Heart Rate Emergency System

By: Alaa Elesh, Ayat Khaled Hosney, Fady Maged Fakhry

Mohannad Khaled Mohamed, Hagar Mohamed Farouk,

Kareem Mohamed Emam.

Table of contents

Chapter	Page
Cover Page	
Table of Contents	1
List of figures	2
• **Abstract**	3
• **Chapter One: Introduction**	4
• **Chapter Two: Hardware and software environment**	7
2.1 Heart rate system hardware	7
2.1.1 GY-MAX30100 Heart Rate Sensor	7
2.1.2 ST7735 TFT Display	9
2.1.3 HC-05 Bluetooth Module	10
2.1.4 STM32F401CCU6 Microcontroller	12
2.2 Heart rate system software	14
2.3 Main control system hardware	15
2.3.1 Buzzer	15
2.3.2 Push Button	16
2.3.3 GPS NEO-6m Module	16
2.3.4 GSM-GPRS SIM900 Module	18
2.3.5 Raspberry Pi Microcontroller	18
2.4 Main control system software	19
2.5 Car prototype system hardware	20
2.5.1 DC Motors and Wheels	21
2.5.2 HC-SR04 ultrasonic sensor	22
2.5.3 L298N dual H-Bridge	22
2.5.4 HC-05 Bluetooth Module	23
2.5.5 STM32F401CCU6 Microcontroller	23
2.6 Car prototype system software	23
• **Chapter Three: Final Prototype and Results**	25
3.1 Final Prototype	25
3.2 Results	27
• **Chapter Four: Conclusion**	29
• **References**	31

List of figures

Figure	page
Figure 2.1: GY-MAX30100 Heart Rate Sensor	7
Figure 2.2: ST7735 TFT display	8
Figure 2.3: HC-05 is a Bluetooth module	9
Figure 2.4: HC-05 Bluetooth module and USB-TTL module connection	10
Figure 2.5: Tera Term tool configurations	10
Figure 2.6: Tera Term tool terminal parameters	10
Figure 2.7: Tera Term tool Serial Port Baud Rate	11
Figure 2.8: Tera Term tool terminal change mode - AT command	11
Figure 2.9: STM32F401CCU6 development board	12
Figure 2.10: Heart rate system flow diagram	13
Figure 2.11: Buzzer	14
Figure 2.12: Push Button	15
Figure 2.13: GPS NEO-6m Module	15
Figure 2.14: GSM-GPRS SIM900 module	17
Figure 2.15: Raspberry Pi 3 Model B+	18
Figure 2.16: Main control system flow diagram	19
Figure 2.17: wheel attached to DC motor	20
Figure 2.18: HC-SR04 ultrasonic sensors	21
Figure 2.19: L298N Dual H Bridge Stepper Motor Driver Board	21
Figure 2.20: Car prototype system flow diagram	23
Figure 3.1: heart rate system 2D AUTOCAD design	24
Figure 3.2: heart rate system PCB	24
Figure 3.3: Car prototype 2D AUTOCAD design	25
Figure 3.4: Final Car prototype system	26
Figure 3.5: heart beat values (in contact)	26
Figure 3.6: heart beat values (no contact)	26
Figure 3.7: Serial Bluetooth terminal mobile application	27
Figure 3.8: SMS help message	28

Abstract

Many accidents on the roads can be caused by sudden changes in the physical conditions of drivers. Thus, it is important to have a monitoring system that can detect these changes in case of any abnormalities. However, this system must not restrict or interfere with the driver's action.

In this project we will present a heart rate emergency monitoring system that can check the driver's heart rate and take action depending on it. When the value of heart beat is normal, the driver continues driving the car smoothly. However, when it is abnormal autonomous driving will take action to park the car and prevent any further damage to the driver and his surroundings.

Our system consists of three main sub systems. The first one is called heart rate system. It is composed of Max30100 heart rate sensor to sense the driver's heart rate, STM32F401CCU6 - ARM based- microcontroller to convert the signal from the sensor to heart beats, ST7735 TFT display to show the heart beat values, HC-05 Bluetooth module to send the measurements using UART communication protocol to Raspberry Pi microcontroller in the second sub system. The second sub system is called main control system. It is composed of Raspberry Pi microcontroller to check the heart beats, buzzer to start an alarm in case of abnormal condition, a GPS NEO-6m Module to get the driver's location, GSM-GPRS SIM900 module to send the driver's location in an SMS help message to his family members, and a push button to stop the alarm if the condition was not actually serious. The third sub system represents a car prototype system that is composed of ARM based STM32F401CCU6 microcontroller alongside with 4 HC-SR04 ultrasonic sensors, 4 DC Motors and 4 wheels, 2 L298N dual H-Bridge attached to a printed car body. To mimic a normal driving condition, android mobile application (Serial Bluetooth terminal) and Bluetooth module connected to the car are used. On the other hand, autonomous driving to park the car will be implemented by the ARM microcontroller with the data collected form the ultrasonic sensors.

Chapter One: Introduction

Many accidents are caused by sudden changes in the physical conditions of drivers. A report shows that road fatalities are lower, but still quite significant, the researchers say. Nationally, fatalities from road crashes per 100,000 population is 10.9, compared to 34.4 for Alzheimer's, 43.7 for stroke, 48.2 for lung disease, 185.4 from cancer and 197.2 from heart disease [1]. Thus, traffic accidents caused by heart disease seem to be an important problem.

In order to solve such a problem, many systems have been developed to implement an emergency procedure for taking away driving control from a driver when his physical condition suddenly changes [2,3]. In automotive industry, vehicles are equipped with such an emergency system [4,5,6,7,8 ,9,10,11]. For example, an emergency system of Mercedes-Benz has several steps. When the driver is no longer interacting with the steering wheel, the system flashes a light and sounds a tone to alert the driver to return his/her hands to the steering wheel. If the driver still does not respond, the system applies the brakes. As it slows, the system maintains the lane in which the vehicle is already traveling [10].

In Japan, the Ministry of Land, Infrastructure, Transport, and Tourism published a "health management manual for drivers of commercial vehicles" in 2010 (revised in 2014) [12]. The manual states that devices and software to check a driver's health are required as part of daily management procedures. Therefore, it is expected that driver-state monitoring systems will be developed in the future. Such systems are required not only to monitor physical health but also to take control of the driving of the vehicle, according to the Society of Automotive Engineers (SAE)'s standards for autonomous driving level 3 [13]. These systems are characterized by the acquisition of the driver's biometric information to understand his condition and reflect the results of vehicle control.

Biometric information includes in vivo information such as an electrocardiogram (ECG), blood pressure levels, and visceral fat levels; and ex vivo information such as exercise levels, sleep patterns, and diet. It is very important to evaluate, associate, and analyze these data appropriately. The instantaneous heartbeat, which is calculated from the ECG, is a very special information. In the normal condition, the instantaneous heart rate fluctuates within a certain range of variation. This range is called heart rate variability (HRV). Analyzing of HRV allows the evaluation and categorization of many heart diseases. Table 1 shows the general and traditional methods of measuring heart rate based on Reference [14].

Method	Technique	Device	Weak Point
ECG (Electrocar-diogram)	Measures the electrical pulses generated by the body during each cardiac cycle	Electrodes	Difficult to position correctly, affected by body movement, electrodes become loose when the body is sweaty
Sphygmomano-meter	A method for measuring changes in arterial pressure that vary with heart pulsation	Sphygmomano-meter	Affected by body movement
Cardiogram	Measure the sound generated by the pulsation of the heart	Finger, stethoscope, microphone	Affected by hand and finger movement
Photoelectric pulse	Near-infrared light is applied to the skin surface and the reflected light is received by a photodiode or other device	Smartwatch	Affected by body movement, touching condition of fingers, skins, etc.

Table 1: Measuring heart rate methods and weak points of the general and traditional method.

According to Table 1, general and traditional methods of measuring heartbeat restrict or interfere with the driver's action. Therefore, these methods are not suitable to measure heartbeat in a vehicle. However, certain wearable-type and nonwearable-type systems do not restrict the driver's action. Therefore, they are currently in use in automotive industry. Wearable types include a person wearing a wristwatch-type, clothes-type, or ring-type systems. Nonwearable types include systems that measure the driver's condition at the steering wheel, seat, or other places the driver touches while driving. Nowadays, there are many watch-type wearable devices, and devices such as the Apple Watch [15], Empatica E4 [16], Garmin Watch [17], Fitbit surge [18], and Jawbone UP3 [19] are sold on the market. The methods used for measuring heartbeat in these devices are nearly the same. The blood-flow to the wrist increases with the heartbeat, and the green light used in the device is well absorbed. In contrast, the green light is less likely to be absorbed at the time between heartbeats. By using this principle, the heartbeat is measured by flashing an LED light behind the watch several hundred times per second [20]. In addition, such devices can be linked to smartphones to share data.

