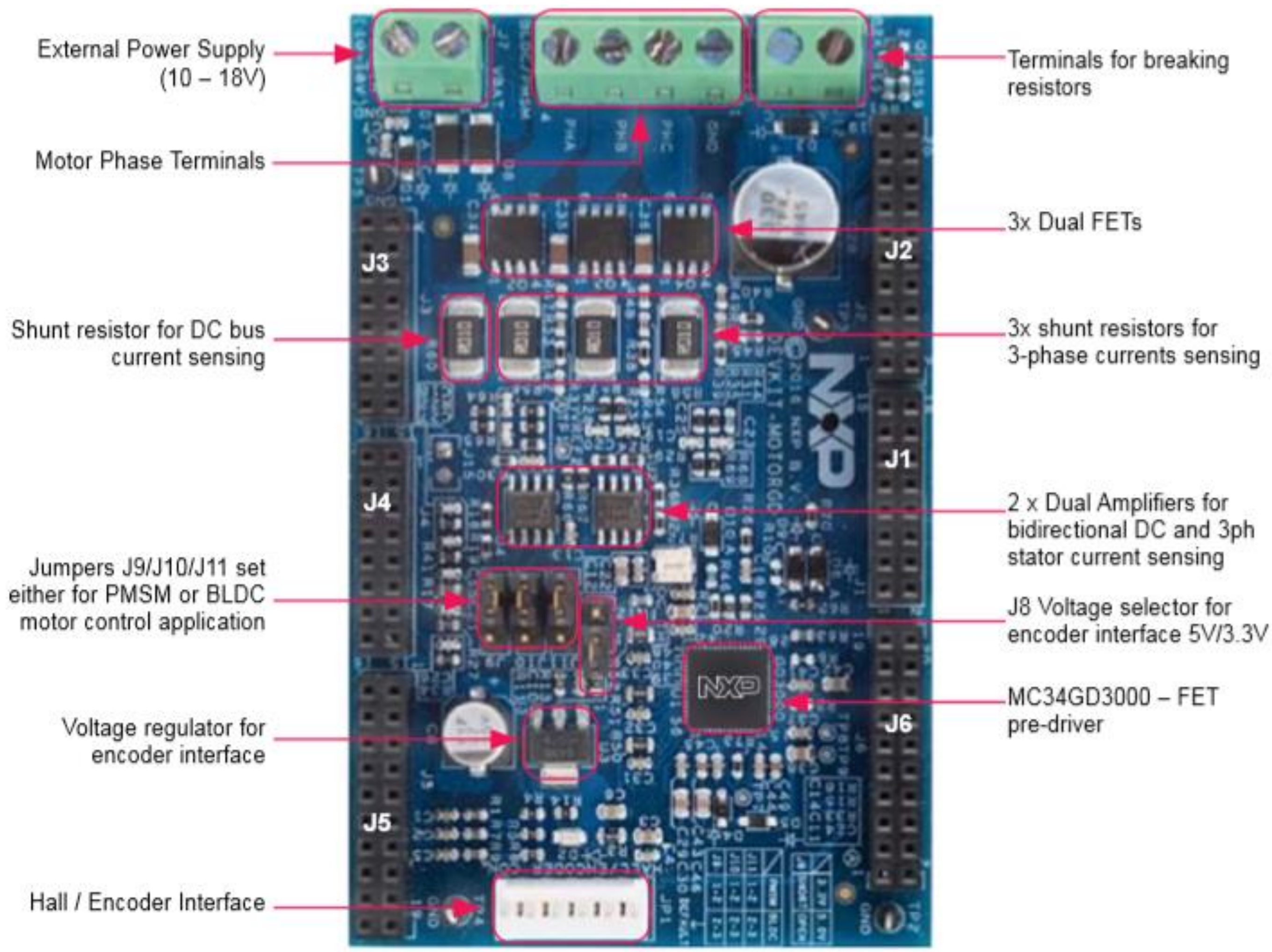

Figure 7: DEVKIT-MOTORGD

S32K144 with DEVKIT-MOTORGD:

