Radar Vital Signs Monitoring System

Technical Document

Contents

1. Introduction

Currently the main focus of mm-wave radar applications are geared towards the automotive market. However, this technology has been quickly digressing towards healthcare applications ranging from cancer imaging to glucose monitoring. One emerging application area is remote non-contact monitoring of human vital signs. This is important application in a sense of provide ease of use, contactless and cost-effective solution. The foundational phenomenon, that allows the measurement of these vital signs i.e. heart rate and breathing rate, is Doppler's effect and is summarized in figure 1. This is contrary to methods like ECG, that are contact based methods.

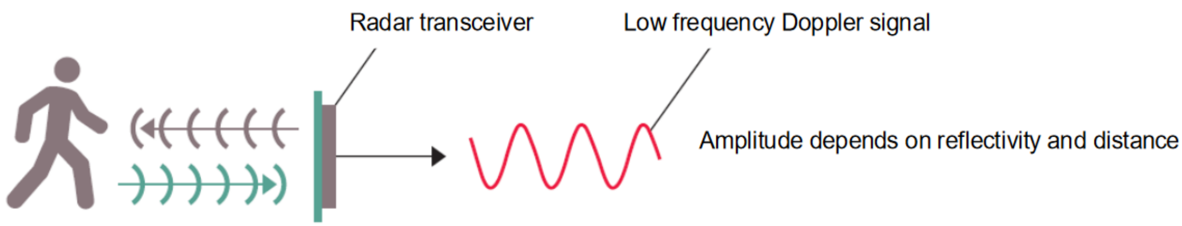


Figure 1 Under Doppler's effect, the reflected signal contains information of velocity and distance from the target

1.1. Problem Description

The problem with traditional vital signs measurement devices is that firstly they only measure either measure either heart rate or breathing rate. Secondly, they do not have ease of use in terms of following a physical procedure to operate the device to measure the corresponding vital sign. Thirdly, in wearable forms of these technology as watches, is only prone to a singular application, not flexible to diverse applications. Fourthly, there are possible security and privacy vulnerabilities in existing products from the communication

1.2. Objective

We wish to construct a vital sign monitoring as a product that measures the heart rate and breathing rate (referred to as vital signs) from a target human being, within the distance of 10cm-1m from the target, using radar technology. It is depicted in figure 2. The results of heart rate and breathing rate are wireless transmitted through Bluetooth low energy and shown on smartphone (Android and iOS) app.

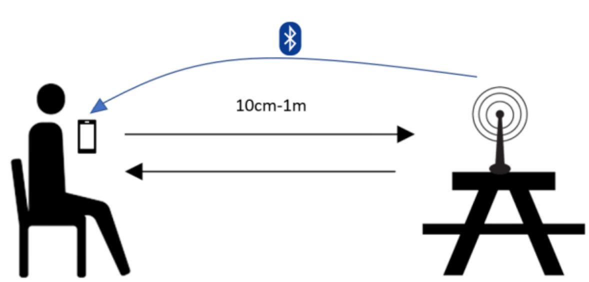


Figure 2 Radar product measures the vital signs of the person and transmits the results to smartphone that displays the results

2. Background

2.1. Continuous Wave Doppler Radar

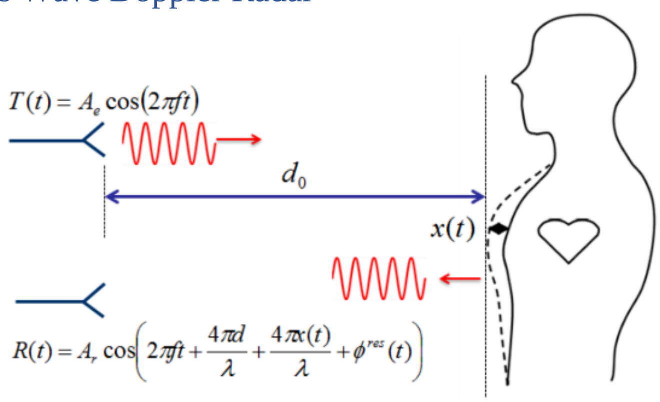


Figure 3 Representation of how radar measures the chest movements associated to breathing and heart beats