By using the concept of wearable-type heart beat measurement system, we decided to implement a heart rate emergency monitoring system that can check the driver's heart rate and take action depending on it. When the value of heart beat is normal, the driver continues driving the car smoothly. However, when it is abnormal, autonomous driving will take action to stop the car and prevent any further damage to the driver and his surroundings.

Our system consists of three main sub systems. The first one is called heart rate system. It is composed of Max30100 heart rate sensor to sense the driver's heart rate, STM32F401CCU6 - ARM based- microcontroller to convert the signal from the sensor to heart beats, ST7735 TFT display to show the heart beat values, HC-05 Bluetooth module to send the measurements using UART communication protocol to Raspberry Pi microcontroller in the second sub system. The second sub system is called main control system. It is composed of Raspberry Pi microcontroller to check the heart beats, buzzer to start an alarm in case of abnormal condition, a GPS NEO-6m module to get the driver’s location, GSM-GPRS SIM900 module to send the driver’s location in an SMS help message to his family members, and a push button to stop the alarm if the condition was not actually serious. The third sub system represents a car prototype system that is composed of ARM based STM32F401CCU6 microcontroller alongside with 4 HC-SR04 ultrasonic sensors, 4 DC Motors and 4 wheels, 2 L298N dual H-Bridge attached to a printed car body. To mimic a normal driving condition, android mobile application (Serial Bluetooth terminal) and Bluetooth module connected to the car are used. On the other hand, autonomous driving to park the car will be implemented by the ARM microcontroller with the data collected form the ultrasonic sensors.

Chapter Two: Hardware and software environment

2.1 Heart rate system hardware

Our heart rate system contains 4 main hardware components. These components include:

1. GY-MAX30100 Heart Rate Sensor
2. ST7735 TFT Display
3. HC-05 Bluetooth Module
4. STM32F401CCU6 Microcontroller

In this section we will be discussing each component in details.

2.1.1 GY-MAX30100 Heart Rate Sensor

The speed of the heartbeat also knowing as heart rate (HR) is measured by the number of beats the heart can produce per minute (bpm) [21]. HR can differ based on the body's needs like: the need of more oxygen and excretion of carbon dioxide. Usually, it is determined by the activity of the sinoatrial node (SA node) in the wall of the right atrium [21]. The SA node works automatically due to the spontaneous changes in Ca^{++}, Na^{+}, and K^{+} conductance. These changes are resulting from some activities such as: physical exercise, anxiety, stress, and drugs. The normal range of human HR is from 60-100 bpm when at rest. HR will decrease below the normal range mainly due to the activation of the vagus nerve, which innervates the SA node. Usually, at rest, there is a significant vagal tone on the SA node that causes the resting heart rate to be between 60 - 80 beats/min. A dysfunction in the SA node can lead to many heart diseases such as: sinus Bradycardia, sinus Tachycardia, or sick- sinus syndrome.

Human pulse is found from many points on the body by pressing the artery with the index and median fingers thus; the pulse of the artery moves to the surface and can be measured [22]. The pressing is often done against the underlying structures such as bone. While measuring the pulse rate, the thumb should not be used because its strong pulse may interfere with the right perception of the target pulse. You can measure the HR from: carotid artery (neck), femoral artery (thigh), aortic artery (behind the knee), abdominal aorta (on the abdomen), top of the heart (chest), superficial temporal artery (temple) and facial artery (lateral margin of the lower jaw) [22].

In this project, GY-MAX30100 Heart Rate Sensor is used to sense the heartbeats (shown in figure 2.1). The MAX30100 is an integrated pulse oximetry and heartrate monitor sensor solution. It combines two LEDs (one emitting a red light, another emitting infrared light), a photodetector, optimized optics, and low-noise analog signal processing to detect pulse oximetry and heart-rate signals. To detect heart, only the infrared light is needed. Both the red light and infrared light is used to measure oxygen levels in the blood. When the heart pumps blood, there is an increase in oxygenated blood as a result of having more blood. As the heart relaxes, the volume of oxygenated blood also decreases. Ultimately, by knowing the time between the increase and decrease of oxygen-rich blood, the device calculates the pulse rate. It turns out, oxygenated blood absorbs more infrared light and passes more red light while deoxygenated blood absorbs red light and passes more infrared light. This is the main function of the MAX30100: it reads the absorption levels for both light sources and stored them in a buffer that can be read via I2C.

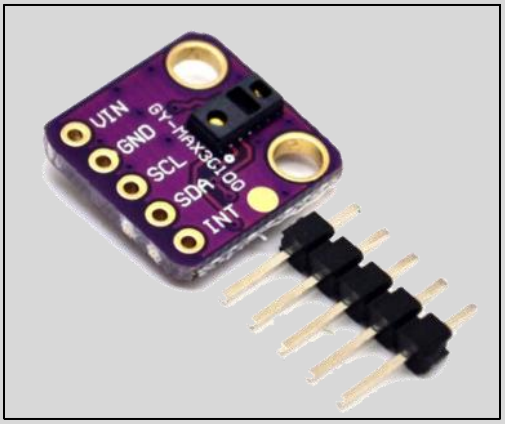

Figure 2.1: GY-MAX30100 Heart Rate Sensor

2.1.2 ST7735 TFT Display

ST7735 TFT display (shown in figure 2.2) can display a wide range of colors. It uses the SPI protocol for communication and has its own pixel-addressable frame buffer which means it can be used with all kinds of microcontroller and you only need 4 I/O pins. it also comes with an SD card slot on which colored bitmaps can be loaded and easily displayed on the screen.

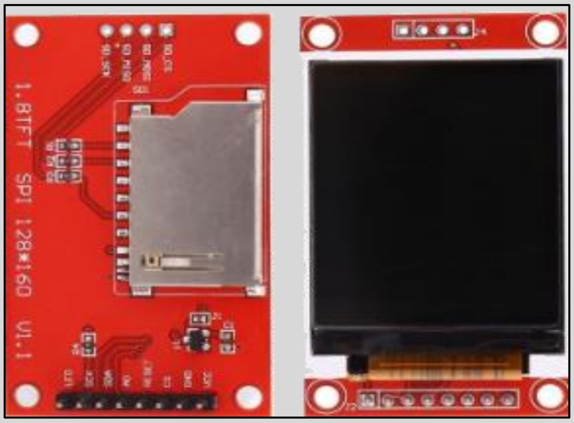

Figure 2.2: ST7735 TFT display

Main features of the display include:

- 1.8″ diagonal LCD TFT display
- 128×160 resolution, 18-bit (262,144) color
- 4 or 5 wire SPI digital interface
- Built-in microSD slot – uses 2 more digital lines
- 5V compatible! Use with 3.3V or 5V logic
- Onboard 3.3V @ 150mA LDO regulator
- 2 white LED backlight, a transistor connected so you can PWM dim the backlight
- 1×10 header for easy breadboarding
- 4 x 0.9″/2mm mounting holes in corners
- Overall dimensions: 1.35″ x 2.2″ x 0.25″ (34mm x 56mm x 6.5mm)
- Current draw is based on LED backlight usage: with full backlight draw is ~50mA

2.1.3 HC-05 Bluetooth Module

HC-05 is a Bluetooth module (designed for wireless communication) is an easy-to-use Bluetooth SPP (Serial Port Protocol) module. This module can be used in a master or slave configuration. It Works with UART communication protocol.

HC-05 (shown in figure 2.3) has a red LED which indicates connection status (whether the Bluetooth is connected or not). Before connecting, this red LED blinks continuously in a periodic manner. When it gets connected to any other Bluetooth device, its blinking slows down to two seconds. The data transfer rate can vary up to 1Mbps is in the range of 10 meters. This module works on 3.3V. We can connect 5V supply voltage as well since the module has on board 5 to 3.3 V regulator.

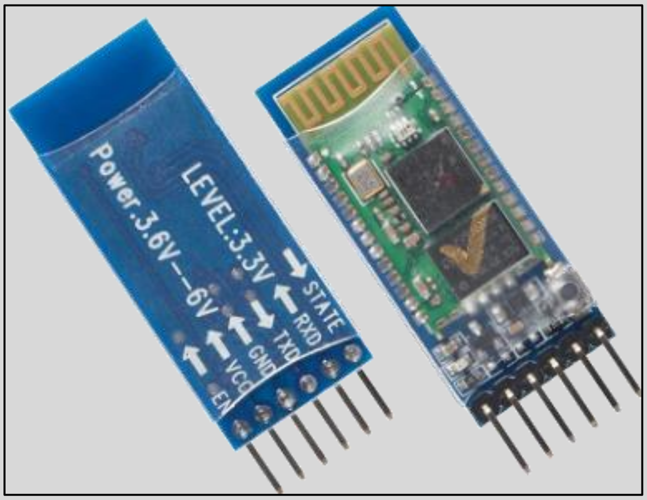

Figure 2.3: HC-05 is a Bluetooth module

In order to transmit the heart rate measurements using HC-05, we need to configure it as Master. To make such a configuration, we used a USB-TTL module converter alongside HC-05 Bluetooth module and a serial terminal software called Tera Term.

