LIDAR BASED MOTOR CONTROL

Akash Sunil Gaikwad (2000116696)
Surya Kollazhi Manghat (2000257775)
Sarada Gajjala (0003453412)

ECE-59500 – Embedded Autonomous Systems
Electrical and Computer Engineering

Indiana University – Purdue University Indianapolis.
420 University Blvd, Indianapolis, IN 46202
Year: Spring 2018

PROJECT REPORT

Lidar Based Motor Control

BY
Akash Sunil Gaikwad,
Surya Kollazhi Manghat,
Sarada Gajjala

Indiana University – Purdue University Indianapolis.
Year: Spring 2018

DECLARATION

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Signature:

Name: Akash Sunil Gaikwad (2000116696)

Signature:

Name: Surya Kollazhi Manghat (2000257775)

Signature:

Name: Sarada Gajjala (0003453412)

DATE: 03/27/2018

ACKNOWLEDGEMENT

We would like to express our gratitude and appreciation to all those who gave us the possibility to complete this report. A special thanks to our Project Guide, Professor Mohamed El-Sharkawy, whose help, stimulating suggestions and encouragement, helped us to coordinate our project especially in writing this report. given us his full effort in guiding the team in achieving the goal as well as his encouragement to maintain our progress in track.

We would also like to acknowledge with much appreciation of the crucial role of the University library and University Labs, which gave the permission to use all required resources to complete our project.

CONTENTS

ABSTRACT

The main goal of this project is to build a system which can control the rpm of the BLDC motor according to the distance of obstacle present. This project uses two S32K144 Evaluation boards, the communication between these boards is done by CAN protocol. In the scope of this project the distance of the object placed in front of the sensor is calculated using Garmin LidarliteV3 sensor and is updated to one of the S32K144 EVB boards. Using the CAN protocol that is established between two S32K144 boards the data is transferred to the other board which has BLDC motor connected to it. Using the Closed Loop Motor control application with Hall sensors, we controlled the RPM of the motor according to the distance of the obstacle calculated by the sensor. The future work on this project involves using of 3D Lidar instead of using LidarliteV3, controlling four motors to avoid obstacle, casualty situation and integrating multi sensor fusion techniques to improve the detection accuracy.

CHAPTER 1

INTRODUCTION

The main aim of this project is to build a system that can detect presence of objects in the field of the system and calculate the distance of that object using Lidar sensor and sends the control signal to the motor to reduce its speed according to the distance range set to the system. This system is built using two S32K144 Evaluation boards, GARMIN LidarliteV3 sensor, Linix 45ZWN24-40 Motor and FRDMK64 Evaluation board. The communication between the two S32K144 EVB's is implemented using FlexCAN protocol. The LidarliteV3 sensor detects the object and gives the distance of the object from the sensor. This sensor is interfaced with FRDMK64 board and the output of the Lidar data is transferred to one of the S32K144 EVB's via UART protocol. Now the data is transferred from one S32K144 to the other board which has BLDC motor on it using CAN protocol. We have used Model Based Design Toolbox to design the system and FreeMASTER to view the Lidar output that is the distance of the object calculated by the sensor. We have used closed loop motor control application with 3 Hall sensors to control the RPM of the motor. Using the Lidar data the motor RPM is controlled on the slave S32K144 using BLDC closed loop method. If the distance of the object is nearer to the field of the system then the motor will be reduced to run on minimum speed and if there is no object present or distance of object is greater than a limit, then the motor will allowed to run on its maximum speed.

CHAPTER 2

REQUIREMENTS FOR THE PROJECT

2.1. Hardware

1. S32K144 Evaluation board

Integrated open-standard serial and debug adapter (OpenSDA) with support for several industry-standard debug interfaces

- Small form factor size supports up to 6” x 4”
- Arduino™ UNO footprint-compatible with expansion “shield” support
- Integrated open-standard serial and debug adapter (OpenSDA) with support for several industry-standard debug interfaces
- Easy access to the MCU I/O header pins for prototyping
- On-chip connectivity for CAN, LIN, UART/SCI.
- Flexible power supply options

1. MicroUSB or
2. External 12V power supply

S32K144 Evaluation Board

2. GARMIN LidarliteV3

Lidar is a surveying method that measures distance to a target by illuminating the target with pulsed laser light and measuring the reflected pulses with a sensor. Lidar uses ultraviolet, visible, or near infrared light to image objects. It can target a wide range of materials, including non-metallic objects, rocks, rain, chemical compounds, aerosols, clouds and even single molecules. A narrow laser beam can map physical features with very high resolutions.

GARMIN LidarliteV3

GARMIN LidarliteV3 hardware description:

- Range (70% reflective target) : 40m(131 ft.)
- User Interface : I2C Interface
- I2C Interface : Fast-mode(400 Kbit/s)
- Power : 5Vdc nominal(4.5 Vdc min.,5.5 Vdc max)
- Size : 20X48X40 mm
- Resolution : +/- 1 cm(.4 in)

3. Linix 45ZWN24-40 Motor:

BLDC motors are very popular in a wide array of applications. Compared to a DC motor, the BLDC motor uses an electric commutation, replacing the mechanical commutator and making it more reliable than the DC motor. In BLDC motors, rotor magnets generate the rotor's magnetic flux, allowing BLDC motors to achieve higher efficiency. Therefore, BLDC motors may be used in high-end white goods such as refrigerators, washing machines, dishwashers, etc., high-end pumps, fans, and other appliances that require high reliability and efficiency.