In a continuous wave (CW) Doppler radar vital-sign detection system, a sinusoidal signal $T(t) = A_e \cos(2\pi f t)$ at carrier frequency f is transmitted towards a human body, located at a certain distance d_o. The signal is reflected by the chest, whose movement $x(t)$ is due to both heart beating and respiration. The reflected signal $R(t)$ is demodulated by an IQ quadrature receiver to avoid null-point detection issues. The two baseband signals I and Q are of the form

$$I(t) = A_r \cos\left(4\pi\lambda x(t) + 4\pi\lambda d_o + \theta(t)\right)$$
$$Q(t) = A_r \sin\left(4\pi\lambda x(t) + 4\pi\lambda d_o + \theta(t)\right)$$

and are modulated by physiological movements x(t) of the human body. θ(t) is defined as total residual phase of the radar system. λ=5 mm is the wavelength at f=60 GHz. The physiological movements are represented by the sum of two single-tone sinusoidal signals, $x(t) = x_r(t) + x_h(t) = m_r \sin(2\pi f_r t + \phi_r) + m_h \sin(2\pi f_h t + \phi_h)$ m_r and m_h describe the movement amplitude of respiration and heartbeat, respectively, f_r and f_h represent the rate of movement, and ϕ_r and ϕ_h are the initial phases for each movement.

The above I and Q signals are referred to as in-phase and quadrature signals, and they are of main interest as far as the signal processing in concerned to extract heart and breathing movements.

2.2. BLE beacon

Bluetooth beacons are a low-cost, low-power, and location-based technology that uses the Bluetooth Low Energy (BLE) protocol in conjunction with small, BLE-enabled hardware devices (beacons). Bluetooth beacons are one-way transmitters to a smartphone or other device, which require a specific mobile application to be installed to interact with the beacons. There are two significant beacon standards: iBeacon, developed by Apple, and Eddystone, developed by Google. These small portable transmitters, of which there are various suppliers, can be installed indoors or outdoors and have a battery life of approximately three years. The beacons can broadcast Bluetooth signals containing several bytes of information together with their universally unique identifier (UUID) to their surroundings.

Bluetooth beacons with additional encryption layer are a common way to send sensor data. The reason for using beacon protocol instead of alternative Bluetooth serial protocols like RFCOMM is that this would eliminate the need for pairing, while ensuring security and privacy. Thus adoption of this would provide an time and energy efficient Bluetooth communication between two devices.

3. Methodology

3.1. Hardware

3.1.1. BGT24LTR11-XMC4700 (Sense2Gol Pulse)

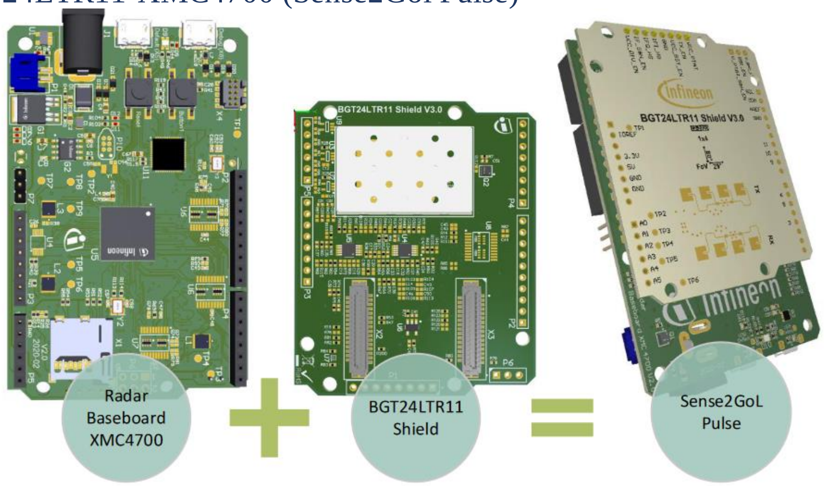


Figure 4 BGT24LTR11-XMC4700 Board

The Sense2GoL Pulse (S2GLP) radar demo kit is a demonstration platform for Infineon's silicon-germanium 24 GHz radar chipset BGT24LTR11. It consists of two boards: the main Radar Baseboard XMC4700 and a radar front-end board BGT24LTR11 Shield. This is depicted in figure 4.

