Faculty of Engineering

Communication and Computer Department

2021

Car Anti-Theft System

Under Supervision: Prof. Dr El-Sayed Mostafa

Prepared by:

- Ahmed Ezz El-Din Ahmed
- Ahmed Sherif Mohammed
- Ahmed Tarek Moustafa
- Ehab Khaled Antar Ahmed
- Ibrahim Anwar El-Araby
- Rami Adel Abdulrazek

Table of Contents

1- Acknowledgement 03

2-Abstract 04

Chapter 1

1.1 - Overview...08

1.2 - Aim of the project.....................................08

1.3 - Requirements..09

1.4 - Assumption and dependencies..........................10

1.5 - Project schedule.......................................11

1.6 - Report contents.......................................12

Chapter 2

2.1 - Overview..15

2.2 - GPS Technology....................................15

2.3 - GSM Technology...................................16

2.4 - GPRS Service..16

2.5 - HTTP...17

2.6 - ADXL335 17

Chapter 3

3.1 - Overview……………………………………………...19

3.2 - General system block diagram………………….19

3.3 - System main components……………………...20

3.4 - System flow charts……………………………...32

3.5 - Data flow diagram……………………………....33

3.6 - System functions………………………………...33

Chapter 4

4.1 - Overview………………………………………36

4.2 - System use cases……………………………...36

4.3 - System sequence diagram………………………38

Chapter 5

5.1 - Overview………………………………………42

5.2 - Sub-System Testing……………………………..42

5.3 - Installation and preparing the system and testing Scenarios…………………………………………42

Chapter 6

6.1 - Overview……………………………………….49

6.2 - Conclusion and Achievement……………………49

6.3 - Challenges…………………………………….49

Acknowledgement

First of all, we would like to express our deepest gratitude and full thanks to almighty Allah for giving us the strength and the composure to complete our senior project, with its complexities and difficulties, it was not an easy task at all.

We convey our sincere gratitude to our Academic Supervisor Prof. DR. El-Sayed Mostafa, Associate Professor of Communications and computer department, Faculty of engineering, Helwan University. Without his kind instructions, even though it took up a lot of his precious time, this project wouldn't have been completed.

In every phase of the project his supervision and guidance shaped this report to be completed perfectly.

Abstract

A vehicle tracking system is very useful for tracking the movement of a vehicle from any location at any time. An efficient vehicle tracking system is designed and implemented for tracking the movement of any equipped vehicle from any location at any time. The proposed system made good use of popular technology that combines a smartphone with an ATMEGA32 MC. The designed in-vehicle device works using Global Positioning System (GPS) and Global System for Mobile Communication (GSM) technology that is one of the most common ways for vehicle tracking. The device is embedded inside a vehicle, these coordinates are to be determined and tracked in real time. ATMEGA32 is used to control the GPS receiver and GSM module. The vehicle tracking system uses the GPS module to get geographic coordinates at regular time interval. The GSM module is used to transmit and update the vehicle location to a server. This paper gives minute by minute update about vehicle location by sending the coordinates of the car through GSM modem to the server.

It contains latitude and longitude of the location of vehicle. ATMEGA32 gets the coordinates from GPS modem and then it sends this information to user's mobile application.

And then the mobile application displays location of the vehicle on an android phone. Thus, the user is able to continuously monitor a moving vehicle on demand using smartphone.

1

CHAPTER ONE

INTRODUCTION

1.1 Overview

1.2 Aim of the project

1.3 Requirements

1.4 Assumption and dependencies

1.5 Project schedule

1.6 Report contents

1.1 Overview

In this chapter, we will introduce the main idea for our project, the aim we try to achieve from this system, the main requirement that will be needed in our work, some of the assumption and dependencies that we will rely on, we also show general block diagram for our system.

1.2 Aim of the project

The project's main idea is to provide a car tracking system, which gives a clear vision about car movements through its travel. Our system will enable the customer to easily track their cars, over country or even worldwide, and for companies with a large number of vehicles, knowing where customer's cars are at every moment in real time and know the speed of traveling at any interval. Our system also provides another layer of security as the owner of the car can, at will, send a request through the mobile application to bring the car to a complete stop in case the car is under attempt of theft.

1.3 Requirements

The requirements of our project will be described in two parts the functional and nonfunctional requirements as follow:

1.3.1 Functional requirements:

The system must track user's car in real time, over its travel at any time, this will require a hardware to receive the GPS coordinates, and send it through communication network to the server through the GSM module.

The system must be able to track the car and this requires server software that deals with coordinates and geographic maps.

1.3.2 Non-Functional requirements:

- Usability, the system will be used by the car's owner or company employees, so the system needs to be easy to use.
- Availability, the system will be available for any person that wants to use it, and also, it should be available at any place during car travel, so we rely on the GPS system that has wide spread in our countries.
- Reliability, the system must be reliable, so it can give you a proper location of your car, with acceptable minimum error.

And for all these requirements to be achieved we will need the following hardware and software:

Hardware:

- A smart phone.
- GSM network with GPRS capability
- Main server.
- GPS module.
- Microcontrollers and other needed peripherals.

Software:

- Mobile programming language (Dart Programming).
- Programming software for map display.
- Graphical user interface (GUI).
- Codes that run on the microcontrollers.

1.4 Assumption and dependencies

The following assumptions are considered in our project:

- the car that will be tracked will have a device fitted with a GPS module to provide GPS service and receive location data.
- Longitude and latitude coordinates received from the GPS satellite are determined using a number of GPS satellites and will be sent using GSM network service to the central server.

1.5 Project Schedule

- Stage 1: Select the idea

Determine the idea of the project, the motivation, and the main objective we intend to achieve.

- Stage 2: Preparing for the project

In this stage, deeper determination of the tasks, and steps we want to perform, are done.

- Stage 3: Project Analysis

In this step, a study of the all possible design's options is to determine our own design.

- Stage 4: Determine the project's requirement after we determine our design scheme, we specify all the needed requirement for the user and the system, software and hardware. And try to bring them to be ready for the implementation stage.

- Stage 5: Studying the Principles

This stage of the project is necessary to study the GSM, GPS, GPRS, mobile programming language, main server programming language.

- Stage 6: Documentation Writing

Documenting the project will begin from the first stage to the last stage.

- Stage 7: make the hardware available

In this stage, the needed hardware mobiles and main server's PC will be brought for the next step

- Stage 8: build up the software

The programming of the project code is started and will be downloaded to the mobile and the central server. Make sure the needed maps are existing.

- Stage 9: testing the system

  In this stage the car will move, tracking will start, feedback will be provided.

- Stage 10: Writing Documentation

  The documentation will continue from the first stage to the last one in parallel.

1.6 Report contents

This report includes the following sex chapters:

- **Chapter one:** Introduction.

  In this chapter, we introduce a general overview about the project, the main idea of the project, requirements, assumption and dependences, the time schedule, and the estimated cost.

- **Chapter two:** Theoretical Background.

  This chapter includes the Theoretical background related to the main idea of the project, technologies, hardware, and software used in the project.

- **Chapter three:** Project Conceptual Design.

  In this chapter we talk about the design concepts, the general block diagram that shows how the system works, the system's flow chart, the data flow diagram, and the functional block diagram that mentions all the implemented functions in the system.

- **Chapter four:** Detailed Design.

  In this chapter we talk about the detailed software engineering design including UML model diagram for both mobile and server application software classes. Then system use case diagram will be described, also the sequence diagram for use cases will be mentioned.

- **Chapter five**: System Testing.

  In this chapter, the whole testing stage will be described, that means a test of an entire interconnected set of components and software for the purpose of determining proper functions and achieving the desired goals of the system. We will talk about testing of each part of the system, describe the scenarios, represent each test and its errors, challenges and modifications, and also, we will figure out the error rate.

- **Chapter six:** Conclusion and Recommendations

  In this chapter, we will mention what we achieved in this project and the conclusion for all things that we have done, also we will talk about the challenges that we faced and ending with recommendation needed for the future work.

2

CHAPTER TWO

THEORITICAL BACKGROUND

2.1 Overview

2.2 GPS Technology

2.3 GSM Technology

2.4 GPRS Service

2.5 HTTP

2.6 ADXL335

2.1 Overview

In this chapter we will mention the basic theoretical information, some technologies, and learn about some devices that will be used in our project in order to use them in an appropriate way and take the desired advantages of them to make it easy for the reader to understand and interact with the project.