Configuration steps include the following:

1. Connect the two modules together as shown in figure 2.4.

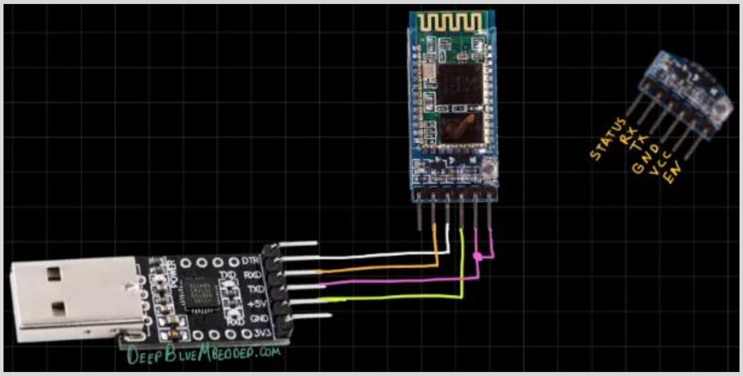

Figure 2.4: HC-05 Bluetooth module and USB-TTL module connection

2. Open Tera Term tool and choose the following configurations shown in figure 2.5.

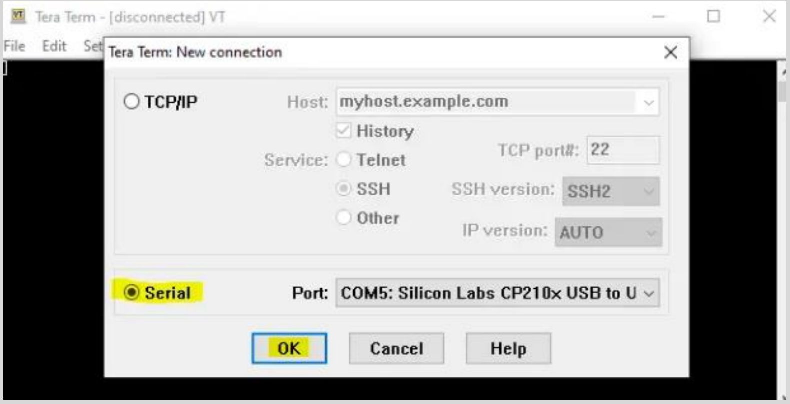

Figure 2.5: Tera Term tool configurations

3. Next, Go to Setup -> Terminal and choose the following parameters shown in figure 2.6.

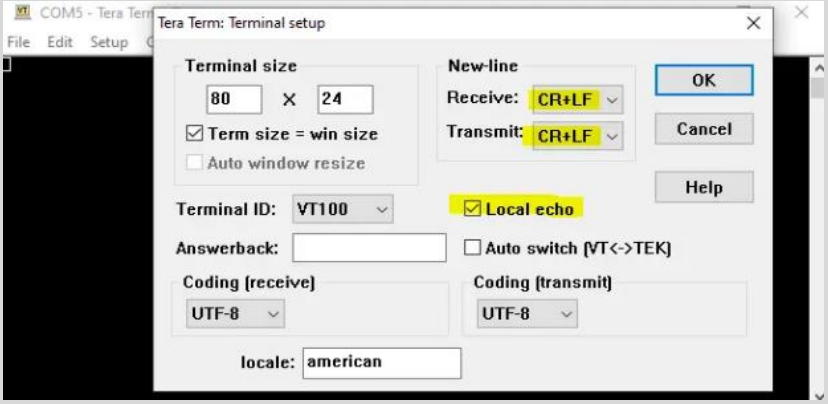

Figure 2.6: Tera Term tool terminal parameters

4. After that, Go to Setup -> Serial Port and choose 38400 Baud rate as shown in figure 2.7.

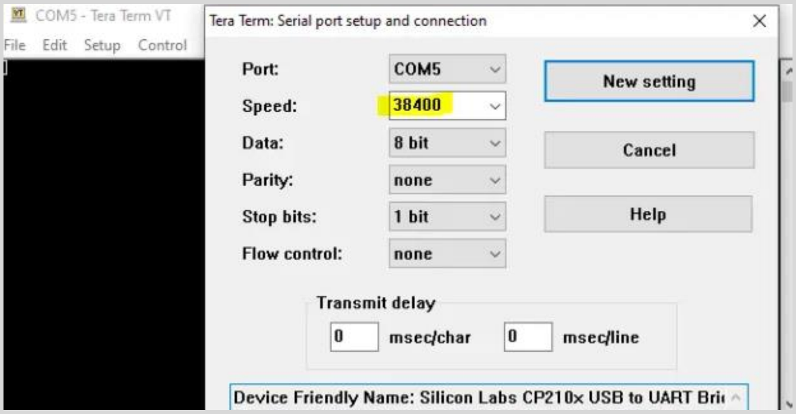

Figure 2.7: Tera Term tool Serial Port Baud Rate

5. Finally, change HC-05 mode to master: by typing "AT+ROLE=1" in terminal below as shown in figure 2.8.

Figure 2.8: Tera Term tool terminal change mode - AT command

2.1.4 STM32F401CCU6 Microcontroller

STMicroelectronics (STMicroelectronics) Group was founded in June 1987 as a merge between the Italian SGS Thomson Microelectronics and French semiconductor companies. Since 1999, ST has been one of the world's top ten semiconductor companies. STM32 is a family of 32-bit microcontroller integrated circuits by STMicroelectronics. The STM32 family consists of seven series of microcontrollers: F4, F3, F2, F1, F0, L1, L0, and W. Each STM32 microcontroller series is based upon a Cortex-M4F, Cortex-M3, Cortex-M0 +, or Cortex-M0 ARM processor core. The Cortex-M4F is conceptually a CortexM3 plus DSP and single-precision floating-point instructions. The STM32 F4-series is the first group of STM32 microcontrollers based on the ARM Cortex-M4F core. It was released in September 2011.

Main features of this series include:

- Core: ARM Cortex-M4F core works at a maximum clock rate of 180 MHz.
- Memory: Static RAM consists of up to 192 KB.
- Flash: consists of up to 2048 KB
- Oscillators: consists of internal (16 MHz, 32 kHz), external (4 to 26 MHz, 32 to 1000 kHz)

From this series we will be using STM32F401CCU6 development board (shown in figure 2.9).

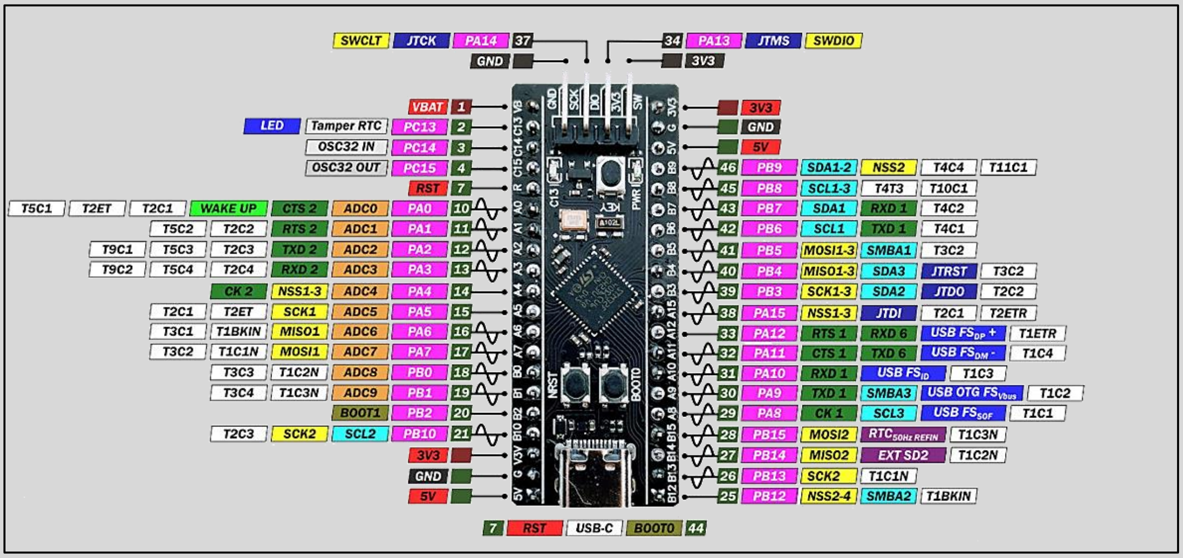

Figure 2.9: STM32F401CCU6 development board

Main features of this board include the following:

- Core: ARM 32 Cortex-M4 CPU.
- Debug mode: SWD (can be programmed with ST-LINK V2)
- CPU Internal Maximum Clock: 84MHz
- External Clock Source: 25MHz crystal
- 256K flash memory, 64K SRAM
- Connectivity: 3 I^{2}C, IrDA, LIN bus, SDIO,3 SPI, 3 UART/USART, USB OTG
- Peripherals: Brown-out Detect/Reset, DMA, I^{2}S, POR, PWM, WDT
- 2.0-3.6V power, I/O
- 36 I/O pins
- A/D Converters

2.2 Heart rate system software

In this section, we will be discussing the logical flow diagram of our heart rate system in details. A flow diagram shown in figure 2.10 demonstrates the logic that is used to program the heart rate system. First, readings of Red and Infra-red signals that was sampled and processed using Max30100 will be sent to STM32F401CCU6 microcontroller using I2C communication protocol. These signals will be covered to heartbeats. After that, they will be displayed on TFT screen (using SPI communication protocol). Also, they will be sent using HC-05 Bluetooth module (using UART protocol) to the main control system that contains Raspberry Pi microcontroller that will be taking some actions depending on the values.

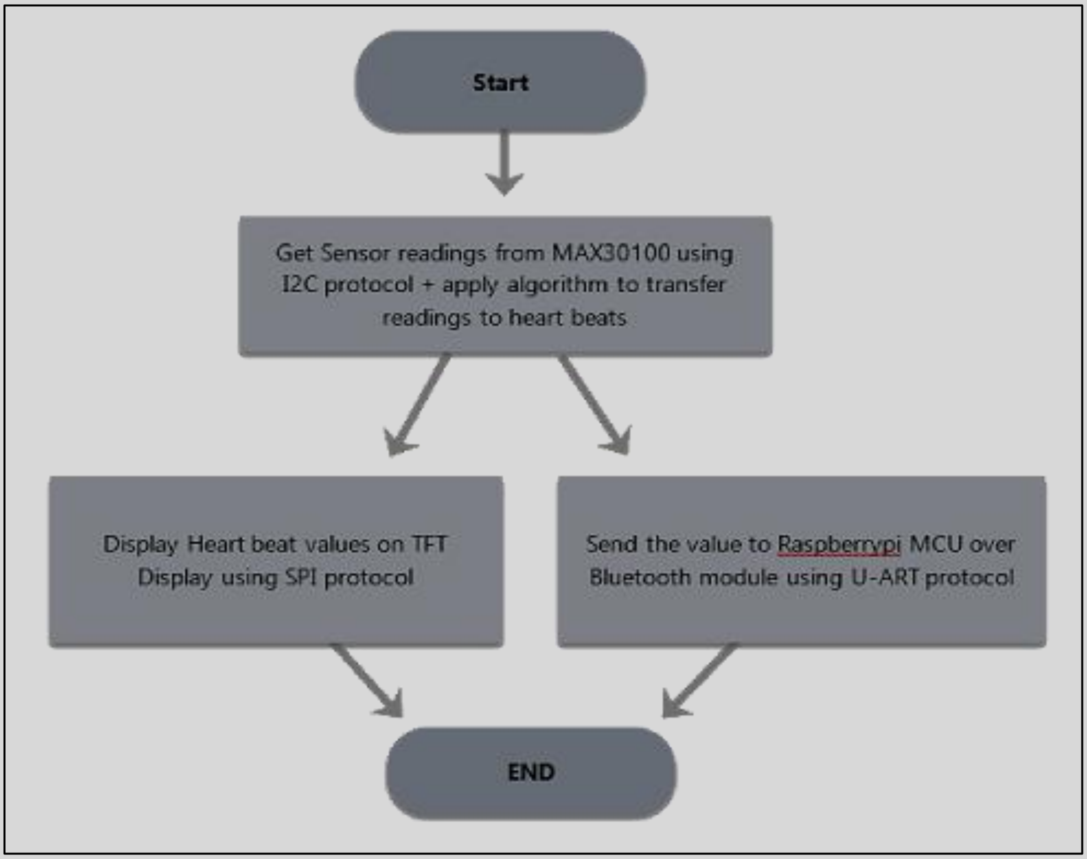

Figure 2.10: Heart rate system flow diagram

2.3 Main control system hardware

Our Main control system contains 4 main hardware components. These components include:

1. Buzzer
2. Push Button
3. GPS NEO-6m Module
4. GSM-GPRS SIM900 Module
5. Raspberry Pi Microcontroller

In this section we will be discussing each component in details.

2.3.1 Buzzer

A buzzer (shown in figure 2.11) is an audio signaling device, which may be mechanical, electromechanical, or piezoelectric (piezo for short). Typical uses of buzzers and beepers include alarm devices, timers, train and confirmation of user input such as a mouse click or keystroke.

In our system we will be using piezo buzzer to give the driver an alert when his heart rate is abnormal. If the driver does not terminate the alert within 30 seconds, a signal will be send to STM32F401CCU6 microcontroller which in return slows down the car movement and start parking. During this operation, GPS NEO-6M module will detect the driver's location and send an e-mail to his family members.

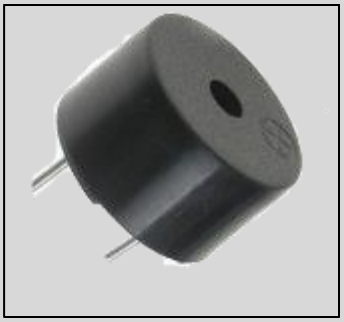

Figure 2.11: Buzzer

2.3.2 Push Button

A push-button (shown in figure 2.12) or simply button is a simple switch mechanism to control some aspect of a machine or a process. Buttons are typically made out of hard material, usually plastic or metal. The surface is usually flat or shaped to accommodate the human finger or hand, so as to be easily depressed or pushed.

In our system we will be using the push button as an interrupt to stop the alert and let driver continue driving the car if his condition was not serious/critical.

Figure 2.12: Push Button

2.3.3 GPS NEO-6m Module

The GPS NEO-6M module (shown in figure 2.13) is a compact, low-power GPS module designed to provide accurate location and time information. We used this Module to get the driver's Location and use it to send an SMS help message to his family members.

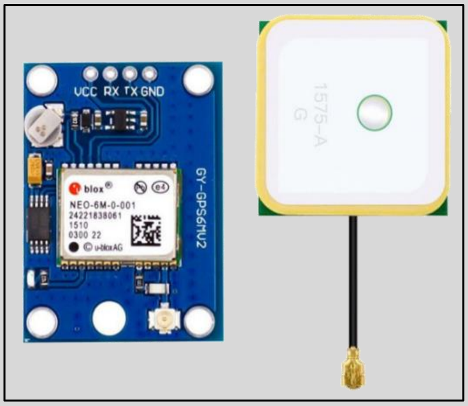

Figure 2.13: GPS NEO-6m Module

Some key features and notes about the module:

- It uses a high-performance GPS chipset, which provides high accuracy and fast positioning.
- It supports multiple satellite navigation systems, including GPS, GLONASS, and QZSS.
- It communicates over a serial interface (UART) with a default baud rate of 9600 bps.
- It designed to operate from a single 3.3V to 5V supply.
- It provides a variety of information including latitude, longitude, altitude, speed, and time.
- Widely used in various applications such as vehicle tracking, personal navigation, and drone positioning, among others.

To interface this module with our Raspberry pi, we will use the following Linux commands:

```
sudo nano /boot/config.txt

dtparam=spi=on

dtoverlay=pi3-disable-bt

core_freq=250

enable_uart=1

force_turbo=1

sudo cp /boot/cmdline.txt /boot/cmdline_backup.txt

sudo nano /boot/cmdline.txt

dwc_otg.lpm_enable=0 console=tty1 root=/dev/mmcblk0p2 rootfstype=ext4elevator=deadline
fsck.repair=yes rootwait quiet splash plymouth.ignore-serial-consoles

sudo reboot

sudo cat /dev/ttyAMA0

ls -l /dev

sudo systemctl stop serial-getty@ttyAMA0.service
```

2.3.4 GSM-GPRS SIM900 Module

GPRS/GSM Module-is a compact and reliable wireless module (shown in figure 2.14). This module can do almost anything a normal cell phone can do, such as sending SMS messages, making phone calls, and connecting to the Internet via GPRS. It is configured and controlled via UART using simple AT commands.