The Radar Baseboard XMC4700 is a generic sensor interface for Infineon's 24 GHz radar sensors. The central unit of the XMC4700 32-bit Arm® Cortex®-M4 based microcontroller can perform radar data processing or forward the sensor data to a USB interface, a serial interface or an Arduino interface. The board is designed to enable customers to carry out prototyping and system integrations, as well as initial product feature evaluations.

The Sense2GoL Pulse firmware is running in the XMC4700 microcontroller and configures the following peripherals:

• Analog-to-Digital Converter (ADC)
• Direct Memory Access (DMA)
• General-Purpose Input Output (GPIO)
• USIC for I2C serial interface
• CCU8/4 for timer and PWM control signals
• Capture and Compare Unit (CCU)
• Hardware interrupts
• USB interface for host communication

On the other hand, the radar shield BGT24LTR11 contains the following:

RF part: consists of the Infineon 24 GHz radar MMIC BGT24LTR11 and includes micro-strip patch antennas for the TX and RX sections.
• S&H part: consists of SPST switches and hold capacitors to sample and hold the analog I/Q signals from the MMIC using a control signal from the microcontroller.
• Analog amplifier part: consists of two amplifier stages, used to smooth the sampled I/Q signals from S&H circuitry and amplify them for the digital part.
• EEPROM part: can be used to store data such as board identifier information, and RF shield hardware settings.

This firmware would be used in this product to extract the I and Q radar signals. This firmware can be programmed with either Dave IDE or Arduino IDE.

3.1.2. Raspberry Pi 4 (Model B)

Figure 5 Raspberry Pi 4 Model B

Raspberry Pi 4 Model B is the latest product in the popular Raspberry Pi range of computers. It offers ground-breaking increases in processor speed, multimedia performance, memory, and connectivity compared to the prior-generation Raspberry Pi 3 Model B+, while retaining backwards compatibility and similar power consumption. For the end user, Raspberry Pi 4 Model B provides desktop performance comparable to entry-level x86 PC systems. This model is shown in figure 5.
Following are its specifications:

- Broadcom BCM2711, Quad core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz
- 1GB, 2GB, 4GB or 8GB LPDDR4-3200 SDRAM (depending on model)
- 2.4 GHz and 5.0 GHz IEEE 802.11ac wireless, Bluetooth 5.0, BLE
- Gigabit Ethernet
- 2 USB 3.0 ports; 2 USB 2.0 ports.
- Raspberry Pi standard 40 pin GPIO header (fully backwards compatible with previous boards)
- 2 × micro-HDMI ports (up to 4kp60 supported)
- 2-lane MIPI DSI display port
- 2-lane MIPI CSI camera port
- 4-pole stereo audio and composite video port
- H.265 (4kp60 decode), H264 (1080p60 decode, 1080p30 encode)
- OpenGL ES 3.1, Vulkan 1.0
- Micro-SD card slot for loading operating system and data storage
- 5V DC via USB-C connector (minimum 3A*)

- 5V DC via GPIO header (minimum 3A*)
- Power over Ethernet (PoE) enabled (requires separate PoE HAT)
- Operating temperature: 0 – 50 degrees C ambient

3.1.3. Power Bank

We will be utilizing a 5V, 3A with USB-C support as power supply for the Raspberry Pi 4. For the product, we will be adopting 'Dr. Prepare 1000mAh' power bank with the required specifications. Its dimensions are 9.398x 6.35x 2.54 cm and weighs 201.282 g. This power bank is shown in figure 6.

Figure 6 5V, 3A Power Bank

3.1.4. Smartphone

The smartphone will be used as the main client by user to access the radar product. It is assumed that smartphone will be running either Android or iOS. Android version should be at least Android 5.0 lollipop and iOS version should be above or equal 13.

