****

Faculty of Engineering and Technology

Electrical and Computer Engineering Department

Graduation Project

ENCS530

Second Semester 2019/2020

**DRIVER ASSISTANCE SYSTEM**

**(AMAN SYSTEM)**

Prepared by

|  |  |
| --- | --- |
| Renad Khaled | 1151536 |
| Aseel Khader | 1152392 |
| Lina Anqawi | 1153280 |

Supervised by

Dr. Abdalkarim Awad

A Graduation Project submitted to the Electrical and Computer Engineering Department in partial fulfilment of the

requirements for the degree of B.Sc. in Computer Engineering

Birzeit University

May– 2020

## Abstract

The Driver Assistance System (DAS) is a hardware-software system that aims to assist car drivers in their daily commute. As the number of cars has increased greatly in recent years, the rate of an accident to occur has increased. Thus, the death toll due to car accidents has increased. Also, various studies proved that drowsiness and fatigue of the driver has been one of the major causes of these road accidents. Car companies adopted driver assistance systems based on drowsiness detection systems, Anti-Lock Braking System, Adaptive Cruise Control System, and many others that aims to assist the driver helping him/her to achieve a safe drive and alerting them before a disaster occurs. However, these different systems may not corporate accurately with the driver, as each driver has his own unique form of running his daily commute. A driver assistance system is introduced in this project, which is based on unique attributes for each driver. Such attributes include the driver’s heartbeats, focus, behavior as eye blinking, yawning and eye closure. These attributes will be considerate to introduce an enhanced and a compatible drowsiness detection system. This system uses three different approaches and techniques that depends on sensors as PPG sensor, camera and mobile application. These three subsystems of the drowsiness detection system aim to reduce the probability of causing accidents in addition to saving drivers’, children’s and all passengers’ lives and attempts to make the vehicle driver achieve a safe drive as much as possible.

In ENCS530, the drowsiness detection system with alarming was implemented using three techniques. These techniques are analyzing physiological signals as heartbeats of the driver using a pulse sensor, image processing techniques for monitoring the driver’s behavior as eye blinking, yawning, and eye closure using a camera, in addition to detecting the concentration of the driver through an android mobile application. This application interacts with the driver to check that the driver is awake by asking the driver some simple mathematical equations and waiting for the correct answers. If the driver answers a wrong answer the application will turn on an alarm.

## المستخلص

أنظمة مساعدة السائق (DAS) هي أنظمة صممت خصيصاً لتساعد السائق خلال رحلاته وتنقلاته اليومية, حيث تشمل في بنائها المكونات البرمجية (Software) والمواد المادية .(Hardware) نلاحظ في السنوات الأخيرة ازدياد أعداد السيارات بشكل كبير الأمر المرافق لزيادة الحاصلة في معدل وقوع الحوادث وعدد القتلى الناتج عنها.

هناك العديد من الدراسات التي أجريت بهدف معرفة أسباب هذه الحوادث لتتمكن البشرية من تلافي خسارة هذا الكم الهائل من الأرواح التي تزهق نتيجة للحوادث الناجمة عن الأخطاء البشرية. أثبتت الدراسات أن النعاس وتعب السائق هو أحد الأسباب الرئيسية وراء هذه الحوادث.

قامت العديد من السيارات باعتماد أنظمة مساعدة للسائق تعتمد على أنظمة اكتشاف النعاس، ونظام الفرامل المانعة للانغلاق، ونظام التحكم التكيفي في السرعة، والعديد من الأنظمة الأخرى التي تهدف إلى مساعدة السائق على القيادة الآمنة وتنبيهه قبل وقوع الكارثة. ومع ذلك، هذه الأنظمة المختلفة لا تخلو من الأخطاء ويمكن ان لا تتعامل بدقة مع السائق، حيث أن كل سائق لديه أسلوبه الفريد الخاص بإدارة تنقله اليومي كيفما يرغب.

هنا في هذا المشروع قمنا ببناء نظام مساعد السائق حيث يستند إلى سمات فريدة لكل سائق بعينه. تشمل هذه السمات نبضات القلب، والتركيز، والسلوك الذي يقوم به السائق عند نعاسه مثل التثاؤب، وإغماض العينين. تم العمل عليها جميعها في نظام أمان لضمان الحصول على نظام اكتشاف النعاس محسن ومتوافق. يستخدم هذا النظام ثلاثة أساليب وتقنيات مختلفة هي الاعتماد على المستشعرات مثل: مستشعر نبضات القلب، الكاميرا، وتطبيق للأجهزة المحمولة. تهدف هذه الأنظمة الفرعية الثلاثة لنظام الكشف عن النعاس إلى تقليل احتمالية وقوع حوادث بالإضافة الى محاولة إبقاء السائق صاحيا ومنتبها أثناء قيادته وذلك لضمان إنقاذ حياته وحياة الركاب الآخرين وتحقيق قيادة آمنة قدر الإمكان.

في ENCS530 تم بناء نظام اكتشاف النعاس مدمج إليه إنذار يعمل عند التأكد بأن السائق أصبح نعساً. يشمل هذا النظام في تركيبه العديد من التقنيات منها ما يقوم بتحليل الإشارات الفسيولوجية كضربات قلب السائق باستخدام مستشعر النبض، و أخرى تشمل معالجة الصور لمراقبة سلوك السائق أثناء إغماض عينيه، وفتح فمه لتثاؤب. تم تدعيم هذا النظام ب Android App يعمل على كشف مدى تركيز السائق وانتباهه.

Table of Contents

[Abstract 2](#_Toc41770080)

[المستخلص 3](#_Toc41770081)

[Chapter 1 Introduction 8](#_Toc41770082)

[1.1. Motivation and overview 8](#_Toc41770083)

[1.2. Problem Statement 9](#_Toc41770084)

[1.3. Project description 10](#_Toc41770085)

[1.4. Project outline 11](#_Toc41770086)

[Chapter 2 Related Work 12](#_Toc41770087)

[2.1. Drowsiness Detection System 12](#_Toc41770088)

[2.2. Accidents Detection and Alarming system 14](#_Toc41770089)

[2.3. Vehicle Heatstroke System 17](#_Toc41770090)

[Chapter 3 Background 19](#_Toc41770091)

[3.1. Drowsiness Detection System 19](#_Toc41770092)

[3.2. Enhanced Driver Assistance System 22](#_Toc41770093)

[Chapter 4 Tools 23](#_Toc41770094)

[4.1. Hardware tools 23](#_Toc41770095)

[4.2. Software tools 24](#_Toc41770096)

[Chapter 5 System Design 25](#_Toc41770097)

[5.1. General Hardware Block Diagrams 25](#_Toc41770098)

[5.2. Overall System Flowchart 26](#_Toc41770099)

[5.3. Drowsiness Detection System 28](#_Toc41770100)

[5.4. Accidents Detection and Alarming system 31](#_Toc41770101)

[5.5. Vehicle Heatstroke Detection System: 32](#_Toc41770102)

[Chapter 6 Implementation 35](#_Toc41770103)

[6.1. Drowsiness diction using USB camera: 35](#_Toc41770104)

[6.2. Drowsiness detection using heart beats: 39](#_Toc41770105)

[6.3. Drowsiness detection using mobile application: 46](#_Toc41770106)

[Conclusion and future work 49](#_Toc41770107)

[References 50](#_Toc41770108)

List of Figures

[Figure ‎2‑1:Proposed drowsiness detection method 14](file:///D:\Documents\2nd%20semester\graduation%20project\final%20report\Gradguation_Final%2317_Report2020.docx#_Toc41770109)

[Figure ‎2‑2:Breaking with and without ABS 14](file:///D:\Documents\2nd%20semester\graduation%20project\final%20report\Gradguation_Final%2317_Report2020.docx#_Toc41770110)

[Figure ‎2‑3:Schematic o Intelligent Cruise Control 15](#_Toc41770111)

[Figure ‎2‑4: ACS system 15](#_Toc41770112)

[Figure ‎2‑5:Node layout 16](file:///D:\Documents\2nd%20semester\graduation%20project\final%20report\Gradguation_Final%2317_Report2020.docx#_Toc41770113)

[Figure ‎2‑6: Sending data by cloud 17](#_Toc41770114)

[Figure ‎3‑1:Plot of PPG signal after performing 1st-order differential operation 20](file:///D:\Documents\2nd%20semester\graduation%20project\final%20report\Gradguation_Final%2317_Report2020.docx#_Toc41770115)

[Figure ‎3‑2: Power spectrum of LF/HF. 20](#_Toc41770116)

[Figure ‎3‑3:68 point of FLP Algorithm 21](#_Toc41770117)

[Figure ‎3‑4:EAR value for eye close/open 21](#_Toc41770118)

[Figure ‎3‑5:YLD 22](#_Toc41770119)

[Figure ‎4‑1:Raspberry Pi Zero Microcontroller 23](#_Toc41770120)

[Figure ‎4‑2:Pulse sensor 23](#_Toc41770121)

[Figure ‎4‑3: Arduino Uno Microcontroller 24](#_Toc41770122)

[Figure ‎4‑4:USB Camera 24](#_Toc41770123)

[Figure ‎5‑1:General system block diagram 25](file:///D:\Documents\2nd%20semester\graduation%20project\final%20report\Gradguation_Final%2317_Report2020.docx#_Toc41770124)

[Figure ‎5‑2: AMAN hardware block diagram 26](file:///D:\Documents\2nd%20semester\graduation%20project\final%20report\Gradguation_Final%2317_Report2020.docx#_Toc41770125)

[Figure ‎5‑3:Overall system’s flowchart 27](file:///D:\Documents\2nd%20semester\graduation%20project\final%20report\Gradguation_Final%2317_Report2020.docx#_Toc41770126)

[Figure ‎5‑4:Drowsiness detection system block diagram 28](file:///D:\Documents\2nd%20semester\graduation%20project\final%20report\Gradguation_Final%2317_Report2020.docx#_Toc41770127)

[Figure ‎5‑5:Drowsiness detection system flowchart 29](file:///D:\Documents\2nd%20semester\graduation%20project\final%20report\Gradguation_Final%2317_Report2020.docx#_Toc41770128)

[Figure ‎5‑6:Drowsiness Detection Algorithm by using Camera 30](#_Toc41770129)

[Figure ‎5‑7:Drowsiness detection system hardware implementation 31](file:///D:\Documents\2nd%20semester\graduation%20project\final%20report\Gradguation_Final%2317_Report2020.docx#_Toc41770130)

[Figure ‎5‑8: Accident detection algorithm 31](file:///D:\Documents\2nd%20semester\graduation%20project\final%20report\Gradguation_Final%2317_Report2020.docx#_Toc41770131)

[Figure ‎5‑10:Block diagram of Vehicle Heat Stroke Detection System 32](file:///D:\Documents\2nd%20semester\graduation%20project\final%20report\Gradguation_Final%2317_Report2020.docx#_Toc41770132)

[Figure ‎5‑11:flow chart of Vehicle Heat Stroke Detection System 33](file:///D:\Documents\2nd%20semester\graduation%20project\final%20report\Gradguation_Final%2317_Report2020.docx#_Toc41770133)

[Figure ‎5‑12:circuit of vehicle heat stroke system 34](#_Toc41770134)

[Figure ‎6‑1:connection, booting 36](#_Toc41770135)

[Figure ‎6‑2:Commands Part 1 37](#_Toc41770136)

[Figure ‎6‑3:Commands Part 2 37](#_Toc41770137)

[Figure ‎6‑4:Commands Part 3 37](#_Toc41770138)

[Figure ‎6‑5:Commands Part 4 37](#_Toc41770139)

[Figure ‎6‑6:Algorithms Code 38](#_Toc41770140)

[Figure ‎6‑7:Result 38](#_Toc41770141)

[Figure ‎6‑8:sensor attached to the object's wrist 39](#_Toc41770142)

[Figure ‎6‑9 from serial to csv file 40](#_Toc41770143)

[Figure ‎6‑10: Raw heart signal from Pulse sensor 40](#_Toc41770144)

[Figure ‎6‑11 preprocessing and filtering signals 41](#_Toc41770145)

[Figure ‎6‑12: Calculating sampling rate 41](#_Toc41770146)

[Figure ‎6‑13:Original and filtered heart signal 41](#_Toc41770147)

[Figure ‎6‑14 process function from heartpy library 42](#_Toc41770148)

[Figure ‎6‑15 Accepted and rejected peaks 42](file:///D:\Documents\2nd%20semester\graduation%20project\final%20report\Gradguation_Final%2317_Report2020.docx#_Toc41770149)

[Figure ‎6‑16: frequency spectrum of heart rate variability 43](file:///D:\Documents\2nd%20semester\graduation%20project\final%20report\Gradguation_Final%2317_Report2020.docx#_Toc41770150)

[Figure ‎6‑17:Plot of original signal and interpolated 43](file:///D:\Documents\2nd%20semester\graduation%20project\final%20report\Gradguation_Final%2317_Report2020.docx#_Toc41770151)

[Figure ‎6‑18: calculating LF and HF 43](#_Toc41770152)

[Figure ‎6‑19 BPM results when drowsy and alert for each object 44](#_Toc41770153)

[Figure ‎6‑20 LF/HF plots showing differences when drowsy and alert 45](#_Toc41770154)

[Figure ‎6‑21:Frequency spectrum of HR 45](#_Toc41770155)

[Figure ‎6‑22: Text to speech code 46](#_Toc41770156)

[Figure ‎6‑23: SetOnTouchListener 47](#_Toc41770157)

[Figure ‎6‑24: OnResults method 47](#_Toc41770158)

[Figure ‎6‑25: Interaction between driver and application 48](#_Toc41770159)

List of Tables

[Table ‎2‑1:Summary result of sensors evaluation 17](file:///D:\Documents\2nd%20semester\graduation%20project\final%20report\Gradguation_Final%2317_Report2020.docx#_Toc41770160)

**List of Abbreviationsa**

|  |  |
| --- | --- |
| ABS | Anti-Lock Braking System |
| ACC | Adaptive Cruise Control (ACC) |
| ACS | Avoidance Collision System |
| ADAS | Advanced Driver Assistance System |
| ECG | Electrocardiogram |
| HL | High frequency |
| HR | Heart rate |
| HRV | Heart rate variability |
| I2C | Inter-integrated circuit |
| IR | Infrared radiation |
| LF | Low frequency |
| LIDAR | Light Detection and Ranging |
| PPG | Photoplethysmography |
| RX | Receiver pins |
| SCL | Serial clock line |
| SDL | Serial data line |
| SVM | Support vector machine |
| TX | Transmitter pins |
| UART | Universal receiver transmitter |
| USART  EAR  YLD | Universal asynchronous receiver transmitter  Eye aspect ratio  Yawing lip distance |
|  |  |

List of Equations

[Equation ‎3‑1:EAR 21](#_Toc41770161)

[Equation ‎3‑2:YLD 22](#_Toc41770162)

# Introduction

Vehicle deaths and injuries problem is a problem that the knowledge of its magnitude, causes, and effects should be known, considered, and never marginalized. Driving while fatigued and drowsy, are considered very dangerous as it is one of the main causes of these accidents. Vehicle and road accidents are not necessarily caused by a crash as they can be non-crash accidents. Recent advanced technologies, aiming to assist drivers and help them to achieve a safe drive and commute for the driver and all passengers in a vehicle, have greatly contributed to making a remarkable decrease in accidents caused by vehicles. However, still, the number of deaths caused by these accidents remains high according to national international statistics. As according to the World Health Organization, deaths caused by road traffic reached 1.35 million approximately annually [1]. world Moving toward finding a real effective solution is still the main challenge that should not be ignored or dismissed. Instead, it should be a high priority and a great responsibility.