Figure 2.14: GSM-GPRS SIM900 module

Some key features and notes about the module:

- Fully compatible with AVR/PIC/Arduino/ARM/FPGA
- Quad-Band 850/ 900/ 1800/ 1900 MHz
- Control via AT commands
- Supply voltage range: 3.1 - 4.2V

2.3.5 Rspberry Pi Microcontoller

Raspberry Pi is a development board in PI series. It can be considered as a single board computer that works on LINUX operating system. The board not only has tons of features it also has terrific processing speed making it suitable for advanced applications.

The Raspberry Pi 3 Model B+ (shown in figure 2.15) is the latest product in the Raspberry Pi 3 range, boasting a 64-bit quad core processor running at 1.4GHz, dual-band 2.4GHz and 5GHz wireless LAN, Bluetooth 4.2/BLE, faster Ethernet, and PoE capability via a separate PoE HAT.

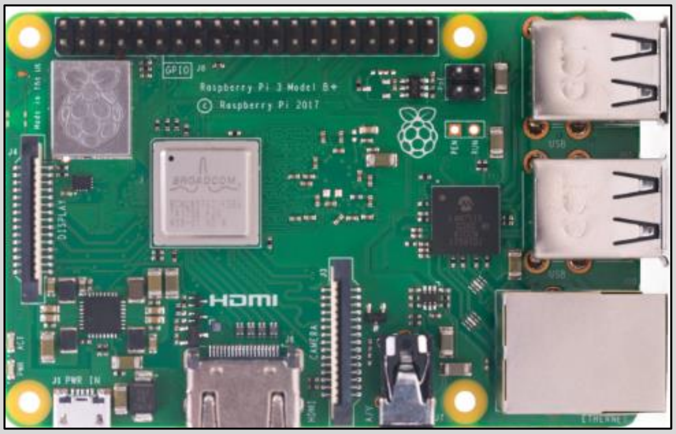

Figure 2.15: Raspberry Pi 3 Model B+

In our system the raspberry pi is the main controller which controls our whole system. First, it gets the heart rate sensor readings and checks if it is normal or not. If the readings were abnormal, it sends a signal to STM32F401CCU6 board of the car system to start parking. Also, it sends the location of the driver using SMS message to his family members.

2.4 Main control system software

In this section, we will be discussing the logical flow diagram of our main control system in details. A flow diagram shown in figure 2.16 demonstrates the logic that is used to program the system. First, Raspberry Pi microcontroller will get the sensor readings and check whether they are normal or abnormal. If they were normal, the driver continues driving the car smoothly. On the other hand, if they were abnormal an audio alert will start playing. If the driver couldn't terminate the alarm within 30 seconds, an interrupt signal will be sent to the STM32F401CCU6 microcontroller of the car prototype system to initiate parking the car. Also, the location of the driver will be sent via SMS message to his family members.

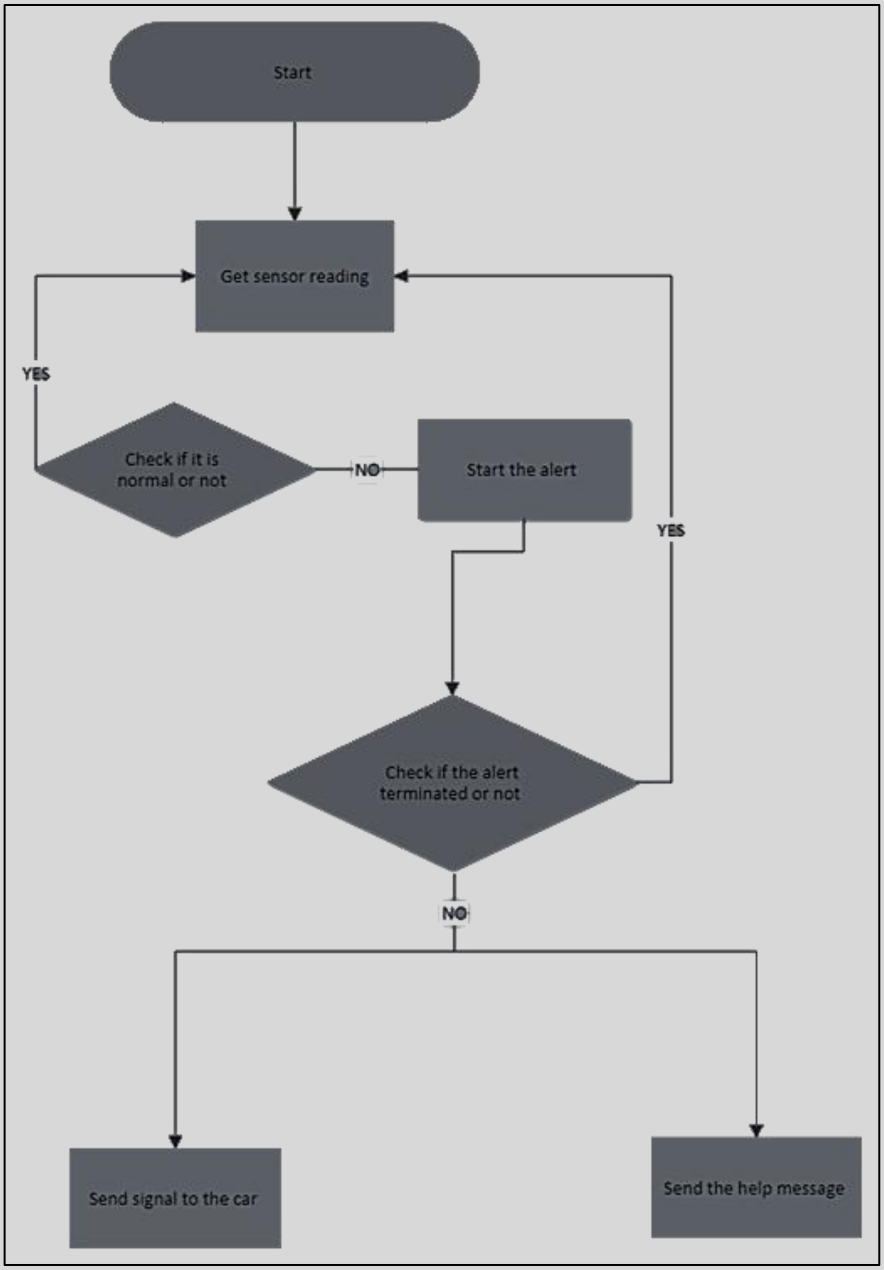

Figure 2.16: Main control system flow diagram

2.5 Car prototype system hardware

Our car prototype system contains 5 main hardware components. These components include:

1. 4 DC Motors with 4 wheels
2. 4 HC-SR04 ultrasonic sensors

3. 2 L298N dual H-Bridge
4. HC-05 Bluetooth Module
5. STM32F401CCU6 Microcontroller

In this section we will be discussing each component in details.

2.5.1 DC Motors and wheels

A DC motor is an electrical machine that converts electrical energy into mechanical energy. In a DC motor, the input electrical energy is the direct current which is transformed into the mechanical rotation.

How does a DC motor work? when kept in a magnetic field, a current-carrying conductor gains torque and develops a tendency to move. In short, when electric fields and magnetic fields interact, a mechanical force arises. This mechanical force will cause the wheel attached to the motor to rotate (shown in figure 2.17).

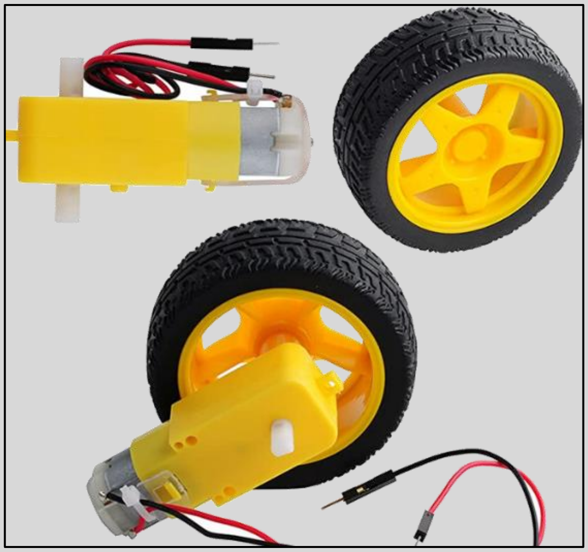

Figure 2.17: wheel attached to DC motor

2.5.2 HC-SR04 ultrasonic sensors

HC-SR04 ultrasonic distance sensor (shown in figure 2.18) provides 2cm to 400cm of non-contact measurement functionality with a ranging accuracy that can reach up to 3mm. Each module includes an ultrasonic transmitter, a receiver and a control circuit. The module has two eyes like projects in the front which forms the Ultrasonic transmitter and Receiver. The Ultrasonic transmitter transmits an ultrasonic wave, this wave travels in air and when it gets objected by any material it gets reflected back toward the sensor this reflected wave is observed by the Ultrasonic receiver module.