Motor GD DevKit

LINIX MOTOR

Linix 45ZWN24-40 Motor Hardware Description:

- Commutation: Electronic commutation based on Hall position sensors (Closed loop control).
- Rated Voltage (V):24 V
- Rated Speed @V :4000 RPM
- Rated Power:40 W
- Longer life and flat speed/torque.
- Maintenance: Less required due to absence of brushless.

2.2. Software

1. Model Based Design Toolbox (MBDT):

The NXP's Model-Based Design Toolbox provides an integrated development environment and tool chain 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. The overview of Model Based Design Toolbox is as follows:

- MATLAB
- SIMULINK/STATEFLOW
- EMBEDDED CODER
- MOTOR CONTROL (MC) TOOLBOX

  Embedded Coder is production code generation environment from MATLAB/Simulink models basic code generator that generates ANSI C source code.

  Simulink is graphical environment for building controls algorithms as well as simulation of these algorithms.

  Stateflow is a special case of Simulink block for state based design and flow chart controls of execution.

  Simulink allows for a basic solver to execute either in discrete time or continuous time modes.
- MBD is popular tool in automotive especially for Motor Control applications.
- The MBDT for S32K Series consists of many Simulink libraries.
- MBD Automatic code generation is another level of abstraction above C code, since algorithm architecture, data typing and optimizations occur at the same level the resistor.

- This voltage passes through an amplifier before it's fed into a 12-bit ADC which then outputs a digital value through the SPI interface.

2. FreeMASTER

FreeMASTER is a user-friendly real-time debug monitor and data visualization tool for application development and information management. This tool supports non-intrusive variable monitoring on a running system. Allows the data from multiple variables to be viewed in an oscilloscope-like display or in a common text format.

- Runtime configuration and tuning tool for embedded software applications.
- Graphical control panel.
- Data capture tool, Connect to target MCU via UART, CAN and JTAG.
- Monitor, show and control the variables in runtime.
- Interface to custom processing in matlab, excel etc.
- It is GUI based plug in tool that provides real time monitoring the motor control system parameters
- The Model Based Design Toolbox has built-in code generation supports FreeMASTER through LPUART interface support
- The main operations done by freeMaster are:

  Connect - It connects to the target MCU via UART

  Monitor - It can display and read the data in run-time

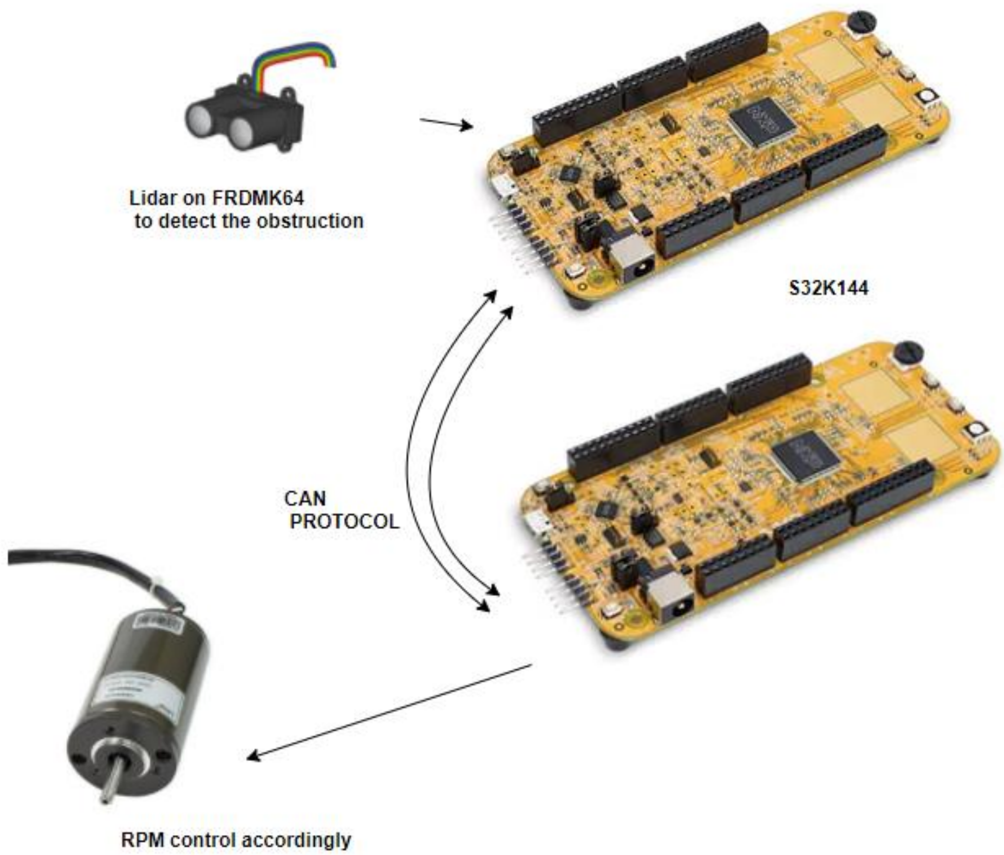
  Control - It controls the motor by sending commands and setting variables