We will talk about the GPS technology, GSM/GPRS technology, a collection of hardware that includes the mobiles, GPS receivers, GPRS modems.

Software that will be used represented by a programming language on the mobile phone, software on the central server in order to receive locations coordinates, and then plot on the geographic maps.

And also, we will talk about the communication technique, which represents an important part in our system, where the data and coordinates will be transmitted from tracking mobile to the main server.

2.2 GPS in our project

We will use GPS to determine the real time location of the moving tracked car, using the GPS receiver in the GPS module. The GPS module sends NMEA messages containing the relative information and process them to calculate the position. The GSM module takes the GPS measurements, and report these to the central server in the network.

2.3 GSM in our project

The usage of the GSM technology in this project is represented by transferring the data from the car that contains the GPS receiver (the GPS module) to the central server. This data will contain the position of the car which can then be viewed on a map on the mobile application.

2.4 GPRS in our project

After determining the location coordinates by the GPS receiver in the GPS module, these coordinates and other additional information will be sent using the GPRS technology implemented in the GSM module, through the cloud, to the central server to be accessible to the mobile application.

The Difference between GSM and GPRS

GSM is an acronym for Global System for Mobile Communications. GSM can only use SMS to transmit data, and it can't be "real-time online" or "charged by volume".

GPRS is the abbreviation of General Packet Radio Service. It is a packet data bearer and transmission service developed on the basis of the GSM system.

The most fundamental difference between GPRS and GSM system is that GSM is a circuit switching system, and GPRS is a packet switching system. Compared to GSM, GPRS has obvious advantages in the bearer and support of data services, and it can more efficiently use wireless network channel resources, it is especially suitable for intermittent, acyclic data transmission, a small amount of data transmission, and infrequent transmission of large capacity data.

The technical advantages of GPRS are also manifested in the following aspects: high transmission rate, high resource utilization, short access time, always online, support IP protocol and X.25 protocol and reasonable charges.

Advantages of GPRS:

- GPRS communication costs are low.
- GPRS data transmission is fast, real-time performance is good.
- GPRS is a networked data transmission that can operate multiple devices simultaneously.

There is no communication module in the center, and it can be directly connected to the Internet. The operation cost of the center is fixed, and the data traffic speed is far greater than the GSM module.

2.5 HTTP in our project

After setting the GPRS to transmit data between the GSM module in the car and main server, we chose HTTP to represent the GPRS service, considering the GSM module as an HTTP client and the cloud server as an HTTP server.

2.6 ADXL335 Sensor

The main usage of the ADXL335 accelerometer sensor in our project is to help us detect if the vehicle is involved in an accident and we can do that by determining a certain acceleration threshold value in the dimension that the car is moving into and if the threshold acceleration has been reached, we can declare that the vehicle has crashed.

3

CHAPTER THREE

PROJECT CONCEPTUAL DESIGN

3.1 Overview

3.2 General system block diagram

3.3 System main components

3.4 System flow charts

3.5 Data flow diagram

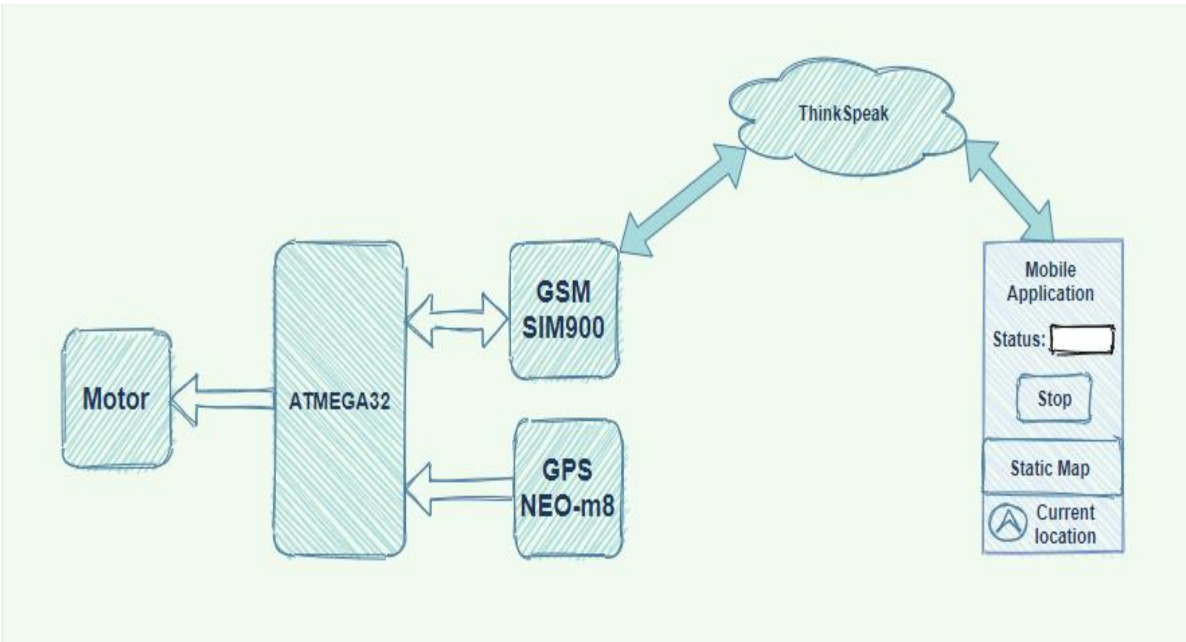
3.6 System functions

3.1 Overview

In this chapter we will describe our system's main parts and the design concepts in some details, we will talk about the system's general block diagram, the system's main components and their related options, system main flow chart, system data flow diagram and the system main functions.

3.2 General system block diagram:

the main parts of our project are: GPS receiver, where locations coordinates will be created, GSM network which is the connection medium through which data will be transferred, and Central server where the data will be received and processed, the MCU is for processing the data and the mobile application is for providing the user interface for the owner of the car



3.3 System main components:

The main components of our system are the GPS Receiver, GSM network, and Central server, all of them in this section will be described in more details, and the options for these components will be also described as follow:

3.3.1 Hardware components

- **ATmega32 microcontroller**

ATmega32 microcontroller is the heart of the project that is used for interfacing with various hardware peripherals. It is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture.

ATmega32 microcontroller is interfaced serially to a GPS module and GSM modem. The GPS module sends a standard NMEA message that contains the coordinates to be processed by the microcontroller. The processed data is sent to the user's mobile through a GSM modem.

This GPS based vehicle tracking system implements RS-232 protocol for serial communication between the microcontroller, GPS and GSM modem. A serial driver IC MAX232 is used for converting RS-232 voltage levels into TTL voltage levels.

The user's mobile number should be included in the source code written for the microcontroller. Thus the user's mobile number resides in the internal memory of the MCU.

- ## GPS

GPS is a space-based satellite navigation system. It provides location and time information in all weather conditions, anywhere on or near the Earth. GPS receivers are popularly used for navigation, positioning, time dissemination and other research purposes.

The GPS consists of satellites that orbit the earth. These satellites are geosynchronous with an orbital period that is the same as the Earth's rotation period. So they maintain exactly the same position with respect to the earth below them. All the GPS satellites transmit radio signals, which are then captured by a GPS receiver and used to calculate its geographical position. A minimum of four satellites may be required to compute the four dimensions of X, Y, Z (latitude, longitude and elevation) and time. GPS receiver converts the received signals into position and estimates time and some other useful information depending on the application and requirements.

GPS determines the distance between a GPS satellite and a GPS receiver by measuring the amount of time taken by a radio signal (the GPS signal) to travel from the satellite to the receiver. To obtain accurate information, the satellites and the receiver use very accurate clocks, which are synchronized so that they generate the same code at exactly the same time.

NMEA Message Description

GPGGA LOG:

$GPGGA,184241.000,1829.9639,N,07347.6174,E,1,05,2.1,607.1,M,-64.7,M,,0000*7C

Name	Example	Units	Description
Message ID	$GPGGA		GGA Protocol Header
UTC Time	184241.000		hhmmss.sss
Latitude	1829.9639		ddmm.mmmm
N/S Indicator	N		N=North, S=South
Longitude	07347.6174		dddmm.mmmm
E/W Indicator	E		E=East, W=West
Position Fix Indicator	1		Fix GPS SPS mode
Satellites Used	05		Range 0 to 12
HDOP	2.1		Horizontal Dilution of Precision
MSL Altitude	607.1	Meters	Mean Sea Level
Units	M	Meters	
Geoid Separation	64.7	Meters	
Units	M	Meters	
Age of Diff. Corr.	-		Null field if DGPS is not used
Diff. Ref Station ID	0000		
Checksum	*7C		
Carriage return Line Feed	<CR><LF>		End of message transmission

“$GPGGA,170017.458,2950.6504,N,03120.7573,E,1,10,4.00,17.8, M,50.0,M,,*54“

$, NMEA message log start

GPGGA, identifier of this log, means global positioning system fix data (time, position, fix type data).