Figure 2.18: HC-SR04 ultrasonic sensors

2.5.3 L298N dual H-Bridge

The L298N is a dual H-Bridge motor driver (shown in figure 2.19) which allows speed and direction control of two DC motors at the same time. The module can drive DC motors that have voltages between 5 and 35V, with a peak current up to 2A.

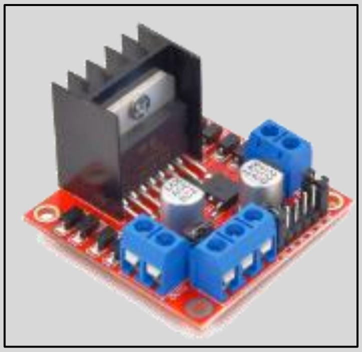

Figure 2.19: L298N Dual H Bridge Stepper Motor Driver

2.5.4 HC-05 Bluetooth Module

Previously discussed in section 2.1.3 page 9.

2.5.5 STM32F401CCU6 Microcontroller

Previously discussed in section 2.1.4 page 11.

In this system, STM32F401CCU6 board will be used to control the speed of the car in two modes. In the First mode (Normal mode), pulse width modulation signals generated by STM32 board will control the direction and the speed of the car depending on commands sent from a mobile application and received by HC-05 Bluetooth module using UART communication protocol. On the other hand, in the second mode (Emergency mode) pulse width modulation signals generated by STM32 board will control the direction and the speed of the car depending on the measured distance by ultrasonic sensors to safely park the car.

2.6 Car prototype system software

In this section, we will be discussing the logical flow diagram of our car prototype system in details. A flow diagram shown in figure 2.20 demonstrates the logic that is used to program the system.

In normal condition (mimicking normal driving), the car is driven using a Bluetooth module that receives instructions from a mobile application and starts an interrupt service routine (ISR). Instructions include: increasing the speed by 20% after receiving character “F”, decreasing the speed by 20% after receiving character “B”, sliding right with same speed when receiving “R”, sliding left with same speed when sending “L”, and stopping the car by receiving character “J”.

In abnormal condition, the main control system sends a low signal (initiates interrupt) to STM32F401CCU6 board that starts executing an ISR to park the car. In this mode, ultrasonic sensors on right, back and front will check the distance continuously. Depending on these measurements, the car will start sliding to the right side of the road. Then, by decreasing speed on specific periods of time, the car starts to brake until it stops and parks safely.

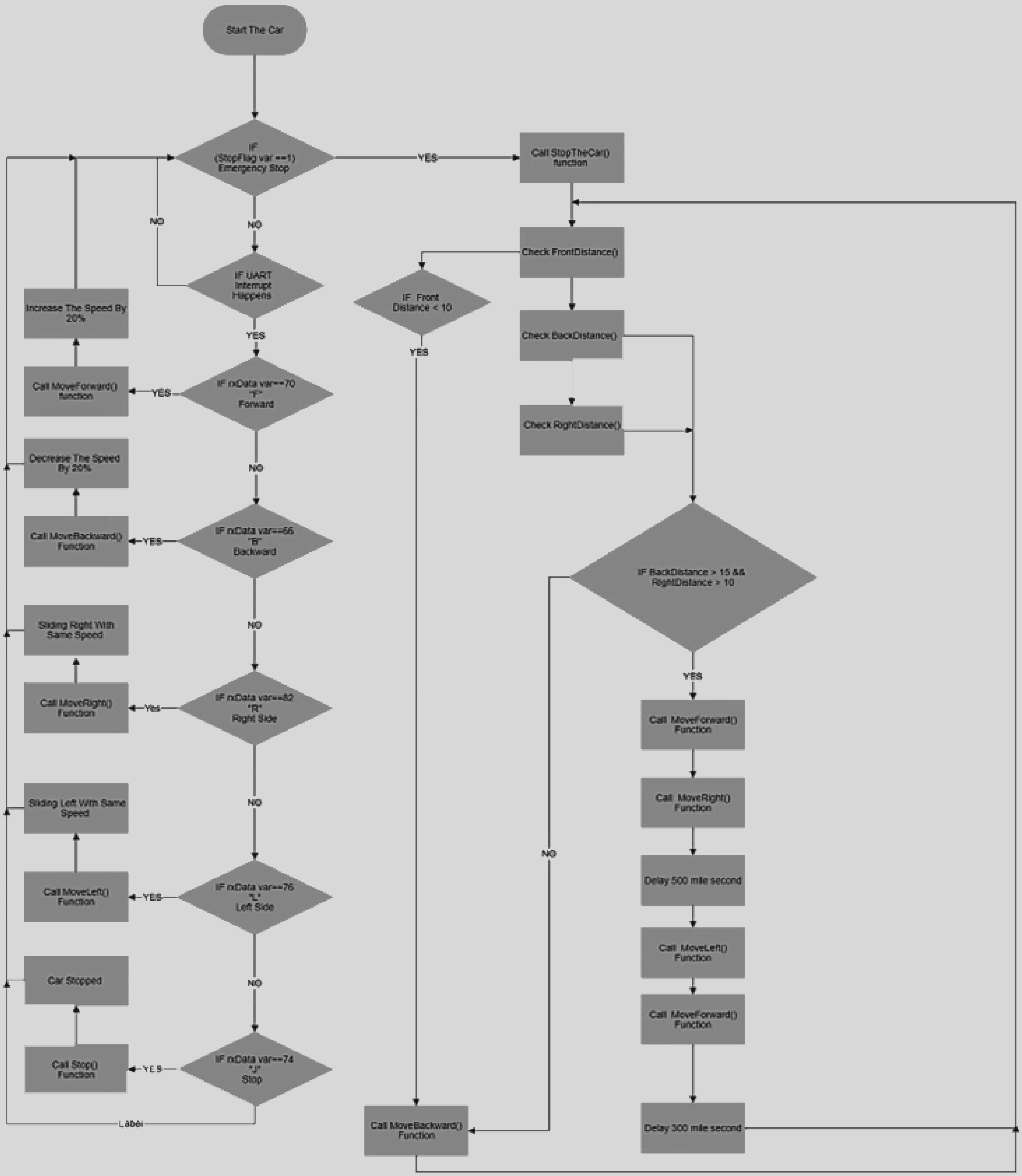

Figure 2.20: Car prototype system flow diagram

Chapter Three: Final Prototype and Results

3.1 Final Prototype

The final prototype of our heart rate emergency system will be composed of 2 main parts. The first part is a box which includes all the components of the heart rate system (shown in figure 3.1). This prototype has a bottom opening to allow GY-MAX30100 heart rate sensor to contact directly with the skin to sense the heartbeats. The top of the box is mainly composed of ST7735 TFT display to show the value of the heartbeats. Inside the box, remaining components of the heart rate system such as STM32F401CCU6 microcontroller, HC-05 Bluetooth module, and batteries are aligned together and attached using a PCB circuit (shown in figure 3.2).