  Share - The variables that are read can be enabled on Matlab to run the hardware

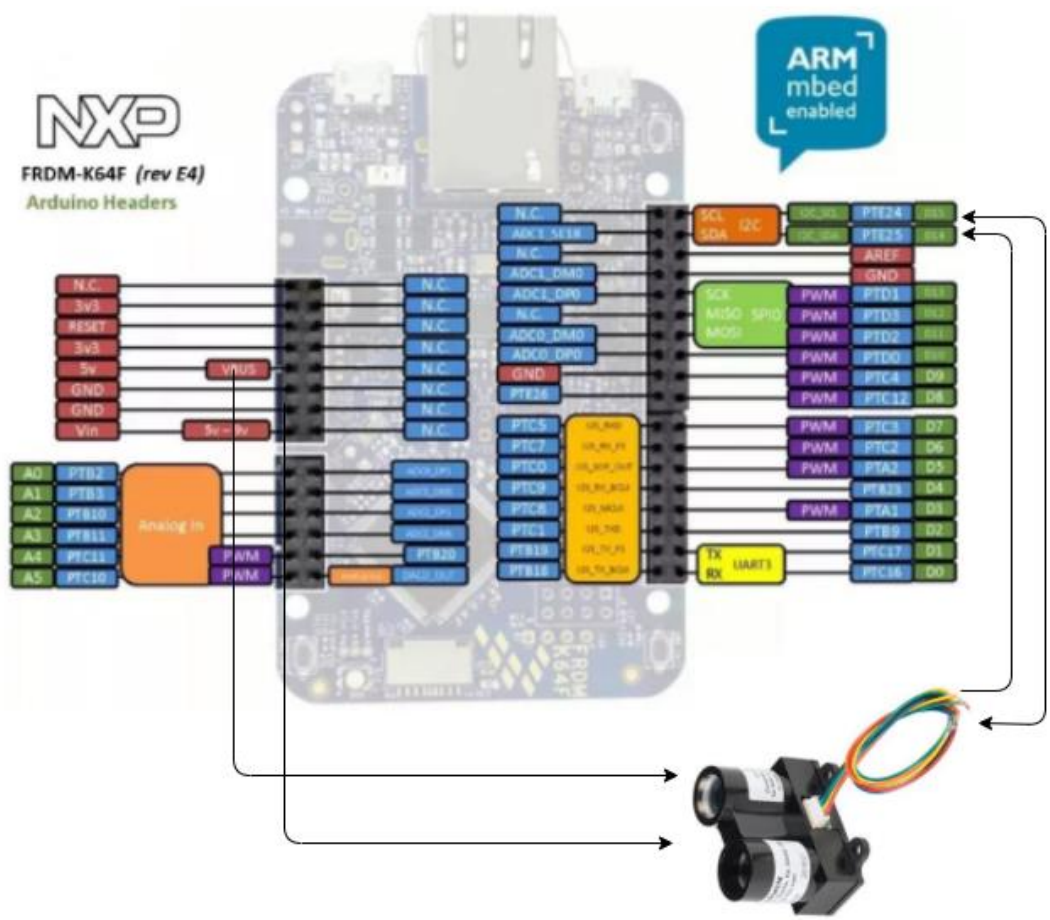
CHAPTER 3

IMPLEMENTATION

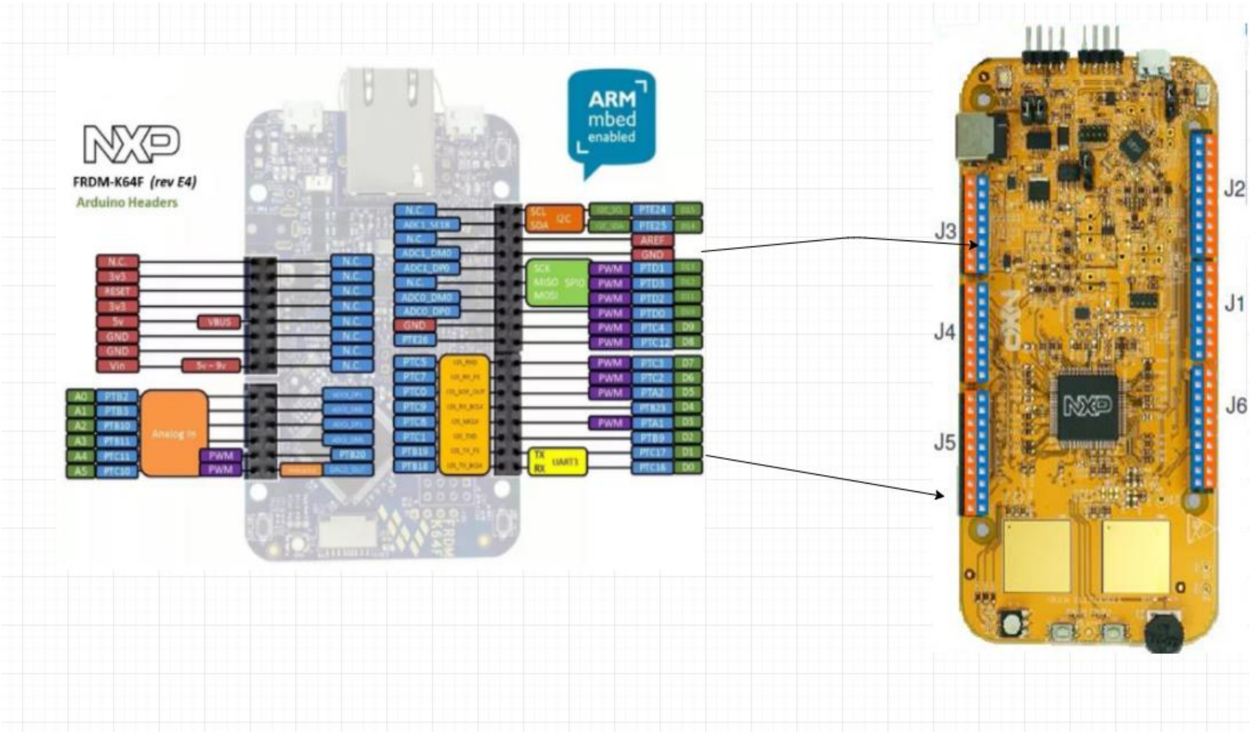
1. System Overview