',', acts as a delimiter to separate different fields in the message

170017.458, indicates the UTC time represented as hhmmss.sss meaning that in this message the time is 17:00:17.458 UTC 2950.6504 indicates the latitude represented as ddmm.mmmm meaning that in this message the latitude is 29 and (50.6504/60) degrees notice that this doesn't give an indication to whether this location exists in the northern hemisphere or the southern hemisphere

N, indicates that this location resides in the northern hemisphere, meaning that when we convert this into degrees it is going to be a positive value

03120.7573, indicates the longitude represented as dddmm.mmmm meaning that in this message the longitude is 31 and (20.7573/60) degrees again, this doesn't give an indication to whether this location exists in the western part of the world or the eastern part

E, indicates that this location resides in the eastern region, meaning that when we convert this into degrees it is going to be a positive value, now we have enough information on getting the exact location which is 29.84417383° latitude, 31.34595467° longitude, residing in the north-east quarter of the world

1, position fix indicator, 1 means that we're getting GPS fix

10, represents the number of satellites in view

4.00, represents the dilution of precision which is a term used to describe the strength of the current satellite configuration on the accuracy of the data collected

17.8, Mean sea level altitude, represents the altitude (or elevation from sea-level)

M, means the unit for measuring altitude is meter

50.0, Geoid separation, reports the height difference between the ellipsodial surface and the geoid's model surface

M, unit for measuring geoid separation is meter blank field, time since last DGPS update, no last update here (blank)*54, blank is diff ref station id

***54,** checksum

And there are other logs like GPGSV, GPRMC, GPVDG, GPGSA

- ## GSM modem

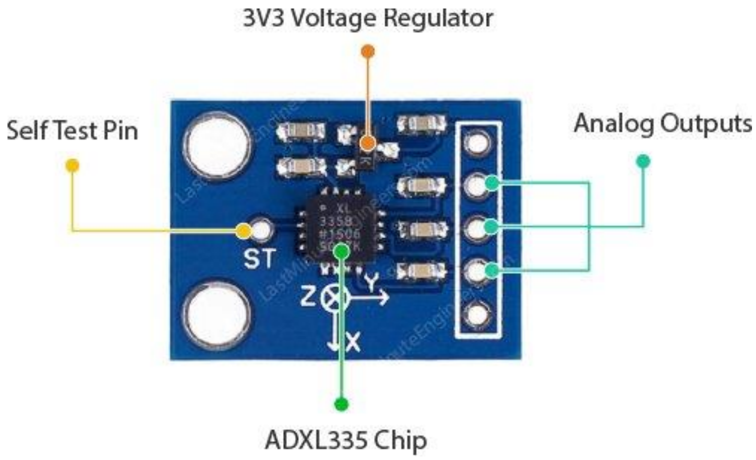
GSM is a standard set developed by the European Telecommunications Standards Institute (ETSI) to describe technologies for second-generation (2G) digital cellular networks.

A GSM modem is a specialized type of modem that accepts a SIM card and operates over a subscription to a mobile operator just like a mobile phone.

GSM modems are a cost-effective solution for receiving and sending SMS messages or data from any server because the sender is paying for the message delivery. To perform these tasks, a GSM modem must support an extended AT command set for sending and receiving SMS messages or data from any server.

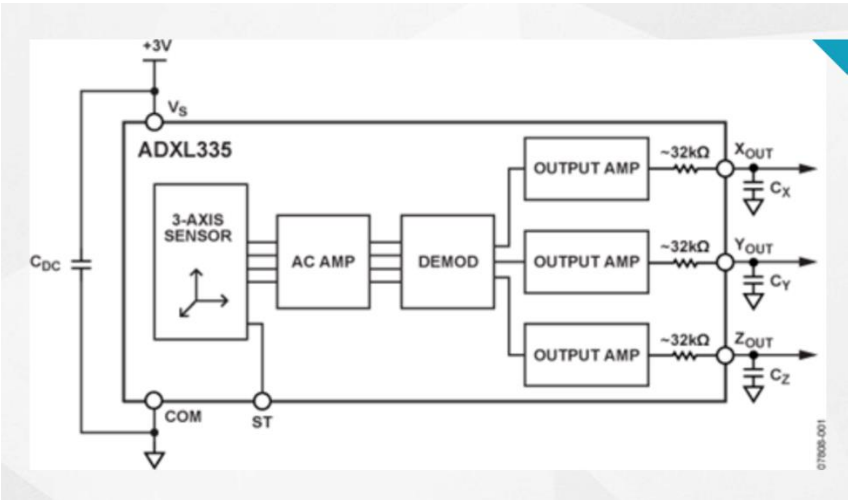
It should also be noted that not all phones support this modem interface for sending and receiving SMS messages, particularly most smartphones like the Blackberry, iPhone and Windows mobile devices.

- ## ADXL335 (Accelerometer Sensor)



The ADXL335 is a small, low power, complete 3-axis accelerometer with signal conditioned voltage outputs. It can measure the static acceleration of gravity in tilt-sensing applications, as well as dynamic acceleration resulting from motion, shock, or vibration.

Sensor's main internal components



Features and Benefits

- 3-axis sensing.
- Small, low profile package.

- 4 mm x 4 mm x 1.45 mm LFCSP.
- Low power – 350 µA (typical).
- Single-supply operation 1.8 V to 3.6 V

General Description

The ADXL335 is a small, thin, low power, complete 3-axis accelerometer with signal conditioned voltage outputs. The product measures acceleration with a minimum full-scale range +-3 g. It can measure the static acceleration of gravity in tilt-sensing applications, as well as dynamic acceleration resulting from motion, shock, or vibration. The user selects the bandwidth of the accelerometer using the Cx, Cy and Cz capacitors at the Xout, Yout, and Zout pins. Bandwidths can be selected to suit the application, with a range of 0.5 Hz to 550 Hz for the Z axis. The ADXL335 is available in a small, low profile, 4mm x 4mm 4 1.45 mm, 16-lead, plastic lead frame chip scale package (LFCSP_LQ).

3.3.2 Software components

AVR ATmega32 has flexible USART, SPI and other communication protocols, which can be used for serial communication with other devices like computers, serial GSM, GPS modules, etc.

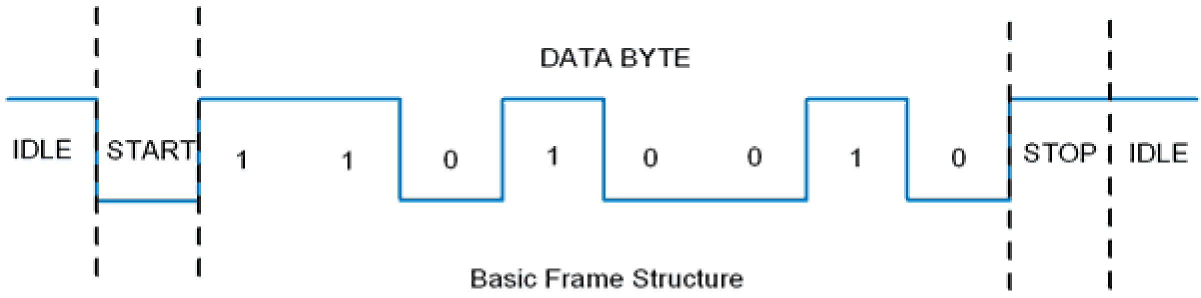
USART

Before beginning with AVR USART, we will walk through the basics of serial communication.

Serial data framing

While sending/receiving data, some bits are added for the purpose of knowing the beginning/ending of data, etc. commonly used structure is: 8

data bits, 1 start bit (logic 0), and 1 stop bit (logic 1), as shown:



There are also other supported frame formats available in UART, like parity bit, variable data bits (5-9 data bits).

Speed (Baud rate)

As we know the bit rate is “Number of bits per second (bps)”, also known as Baud rate in Binary system. Normally this defines how fast the serial line is. There are some standard baud rates defined e.g. 1200, 2400, 4800, 19200, 115200 bps, etc. Normally 9600 bps is used where speed is not a critical issue.