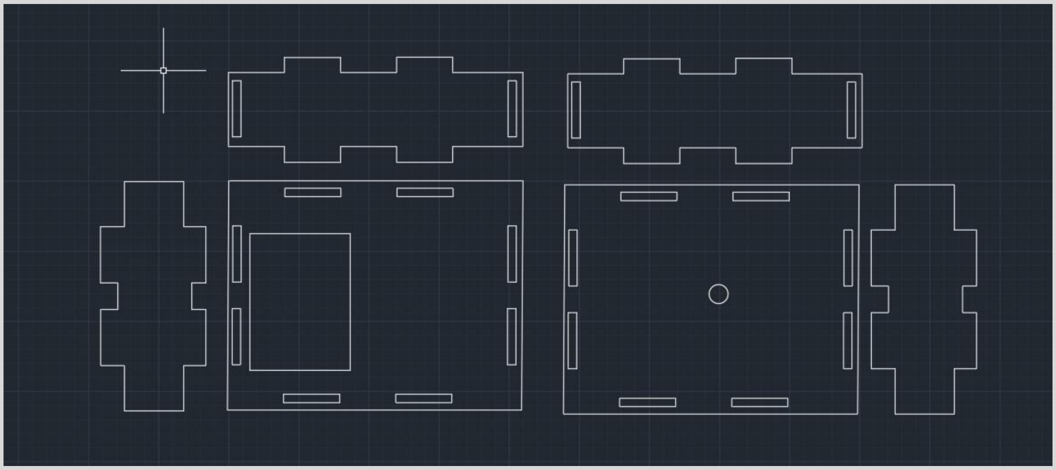

Figure 3.1: heart rate system 2D AUTOCAD design

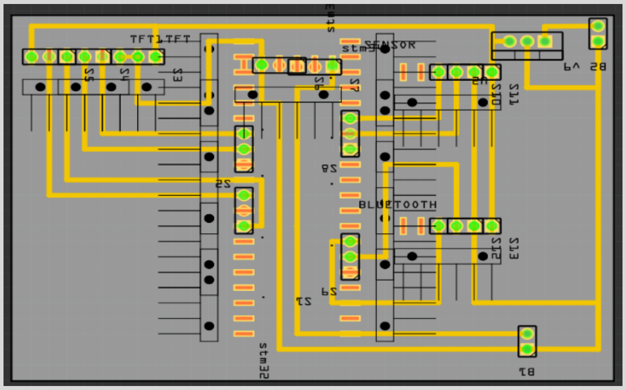

Figure 3.2: heart rate system PCB

The second part is a car prototype which contains both the main control unit and the car system. This prototype is composed of the following: Raspberry Pi microcontroller, buzzer, push button, GPS NEO-6m Module, HC-05 Bluetooth module, STM32F401CCU6 microcontroller, two L298N dual H-Bridge, four HC-SR04 ultrasonic sensors, four DC Motors and four wheels and batteries. Figure 3.3 shows the AUTOCAD 2D designs that were made for the car prototype. Finally, figure 3.4 shows the final car prototype system.

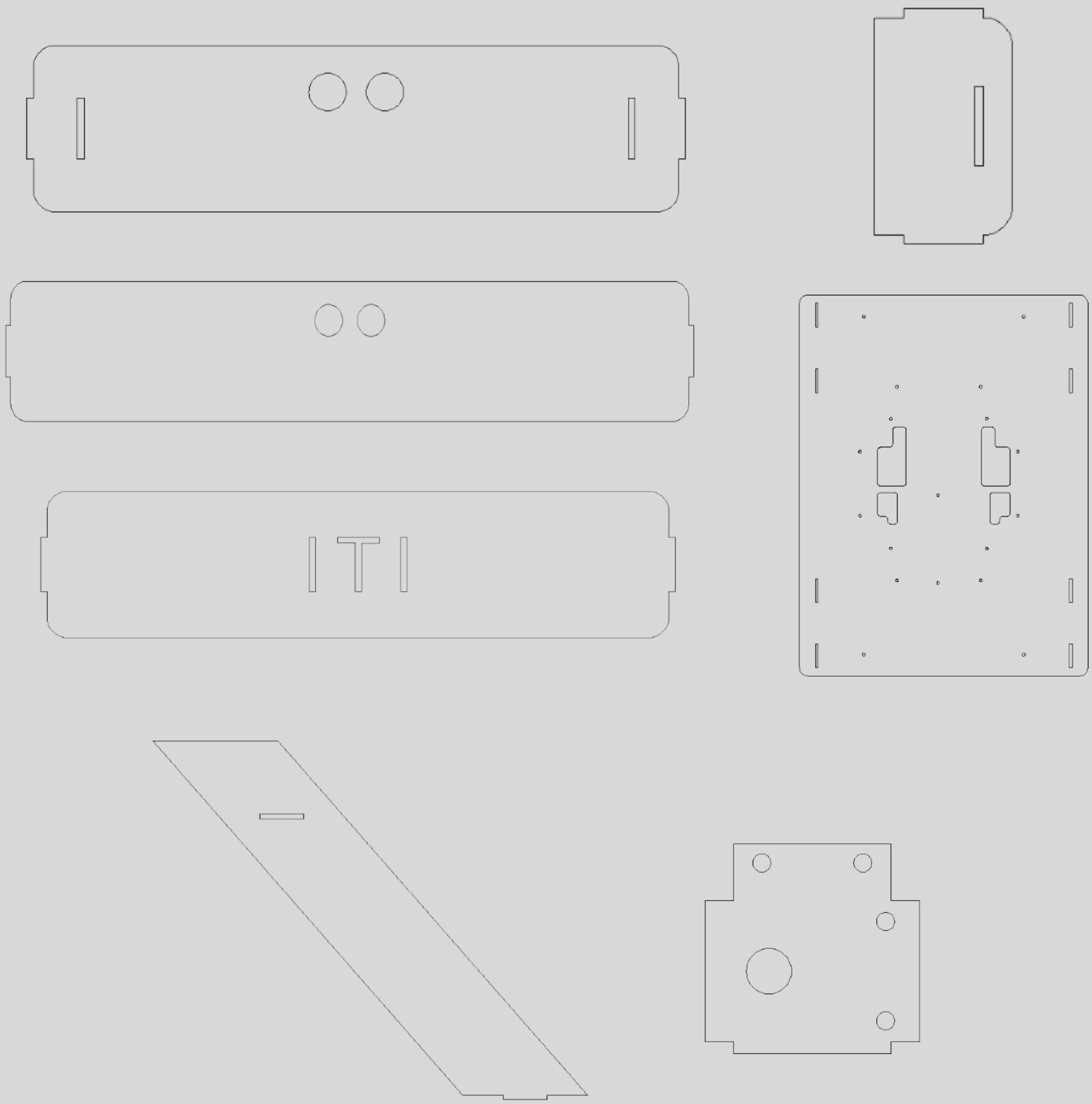

Figure 3.3: Car prototype 2D AUTOCAD design

Figure 3.4: Final Car prototype system

3.2 Results

In this section, we will display the results we got from the 2 main parts of our system.

For the heart rate system, figure 3.5 displays the heartbeat values that were acquired by MAX30100 heart rate sensor and displayed on ST7735 TFT display. The values are within the normal range of human heart rate (60-100 PBM) when the sensor is in direct contact with the skin. In case if there was no contact with the body, no heart rate is detected as shown in figure 3.6.

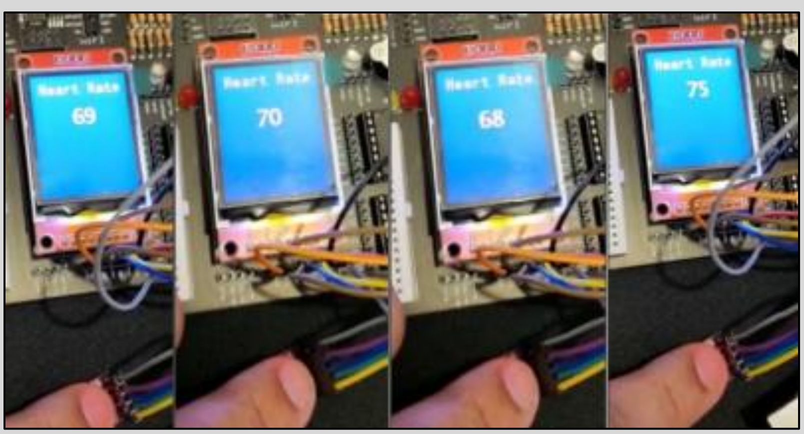

Figure 3.5: heart beat values (in contact)

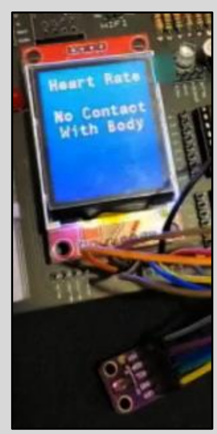

Figure 3.6: heart beat values (no contact)

In the car prototype system, heartbeat values will be sent to Raspberry Pi microcontroller in the to check if they exceeded the normal range or not. In case if the values were normal, the car prototype will be controlled by Serial Bluetooth terminal mobile application as shown in figure 3.7. This application is a line-oriented terminal / console app for microcontrollers with a serial / UART interface connected with a Bluetooth to serial converter to android device. A video that demonstrates the Serial Bluetooth terminal controlling the car movement can be found in: https://drive.google.com/drive/folders/1NXnnQRKjPQ8fyEFTNV7T88ZYiZc9R_ym

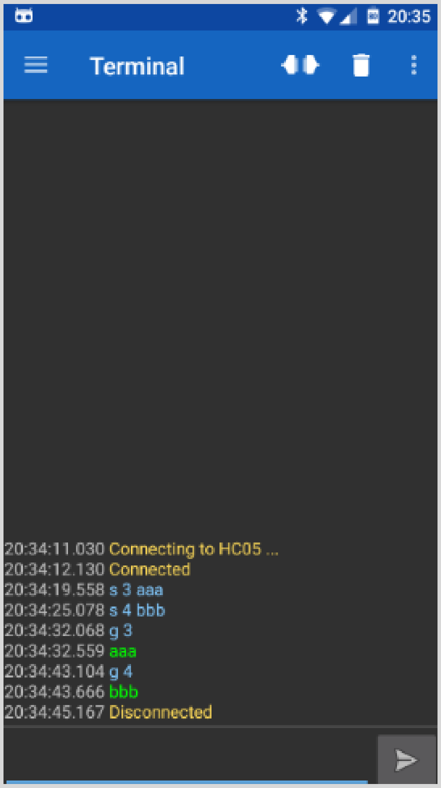

Figure 3.7: Serial Bluetooth terminal mobile application

In case if values were abnormal, an alarm will start to play. If the push button was pressed within 30 seconds, the alarm will stop and the car will continue moving normally using the mobile application. Contrary, if it wasn't pressed, external interrupt signal will be sent to STM32F401CCU6 microcontroller to start the parking sequence. A video that demonstrates the parking sequence can be found in: https://drive.google.com/drive/folders/1NXnnQRKjPQ8fyEFTNV7T88ZYiZc9R_ym

Finally, SMS help message with the driver's location will be sent to a prespecified phone numbers as shown in figure 3.8.