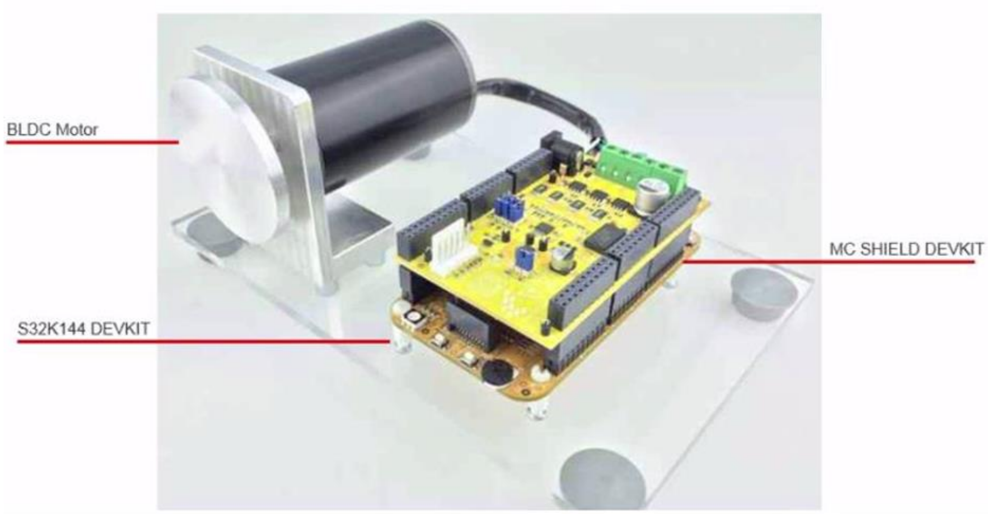
2. Connection Diagram of Garmin LidaliteV3 to FRDMK64 board



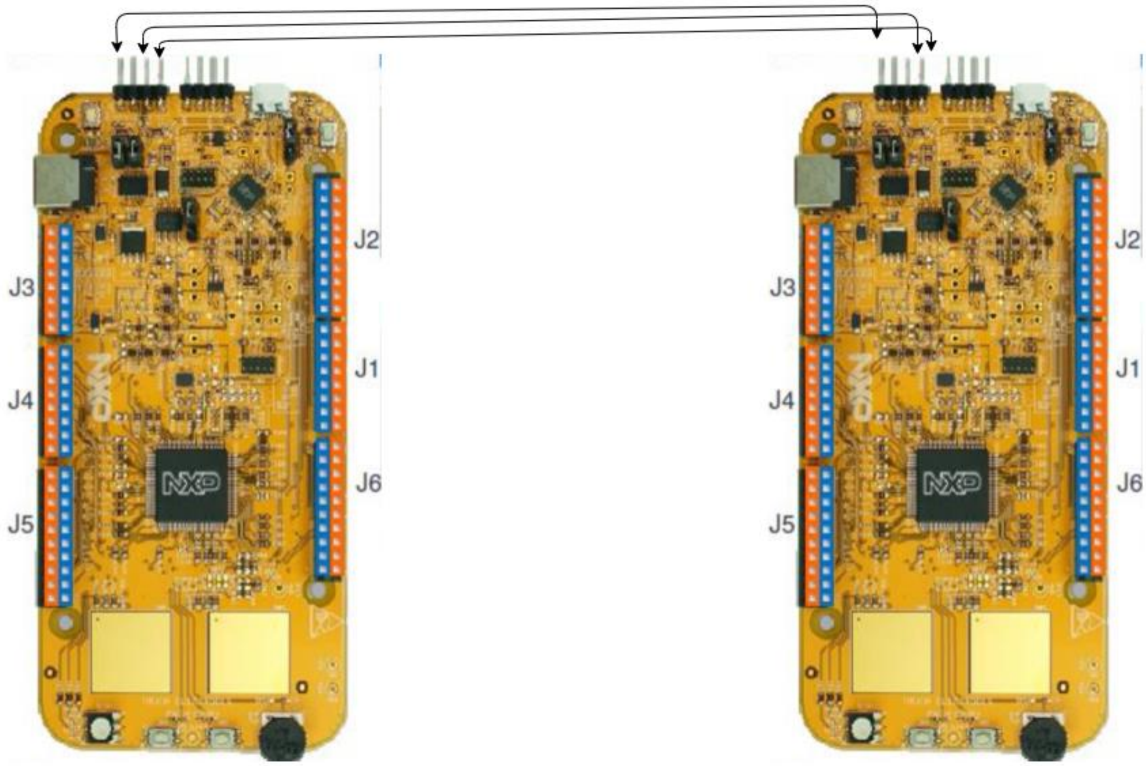
3. Connection Diagram of FRDMK64 board to S32K144 EVB