Wires and Hardware connection

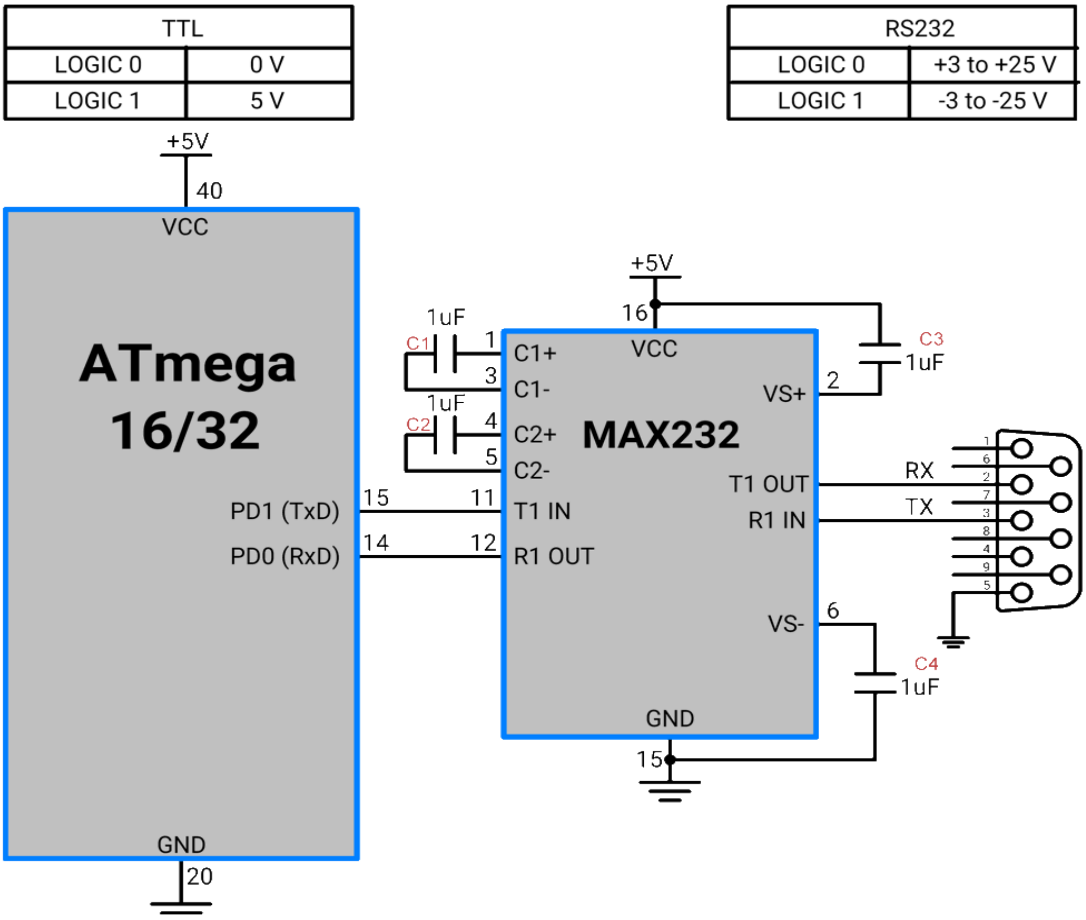
Normally in USART, we only need Tx (Transmit), Rx (Receive), and GND wires.

AVR ATmega32 USART has a TTL voltage level which is 0 v for logic 0 and 5 v for logic 1.

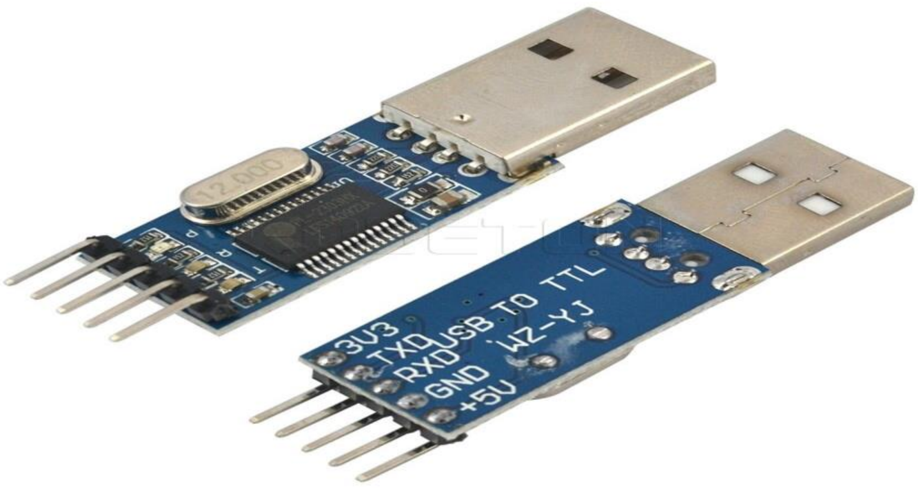
In computers and most of the old devices, RS232 protocol is used for serial communication, where normally 9 pin ‘D’ shape connector is used. RS232 serial communication has different voltage levels than ATmega serial communication i.e. +3 v to +25 v for logic zero and -3 v to -25 v for logic 1.

So to communicate with RS232 protocol, we need to use a voltage level converter like MAX232 IC.

Although there are 9 pins in the DB9 connector, we don’t need to use all the pins. Only 2nd Tx (Transmit), 3rd Rx (Receive), and 5th GND pin need to be connected.



With new PCs and laptops, there is no RS232 protocol and DB9 connector. We have to use serial to the USB connector. There are various serial to USB connectors available e.g. CP2102, FT232RL, CH340, etc.



Serial to USB converter

To see serial communication, we can use a serial terminal like Realterm, Teraterm, etc. By selecting the serial port number (COM port in windows) and baud rate, we can open a serial port for communication.

As stated in the datasheet ATmega32 USART has the following features

- Full Duplex Operation (Independent Serial Receive and Transmit Registers)
- Asynchronous or Synchronous Operation
- Master or Slave Clocked Synchronous Operation
- High-Resolution Baud Rate Generator
- Supports Serial Frames with 5, 6, 7, 8, or 9 Data Bits and 1 or 2 Stop Bits.
- Odd or Even Parity Generation and Parity Check Supported by Hardware
- Data Overrun Detection

- Framing Error Detection
- Noise Filtering Includes False Start Bit Detection and Digital Low Pass Filter
- Three Separate Interrupts on TX Complete, TX Data Register Empty, and RX Complete
- Multi-processor Communication Mode
- Double Speed Asynchronous Communication Mode

SPI

•Serial Peripheral Interface (SPI) bus is a synchronous serial communication interface specification used for short distance communication.

•The SPI bus can operate with a single master device and with one or more slave devices.

•SPI is a full duplex communication protocol.

•SPI is single master multiple slave communication protocol.

•SPI protocol has complete flexibility for the bits transferred as there is no limit for 8-bit word or message size or purpose.

ATmega32 has an inbuilt SPI module. It can act as a master and slave SPI device. SPI communication pins in AVR ATmega32 are:

- **MISO** (Master In Slave Out)

  The Master receives data and the slave transmits data through this pin.

- **MOSI** (Master Out Slave In)

  The Master transmits data and the slave receives data through this pin.

- **SCK** (Shift Clock)

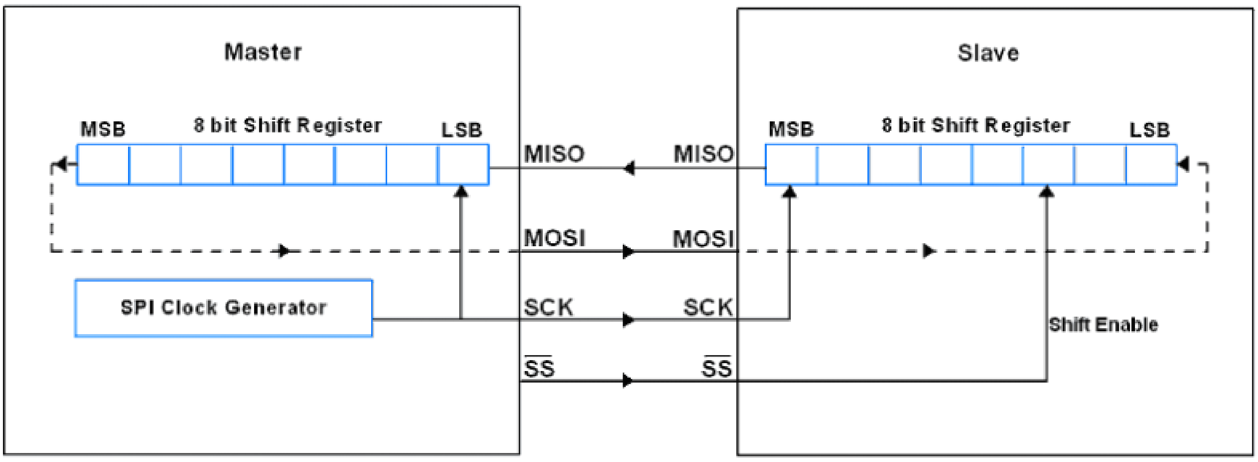
  The Master generates this clock for the communication, which is used by the slave.