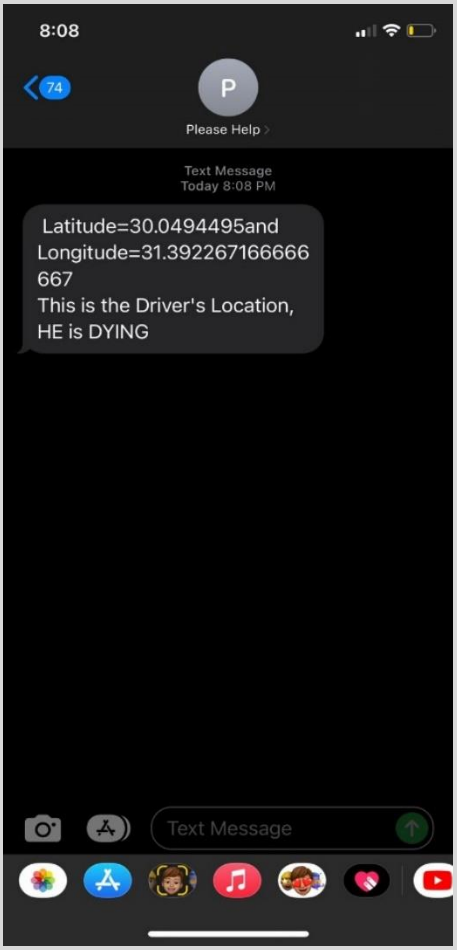

Figure 3.8: SMS help message

Chapter Four: Conclusion

The emergence of wearable devices such as the Apple Watch is considered to be a turning point in the application of heart-rate detection while driving vehicles. Wearable devices such as wristwatches, which can be worn easily without restraining the driver, are responsible for heart-rate detection alone. In this project we deliver a heart rate emergency system that is capable of measuring the driver's heart rate and taking actions based on these measurements. The main features of our system include the following:

- System to sense the driver's heart rate and display it on a screen.
- If heart rate exceeds a defined range, an alarm will start playing. Now we will have to modes:
    - In normal mode, when the driver can stop the alarm, he will continue driving.
    - In emergency mode, when the driver can’t stop the alarm, autonomous driving will control the car until it parks safely and SMS help message with location will be sent to family members.

Of course, we faced some difficulties throughout the project. One was related to the components because some of them were not available, or were very expensive. In addition, some errors in the chosen components were discovered during practical work. We tried to solve this problem by finding other components with the same desired features. So, all these disruptions affected our workflow but we managed to finish on time with all the main features we planned to have from the beginning.

For future improvements, we planned some ideas. First, adding extra sensors to measure parameters like temperature, blood glucose, blood pressure, and SpO2. Second, using more reliable and accurate sensors. Third, adding extra ultrasonic sensors to increase safety while parking the car. Finally, updating the car system over the air to add more features.

Our ultimate goal is to use our heart rate emergency system inside every vehicle to prevent damages that are caused by sudden changes in the physical conditions of drivers.

References

1. Bernie DeGroat, Traffic Deaths Considerable Compared with Leading Causes of Death. [(accessed on 1 June 2021)]. Available online: https://news.umich.edu/traffic-deaths-considerable-compared-with-leading-causes-of-death/

2. Kwon S., Jung C., Choi T., Oh Y., You B. Autonomous Emergency Stop System. IEEE Intell. Veh. Symp. Proc. 2014:444–449. doi: 10.1109/IVS.2014.6856482. [CrossRef] [Google Scholar]

3. Takano M., Morimoto K., Takagi M., Oda T., Nishimura N. Development of an Emergency Stop Assistant System. SAE Tech. Pap. 2019;1025:1. doi: 10.4271/2019-01-1025. [CrossRef] [Google Scholar]

4. LEXUS Homepage, LEXUS Safety Technology. [(accessed on 7 July 2021)]. Available online: https://www.lexus.com/safety

5. Subaru Homepage, Safety Preventive Safety: Eyesight. [(accessed on 6 June 2021)]. Available online: https://www.subaru.jp/levorg/levorg/safety/safety2_2

6. Hino Motors Ltd. Hino Motors Develops World's First Emergency Driving Stop System (EDSS) for Commercial Vehicles to Be Launched on the Hino S'ELEGA This Summer. [(accessed on 31 July 2021)]. Available online: https://www.hino-global.com/corp/news/2018/20180521.html

7. Mitsubishi Fuso Truck and Bus Corp Mitsubishi Fuso Releases 2019 Model Year Aero Queen and Aero Ace Large Coach Buses. [(accessed on 30 July 2021)]. Available online: https://www.mitsubishi-fuso.com/news/2019/02/21/mitsubishi-fuso-releases-2019-model-year-aero-queen-and-aero-ace-large-coach-buses/

8. Cadillac Homepage. Automatic Emergency Braking. [(accessed on 31 July 2021)]. Available online: https://my.cadillac.com/how-to-support/safety/automatic-emergency-braking#:~:text=Helps%20Alert%20and%20Assist%20You,at%20speeds%20below%2050%20mph.

9. BMW Group Homepage. Stopping Safely in an Emergency. [(accessed on 31 July 2021)]. Available online: https://www.press.bmwgroup.com/global/article/detail/T0022635EN/stopping-safely-in-an-emergency?language=en
10. Kelly Blue Book How Mercedes-Benz Active Emergency Stop Assist Works. [(accessed on 31 July 2021)]. Available online: https://www.kbb.com/car-news/how-mercedes-benz-active-emergency-stop-assist/
11. Audi MediaCenter Driver Assistance Systems. [(accessed on 31 July 2021)]. Available online: https://www.audi-mediacenter.com/en/technology-lexicon-7180/driver-assistance-systems-7184
12. Study Group on Analysis of Factors Affecting Traffic Accidents in the Automobile Transport Business, Automobile Bureau, Ministry of Land, Infrastructure, Transport and Tourism, Health Management Manual for Drivers of Commercial Vehicles. [(accessed on 6 June 2021)]; Available online: https://wwwtb.mlit.go.jp/tohoku/jg/manual_kenkoukannri.pdf (In Japanese)
13. Arakawa T. Application of Machine Learning for Driver State Detection Technology. Automot. Technol. 2021 in press (In Japanese) [Google Scholar]
14. ROHM Co. Ltd. Pulse Wave Sensor. [(accessed on 1 September 2021)]. Available online: https://www.rohm.co.jp/electronics-basics/sensors/sensor_what3 (In Japanese)
15. Check Your Heart Rate on Apple Watch. [(accessed on 16 July 2021)]. Available online: https://support.apple.com/guide/watch/heart-rate-apda88aefe4c/watchos
16. Get Started with Your New E4 Wristband. [(accessed on 25 August 2021)]. Available online: https://www.empatica.com/get-started-e4
17. Heart Rate Transfer Mode: How to Set and Use. [(accessed on 6 June 2021)]. Available online: https://support.garmin.com/ja-JP/?faq=8G7NCxDkuo6bUMb6YTY8j6 (In Japanese)

18. Fitbit, Fitbit Surge. [(accessed on 30 July 2021)]. Available online: https://www.fitbit.com/pl/shop/surge

19. Superwatches, Jawbone Smartwatches (UP, UP2, UP3, UP4) [(accessed on 30 July 2021)]. Available online: https://www.superwatches.com/jawbone-smartwatches/

20. Monitor Your Heart Rate with Apple Watch. [(accessed on 16 July 2021)]. Available online: https://support.apple.com/en-us/HT204666

21. R, Klabunde,"Control of Heart Rate," Image for Cardiovascular Physiology Concepts". [Online]. Available: http://www.cvphysiology.com/Arrhythmias/E010. [Accessed: 08-Dec-2017].

22. Wikipedia, "Heart rate," 11-Dec-2017. [Online]. Available: https://en.wikipedia.org/wiki/Heart_rate. [Accessed: 13-Dec-2017].