3.1.5. Logical Flow

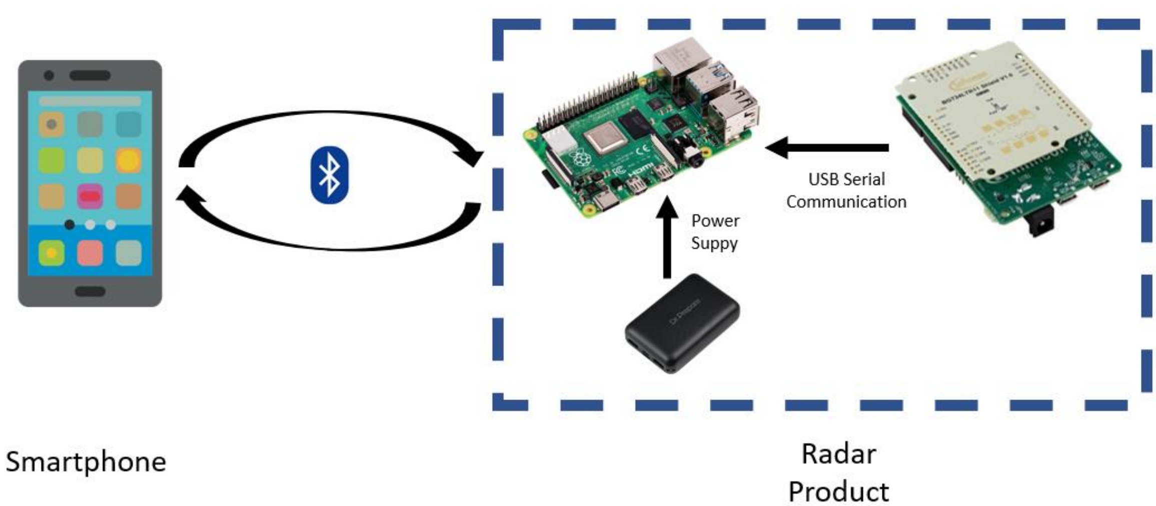


Figure 7 Setup of Radar Monitoring System

The proposed vital signs monitoring system comprises of a smartphone app (Android and iOS compatible) and a radar box (comprised of Raspberry Pi 4 and BGT24LTR11-XMC4700 board). This is shown in figure 7. The logical flow of the monitoring system begins with smartphone app, that sends a Transmission ID (T_{ID}) as an Eddystone UID to the Raspberry Pi 4 (which has an installed Bluetooth module). On reception of the T_{ID}, the Raspberry Pi enables serial USB communication with the BGT24LTR11-XMC4700 board and received I/Q signals from the radar board at 150 Hz sampling frequency for 21 seconds.
After the collection of data, the raspberry Pi 4 measures the heart rate (HR) and breathing rate (BR) via frequency domain and time domain method (described in section 3.2). The resultant rates are truncated to two decimal places and concatenated as ($HR||BR$). This result is encrypted via AES encryption using the key as T_{ID} and transmitted by the Raspberry Pi 4 as a iBeacon UID as R_{ID}.
The smartphone receives the R_{ID} and decrypts it using T_{ID} as the key to get the HR and BR back, which is displayed on the UI of the smartphone. This logical flow is represented in figure 8.