  The only master can initiate a serial clock.

- **SS** (Slave Select)

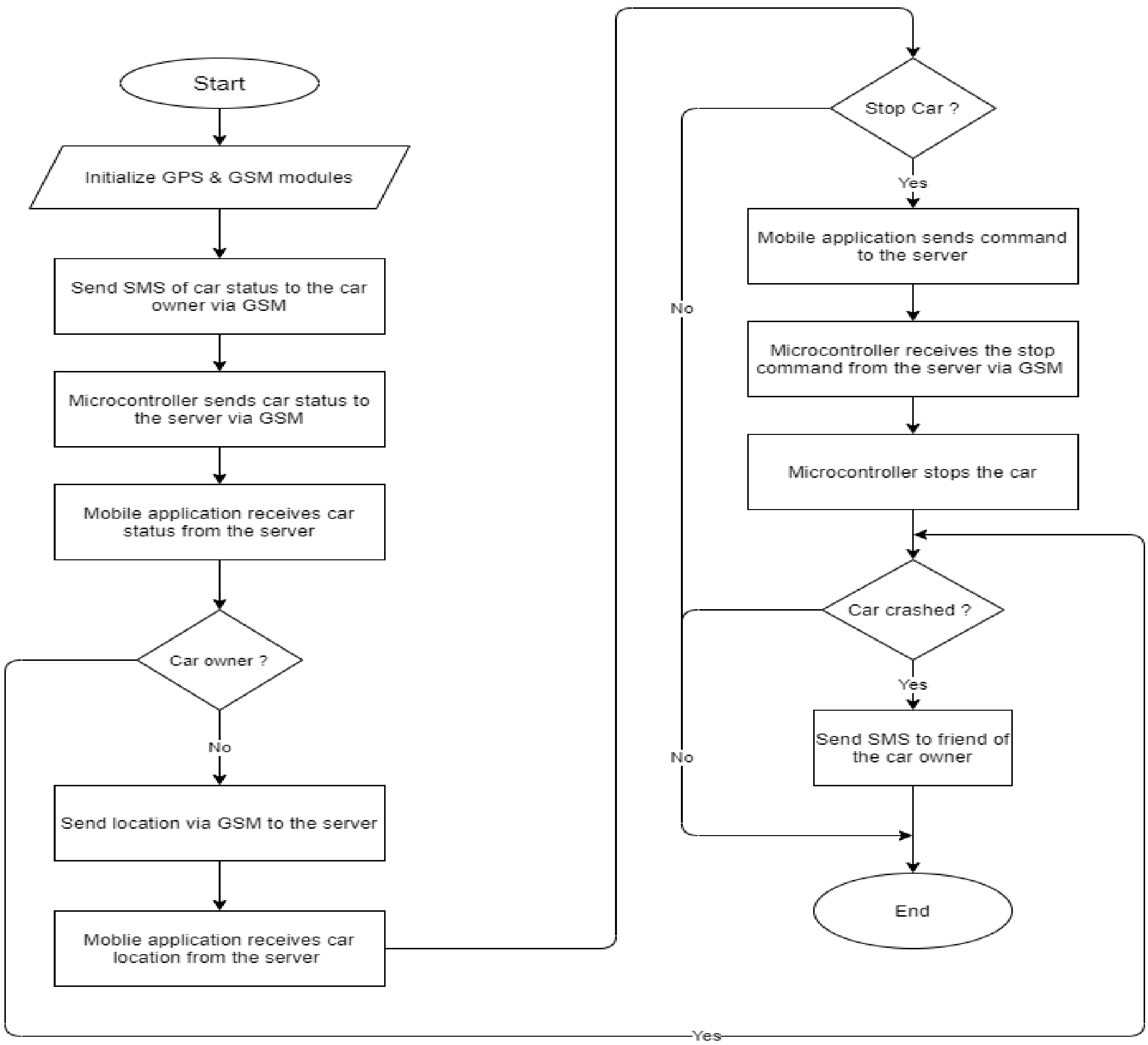
  Master can select slaves through this pin.

The interconnection between master and slave using SPI is shown in the below figure.



SPI Master Slave Interconnection

3.4 System flow chart:

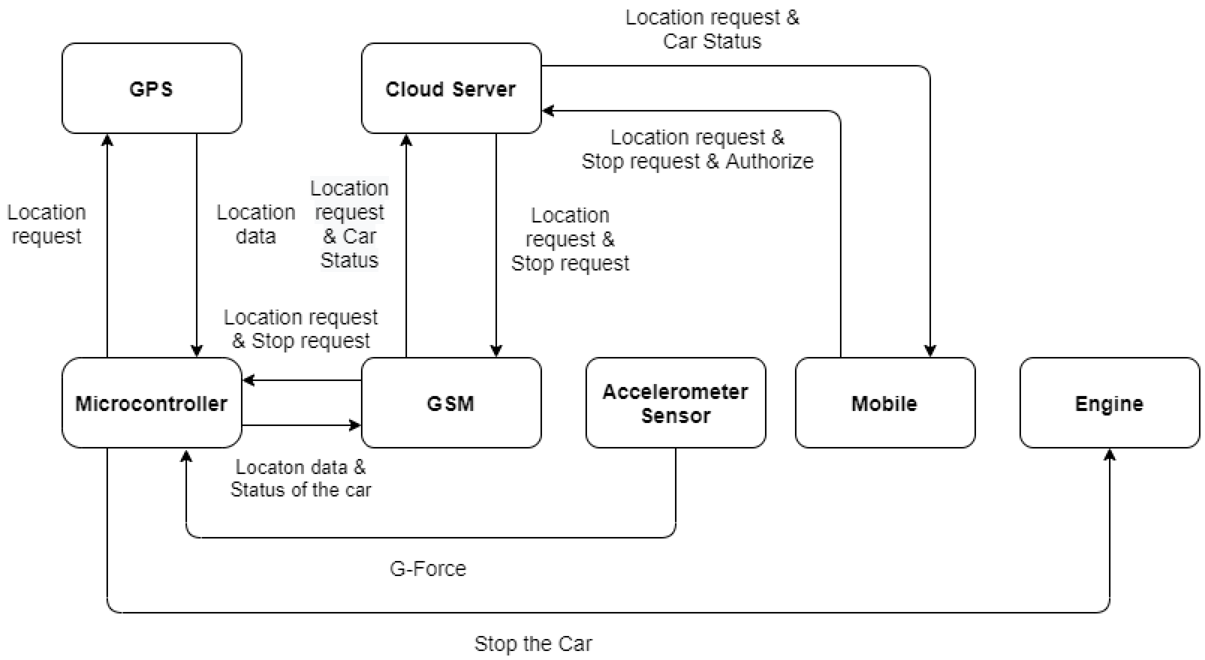


1- When the engine runs, a message is sent to the vehicle's owner (mobile application) through the GSM module that is embedded in the car.

2- This msg conveys information about the location of the car in terms of longitude and latitude and is sent through the GSM module.

3- The user now can track the car's location via their mobile application then they either can stop the car or just by knowing the location of the car the owner can call the Police for help.

3.5 Data Flow Diagram:



3.6 System Functions:

Here we mention more details about main parts in our project, and the main functions that will be implemented.

3.6.1 Mobile Functions:

The mobile application provides the following functionalities:

- Create user interface: We built an application to be installed on the user's phone, this application must be usable and easy for the users to interact with it.
- Get registration information: For any user, in order to register in the system, some information is necessary to be supplied, this information consist of: username and password.
- Get coordinates: This is done by sending a request to the system to receive the coordinates, of the tracked car, from the GPS satellites. Then, read the stored coordinates, on the map on the mobile application.
- request data from the main server: send requests to the MCU through the cloud server and the GSM module to request the location and status if the

car.