## Motivation and overview

For many years, the problem of vehicles accidents crashing and non-crashing ones has been considered as a fundamental and a serious issue globally, and this problem is expected to continue in the foreseeable future if no serious actions and considerable changes related to the causes of these accidents were took or applied. According to the World Health Organization (WHO) the number of annual road traffic deaths remains unacceptably high, as it has reached 1.35 million worldwide, it highlighted or considered traffic deaths as leading killers of people especially people aged 5-29 years [1]. In general, vehicle accidents, injuries, and deaths can occur and cause innumerable reasons, as they can be results of drowsy driving, speeding, driving under the influence of drugs or alcohol, children left in a hot car, etc. Accidents caused by drowsiness while driving are considered as one of the main causes of vehicle accidents. It is also considered as dangerous as drunk driving. There are too many factors that causes drowsiness while driving as driving for a long time, long distances, lack of sleep, etc.

The concept of Driver Assistance systems, which is also can be referred to as the Advanced Driver Assistance System (ADAS) will be discussed, studied, analyzed, and used to move toward finding a satisfiable solution that ensures a safe drive to protect the driver and passengers in the vehicle. In order to achieve that, we proposed in the previous course ENCS520 a big compatible system that consists of three different subsystems of ADAS. These systems were, drowsiness detection system, heatstroke detection system and accident detection system. But here, in this course ENCS530, we will introduce an implemented compatible system for drowsiness detection while driving based on three different techniques. The drowsiness detection system aims to detect if a driver is drowsy and alerts him/her at the right moment before a disaster occurs. This system depends on the driver’s heartbeat rate measures by a specific sensor called photoplethysmogram (PPG), the driver’s behavior while driving as eye blinking, yawning, and eye closure which is monitored by a camera, in addition to an interaction between the user and a mobile application.

In general, ADAS has been introduced widely as a promising solution trying to contribute to preventing, limiting, and reducing the devastating results of accidents and problems caused by vehicles. The drowsiness detection system in one of the ADAS systems that rely on information provided by some sensors, cameras and other tools. Fortunately, driver assistance systems have really contributed to a remarkable decrease in traffic accidents, especially the ones that promote comfort and economy, as they help to avoid these accidents by assisting drivers and offer support technologies to help them in their driving task continuously.

In the following sections, we will see extensively, an overview about the drowsiness detection system, its implementation and results.

## Problem Statement

Despite that existing current ADAS systems have made really noticeable progress in recent years, they are still facing some challenges that make them far from meeting all the requirements demanded or needed varying between price, safety, complexity, and efficiency, which makes it very limited. Although ADAS can really solve or at least reduce the probability of a problem, accident, or disaster to occur, we need to shed light on three severe problems that we are aiming to solve them in one compatible system.

Accidents caused by drowsy driving, which is a dangerous combination of sleepiness and driving that happens when the driver has not slept enough or too tired to remain alert, affects his/her ability to drive safely even if the driver doesn’t fall asleep. The U. S. National Highway Traffic Safety Administration (NHTSA) reports that 100,000 vehicle crashes are caused by drowsiness. In addition, NHTSA reported that more than 1,500 deaths are occurred per year [2]. The drowsiness detection system was proposed to solve this problem and alert the driver before getting into trouble. The system we will focus on is the one which detects drowsiness by taking the decreasing heart rate measurements as (Beats Per Minute) BPM and LF/HF ratios as a sign of drowsiness by the PPG sensor. Also, drowsiness while driving will be detected through a camera that monitors the driver’s behavior. In addition to an application called AMAN.

The drowsiness or fatigue while driving problem will be solved in one low-cost, enhanced and effective system that combines three subsystems, using three different techniques that will be all described extensively in the upcoming sections.

## Project description

The goal of this project is to create and develop and drowsiness detection system that functions with an alarming system. This system is integrated with a mobile application in which we called the application **AMAN**.

The drowsiness detection system consists of three subsystems; detection through camera, heartbeat bracelet, and the mobile application. Each subsystem is desired to serve a different purpose. However, these three subsystems aim to achieve the same goal once they are connected together.

The aim of the system is to detect early signs of drowsiness on the driver during the driving session and deal with the situation efficiently to avoid any sort of accidents or mistakes.

The first subsystem, the camera detection, aims to detect drowsiness through the use of the camera. The camera will constantly video stream and resample the face as multiple points. The system shall evaluate the expressions of the driver and use the collected data to determine if the driver is in a drowsiness state. As a result, there will be an alarm created by AMANapplication that will occur if the drowsiness has been detected.

The second subsystem, the heartbeat bracelet, aims to detect drowsiness through a worn bracelet. The bracelet collects multiple heartbeat data constantly and analyzes its signals. If the analyzed signals suggest that the person is in a drowsiness state, there will be an alarm created by AMANapplication.

In rare cases, the mentioned subsystems can alarm about a drowsiness detection, while the drowsiness isn’t occurred at all. To avoid such cases, the AMAN applicationwill provide an easy mathematical question to ensure that the driver is focused and conscious; and this is the third subsystem that aims to serve as an interaction unit between the driver and the application.

## Project outline

This report is structured in 6 chapters so far. Each chapter focuses on a specific section of the overall project. Chapter 1 provides a general introduction, motivation and overview, problem statement, and a brief description of the project with the main concepts. Chapter 2 presents the related work, which provides some existing related examples and work. Chapter 3 displays the background, which supplies a detailed description of each separate system of the three mentioned systems. In addition, it provided description of the enhanced compatible system. Chapter 4 manifests the tools used in the project, including the hardware and software tools. Chapter 5 stores the hardware design and implementation used to build the overall system in detail illustrated by a flowchart, hardware block diagrams and hardware implementations. The last chapter which is the implementation chapter, that has all implementation of drowsiness detection using camera, heartbeats and mobile application.

# Related Work

Advanced driver assistance systems are designed to assist drivers for safe and better driving. These systems fall under these categories:

* **Adaptive systems** that adapt according to input from the around environment.
* **Automatic systems** that perform certain functions that the driver cannot perform without causing damage,
* **Monitoring systems** that use sensors to monitor the nearby area or drive the vehicle and estimate if correction is wanted.
* **Warning systems** that alert the driver to problems found or likely to exist in the driver's drive or in the driving of others.

Many advanced driver assistance systems are subsystems or combinations of these categories, in this chapter, we will talk about some of the advanced systems that fall under these categories, in this report our effort focused on the drowsiness detection system.

## Drowsiness Detection System

Drowsy driving causes for at least 100,000 vehicle accidents annually, resulting in approximately 40,000 nonfatal injuries and 1,550 mortalities as the U.S. National Highway Traffic Safety Administration (NHTSA) reported [3]. As a result, many efforts have been reported and systems were developed in order to detect drowsiness and sleepiness while driving. Drowsiness detection systems approaches can be categorized into three categories [4].

* **Imaging processing techniques**: this approach is based monitoring the driver’s behavior as eye blinking, yawning and eye closure and other physical movements through cameras [5].
* **Physiological signal detection techniques:** This approach is to use sensors as Photoplethysmography (PPG), electrooculography (EOG), and electrocardiogram (ECG) to measure physiological changes of drivers through body signals [4].
* **Vehicle-based metrics:** these metrics are based on vehicles signals **as** steering wheel angle, pressure on acceleration pedal, and lane position signals [5].

The two systems below show different approaches in detecting drowsiness from two categories. The first one uses imaging processing techniques and the other uses physiological signal detection techniques.

A computer vision-based system to detect driver’s drowsiness was proposed. It takes a video input by a camera and then detects the driver’s face and eyes. Iris region extraction, morphological operations, and finally PERCLOS for left and right iris estimation was done. PERCLOS measure is the percentage of total time that eye is closed during a time interval. This approach is illustrated in the flowchart in figure 2-1. Three threshold values of PERCLOS were used P60, P70, and P80. The results of the system showed that PERCLOS value when the driver is drowsy is higher than when the driver is not, So, it can be a good way to detect drowsiness while driving. But a limitation of this system is when the face is unrecognized due to distraction or tendency of driver to rotate his face sideways [5].

The other system is the Heart Beat Based Drowsiness Detection System which uses body signals and the change of the human heart rate in the case of sleepiness or drowsiness of the driver in order to detect drowsiness. Heartbeat sensor called Photoplethysmography (PPG) which detects volumetric changes of blood using a simple optical technique was used for heart beat measurements. In general, it was found that when a human enters the relaxation phase, the heart rate will be reduced by 8 beats per minute approximately. So, in the first 5 seconds after wearing or placing the sensor of the driver’s finger the system will perform calibration by calculating the average heart rate. The average heart rate calculated will be displayed on a screen that detects that the calibration process is successful.  Then, the system will run and start taking heart rate in Real-time. The value of heart rate will then be compared with the average value. Drowsiness is detected if the heart rate is 8 beats below the average value. The system uses Arduino Nano and Odroid XU4 as the processing unit and an LCD to display the output. The system showed good results and accepted accuracy of 96.52%, but has some limitations. The sensor need not to be moved from the finger to give good results so it should be attached properly. Adding a physiological sensor as ECG or other is suggested in order to improve the accuracy of drowsiness detection [6].

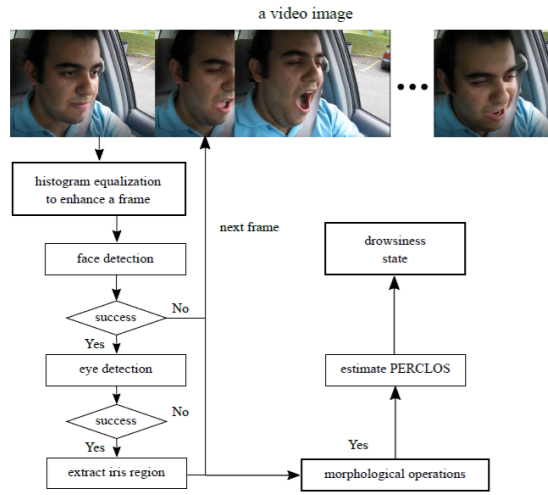


Figure ‑:Proposed drowsiness detection method

## Accidents Detection and Alarming system

There are several methods that have been applied in the last decades to detect, avoid, and alarm if the car accident occurred, like ACC, ACS, and ABS.

* **Anti-Lock Braking System (ABS)**

The ABS is used to continuously apply optimal braking pressure for each wheel, so it will be enough to not lock the wheels, which ensures maintaining contact with the road surface traction especially in slippery surfaces. So, it gives the driver the ability to control the vehicle in an emergency situation in critical time. The figure 2-2 below shows the difference between breaking with and without ABS in cars [7].

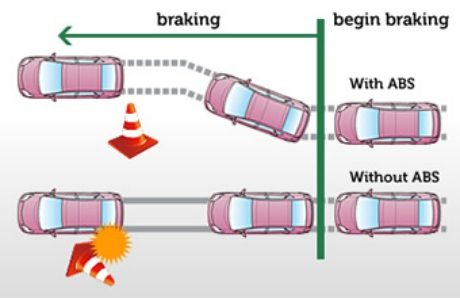


Figure ‑:Breaking with and without ABS

* **Adaptive Cruise Control (ACC)**

ACC is an automotive system that breaks the vehicle when it detects a car that is approaching another vehicle ahead and then accelerates when the traffic allows it to through a radar system. This is illustrated in the figure 2-3 below.

ACC system only provides some help to the driver but it doesn’t control anything in the car [8].

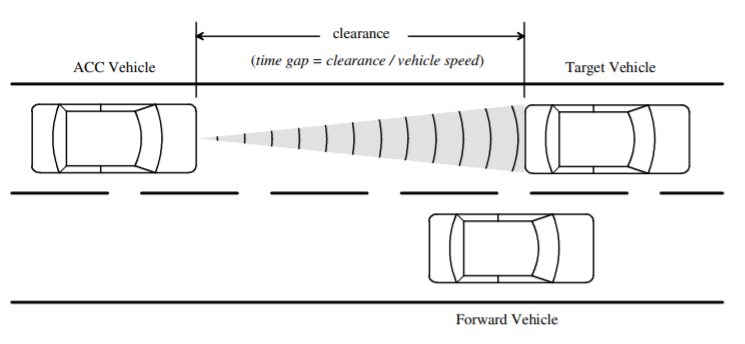


Figure ‑:Schematic o Intelligent Cruise Control

* **Avoidance Collision System (ACS)**

ACS is an automobile safety system designed to prevent or reduce the severity of a collision [9]. Once this system detects an accident, it immediately sends a warning to the driver or makes an action when there is no time for warning. So, it controls the car deviation and direction to prevent crashes. As shown in the figure 2-4 below.

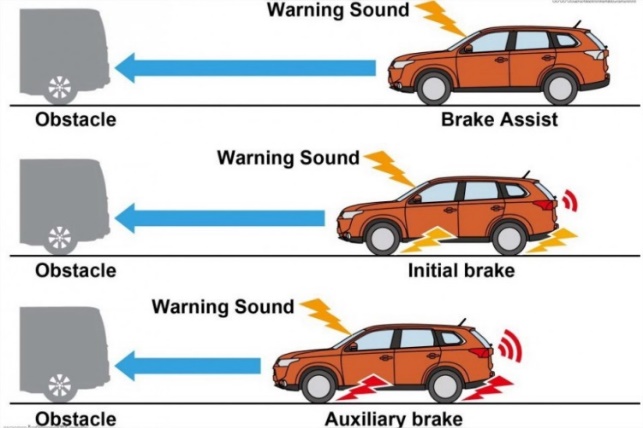


Figure ‑: ACS system

* **Other related systems**

Some papers tried to implement a real-time traffic accident detection system using a wireless sensor network and radiofrequency identification technologies [10]. It gets the data before the accident occurs and sends it as a signal to monitoring stations that tracks the location where the accident has occurred which will send an alert to the authorities concerned. the system designed in a way that gives the main objective of the RTTAD system which includes a speed sensor, microcontroller, RF module, RFID reader, and a status sensor as shown in the figure 2-5 below.

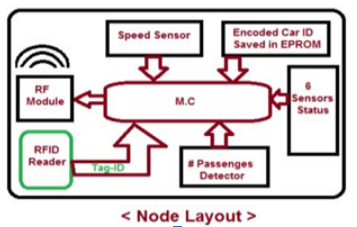


Figure ‑:Node layout

For example, the system collects the data and sends it to the right stations to give the immediate appropriate help. According to the collected data, the number of passengers in the vehicle will be counted so, it can use this data to know the number of ambulance cars needed for injured persons.