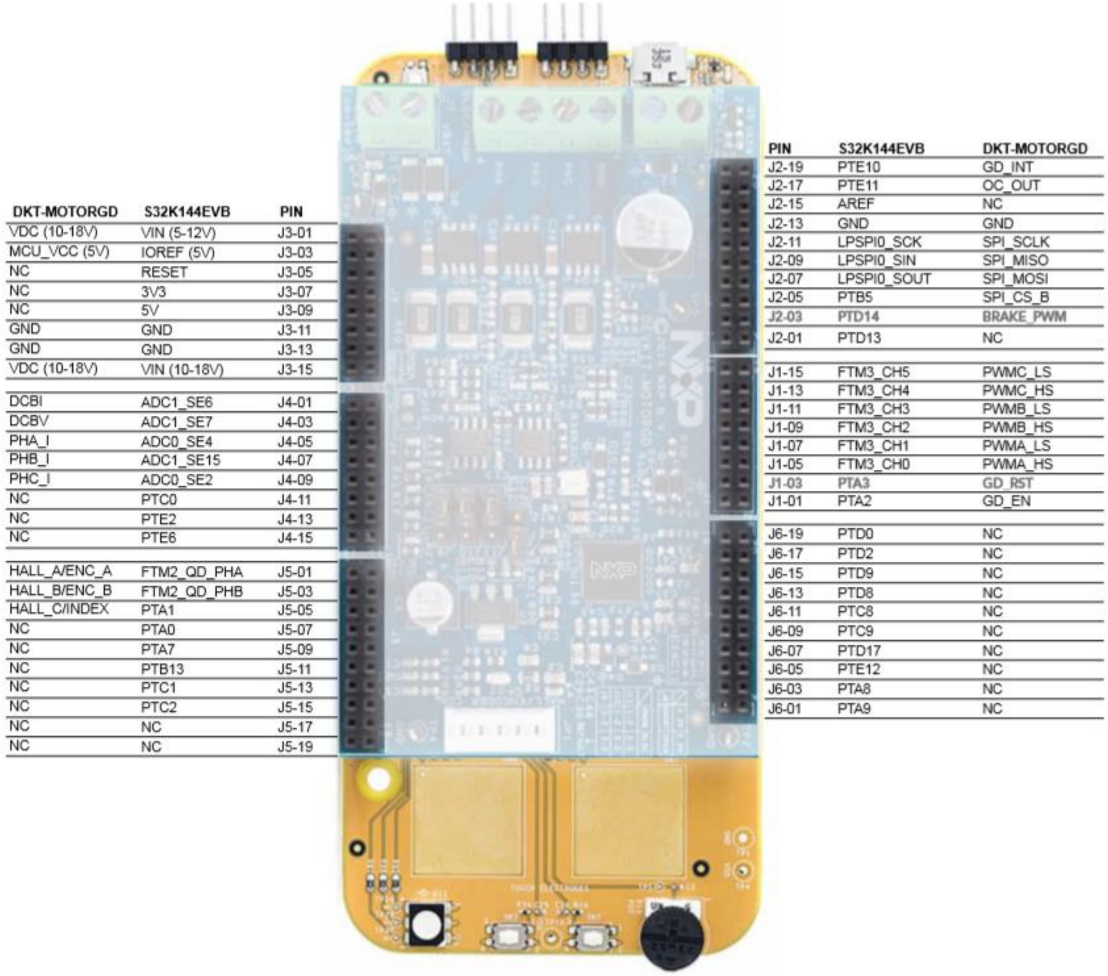

Figure 8: S32K144 with DEVKIT-MOTORGD [4]

BLDC motor:

In this paper, we will be controlling the speed of an electric vehicle by showing a controlling of an electric motor. Electric motor comes in two types: Brushed and Brushless. For this project, we will be using Brushless DC motor because brushless dc motors are more advantageous than the brushed dc motor in terms of maintenance requirement and life expectancy. BLDC motors are referred to as brushless permanent magnet, permanent ac motor, permeant magnet synchronous motor, etc. BLDC motor also known as electronically commutated motors , are synchronous motors powered by DC electricity via an inverter or switching power supply which produces an AC electric current to drive each phase of the motor via a closed loop controller. The controller provides pulses of current to the motor windings that control the speed and torque of the motor. Trapezoidal control is one type of commutation method used to turn a motor where only two phase windings will conduct current at any one time. With direction also to consider, that leaves six possible patterns. Hall sensors detect the rotor positing returning a digital or analog signal. The information is used to move the stator magnetic field in the next position.

Figure 9: BLDC motor [5]

UART Communication:

A universal asynchronous receiver and transmitter is a computer hardware device for asynchronous serial communication in which the data format and transmission speeds are configurable. The electric signaling levels and methods are handled by a driver circuit external to the UART. The universal asynchronous receiver-transmitter (UART) takes bytes of data and transmits the individual bits in a sequential fashion. At the destination, a second UART re-assembles the bits into complete bytes. Each UART contains a shift register, which is the fundamental method of conversion between serial and parallel forms. Serial transmission of digital information (bits) through a single wire or other medium is less costly than parallel transmission through multiple wires. The data frame is as given below:

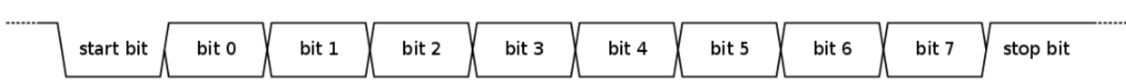

Figure 10: Data Frame[6]

Software description:

MATLAB- Motor Based Development Toolbox:

The NXP's Model-Based Design Toolbox provides an integrated development environment and toolchain for configuring and generating all of the necessary software automatically (including initialization routines and device drivers) to execute complex applications (e.g.: motor control algorithms, communication protocols CAN, SPI, I2C, UART, and sensor-based applications) on NXP MCUs. The toolbox includes integrated SIMULINK embedded target for NXP MCUs, peripheral device blocks and drivers, the Math and Motor Control library set and bit-accurate simulation results and provides built-in support for Software and Processor-in-the-Loop (SIL and PIL) simulations, which enables fast prototyping, verification and validation on real targets for the algorithms developed in MATLAB environment. [7]

MBED:

Mbed is a platform and operating system for internet-connected devices based on 32-bit ARM Cortex-M microcontrollers. Such devices are also known as Internet of Things devices. The project is collaboratively developed by Arm and its technology partners. Applications for the Mbed platform can be developed using the Mbed online IDE, a free online code editor and compiler. Only a web browser needs to be installed on the local PC, since a project is compiled on the cloud, i.e. on a remote server, using the ARMCC C/C++ compiler. The Mbed IDE provides private workspaces with ability to import, export, and share code with distributed Mercurial version control, and it can be used also for code documentation generation. Applications can be developed also with other development environments such as Keil µVision, IAR Embedded Workbench, and Eclipse with GCC ARM Embedded tools. [8]

Tera Term:

Tera Term is an open source, free, software implemented, communication program. In our project Tera Term is used to view the distance between the object and the sensor.

Methodology:

BLOCK DIAGRAM:

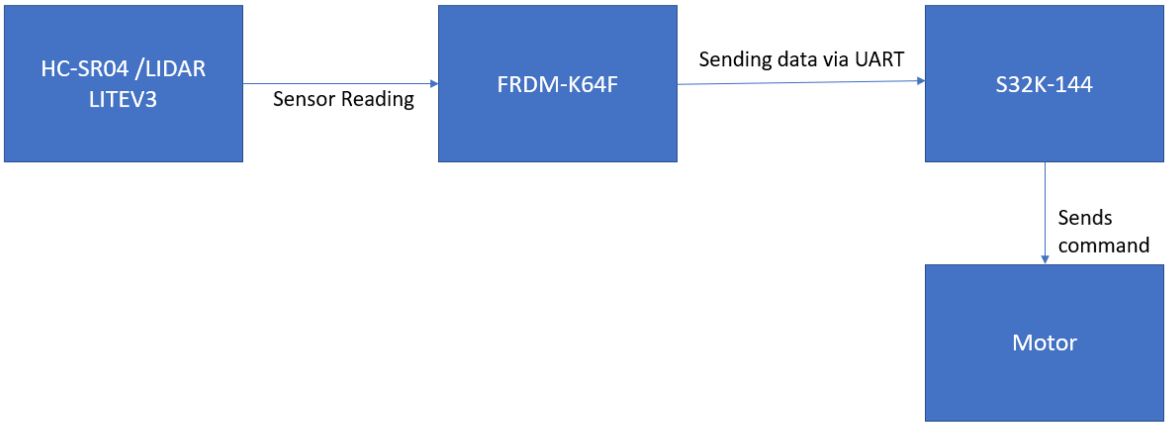

Figure 11: Hardware Block Diagram

Block Diagram Description:

The LIDAR/HC-SR04 (Ultrasonic sensor) is used to get the range between the object and the sensor. This range is then given to the FRDM-K64F. FRDM-K64F calculates the distance in cm and is transmitted to S32K-144 via UART. The S32K-144 checks the distance and controls the speed of the motor. When the distance between a object and the sensor is less than 30cm the motor stops and when its greater than 30 cm the motor rotates.

Hardware and software connections and Procedure:

Step 1:

The LIDAR/HC-SR04 is connected to the FRDM-K64F board by giving the below connections,

LIDAR:

LIDAR	FRDM-K64F
Power	5V
Gnd	Gnd
SCL	D15
SDA	D14

HC-SR04:

HC-SR04	FRDM-K64F
Power	5V
Gnd	Gnd
Trigger	D8
Echo	D12

Step 2:

A code (attached in the appendix below) is written and compiled using Mbed compiler to get the sensor values in cm and to transmit these values to the S32K144 board via UART. The code is then burned onto the FRDM-K64F board. The distance in cm can be viewed using Tera Term.

Step 3:
The UART connections are made between the FRDM-K64F and S32K144 as given below:

FRDM-K64F	S32K144
D1- Transmitter	PTC2 – Receiver
D0 – Receiver	PTC3 – Transmitter

Step 4:
A Simulink model (attached in the appendix below) is created using MBDT toolbox, which receives the distance via UART and controls the speed of the motor. This code is built and dumped onto the S32K144 board. Once the code is built, we can see that when the distance between the object and sensor is less than 30cm, the motor slows down and once the distance between the motor and the sensor becomes greater than 30cm, the speed of the motor starts increasing. The speed of the motor is controlled by increasing or decreasing the duty cycle of the Pulse width modulation which is given to the phase coils inside the BLDC motor. The hall sensors inside the motor are used to determine the position of the permanent magnet (inside the motor) and two of the 3 phases are charged at a time.
Connections are as given in the image below:

Figure 12: Hardware Connection of LIDAR LITEV3

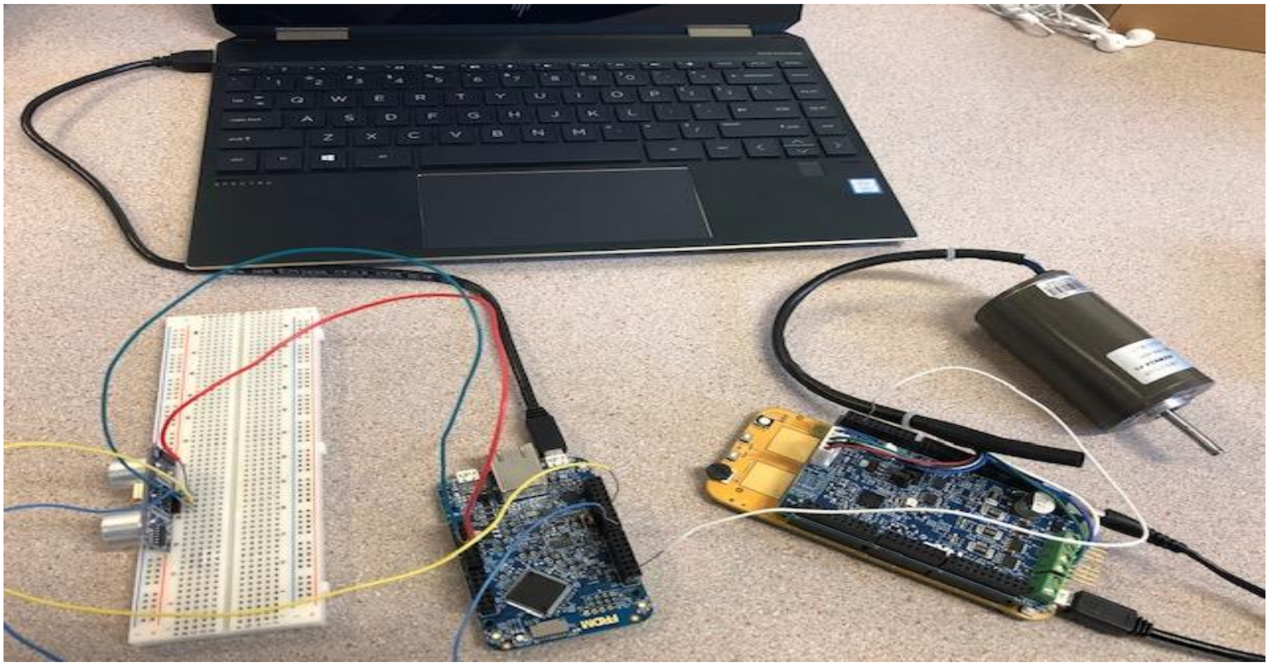

Figure 13: Hardware Connection of HC-SR04

Result

Future Work

The prototype can be adapted for several other appliances and expanded as needed. We can also add a warning audio with the LED warning. Also once the collision is detected we can tighten the seat belt, adjust the seat and head rest to minimize injuries during collision. Sensors like LIDAR, RADAR, Temperature and Humidity sensor, Rain sensor, Pressure Sensor can be added to this prototype. It can also be integrated with cloud. The data collected by these sensors can be stored and live monitored by sending it to a cloud which can be accessed by smart devices. Furthermore, additional functionalities can be added to this prototype. The driver can be notified if an obstacle or a blind spot is detected. The RPM of the motor and its braking time can be controlled in accordance with the load.

Conclusion

This prototype was successfully able to collect data from the ultrasonic sensor/LIDAR and determine the distance and was able to control the motor on basis of those distance values. Also enabled the led to change color as a warning when the object is closer to the sensor.

Appendix:

Lidar.txt ultrasonic.txt EC595_Project1_s32k14.slx

References:

[1] https://cdn.sparkfun.com/datasheets/Sensors/Proximity/HCSR04.pdf
[2] https://os.mbed.com/media/uploads/GregC/frdm-k64f_ug_rev0.1.pdf
[3] https://www.digikey.com/en/product-highlight/n/nxp-semi/s32k144evb-eval-board-for-s32k1-mcu-portfolio
[4] https://www.nxp.com/docs/en/quick-reference-guide/MTRDEVKSPNK144QSG.pdf
[5] https://en.wikipedia.org/wiki/Brushless_DC_electric_motor
[6] https://en.wikipedia.org/wiki/Universal_asynchronous_receiver-transmitter
[7]https://www.nxp.com/support/developer-resources/run-time-software/automotive-software-and-tools/model-based-design-toolbox:MC_TOOLBOX
[8] https://en.wikipedia.org/wiki/Mbed