3.6.2 Main server functions:

There is a specific mechanism the main server follows to continue the system build.

And this mechanism consists of the following main functions:

- Receive data: each data or coordinates that is posted from the GSM module will be received on the server side, through the HTTP connection mechanism.
- Plot cars movement on map: each time new coordinates are received, it will be processed to be represented accurately on tracked geographic map in our system, then update the current tracking car's path for the car.
- View cars history: the system must be able to save the car's registration data, and store the history for the car over any time they previously connected to the system, and clear way to retrieve this history.
- Waiting for new connection: during all process of the system, it must still able to receive the active car's coordinates, and transmit them to the mobile application to be displayed on the map, without any reflection in the processes sequence or connections.

4

CHAPTER FOUR

DETAILED DESIGN

4.1 Overview

4.2 System use cases

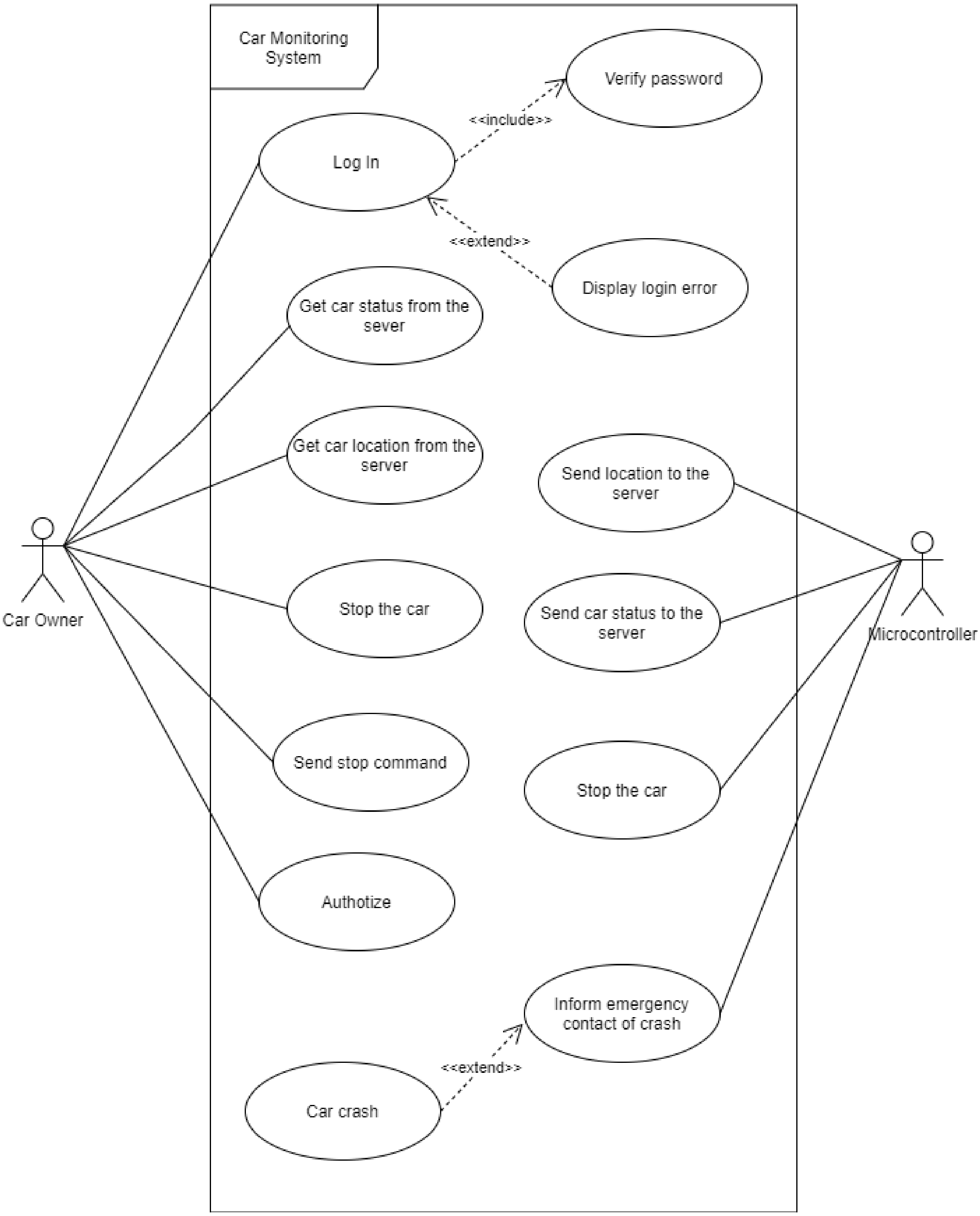
4.3 System sequence diagram

4.1 Overview

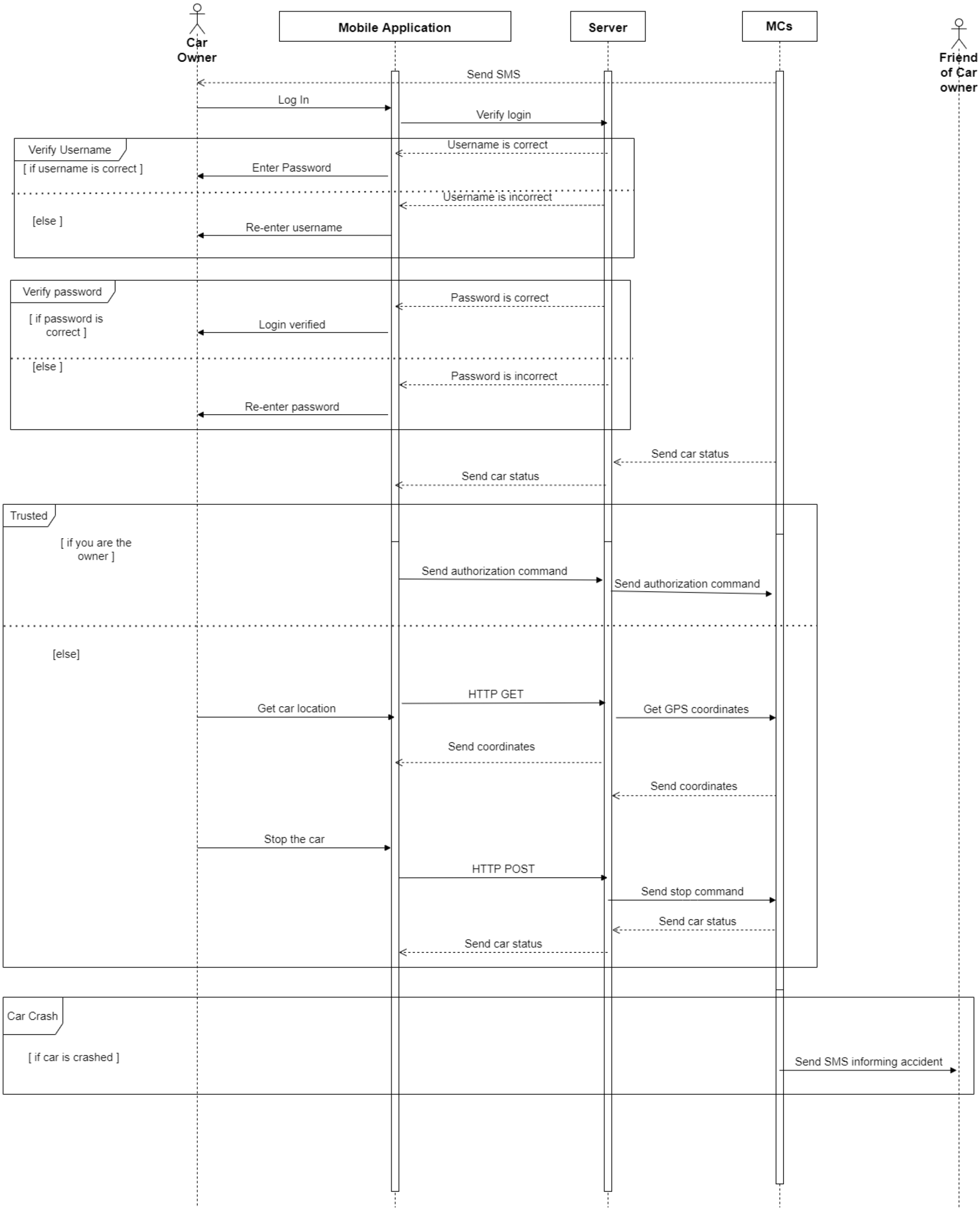
In this chapter, the software engineering details will be mentioned and described, including UML model diagram for both mobile and server application software classes, with implementation for these classes, then the system use case diagram will be described, also the sequence diagram for use cases will be mentioned.