The system has great results, but the implementation is expensive since it has a router algorithm that will be developed in the router itself to detect if there is a packet that has been sent to any available street router which contains “help” message so that means an accident has occurred and then the router will send the message to the coordinator.

There is another paper that have the same concept but instead of dealing with routers it will deal with the concept of IOT-Internet Of Things- with an automatic cache response manufactured within the car, that use it to communicate with the driver using the cloud server which will be paid for the provider for an emergency [11].

This system depends on the operator that communicates with the driver, which will get the driver information and his situation that is already enrolled in, so the system will process this data and pass it to the accident detection algorithm which is installed in the Raspberry Pi that will control the next step. This will be done by sending data collected from the GPS location to the cloud using WIFI as shown in the figure 2-6 below, and then the cloud sends this data to the ambulance.

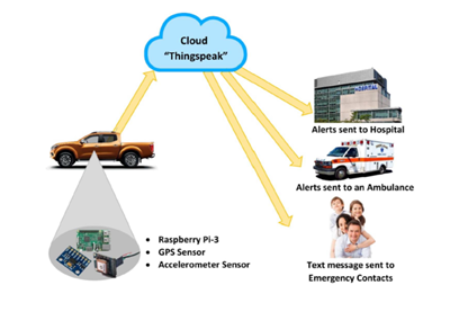


Figure ‑: Sending data by cloud

The system has good results since it communicates directly with emergency service and providing it with all the needed information about the driver condition, but at the same time, it is not flexible for all users to use the cloud only if he has a credential to access.

## Vehicle Heatstroke System

Recently, many research and studies have been carried out to save children's lives while in closed cars and to ensure that they are not exposed to a thermal shock that leads to their death, as researchers have built and develop reliable systems

One of these systems is Development of an Automatic Vehicular Heatstroke Detection System, this system detects a human presence inside the closed vehicle and constantly checks the temperature inside the vehicle, If the temperature reaches a dangerous temperature that puts the child's life at risk, the system will alert the parent.

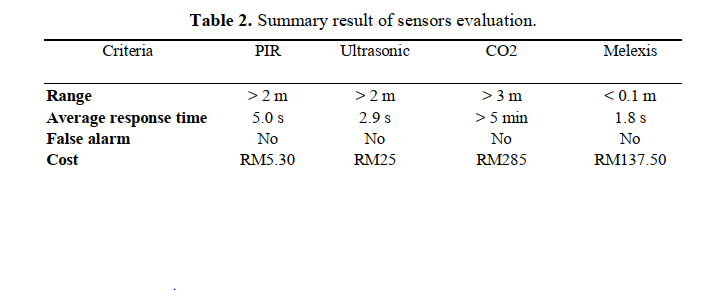
The system consists of three basic modules Sensors module, processor, and response module. To build a sensors module, researchers compared many sensors that detect human presence to ensure the construction of a reliable high-resolution system, and eventually the ultrasonic sensor and motion sensor were selected to achieve the best results in the two meters range show in the table 2-1 below [12].

Table ‑:Summary result of sensors evaluation

However, this system stops when it alerts parents to the seriousness of the child's situation and does not actually intervene to save the child's life, just the opposite of what the Affordable System for Alerting, Monitoring and Controlling Heat Stroke inside Vehicles does.

The Affordable System for Alerting, Monitoring and Controlling Heat Stroke inside Vehicles consist of three parts Monitoring unit, Coordinate unit, and Real-time control unit. They all work consistently for child safety. The monitor unit contains the sensing devices and controller. The controller collects data from sensors located inside the car to sense environmental conditions for monitoring the heatstroke's causes. The control unit will take effective action to overcome the heatstroke problem. The sensor device includes a temperature sensor, a presence detection system, a real-time clock, and a GPS receiver module. The coordinator unit will communicate data to the remote server, and this data will be available for specific end users [13].

# Background

This chapter gives the conceptual background for related topics for our system that works concurrently to get the best results. Our enhanced Driver Assistance System includes three subsystems of drowsiness detection that depends on each other:

* Drowsiness detection based on heart beats signals.
* Drowsiness detection using a camera.
* Drowsiness detection using the mobile application.

At the end of this chapter, we will include how our enhanced drowsiness detection will work together.

## Drowsiness Detection System

Drowsiness detection while driving is so important for the driver’s and passengers’ safety. As a result, many researches developed different techniques, technologies, approaches and systems to detect drowsiness. Once the driver is detected as sleepy or drowsy the driver is alerted in many ways, a buzzer is one of them. Using and analyzing physiological signals such as brain wave, heart rate, pulse rate, and respiration rate signals are considered as one of the most reliable approaches for drowsiness detection [3]. And the other approach is by using facial analysis like yawing and closing eyes that it will be extracted and to use it in detection drowsiness.

### Drowsiness detection based on heart signals

Several studies proved that there is a relation between heart rate (HR) and drowsiness. HR varies significantly between the drowsy state and alert state. So, it is a vital sign that ought to be investigated to detect drowsiness. Furthermore, other studies have confirmed that heart rate variability (HRV) based methods can recognize driver drowsiness. HRV is defined as the constant change of the interval between heart rate that is measured through special devices. Electrocardiography (ECG) is known as a preferred technology to derive heart rate and HRV information. But ECG can be highly affected by movements, contact or sweat. It is also uncomfortable to be used while driving, as it requires correct positioning, electrodes stuck on the driver’s body as well as a conductive gel on the electrodes [14]. As an alternative way of measuring HRV and to overcome ECG problems, Photoplethysmography (PPG) sensor can be used. In our system, HRV will be calculated by analyzing a time series of beat-to-beat intervals that are derived from a pulse wave signal measured via a pulse sensor that uses the PPG technology [15]. This sensor will be attached at the driver’s wrist in a bracelet form. PPG is a low-cost and noninvasive sensor that has no electrical interaction with human body. It uses an optical technique for sensing the cardiovascular blood volume pulse and makes measurements on the surface of the skin. It can be attached to finger, wrist, earlobe or other areas where it has contact with skin [16] [15]. PPG signals should be analyzed in time and frequency domains in order to extract features and compare drowsy and alert signals and then classify them with an appropriate classifier as support vector machine (SVM). It is preferred to take the first order differential operation of PPG signal in order to remove the artifacts caused by driver's movement which helps the extraction of peak-to-peak intervals of the signals [15]. The plots of PPG signals after performing first order differential operation can be shown in figure 11. HRV is usually grouped into three groups by means of FFT-based power spectrum density:

* **Very low frequency**: VLF: 0.003–0.04 Hz.
* **Low frequency**: LF: 0.04–0.15 Hz.
* **High frequency**: HF: 0.15–0.4 Hz.

The HRV power spectrum for drowsy and alert driving in figure 3-2. shows that the LF/HF ratio increases while drowsy driving.



Figure ‑:Plot of PPG signal after performing 1st-order differential operation

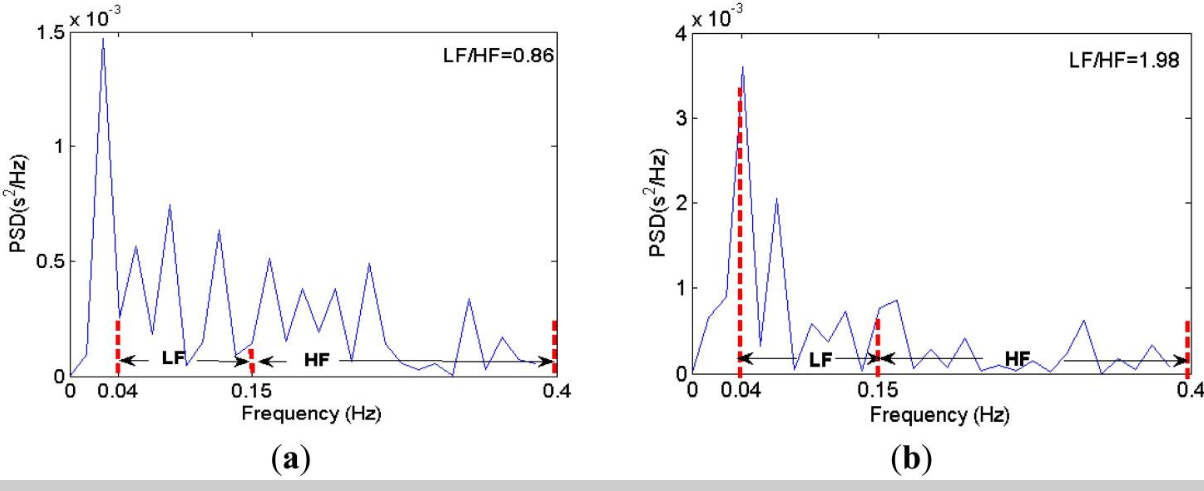


Figure ‑: Power spectrum of LF/HF.

1. *shows alert driving; (b) shows drowsy driving* [15]

### Drowsiness detection using camera

Facial expression will be extracted by using several Landmark classifiers which will estimate the EAR -Eye Aspect Ratio-, YLD - Yawing Lip Distance – so it can lead to detecting drowsiness and then send a notification or warning through the AMAN app which is already the driver installed it on his smart phone. The system will continuously check the driver's eye and the yawing state by using these algorithms:

* **Facial Landmark Prediction (FLP)Algorithm:**

Every human face has a 68-point spared on the whole face, so these points will be stored in a matrix, then pass it through this algorithm which will help to locate the eyes and the mouth position. As shown in figure 3-3.



Figure ‑:68 point of FLP Algorithm

* **Eye Aspect Ratio (EAR) Algorithm:**

Eye aspect ratio is the ratio between the vertical line and the horizontal one for the eye when the eyes are open the EAR ratio will be above of 0.2 for typical Middle east people as shown in equation 3-1.

Equation ‑:EAR

However, when the eyes are closed, the EAR will almost fall to 0.13. As shown in figure 3-4.

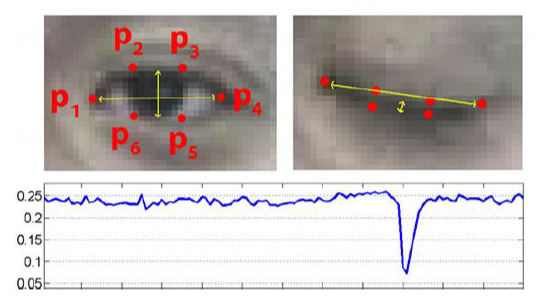


Figure ‑:EAR value for eye close/open

* **Yawing Lip Distance (YLD) Algorithm:**

YLD algorithm as shown in equation 3-2, calculates the distance between the top lip and lower lip when the distance is above 53 as shown in figure 3-5, which it indicates that the driver is yawing otherwise it will be either talking or silent.

Equation ‑:YLD

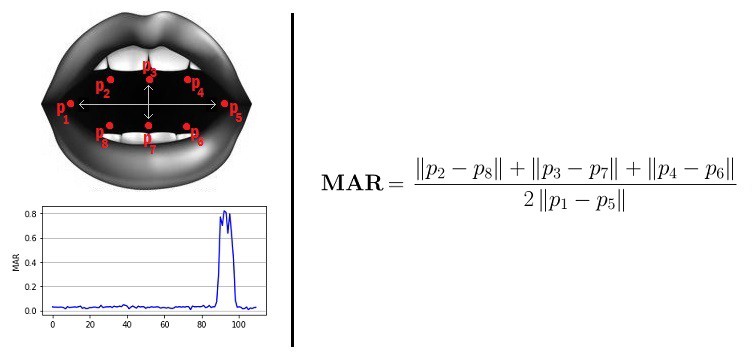


Figure ‑:YLD

## Enhanced Driver Assistance System

Our drowsiness detection system is made up of three subsystems drowsiness detection based on heat beats signals, drowsiness detection using a camera, and drowsiness detection using mobile Application. These systems work consistently together to enhance the safety of the driver and passengers. By monitoring and checking the output of the subsystems mentioned above, and activate the alerting unit when drowsiness occurs. Our enhanced system works completely separately from the body of the vehicle, as it is not a built-in system, this is to ensure reduced fault ratios as the system can be replaced at no additional cost.

# Tools

We are designing a system that aims to keep the driver and passengers in your vehicle safe by achieving the detection of the drowsiness through three different ways: by using a camera, heartbeats bracelet, and a mobile application. So, the overall system will need different simple, and various tools, hardware and software ones. These tools are listed below:

## Hardware tools

### Drowsiness Detection subsystem

* **Smart Phone**

Smartphones have a lot of useful capabilities and properties. Some of these capabilities can be used to implement drowsiness detection through an application.

* **Raspberry Pi Zero**

A mini microcontroller that will be attached to the bracelet of Drowsiness Detection System. Raspberry pi runs Raspbian operating system. WIFI and Bluetooth properties will be used in order to connect with other microcontrollers. Also, it will be used for data processing and other functions.



Figure ‑:Raspberry Pi Zero Microcontroller

* **Pulse sensor**

Pulse sensor is developed based on PPG techniques. PPG techniques uses a light-based technology in order to detect in living tissues, the amount of arterial blood volume changes.

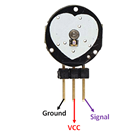


Figure ‑:Pulse sensor

* **Arduino**

This microcontroller can act or used as an ADC when it is wired with a microcontroller that cannot read analog values, as it has a built-in analog to digital convertor.



Figure ‑: Arduino Uno Microcontroller

* **USB Camera**

Rather than camera module with Raspberry Pi, we can use a standard USB webcam to take pictures and videos on the Raspberry Pi



Figure ‑:USB Camera

## Software tools

* **PyCharm (Python IDE)**

Pycharm is a development environment for Python which is integrated and supports different ways of stepping through the code. It will be used for processing and analyzing the HR signals.

* **Arduino IDE**

Arduino Ide is an open-source software tool it is used to write code and upload it to the board easily.

* **Android Studio**

is the official integrated development environment (IDE) for Google's Android operating system writing platform that makes it easy for developers to write source code, allows the developer to preview its application format to various screen measurements in real-time during development. [17]

# System Design

## General Hardware Block Diagrams

Figure 5-1 below, shows our system AMAN, that we proposed in the introduction of the graduation project. As mentioned before, AMAN consists of three main subsystems drowsiness detection system, accident detection, and heatstroke detection system. In the graduation project, we decided to focus on the drowsiness detection system and implement it.