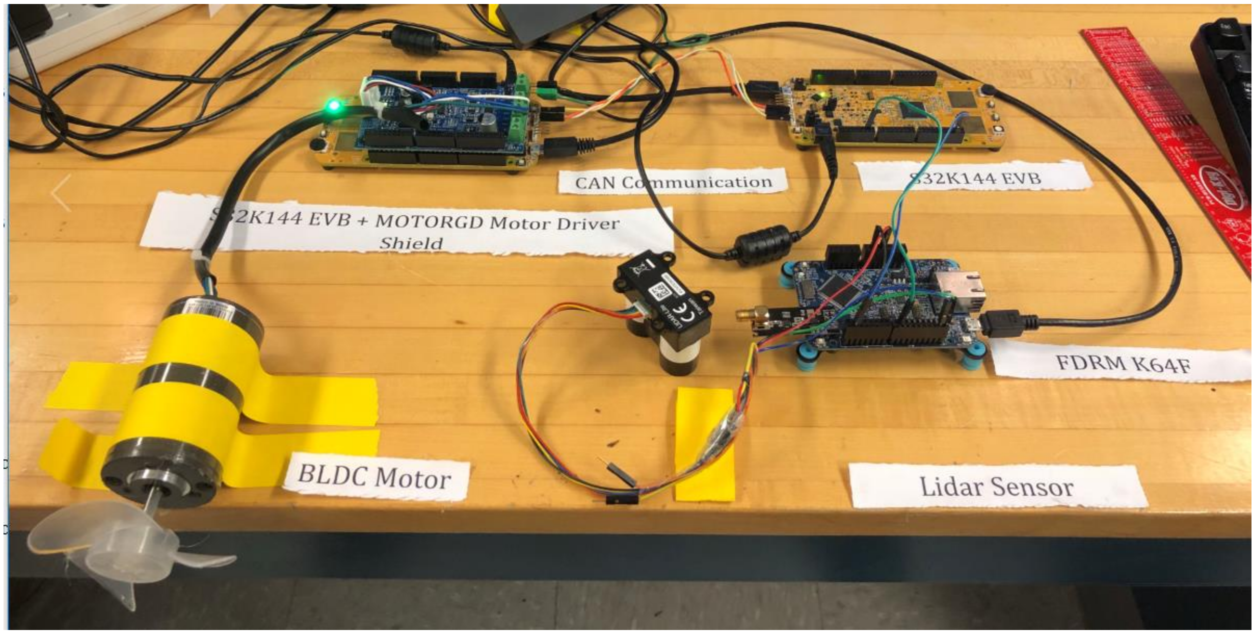


4. Connection Diagram of S32K144 EVB to MOTORGD Dev kit



5. Connection Diagram of CAN Communication Between two S32K144 EVB's

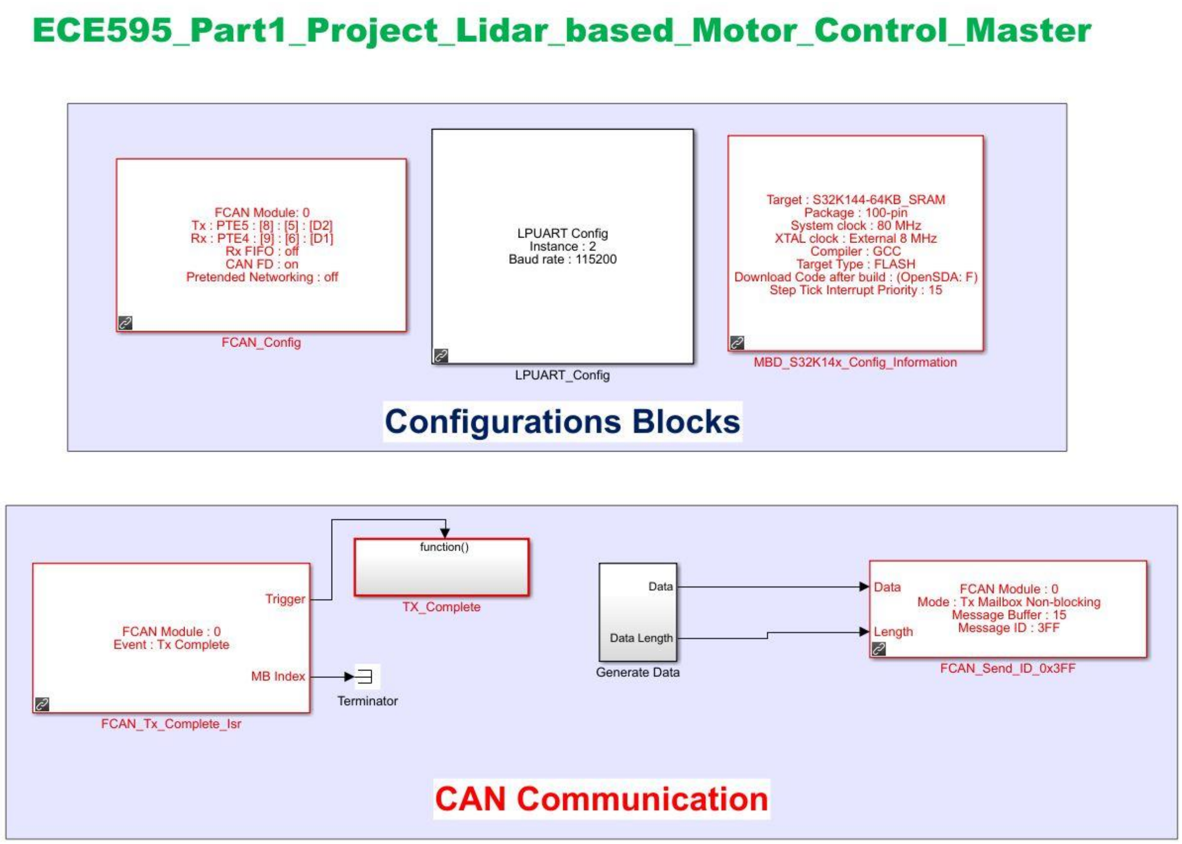




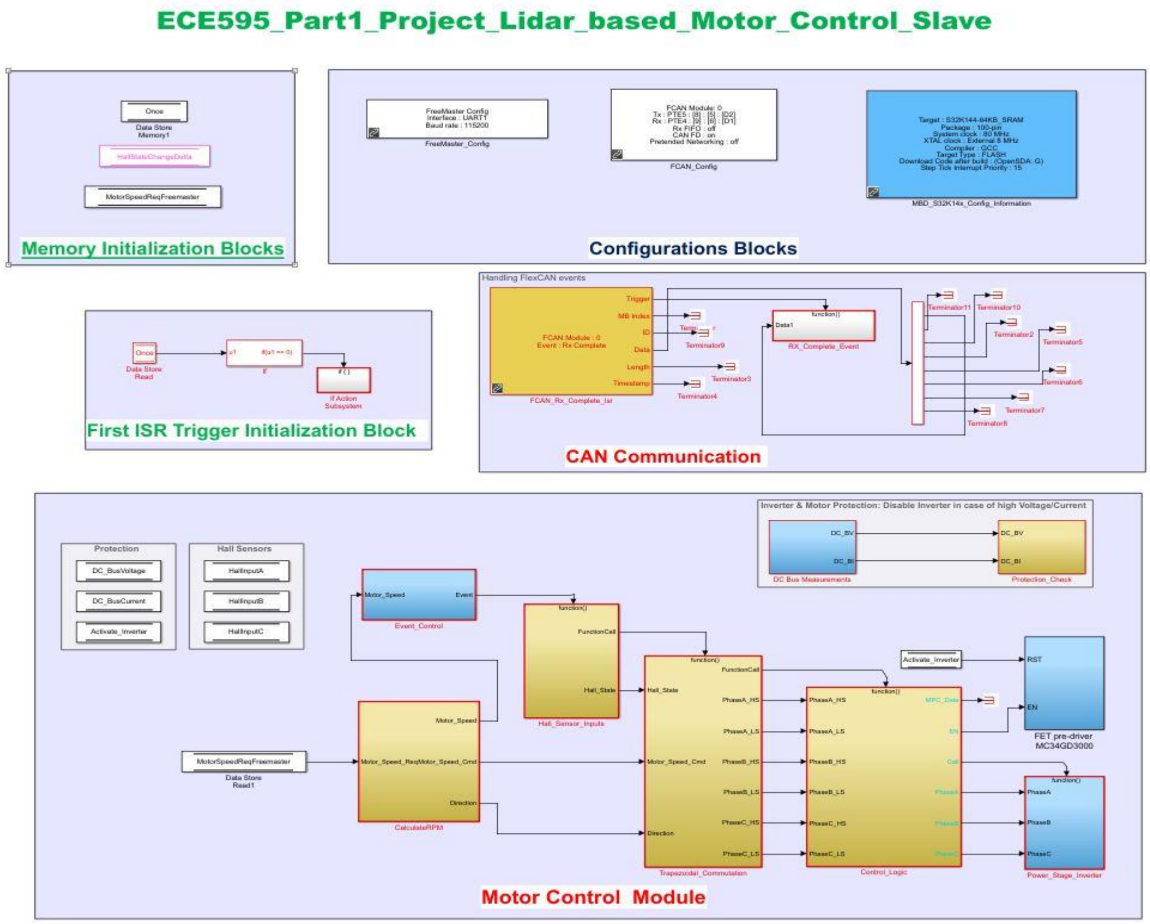
Actual setup

1. The Lidar Lite V3 connected to the FRDMK64F using I2C board to detect the presence of object and the distance of the object from the sensor.
2. The distance of the object calculated by the sensor is then sent to S32K144 (PTD6) using the UART interface.
3. This distance is mapped in to different rpm of BLDC motor. According to this distance, the speed of the motor will be controlled.
4. Connect the CAN High, Low and Ground pins of 2 S32K144. (Refer the connection diagram).
5. Power up the S32K144 from external supplies.
6. Connect the DEVKIT MotorGD and the Linix-BLDC motor to the slave S32K144.
7. Power up the MotorGD shield using 15V, 2A power adapter.
8. Make the S32K144 EVB to work on closed loop control with hall sensors.
9. We will be using Model Based Design Toolbox for S32K Series from matlab to model our system and implement it on target MCU.