4.2 System use cases

The use case diagram represents the user's interaction with the system showing the relation between the user and use cases. hence, in use case diagram of "Car monitoring" we see the user interacts with authentication screen to login to the application, looks at the status field in the application that shows car status, get location on a map and sending commands to stop the car. From the microcontroller view, the microcontroller can send car status to the server through GSM and send car location (coming from the GPS) to the server via GSM.


Car Monitoring System
Verify password
<<include>>
Log In
<<extend>>
Display login error
Get car status from the sever
Get car location from the server
Send location to the server
Car Owner
Stop the car
Send car status to the server
Microcontroller
Send stop command
Stop the car
Authotize
Inform emergency contact of crash
<<extend>>
Car crash

4.3 System sequence diagram



Sequence Diagrams are interaction diagrams that detail how operations are carried out. They capture the interaction between objects in the context of a collaboration. the full description of the project is defined in the sequence diagram, first of all the user tries to login with specific user name and password, an authentication is performed then, if the user is authenticated, a control screen will appear to the user that allows him to control his car by sending stop command to the server to stop the car, see the car status either active or inactive, see static map showing current location or open interactive map that shows car's location. when the user sends a command then the sever receives this command, sends response to the user and performs a certain task for example: if the user presses on "Stop the car" button, an HTTP request is sent to the server with predetermined value that the microprocessor receives via the GSM and stops the engine of the car.

User log in the mobile application, the application verifies the e-mail and password if they are correct, it allows the user to access the control screen to monitor the car whether get the car location and status or stop the car else if the e-mail is wrong the application will display message "could not find user with this e-mail" else if the password was wrong the application will display message "Invalid password".

- Control screen contains:

  a- "Stop the car" button which sends stop command to the server that interrupts the microcontroller, then the microcontroller stops the engine.

  b- "Status" field that gets car status from the server, the server gets the status from the microcontroller via GSM it might be active or inactive.

c- Static map which is not interactive map so it just displays screenshot of the car location.

d- "Get current location" which sends a command to server, then the microcontroller gets the command from the server then gets coordinates from the GPS and send the coordinates to the server via GSM.

5

CHAPTER FIVE

SYSTEM TESTING

5.1 Overview

5.2 Sub-System Testing

5.3 Installation and preparing the system and testing Scenarios

5.1 Overview

In this chapter, the whole testing stage will be described, including a test of an entire interconnected set of components and software for the purpose of determining proper functions and achieving the desired goals of the system. We will talk about testing of each part of the system, describe the scenarios, represent each test and its errors, challenges and modifications, and also, we will figure out the error rate.

5.2 Sub-System Testing:

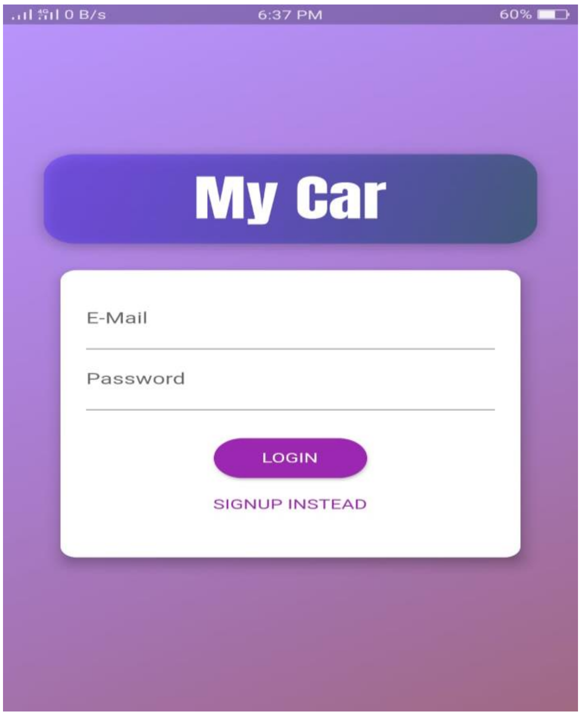
The main aim of this testing part is to test the main operations in the system. There are five main operations in the system. The first one is the getting the correct coordinates.

The second operation is posting data to the main server. The third is receiving the data and processing it to register car and plot it's coordinates to get its location. The last one is allowing access to history data base at any time. All the operations mentioned before are tested and meet the expectations.

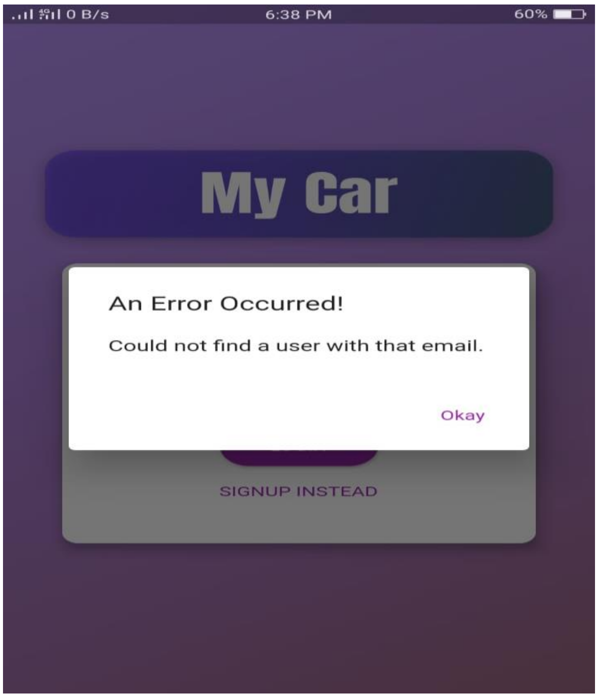
5.3 Installation and preparing the system and testing Scenarios

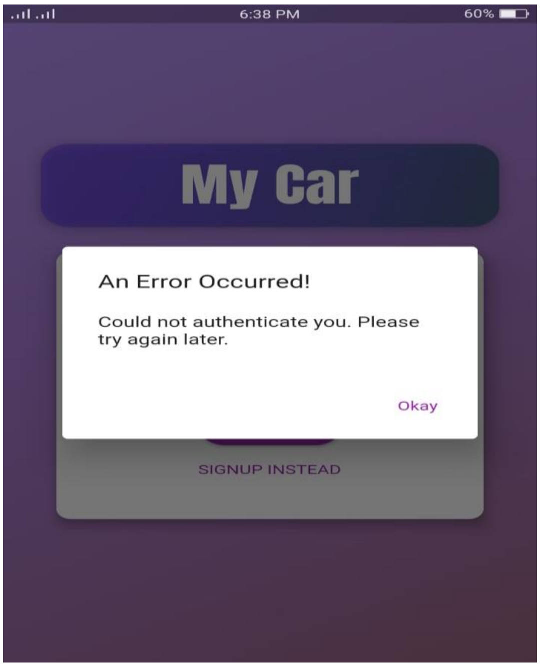
At this stage, all system parts must be ready to be tested at sides of mobile application at the car to be tracked and controlling it.