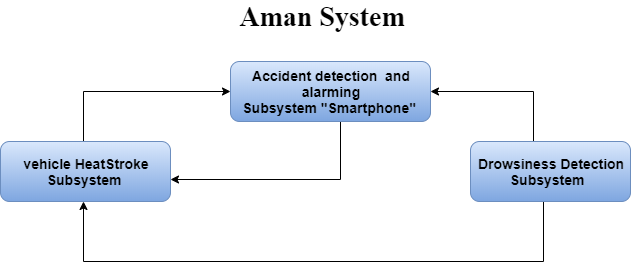


Figure ‑:General system block diagram

The general AMAN proposal contains three main microcontrollers; each has a role to play. The smartphone is the one that is used for sensing, analyzing, and detecting an accident. As shown in the block diagram below, the smartphone also receives data from the other two microcontrollers. It receives the driver’s status which is detected by the raspberry pi in the drowsiness detection system, and the child status in the car (if detected) in critical situations from the heatstroke system. Besides that, the heatstroke system microcontroller is used also to help the drowsiness and accident detection system to start operating at the right time. The right time mentioned here is the moment when the driver is detected to be in the car, it is detected through a continuous check of Bluetooth connection between the three microcontrollers. The microcontroller in the drowsiness detection system will mainly be used in order to process the drowsiness detection algorithm through analyzing signals received from the pulse sensor.

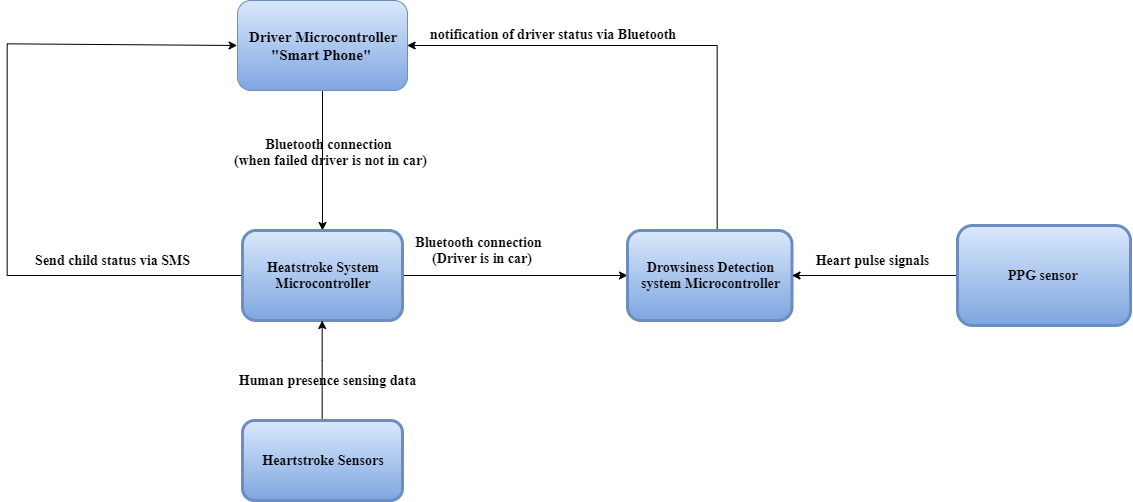


Figure ‑: AMAN hardware block diagram

## Overall System Flowchart

The flowchart shown below describes the overall system combining all the three subsystems, it shows the flow of each system starting from installing the mobile application. The user of AMAN should install the mobile app, fill his data and credentials through different ways then the system is ready to be used. As shown in the flowchart, drowsiness and accident detection systems start once the driver is detected to be inside the car while the heatstroke system starts once the driver leaves the car and no more Bluetooth connection is available with the smartphone.

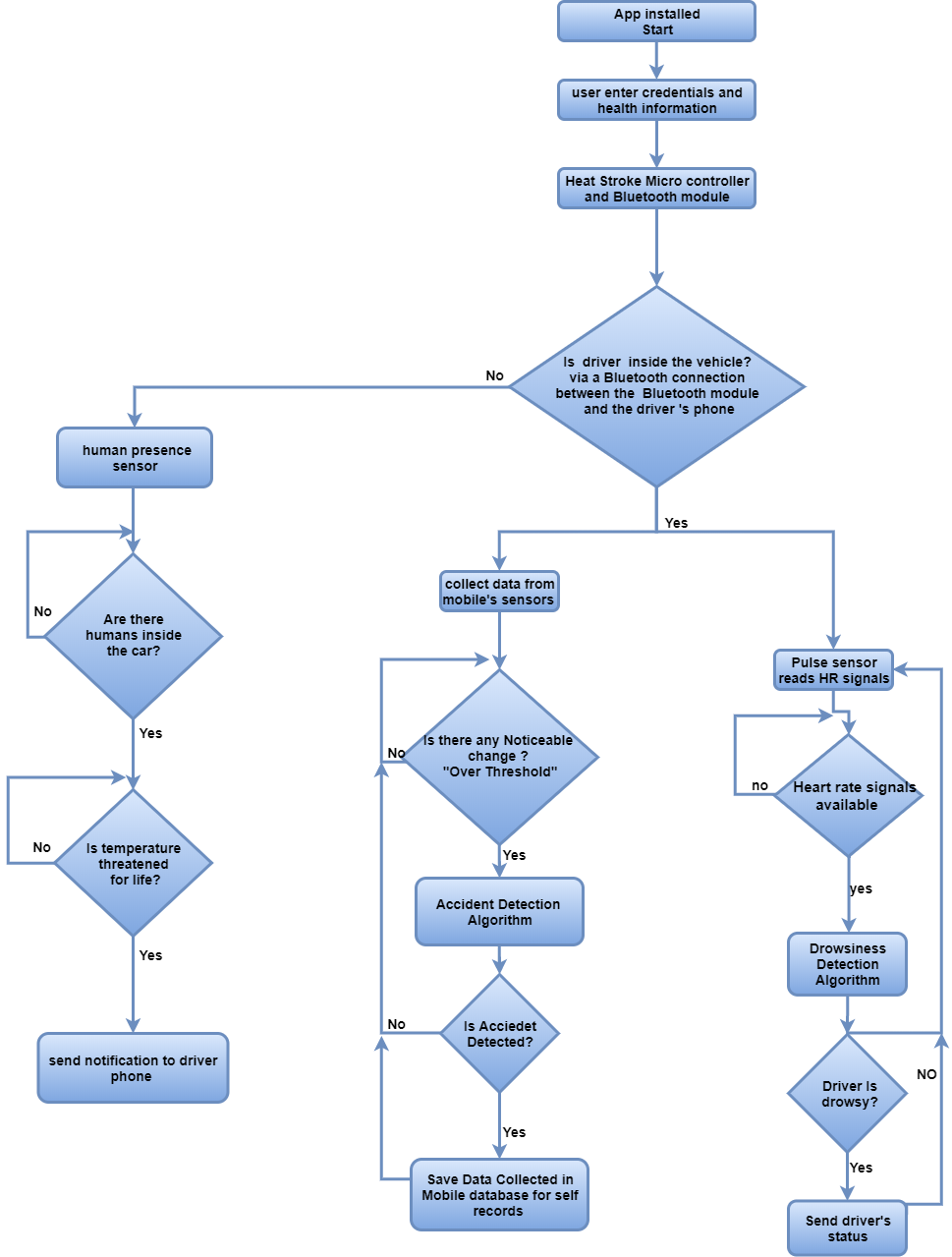


Figure ‑:Overall system’s flowchart

## Drowsiness Detection System

### Components of the system

Drowsiness Detection System consists mainly of the sensing part and the analyzing part. The sensing part senses the human pulse through the pulse sensor that uses PPG technology. The pulse sensor will be attached to the driver’s wrist. The analyzing part gets digital and analog data from the sensing part in order to detect drowsiness points and to detect the driver’s status drowsy or alert. An Arduino is used as an analog to digital converter (ADC) to convert analog signals to digital signals. The microcontroller in this system which is the raspberry pi zero will be the processing and analyzing part that at the end will send a notification of the diver’s status to the smartphone by Bluetooth connection. The smartphone should run the buzzer through the mobile application when the driver is detected as drowsy.

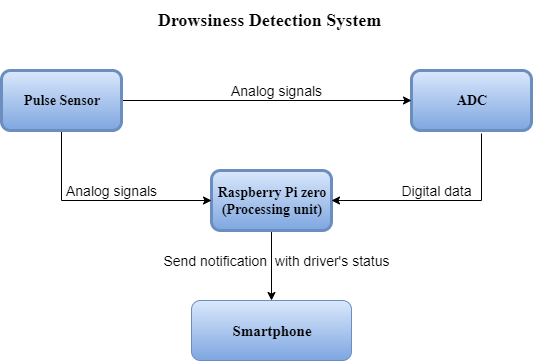


Figure ‑:Drowsiness detection system block diagram

### How the system works?

In the proposed design of the drowsiness detection system, starts when the driver is detected to be in the car. This is done through checking Bluetooth connection with a microcontroller that is placed in the vehicle which is the Arduino that is a part of the heatstroke system. Once the driver is detected to be inside the car, pulse sensor starts taking heart rate measurement. The signal that the pulse sensor which uses the PPG technology is then received by the microcontroller which is the raspberry pi zero in our case. Digital reading through analog signals can be observed through an analog to digital convertor ADC. Signals received by the microcontroller are then processed and analyzed in time and frequency domains. BPM and LF/HF ratios are then calculated to detect drowsiness points. Drowsiness is detected when LF/HF ratio and BPM shows a noticeable decrease. Once a drowsiness status is detected, the driver’s status is sent to the smart phone by Bluetooth in order to start a buzzer.

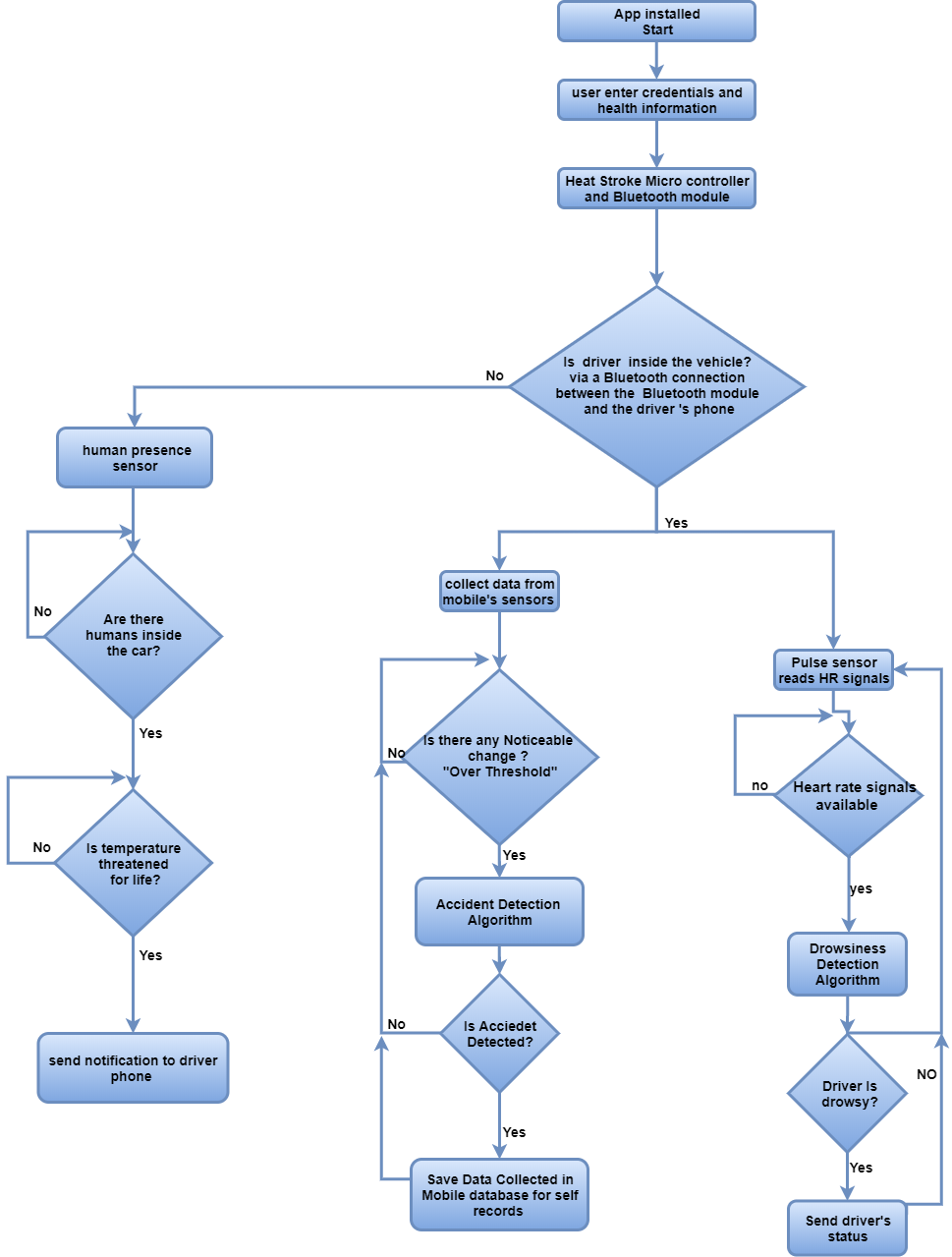


Figure ‑:Drowsiness detection system flowchart

Since the facial expressions give a lot of meaning like- closed eyes and yawing-, we will use it as another way to detect drowsiness by using USB camera so it will be turned on when the system detects the driver inside the car and start video streaming to capture the status of the driver if his eyes are closed or mouth is opened for yawing. Once a drowsiness status is detected, the driver’s status is sent to the smartphone by Bluetooth to start a buzzer as shows figure 5-6.

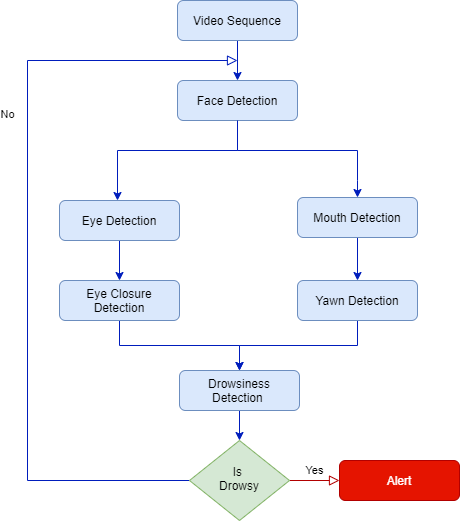


Figure ‑:Drowsiness Detection Algorithm by using Camera

When the application receives the notification of the driver's status from both other systems, the application creates a random mathematical equation to test the driver's focus and attention. After the driver answers the equation, the application checks the correctness of the driver's answer. If the driver answers correctly, the application will close itself. Otherwise, the application will turn on an alarm and it will not turn off until the driver answers correctly.

### System Connection

As mentioned before, the main components of the drowsiness detection system are the pulse sensor and the microcontroller. The figure 5-7 below shows the connections between the pulse sensor, ADC and raspberry pi. Definitely, a power supply with a stable 5V that won’t vary or sag is the best way to power up our system.