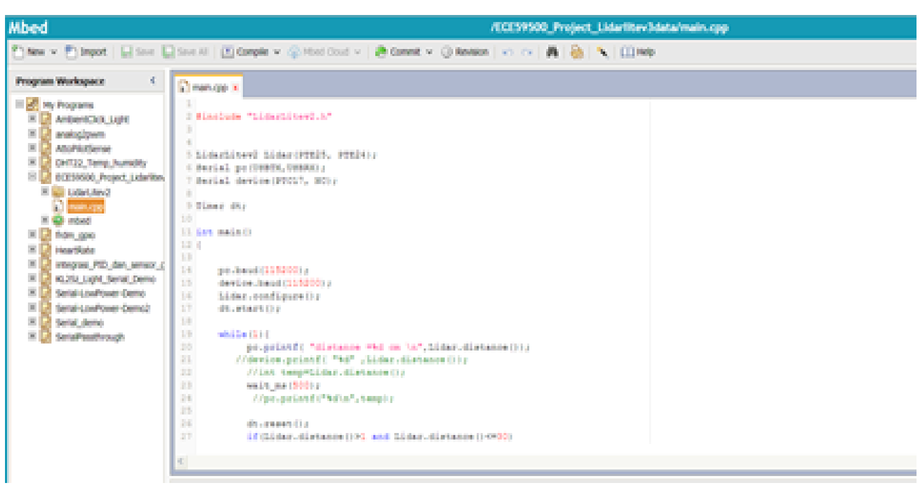
Master Block



Slave Block



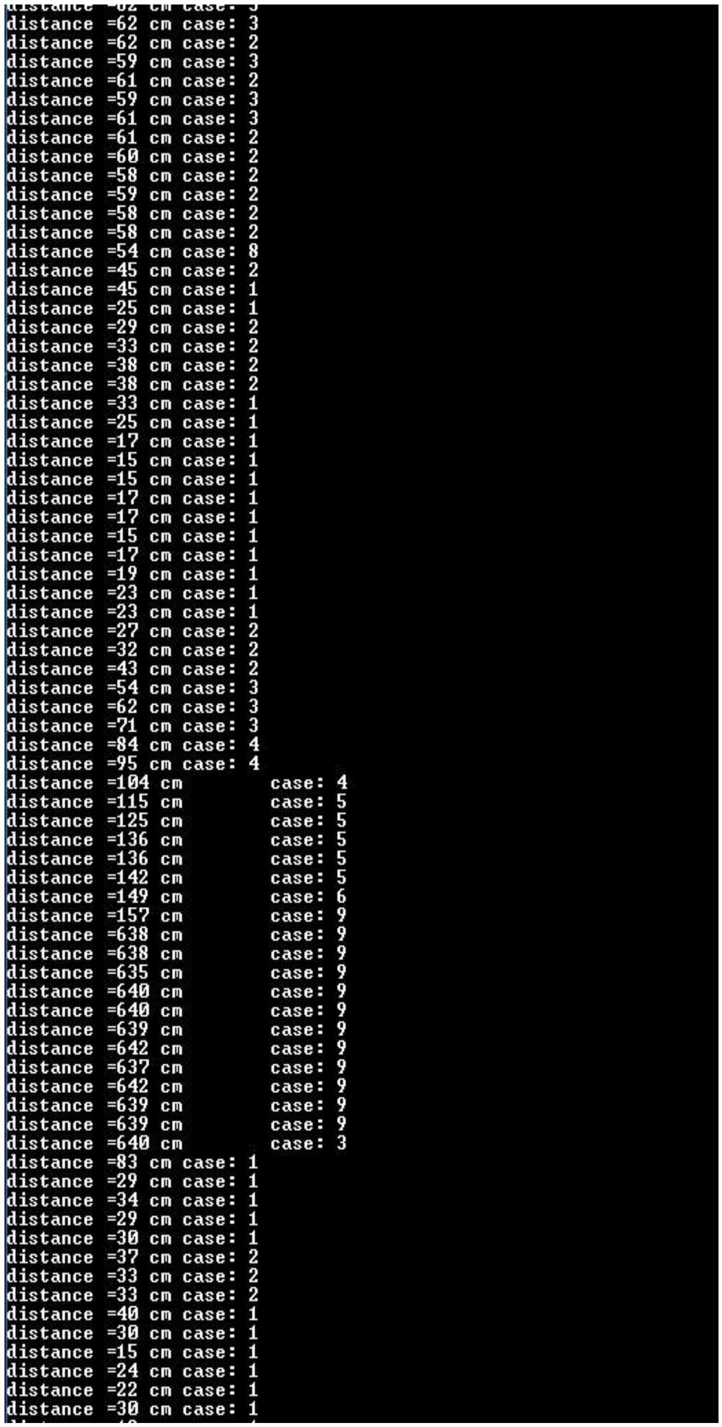
Mbed Code



CHAPTER 4

RESULT & CONCLUSION

Lidar Data

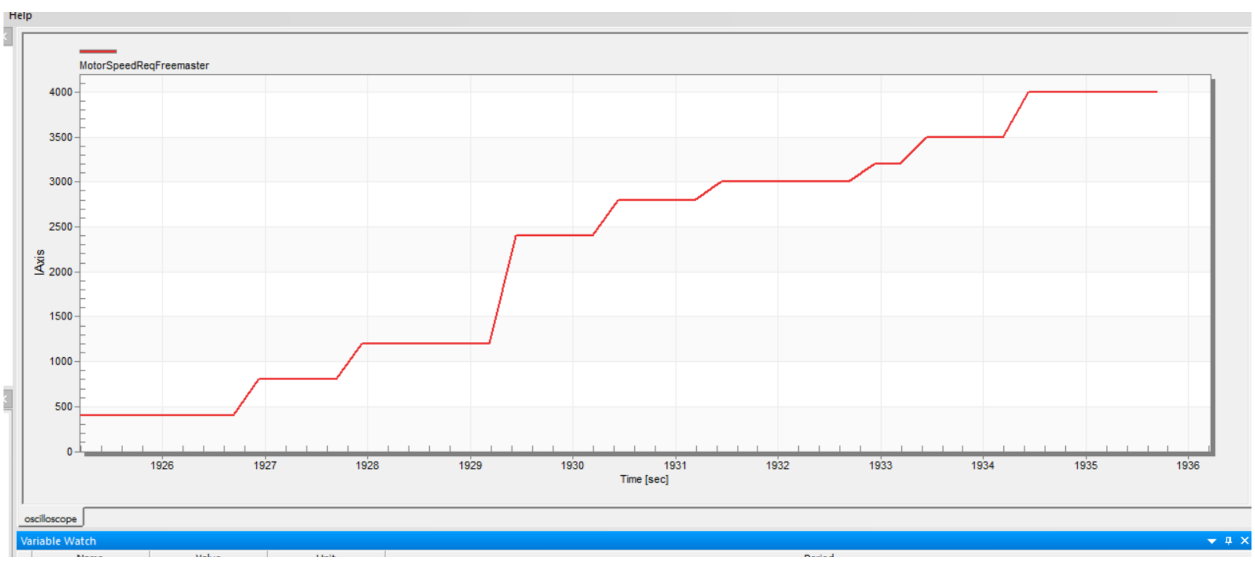


```
distance =62 cm case: 3
distance =62 cm case: 2
distance =59 cm case: 3
distance =61 cm case: 2
distance =59 cm case: 3
distance =61 cm case: 3
distance =61 cm case: 2
distance =60 cm case: 2
distance =58 cm case: 2
distance =59 cm case: 2
distance =58 cm case: 2
distance =58 cm case: 2
distance =54 cm case: 8
distance =45 cm case: 2
distance =45 cm case: 1
distance =25 cm case: 1
distance =29 cm case: 2
distance =33 cm case: 2
distance =38 cm case: 2
distance =38 cm case: 2
distance =33 cm case: 1
distance =25 cm case: 1
distance =17 cm case: 1
distance =15 cm case: 1
distance =15 cm case: 1
distance =17 cm case: 1
distance =17 cm case: 1
distance =15 cm case: 1
distance =17 cm case: 1
distance =19 cm case: 1
distance =23 cm case: 1
distance =23 cm case: 1
distance =27 cm case: 2
distance =32 cm case: 2
distance =43 cm case: 2
distance =54 cm case: 3
distance =62 cm case: 3
distance =71 cm case: 3
distance =84 cm case: 4
distance =95 cm case: 4
distance =104 cm        case: 4
distance =115 cm        case: 5
distance =125 cm        case: 5
distance =136 cm        case: 5
distance =136 cm        case: 5
distance =142 cm        case: 5
distance =149 cm        case: 6
distance =157 cm        case: 9