Mobile application:

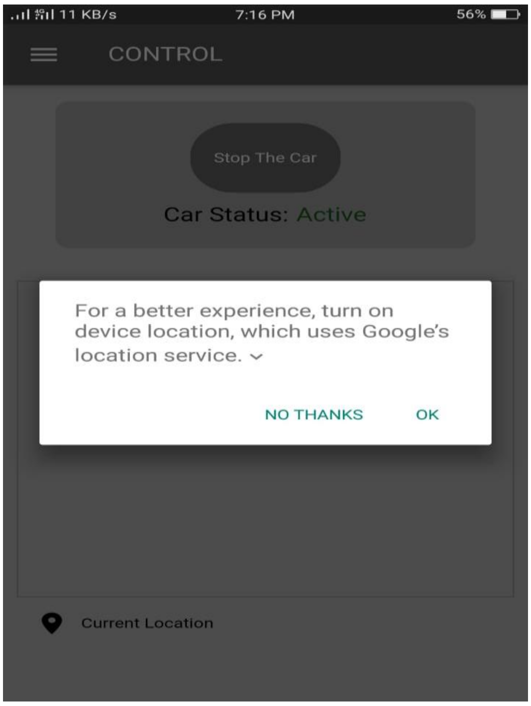


The authentication screen where the user can sign up or log in to the application

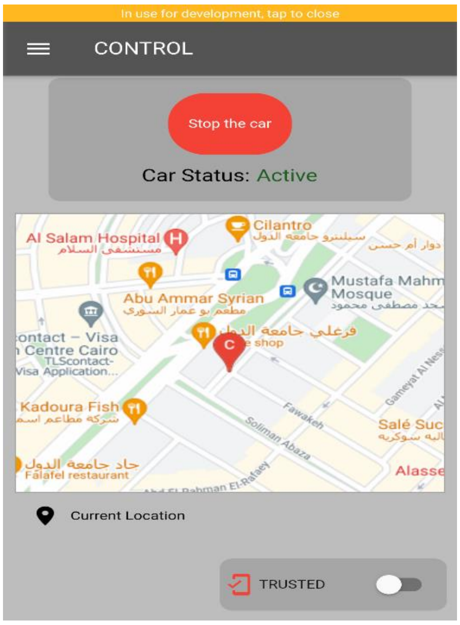




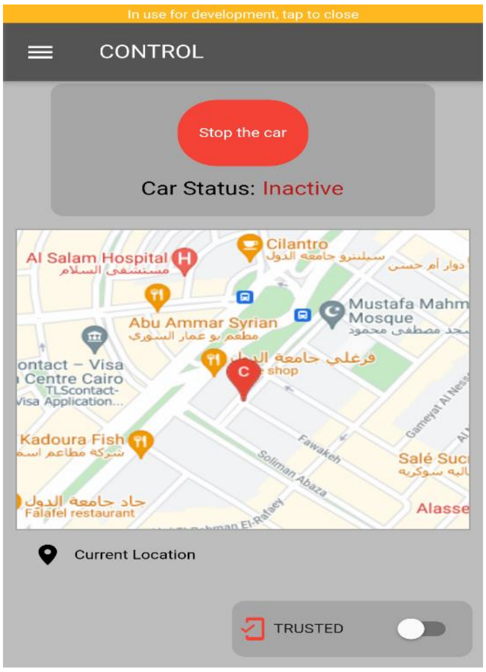
The server check if there is internet connection or not, also checks the validation of e-mail and password.



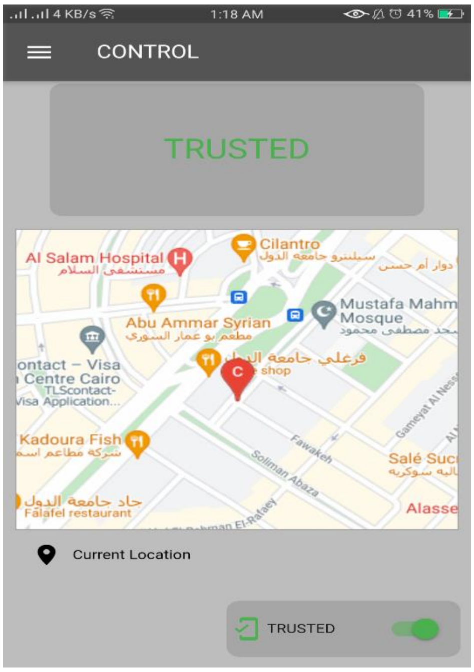
When user press on Current location button, the app requests permission to use GPS.



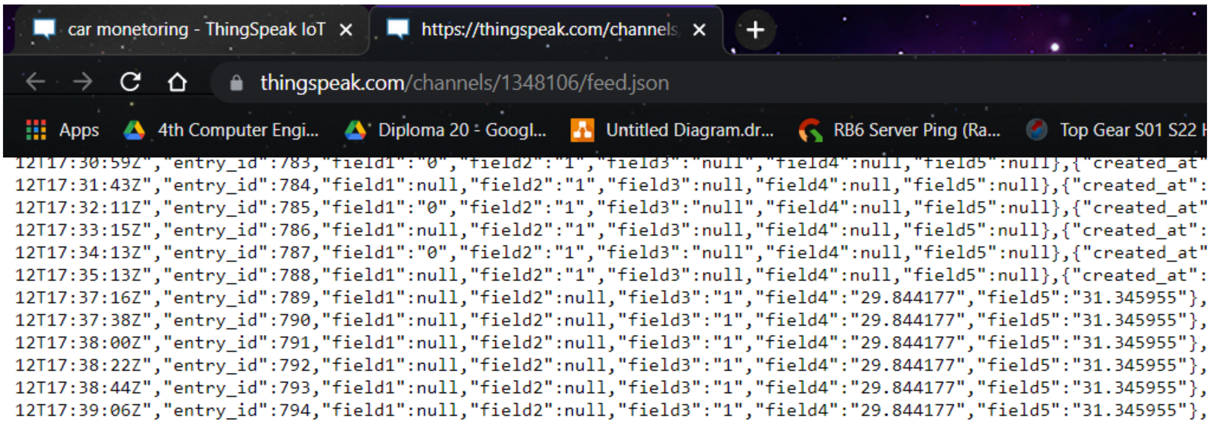
The user permits the application to use GPS then the application displays a static map (non-interactive map) with the current location.



When the GSM sends the status of the car, the application displays a text that contains the status of the car in a text field (active or inactive status).



If the owner of the car authorizes the access to the car, the mcu will stop sending the coordinates and status of the car to save resources.



When the GPS gets the coordinates, the GPS sends latitude and longitude of current location to the microcontroller that sends the coordinates to the server via the GSM to be stored in field4 (latitude) and field5 (longitude).

The microcontroller also sends the car status to the Thingspeak server in field3, (1 means the car is inactive, 0 means that car is active).

field1 carries the data for stopping the car, field2 is for the "trusted" task.

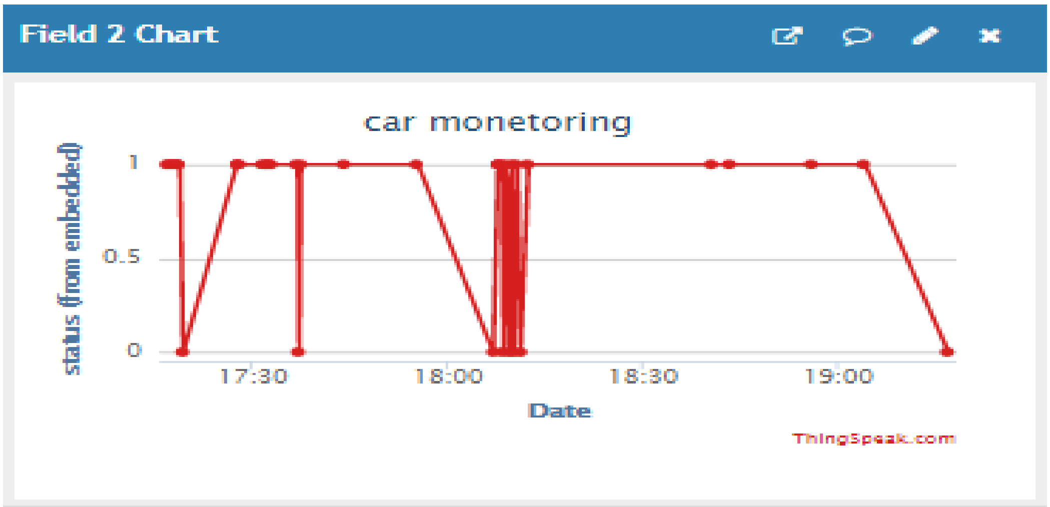


Chart shows the changing of car status (0 and 1)

6

CHAPTER SIX

CONCLUSION AND RECOMMENDATION

6.1 Overview

6.2 Conclusion and Achievement

6.3 Challenges

6.1 Overview

In this chapter, we will mention what we achieved in this project and the conclusion for all things that we have done, also we will talk about the challenges that we faced.

6.2 Conclusion and Achievement

Almost all the goals of our system have been achieved. In this section the main achievements of the system are discussed and the ways of achieving it.

- We built a mobile application to give the accurate location, of the travelling car and send it to the main server.
- We built an accessible database for the car's information and travelling history, that can be retrieved any time.

6.3 Challenges

In this project we have faced some critical points that still need a solution for them. Such as: - the GPS module needs a very clear view of the sky to get a cold fix and we've face some difficulties getting good satellite coverage. - GSM