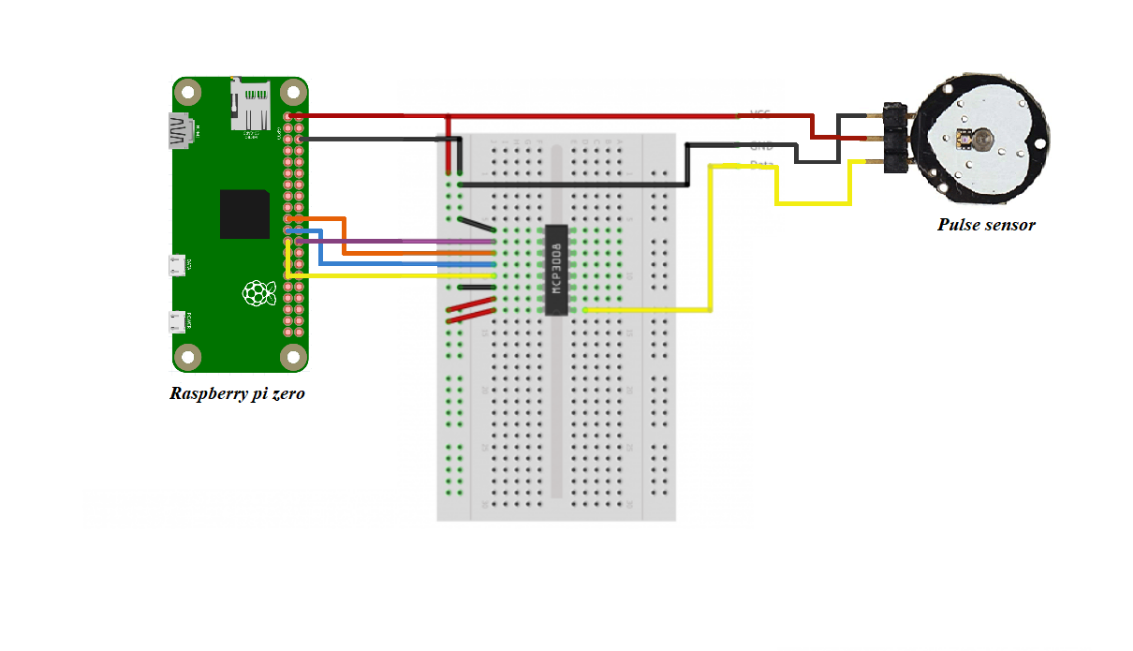


Figure ‑:Drowsiness detection system hardware implementation

## Accidents Detection and Alarming system

As mentioned before, the smartphone is the part of the system that has a direct interaction with the user through the AMAN app.

AMAN app provides a safe drive through the built-in algorithm shown below that detects accidents and performs alarming. An accident is detected when all or some conditions that are related to vehicles speed, that is received from the accelerometer and gyroscope sensors of smartphone, are met.

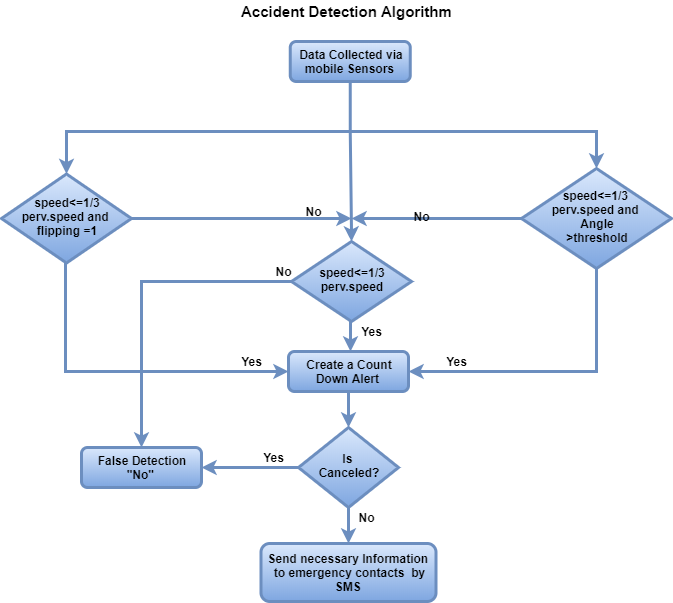


Figure ‑: Accident detection algorithm

Once an accident is detected, a countdown alarm pops up on the smartphone through the app to ensure that the driver is conscious. The countdown alarm should stop only if the driver cancelled it within 30 seconds. However, if the countdown alarm was not cancelled, the system will detect that the driver unconscious. So, the application will access the emergency information entered by the user. The application will use the contacts in the emergency list to notify them of the driver condition. At the same time, blood type of the driver and the location where the accident occurs will be sent to the ambulance, in order to ensure a fast response which increases the probability of saving a life.

AMAN application also sends notifications when drowsiness is detected. This is done through the data received from the subsystem drowsiness detection.

## Vehicle Heatstroke Detection System:

### Components of the system

The system consists of several components that work together effectively: Arduino Uno, hc-05 Bluetooth module, grove-human presence sensor, and sim900A GSM module as shown in figure 5-10 below.

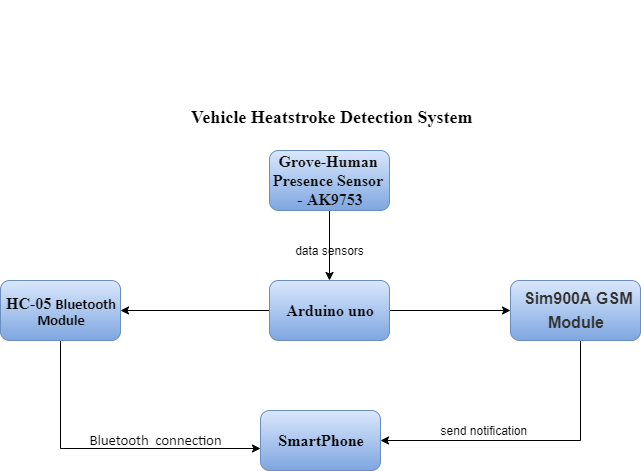


Figure ‑:Block diagram of Vehicle Heat Stroke Detection System

### How the system works?

The system only operates when the driver is not in the vehicle. Because the driver is in the vehicle, the system is no longer needed because the driver is able to monitor children and ensure that they are not exposed to a heatstroke.

We can make sure that the driver is in the vehicle via a Bluetooth connection. Assuming that the driver's phone is always with him. If the system's microcontroller is able to communicate with the driver's phone via HC-05 Bluetooth module, it means that the driver is near or inside the vehicle. The system will then shut down. Else, when the system begins to operate, the first condition that the system is checking is that the child is in the vehicle. If the result is, there is no human presence inside the vehicle, the system will wait until detecting human presence or the driver becomes inside the car. But when the result becomes positive and there is a human presence inside the vehicle, the system examines the second condition, which is monitoring the temperature and ensure that it does not threaten human life. If the temperature approaches a life-threatening temperature, the system sends a message to the driver’s phone by using sim900A GSM module. Else, if the condition is not met, the system will continue to monitor the temperature. In both conditions, the system depends on reading the human presence sensor to detect heat stroke as shown in figure 5-11 below.

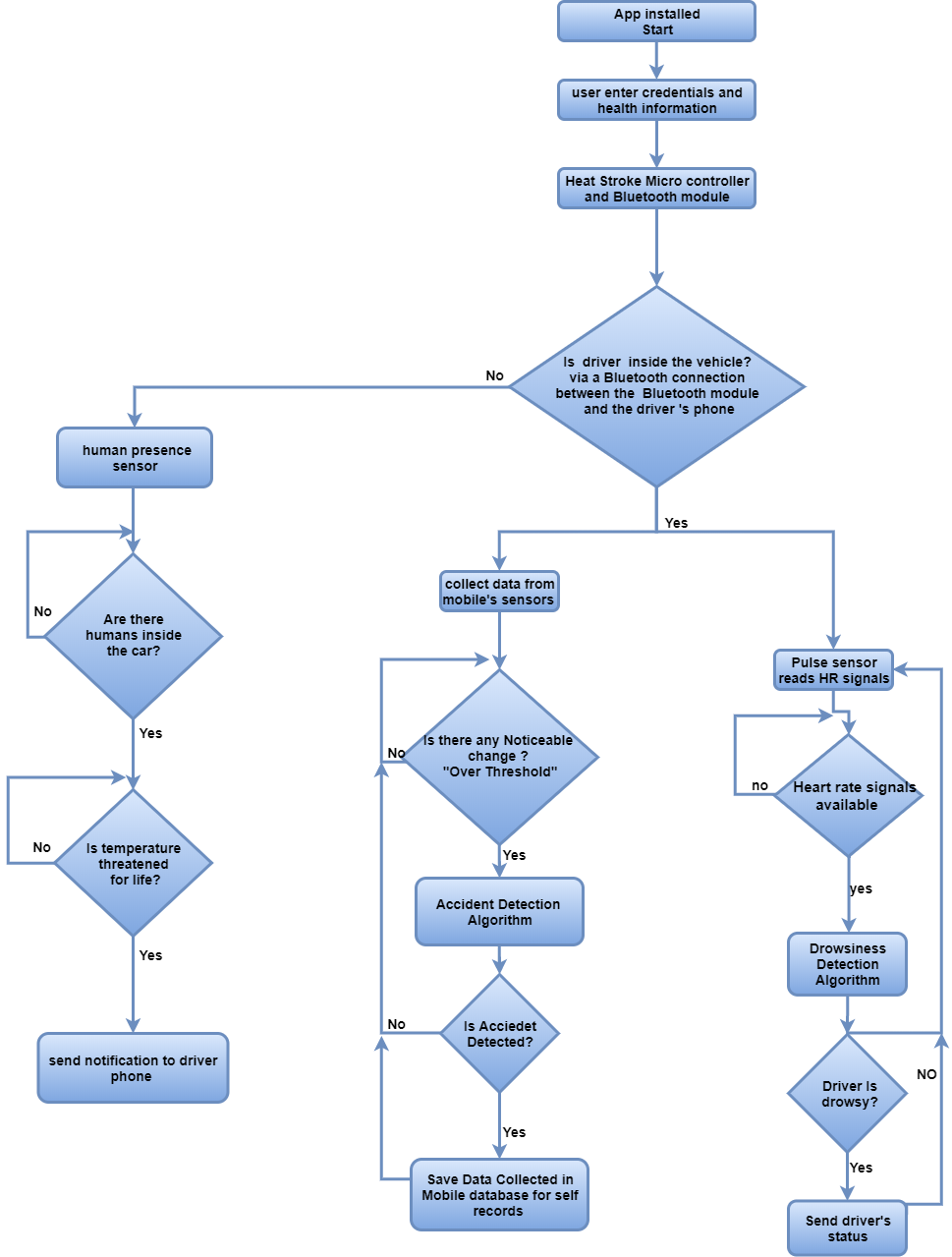


Figure ‑:flow chart of Vehicle Heat Stroke Detection System

### System Connection

**Grove-human presence sensor**

As shown in the figure 5-12 below, the Grove - Human Presence Sensor is connected to Arduino Uno by connecting A4 and A5 Arduino’s pin to SDL and SCL Human Presence Sensor’s pin respectively. In order to make this connection, I2C protocol.

**Sim900A GSM module**

Uses USART communication to communicate with microcontroller. Pin7 and 8 of **Arduino** are connected with Rx and Tx of GSM respectively. As shown in the figure 5-12 below.

**Hc-05 Bluetooth module**

Connection can be done by UART port of Arduino board. Pin numbers 0 and 1 are used for those purposes. TX of Arduino is connected to Rx of HC-05 and RX of Arduino to Tx of HC-05. As shown in the figure 5-12 below.

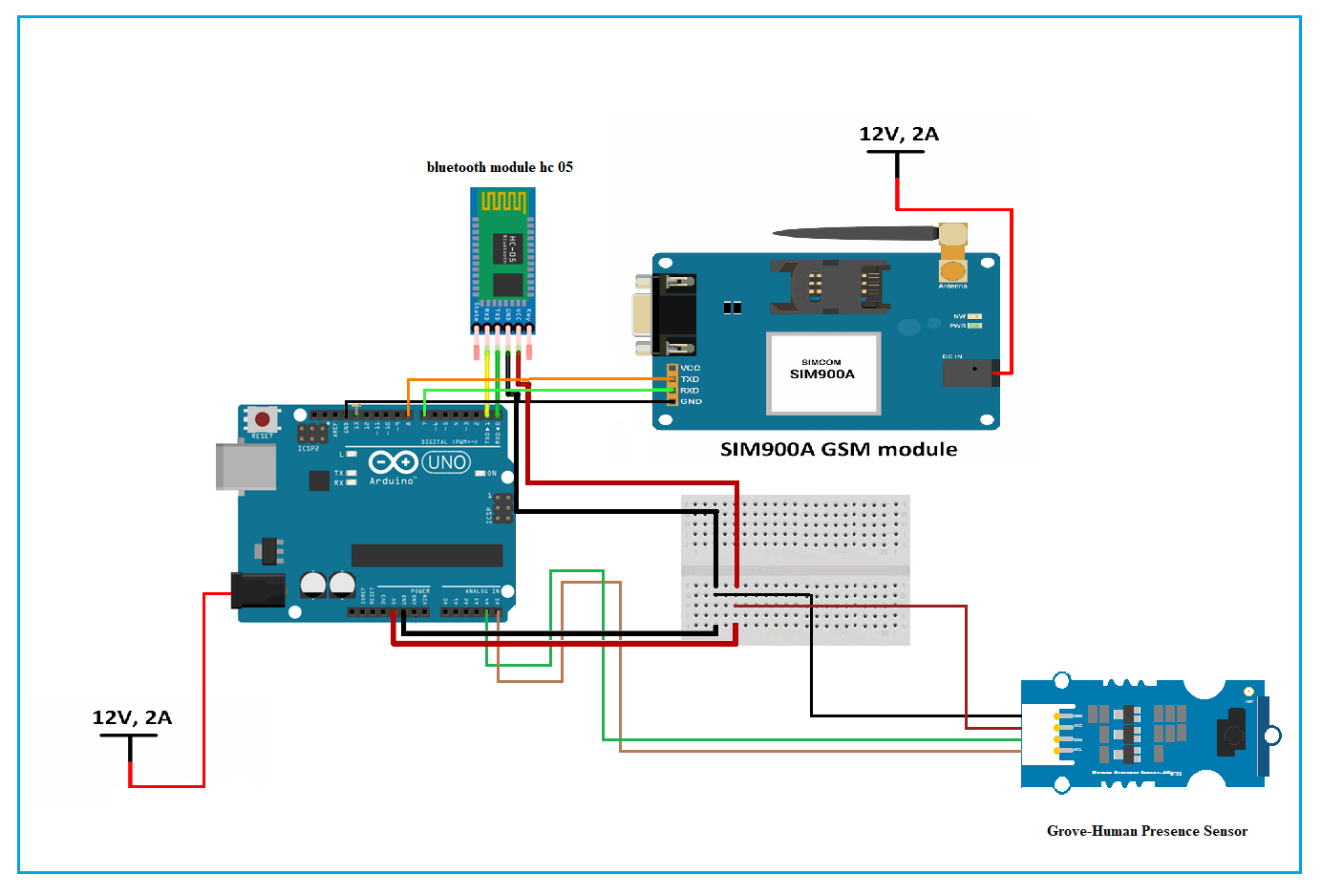


Figure ‑:circuit of vehicle heat stroke system

# Implementation

As we mentioned in the sections before, the proposed AMAN system was designed to be a big compatible system that includes three different subsystems. Based on the main goal of AMAN which is enhancing safety while driving, and the findings that has been drawn by many researches which concludes that drowsiness is the main cause of accidents, we decided to focus on the drowsiness detection system in ENCS530. Three different techniques were used to implement the drowsiness detection system. We can see the detailed implementation and results for each subsystem of the drowsiness detection system in the following sections.