distance =638 cm        case: 9
distance =638 cm        case: 9
distance =635 cm        case: 9
distance =640 cm        case: 9
distance =640 cm        case: 9
distance =639 cm        case: 9
distance =642 cm        case: 9
distance =637 cm        case: 9
distance =642 cm        case: 9
distance =639 cm        case: 9
distance =639 cm        case: 9
distance =640 cm        case: 3
distance =83 cm case: 1
distance =29 cm case: 1
distance =34 cm case: 1
distance =29 cm case: 1
distance =30 cm case: 1
distance =37 cm case: 2
distance =33 cm case: 2
distance =33 cm case: 2
distance =40 cm case: 1
distance =30 cm case: 1
distance =15 cm case: 1
distance =24 cm case: 1
distance =22 cm case: 1
distance =30 cm case: 1
```

Distance – RPM Mapping

Distance(in cm)	RPM
0-30	0
30-60	400
60-90	800
90-180	1200
180-240	2400
240-270	2800
270-300	3200
above 300	4000

Motor Speed on FreeMASTER



CONCLUSION:

The S32K144 Evaluation board was used to collect the data from the Garmin LidarliteV3 sensor and this data is sent to the other S32K144 board which has BLDC motor using CAN protocol. Using closed loop motor control with sensors, the RPM of the motor is controlled according to the distance of the obstacle present.

CHAPTER 5

REFERENCES

1. https://www.nxp.com/pages/model-based-design-toolbox:MC_TOOLBOX
2. https://developer.mbed.org/compiler/#nav:/;
3. https://developer.mbed.org/cookbook/Homepage