## Drowsiness diction using USB camera:

In this section, we used a USB camera with RPI to detect a drowsy condition of the driver, by using a real-time algorithm to detect eye blinks and yawing in a video stream, which it will be in a steady-state in the front of the driver seat.

### Software Description:

The programming language that is used in this system is Python since it contains many features and supports multiple programming paradigms, including object-oriented, imperative, functional, and procedural and has many standard libraries. Some of the libraries being used are:

1. OpenCV: comes with a trainer as well as a detector, and it allows you to create your own classifier [18].
2. Dlib: is a toolkit for real-world machine learning and data analysis application [19].
3. Pillow: also known as PIL stands in the python imaging library which is used to open, manipulate and save images in a different format [20].
4. Face Recognition: is considered to be the simplest library to recognize and manipulate faces [21].
5. Imutils: it has many functions to make the basic image processing functions such as translation, rotation, resizing... Etc. And much easier with OpenCV and python3 [22].
6. Scipy: it provides many user-friendly and efficient numerical routines, such as routines for numerical integration, interpolation, optimization, linear algebra, and statistics [23].

### Hardware Requirements:

This system needs the following:

* RPI B+ model
* Flash memory 32GB
* Camera USB
* Keyboard, Mouse
* Power supply
* WIFI chip/Ethernet cable
* HDMI cable

First of all, we set up the USB for Raspberry Pi by flash the last version of the RPI image to the USB by using Etcher tool. This means that we will be able to write the image to the USB device, then we remove the SD card and put the USB instead, and then connect it to the power supply, after that we connect the Ethernet cable to the router, and the HDMI cable with TV/any monitor, as shown in the following figure 6-1, and then we noticed that the RPI began booting from the USB which takes almost 10 seconds.



Figure ‑:connection, booting

### Installing requirements:

The process to install OpenCV will take time (around 10 hours, some cases might be around 6 hours), we began with an update and upgrade command as figure 6-2 shows

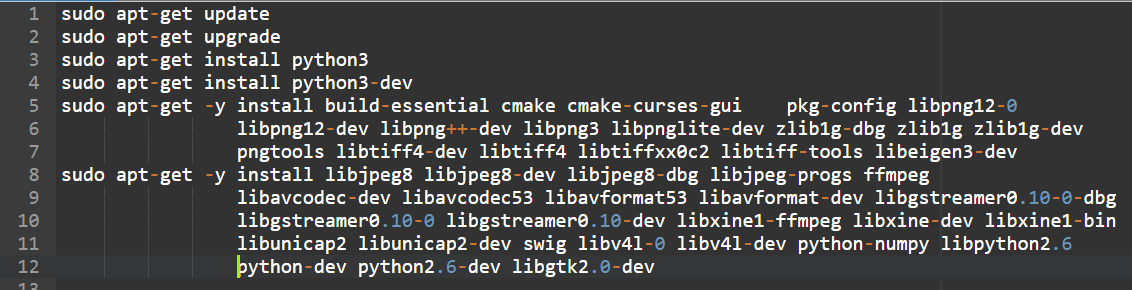


Figure ‑:Commands Part 1

After the dependency’s libraries finished to install, we download the OpenCV using the following commands as Figure 6-3 shows.

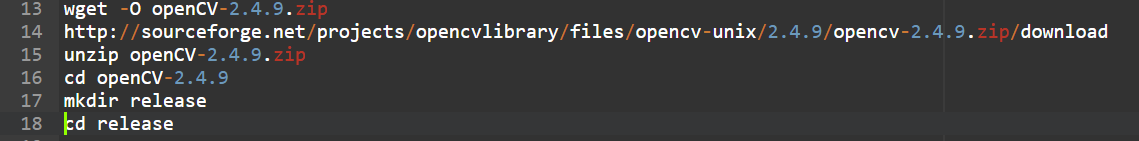


Figure ‑:Commands Part 2

Then we configured OpenCV it’s like setup, we used the default setup by the following command as Figure 6-4 shows



Figure ‑:Commands Part 3

When it has done with setting, we press ‘c’, then we press ‘g’ to generate the make file from the configuration, it almost takes 10 hours, when it’s done, we run the last command takes another 10 hours as Figure 6-5 shows.

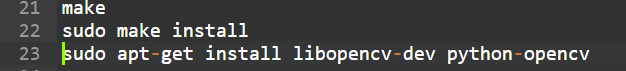
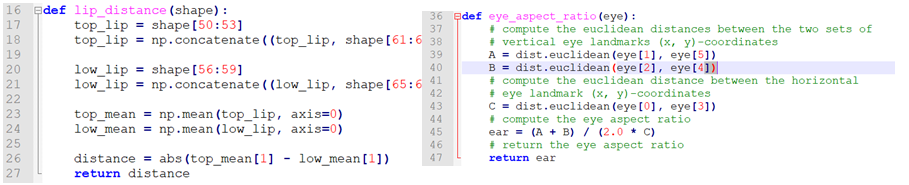


Figure ‑:Commands Part 4

### Results:

As a result of installing all the required libraries and dependencies we started implementing our code for this section using two main concepts as shown in the following figure 6-6.



1. (b)

Figure ‑:Algorithms Code

(a)YLD algorithm throw lip points (b)EAR algorithm throws eyes points

The YLD -Yawing Lip Distance - and EAR –Eye Aspect Ratio- they depend on facial extraction through landmarks classifiers and the outcome of these concepts leads to turn on the alarming system through a notification in the android application that’s already installed in the driver smartphone.

The obtained results were satisfying, since we tested it for different people –female/male- in Ramallah area for several ages, and the system successfully detects the drowsiness as shown in the following figure 6-7. The efficiency of the system increased if the driver brings a USB camera with a higher resolution.

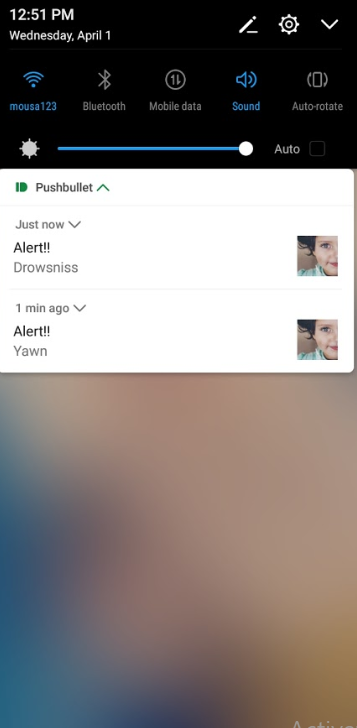


Figure ‑:Result

In the hyperlink below we can see a record for a video streaming that explains the process of the process for the drowsiness system analysis with a USB camera.

[**Hyperlink for Video-result-**](https://drive.google.com/file/d/1l_8PgK_xvQXh2qGULsV2hu-pkgLT6MHV/view?usp=sharing)

<https://drive.google.com/file/d/1l_8PgK_xvQXh2qGULsV2hu-pkgLT6MHV/view>

## Drowsiness detection using heart beats:

In this part, the technique used for detecting drowsiness was through analyzing physiological signals, the heartbeats signals. These signals where obtained through the previously described sensor which is the PPG sensor. Regarding the connection of the sensor with the raspberry pi, an ADC is needed in order to convert analog signals to digital and to make these signals readable. Unfortunately, due to the lockdown and quarantine we were not able to get an ADC but Arduino solved the problem. Knowing that the Arduino has a built in ADC so we decided to use it with the pulse sensor and raspberry pi.

The Pulse Sensor was connected directly to the analog pi A0 of the Arduino, and the Arduino was connected to the raspberry pi. Using the built-in example in the Arduino IDE which is named as “GettingStartedProject” we were able to see raw signals of our sensor. The signals were clearly shown on the serial monitor and plotter.

### Data recording, collecting and saving to files

The data we have is from live recording of heart signals through the Pulse sensor which was attached to the wrist of the object as shown in figure 6-8. We recorded data from objects, and each object recorded one time when alert and the other when drowsy.

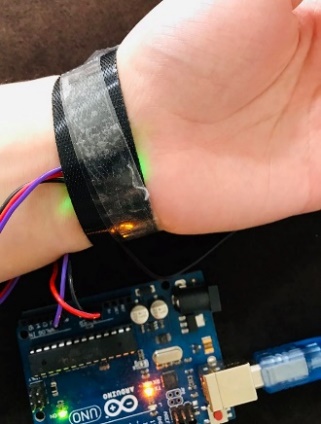


Figure ‑:sensor attached to the object's wrist

The next step was to read digital readings of the pulse sensor signals to a csv file from serial. This was done by a python code using PyCharm as an editor. The libraries needed in order to read from serial into a file adding the time of each recording are the following:

* Serial
* \_datetime
* Csv

By selecting the correct port and the same serial communication speed applied to the Arduino while recording, we can simply read data to a csv file as shown in the code below in figure 6-9. We can see also how we added a time column that contains the time in the format %H: %M: %S.



Figure ‑ from serial to csv file

### Processing and analyzing signals

After reading data to csv file, the next step was to analyze the data we have. Analyzing data was done using a library called HeartPy [24]. This library contains a lot of interesting functions for preprocessing data, filtering and calculating time and frequency measurements of the signals. As we mentioned before, we need to analyze the heart signals and get some measurements that can help us in determining drowsiness. The measurements that we adopted are beats per minute (BPM) and LF/HF ratio.

But before we go any further, let’s see how does PPG signals extracted from Pulse sensor

look like. Figure 6-10 shows raw heart signals recorded by the sensor connected by Arduino.

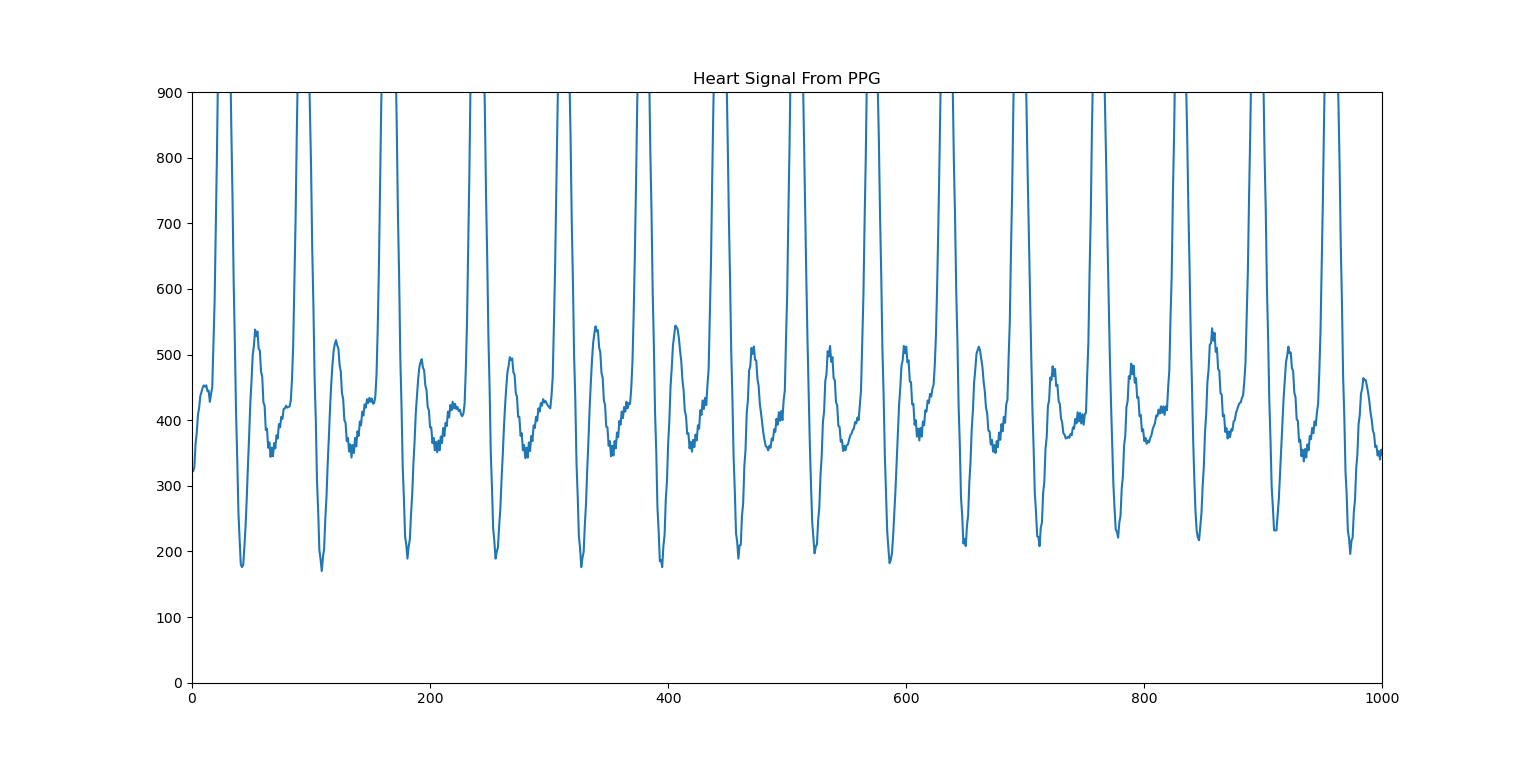


Figure ‑: Raw heart signal from Pulse sensor

In general, the measurement of raw heart signals is always prone to noise. Hence, in order to attenuate or eliminate noise, preprocessing and filtering is usually needed especially when data is obtained from live recordings. A Butterworth low pass filter of third order and with cutoff frequency of 3HZ was used, in addition to a scaling function for the signals to get a pretty good signal to play with. Both filtering and signal scaling were called by the preprocessing function which is shown in figure 6-11.



Figure ‑ preprocessing and filtering signals

The ***fs*** parameter in the preprocessing function stands for the sampling frequency. This sampling frequency was calculated for each signal through the function ***get\_samplerate***. We can see this in figure 6-12



Figure ‑: Calculating sampling rate

We can see clearly in figure 6-13 the filtered data and the difference between the original signal and the filtered one.

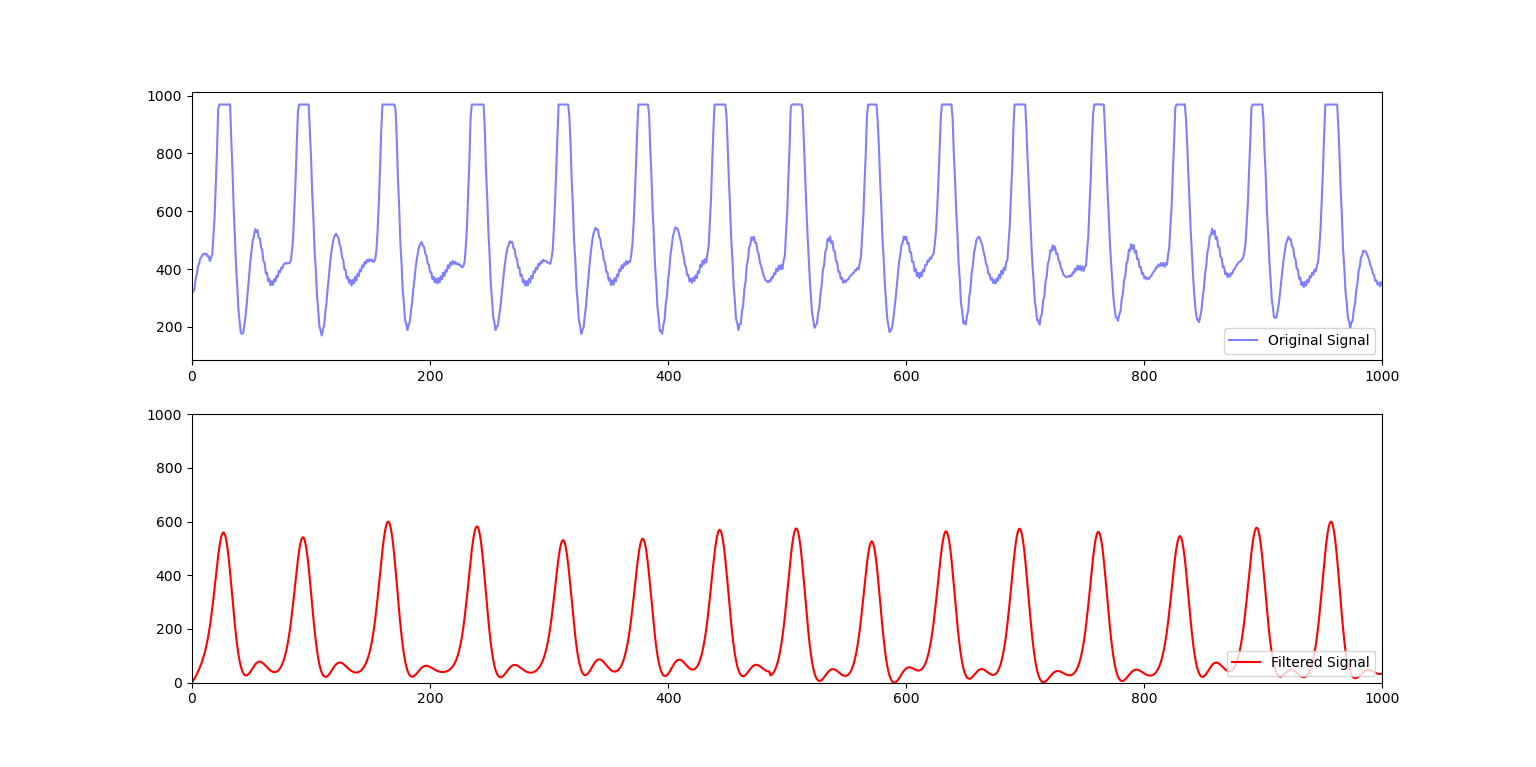


Figure ‑:Original and filtered heart signal

### Measurements from heart signals

Heartpy library provides us with many important and useful calculated measurements that are related to the heart signals. These measurements are computed from detected peaks of the signal. Beats per minute (BPM) from the time-series measurements and the ration high frequency/low frequency from the frequency domain are the measures we are interested in to detect drowsiness. These measurements can be obtained using the ***process*** function which is one of the main functions of heartpy library. This function has 21 parameters and returns 2 variables, working data and measures that contains measurements calculated. In the code below we can see how we used the process function with 5 parameters in figure 6-14.



Figure ‑ process function from heartpy library

The heartpy library provides us with a plot of detected and rejected peaks of the signal by using plotter function passing working data and measures that came from process function. We can see how the plot looks like when we use the plotter function in figure (6-15).

Figure ‑ Accepted and rejected peaks

To get frequency domain measurements, the parameter ***calc\_freq*** should be set to ***True***. The measurement we get then of the frequency domain is the LF/HF ratio. We were interested also with LF and HF separately which can be shown by taking the Fast Fourier Transform (FFT) to the R-R intervals which is calculated from intervals created by two adjacent, accepted peaks. This feature is not built in with the heartpy library. So, we did that with different functions and process. Interpolation or resampling was needed for frequency domain measures. Peaks list and the RR-list measurements from wd array were used in interpolating, which was simply done using the function ***interp1d*** from ***scipy.interpolate*** library in python. Figure 6-17 shows the original signal with the interpolated one. After interpolating FFT was applied, and finally LF and HF were calculated as shown in figure 6-18. In figure 6-16 we can see the plot in the frequency domain which shows us the regions of low frequencies 0.04 to 0.15 Hz and high frequencies which are in the range of 0.18-0.4 Hz.

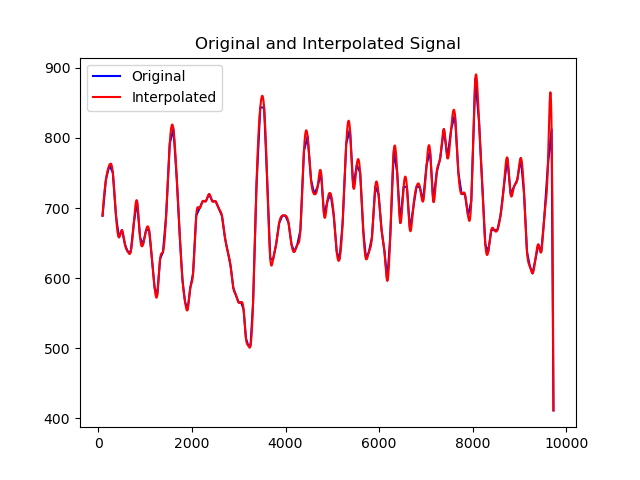
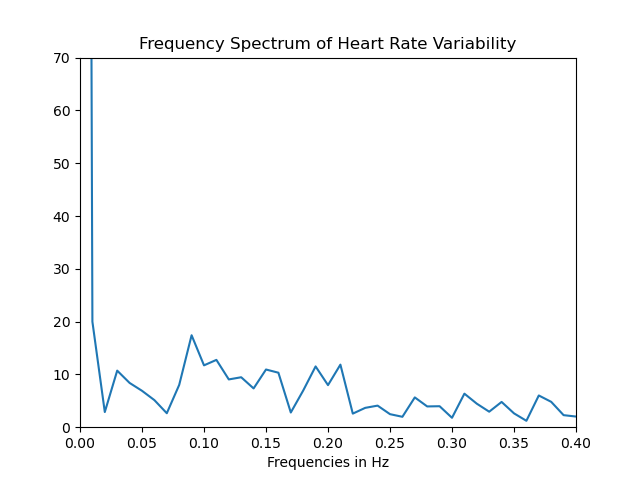


Figure ‑: frequency spectrum of heart rate variability

Figure ‑:Plot of original signal and interpolated



Figure ‑: calculating LF and HF

The last step was to log all data files with their BMP and LF/HF ratio values to an excel file to organize and visualize the results in charts. We can see all results in results section below.

### Results

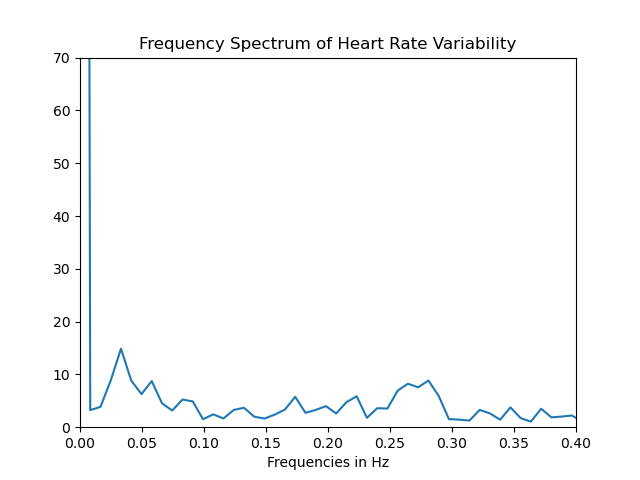
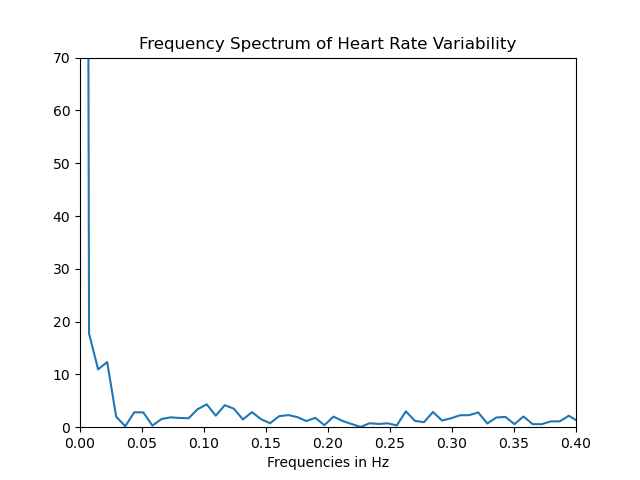
Now, we can see the results based on BPM and LF/HF ratios. The blue columns represent alert objects BPM values, while the orange ones show their BPM when they are drowsy. The graph clearly show how BPM for drowsy object differs from alert ones. We can see that if BPM differs by 6 or more then we can detect that the object is drowsy. For sure we got some errors and some data that shows close BPM values when drowsy and alert. This was expected, as our sensor is not that much accurate, in addition to that recording drowsy data was not easy at all.

Figure ‑ BPM results when drowsy and alert for each object

The lines in the graph in figure 6-20 show the difference between LF/HF ratios between alert and drowsy objects. As we mentioned before, we have some errors due to inaccuracy of the sensor. Despite all, we can see that the alert LH/HF ratios have somehow bigger values than drowsy ones. From the results shown in the data table under the graph, we can say that objects with LF/HF of let’s say 0.5 or greater are probably alert. But again, this result has some errors and collecting more data files could enhance it and make it more accurate.

Figure ‑ LF/HF plots showing differences when drowsy and alert

Finally, we were interested to prove a result that we saw in a paper and mentioned it before in the background chapter (see figure 3-6), that shows differences in frequency spectrums of drowsy and alert signals especially at frequencies 0.04 and 0.15. Fortunately, we got somehow similar results. We can see in figure 6-21 (b) that for drowsy objects, at frequencies of 0.04 and 0.15 the spectrum shows sharper lines than the one of alert object shown in (a).



(a) (b)

Figure ‑:Frequency spectrum of HR

(a) spectrum of alert object (b)spectrum of drowsy object

## Drowsiness detection using mobile application:

Building an AMAN application is the final stage of the driver drowsiness detection system. In the beginning, AMAN application will turn on when gets an alarm notification from the other two subsystems of the drowsiness detection system. We were not able to get a raspberry pi to try this system because of coronavirus's circumstance.

The application initially generates a random number to be the index of equations’ array. This array includes simple mathematical equations, the application will write the equation on the screen and use text to speech instance to make the driver’s Android device read the equation and convert it to audio out via the speaker as shown in figure 6-22.

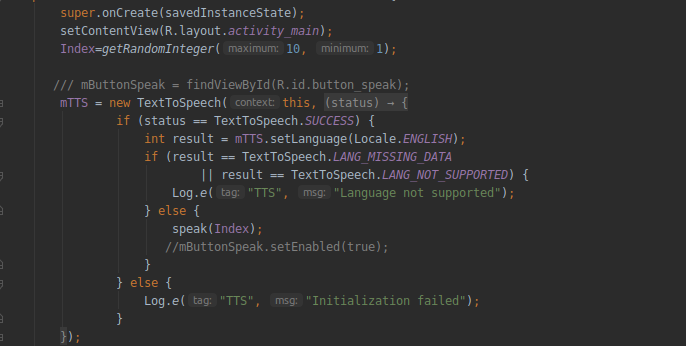


Figure ‑: Text to speech code

After that, the application will be freezing until the driver clicks on anywhere on the App’s screen to allow it to listen to the driver's sound. this approach comes to protect the driver's privacy. Then the app will recognize and analyze the driver's sound. The application will use speech Recognizer Intent to get the driver's answer as shown in figure 6-23.

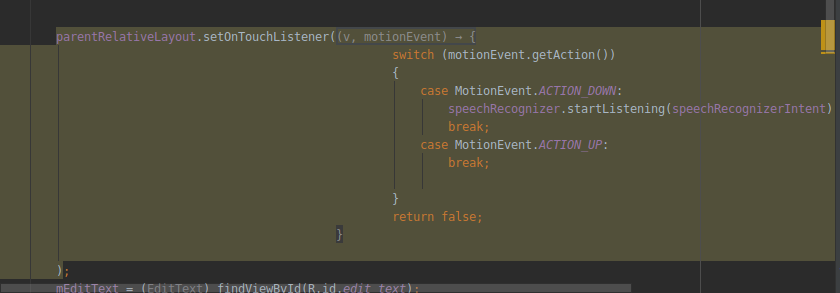


Figure ‑: SetOnTouchListener

The last stage is to verify the driver's answer. by comparing the default equation's answer with the driver's answer. If the driver answers correctly, the application will close itself. Otherwise, the application will turn on an alarm and it will not turn off until the driver answers correctly, as shown in figure 6-24.

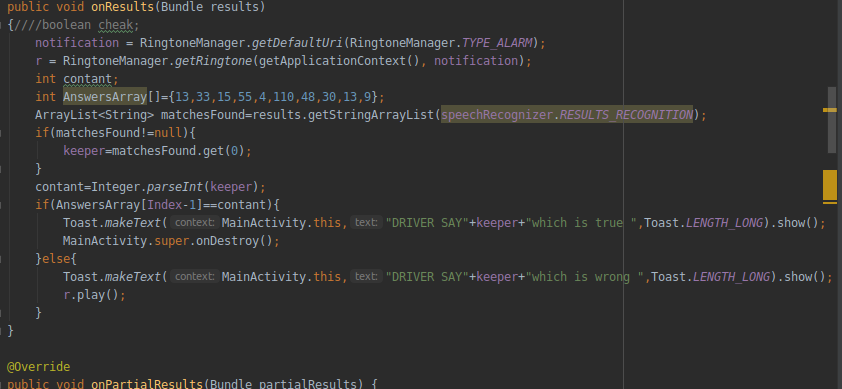
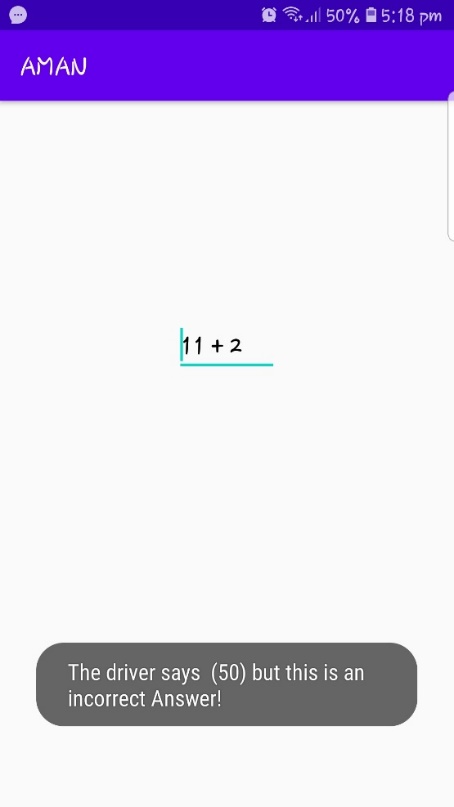
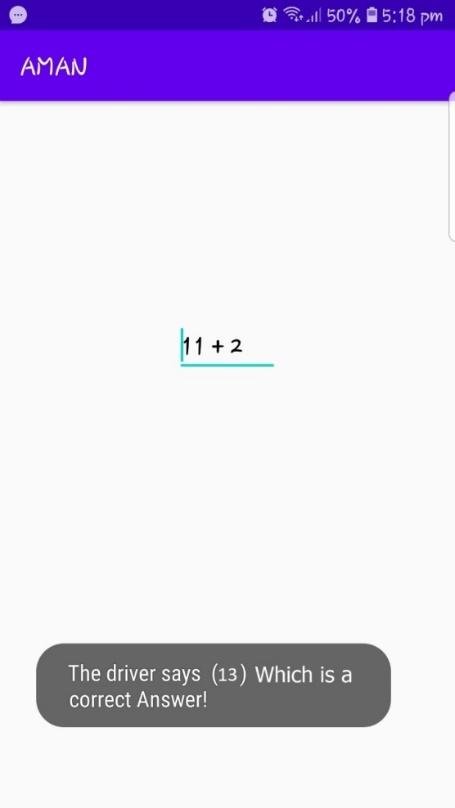


Figure ‑: OnResults method

### Results:

Here we can see the interaction between the driver and the application. The equation will be written on the screen and heard through the speakers. When the driver answers, the app will detect whether if the answer is correct or not. The response of the app will be shown on the screen as shown in figure 6-25.



1. **(b)**

Figure ‑: Interaction between driver and application

(a) when the driver answers correctly (b) when driver answers incorrect

In the Hyperlink below we can see a demo that shows the interacting between the driver and the application and how it turns on an alarm when the answer is incorrect

[**Hyperlink for AMAN app-result-**](https://drive.google.com/file/d/193TmysV7ZcjaBmOX2c8FNXhOJ521etaY/view)

<https://drive.google.com/file/d/193TmysV7ZcjaBmOX2c8FNXhOJ521etaY/view>

# Conclusion and future work

Researches and surveys suggest that nearly 20-50 million around the world suffer injuries or death from car accidents only. Currently, there exist automated systems that are designed to help the driver stay safe on the road. However, the issue still exists where the car isn’t “too safe”. In addition, with the advancement of technology, a mobile application will be developed to interact with the user and provide additional support while driving. All the work carried out in ENCS520 was research, theory, and related work on the concept of the proposed AMAN system in general, and in many respects. A preliminary design, connections, and implementations for the overall system were provided. In ENCS530 we started by the drowsiness detection system and focusing on it in conjunction with the construction of a mobile application. This approach is to ensure a good complete implementation with high accuracy of the drowsiness detection system. The drowsiness detection system was implemented using three different techniques that are physiological techniques, image processing techniques, and a mobile application that interacts with the driver. The results of implementing and using each technique of drowsiness that we mentioned were satisfying.

Unfortunately, we were not able to combine all the drowsiness detection subsystems together because of the quarantine and the lockdown due to the coronavirus. Despite all obstacles we are still looking for combining all these systems together in order to get a good overall result and accuracy for detecting drowsiness.

As a future work we aspire to build and implement the overall AMAN system that we proposed and introduced in ENCS520, in order to have a big compatible system that also aims to enhance safety while driving a car, using different techniques and approaches.

# References

|  |  |
| --- | --- |
| [1] | World Health Organization, [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries. [Accessed 15 November 2019]. |
| [2] | "Journal of Korean medical science," 12 Febreuary 2019. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6393761/. [Accessed 15 November 2019]. |
| [3] | J.-C. Lee and H. Liu, "Development of a Real-Time Driver Health Detection System Using a Smart Steering Wheel," *International Journal of Prognostics and Health Management,* no. 2153-2648, 2018. |
| [4] | S. H. R. L. Bowlds, Y. G. and X. Y. , "Pulse wave sensor for non-intrusive driver's drowsiness dtection," in *31st Annual International Conference of the IEEE EMBS*, Minnesota, 2009. |
| [5] | S. Jaunead and H. Akbar, "Driver Drowsiness Detection Based on Face Feature and PERCLOS," in *Journal of Physics: Conference Series*, Jakarta, 2018. |
| [6] | P. D. Purnamasari and A. Zul Hazmi, "Heart Beat Based Drowsiness Detection System for Driver," in *International Seminar on Application for Technology of Information and Communication (iSemantic)*, 2018. |
| [7] | "Wikipedia," 14 November 2019. [Online]. Available: https://en.wikipedia.org/wiki/Anti-lock\_braking\_system. [Accessed 15 November 2019]. |
| [8] | "Wikipedia," 14 November 2019. [Online]. Available: https://en.wikipedia.org/wiki/Adaptive\_cruise\_control. [Accessed 15 November 2019]. |
| [9] | "Wikipedia," 03 November 2019. [Online]. Available: https://en.wikipedia.org/wiki/Collision\_avoidance\_system. [Accessed 15 November 2019]. |
| [10] | H. M Sherif, . A. S. M and S. A. S. , "Real-Time Traffic Accident Detection System Using Wireless Sensor Network(RTTADS)," 2014. |
| [11] | A. Shaik, N. Bowen, J. Bole, G. Kunzi, D. Bruce, A. Abdegawad and K. Yelamarthi, "Smart Car: An IoT Based Accident Detection System," 2018. |
| [12] | W. J. G. S. F. L. A. J. O. Y. B. a. C. V. S. S N David Chua, "Development of an Automatic Vehicular Heatstroke Detection". |
| [13] | Sumit Garethiya, Himanshu Agrawal,, Suresh. V, Amit Kudale, Gaurav and Shilpa Gite, “Affordable System for Alerting, Monitoring and”. |
| [14] | B. Vescio, M. Salsone, A. Gambardella and A. Quattrone, "Comparison between Electrocardiographic and Earlobe Pulse Photoplethysmographic Detection for Evaluating Heart Rate Variability in Healthy Subjects in Short- and Long-Term Recordings," in *US National Library of Medicine National Institutes of Health*, Catanzaro, 2018. |
| [15] | G. Li and W.-Y. chung, "Detection of Driver Drowsiness Using Wavelet Analysis of Heart Rate Variability and a Support Vector Machine Classifier," *Sensors,* no. 1424-8220, 2013. |
| [16] | "DFROBOT," 2008-20017. [Online]. Available: https://media.digikey.com/pdf/Data%20Sheets/DFRobot%20PDFs/SEN0203\_Web.pdf. [Accessed 12 January 2020]. |
| [17] | "Wikipedia," 27 April 2020. [Online]. Available: https://en.wikipedia.org/wiki/Android\_Studio. [Accessed 10 May 2020]. |
| [18] | "OpenCV," 2020. [Online]. Available: https://opencv.org/. [Accessed 4 April 2020]. |
| [19] | "Dlib," 14 Dec 2019. [Online]. Available: http://dlib.net/. [Accessed 4 April 2020]. |
| [20] | "Pillow," [Online]. Available: https://pillow.readthedocs.io/en/stable/. [Accessed 5 April 2020]. |
| [21] | "face-recognition," pypi.org, 20 Feb 2020. [Online]. Available: https://pypi.org/project/face-recognition/. [Accessed 6 April 2020]. |
| [22] | "imutils," pypi.org, 18 Aug 2019. [Online]. Available: https://pypi.org/project/imutils/. [Accessed 6 April 2020]. |
| [23] | "scipylib," scipy.org, 2020. [Online]. Available: https://scipy.org/scipylib/. [Accessed 7 April 2020]. |
| [24] | P. v. Gent, 14 April 2020. [Online]. Available: https://readthedocs.org/projects/python-heart-rate-analysis-toolkit/downloads/pdf/latest/. [Accessed 1 May 2020]. |
| [25] | "World Life Expectancy," [Online]. Available: https://www.worldlifeexpectancy.com/united-states-road-traffic-accidents. [Accessed 14 November 2019]. |
| [26] | "Adaptive Cruise Control System Overview," 14 April 2005. [Online]. Available: http://sunnyday.mit.edu/safety-club/workshop5/Adaptive\_Cruise\_Control\_Sys\_Overview.pdf. [Accessed 15 November 2019]. |
| [27] | "Sleep Education," [Online]. Available: http://sleepeducation.org/sleep-topics/drowsy-driving. [Accessed 15 November 2019]. |
| [28] | "National Sleep Foundation," [Online]. Available: https://www.sleepfoundation.org/articles/drowsy-driving. [Accessed 15 November 2019]. |
| [29] | "Google Search," [Online]. Available: https://www.google.com/search?biw=1536&bih=674&tbm=isch&sxsrf=ACYBGNR8zRoiJjFxG3z4nBIrK1C-u2Ri7w%3A1573617071455&sa=1&ei=r33LXYywG-Gp8gK4y5moDA&q=drowsy+driving+statistics+in+US&oq=drowsy+driving+statistics+in+US&gs\_l=img.3...51975.55065..55575...0.0..0.1. [Accessed 15 November 2019]. |
| [30] | "Apex Airconditioning," 05 February 2018. [Online]. Available: https://www.apexaircon.com.au/blog/do-cold-temperatures-make-you-sleepy. [Accessed 15 November 2019]. |
| [31] | "Time," 25 May 2018. [Online]. Available: https://time.com/5291550/cars-dangerous-temperatures/. [Accessed 15 November 2019]. |
| [32] | wikipidia, "wikipidia," 8 june 2016. [Online]. Available: www.r.com. [Accessed 10 october 2018]. |
| [33] | M. Mitchell, An Introduction to Genetic Algorithms, London, England: MIT press, 1998. |
| [34] | N. Panwar, S. Shantanu and A. K. Singh, "A Survey on 5G: The Next Generation of Mobile Communication," *Physical Communication,* vol. 18, no. 3, pp. 64-84, 2016. |
| [35] | E. Seidel, "Carrier Aggregation Based Interference Coordination for LTE-A Macro-Pico HetNet," *Nomor Research GmbH,* vol. 77, no. 4, pp. 1-6, 2013. |
| [36] | "healthychildren.org," American Academy of Pediatrics, 16 July 2019. [Online]. Available: https://www.healthychildren.org/English/safety-prevention/on-the-go/Pages/Prevent-Child-Deaths-in-Hot-Cars.aspx. [Accessed 15 November 2019]. |
| [37] | 30 Julu 2019. [Online]. Available: https://edition.cnn.com/2019/07/30/us/hot-car-deaths-child-charts-graphs-2019-trnd/index.html. [Accessed 15 November 2019]. |
| [38] | "American Acaddemy of Pediarics," 01 July 2005. [Online]. Available: https://pediatrics.aappublications.org/content/116/1/e109. [Accessed 18 November 2019]. |
| [39] | "Intelligent system for vehicular accident detection and notification," [Online]. Available: Intelligent system for vehicular accident detection and notification. [Accessed 18 11 2019]. |
| [40] | "Smart2.0," 1 4 2019. [Online]. Available: https://www.smart2zero.com/news/magnetic-field-nanowire-sensor-extends-battery-life-evs. [Accessed 18 11 2019]. |
| [41] | "FierceElectronics," 18 july 2019. [Online]. Available: https://www.fierceelectronics.com/electronics/sensor-creates-its-own-energy-could-be-useful-for-automotive. [Accessed 18 11 2019]. |
| [42] | "Photoplethysmogram," [Online]. Available: https://en.wikipedia.org/wiki/Photoplethysmogram. [Accessed 18 11 2019]. |
| [43] | "Doppler radar for heartbeat rate and heart rate variability extraction," 24-26 11 2011. [Online]. Available: https://ieeexplore.ieee.org/document/6150419. [Accessed 18 11 2019]. |
| [44] | J. Sang-Ho, K. Jin-Myung and K. Dong Kyoo, "Heart Rate Change While Drowsy Driving," *JKMS,* 2019. |
| [45] | K. K. BM, S. Sethi, R. Kumar, A. Shankar and N. Kumar, "Driver Drowsiness Detection System Using Sensors," *International Journal of Informatics and Communication Technology (IJ-ICT),* vol. 6, no. 3, pp. 139-145, 2017. |
| [46] | A. Karaman, S. Z. Tan, B. Li, L. G. H. Thein, R. X. Young and S. Qie, "ThermSafe: Child Heat Injury Prevention in Heated Locked Cars," [Online]. Available: https://docs.google.com/document/d/1LbRCP7lY\_eLP1ho7S1gXEggAeitGY8vf-eCfA-LX7Fk/edit#. [Accessed 18 11 2019]. |
| [47] | [Online]. Available: https://ionicframework.com/. [Accessed 20 January 2020]. |
| [48] | K. Taylor-Smith, "AZO sensors," 8 January 2019. [Online]. Available: https://www.azosensors.com/article.aspx?ArticleID=1110. [Accessed 24 January 2020]. |
| [49] | "GIS Geography," 14 November 2019. [Online]. Available: https://gisgeography.com/lidar-light-detection-and-ranging/. [Accessed 24 January 2020]. |
| [50] | O. Cameron, "voyage," 9 May 2017. [Online]. Available: https://news.voyage.auto/an-introduction-to-lidar-the-key-self-driving-car-sensor-a7e405590cff. [Accessed 24 January 2020]. |
| [51] | "LiDAR," [Online]. Available: http://www.lidar-uk.com/how-lidar-works/. [Accessed 24 January 2020]. |
| [52] | . P. Parmar and A. M. Sapkal, "Real-Time Detection and Reporting of VehicleCollision," 2017. |
| [53] | A. Bin Faiz, A. Imtiaz and M. Chowdhury, "Smart Vehicle Accident Detection and AlarmingSystem Using a Smartphone," Bangladesh, 2015. |
| [54] | K. Gale, "Sensorland," [Online]. Available: https://www.sensorland.com/HowPage051.html. [Accessed 24 January 2020]. |
| [55] | J. Engel, "DIGNITED," 18 May 2017. [Online]. Available: https://www.dignited.com/23114/bluetooth-5-vs-bluetooth-4/. [Accessed 24 January 2020]. |
| [56] | "GSMArena," [Online]. Available: https://www.gsmarena.com/glossary.php3?term=sensors. [Accessed 25 January 2020]. |