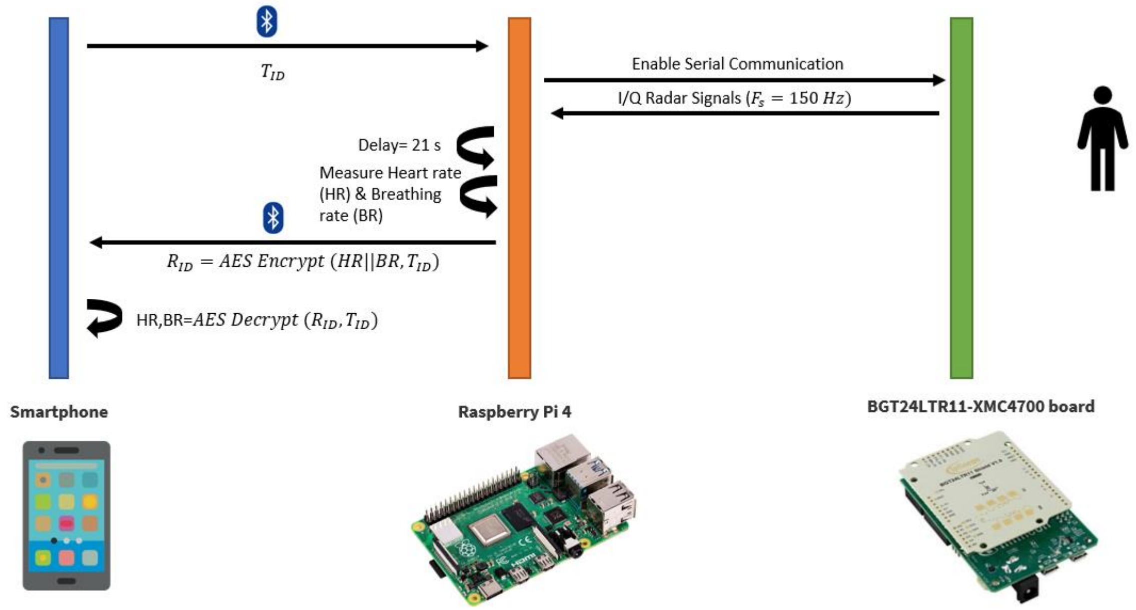


Figure 8 Sequence diagram for vital signs monitoring system

3.1.6. ECG as Validation

In this radar based vital signs measurement system, ECG based vital signs measurement system is introduced as an alternate method to provide validation. The setup comprised of ECG 3-channel electrodes, connected to a AD8232 module. The connection of 3 electrodes to body placement is shown in figure 9.

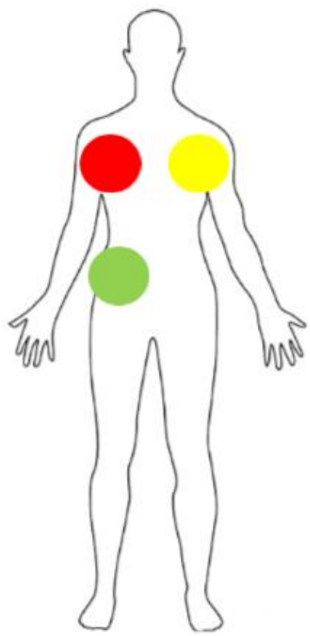


Figure 9 Electrode Placement Topology

NodeMCU ESP8266 collects the ECG signals from it at 360 Hz sampling frequency and this data and transports the data to PC for application of signal processing to extract heart rate and breathing rate. The setup is shown in figure 10.

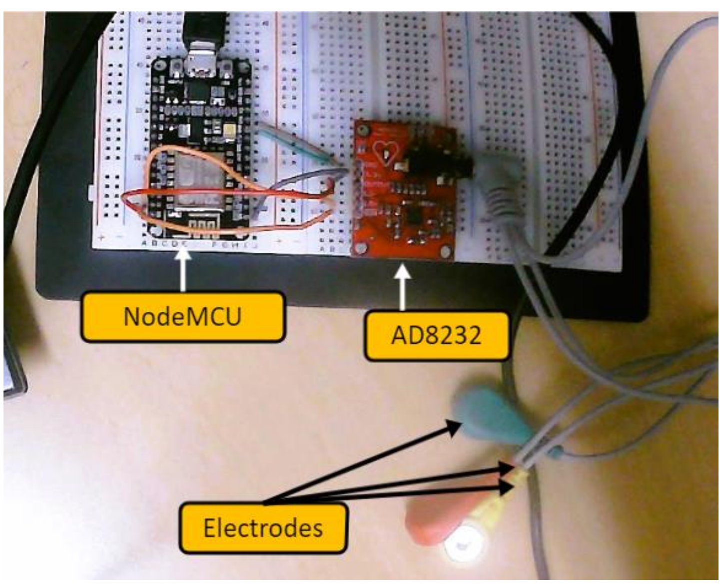


Figure 10 Setup of ECG measurement system

3.2. Software

3.2.1. Architecture Flow

The architecture flow of the radar product comprises of radar side and smartphone side. The radar side comprised of Sense2GoL Pulse (S2GLP) radar demo kit and raspberry pi 4, which is mainly responsible for collecting radar signals from a target and perform advanced signal processing to determine the vital signs. These results are then communicated to the smartphone side via Bluetooth low energy in beacon protocol. As discussed in section 2.1, the pulsed Doppler radar (Sense2Gol Pulse kit in our case) provides the in phase and quadrature signals (I and Q), which are further proceeded towards signal processing. The key technique in signal processing of radar signals is variational model decomposition, which decomposes the signal into intrinsic modes, that are fundamentally sinusoidal with slow time varying amplitude and frequency. The variational mode decomposition in our case decomposes the signal into 5 modes, of which mode-2 and 3 are used later for further signal processing. The architecture flow of radar system in shown in figure 11.

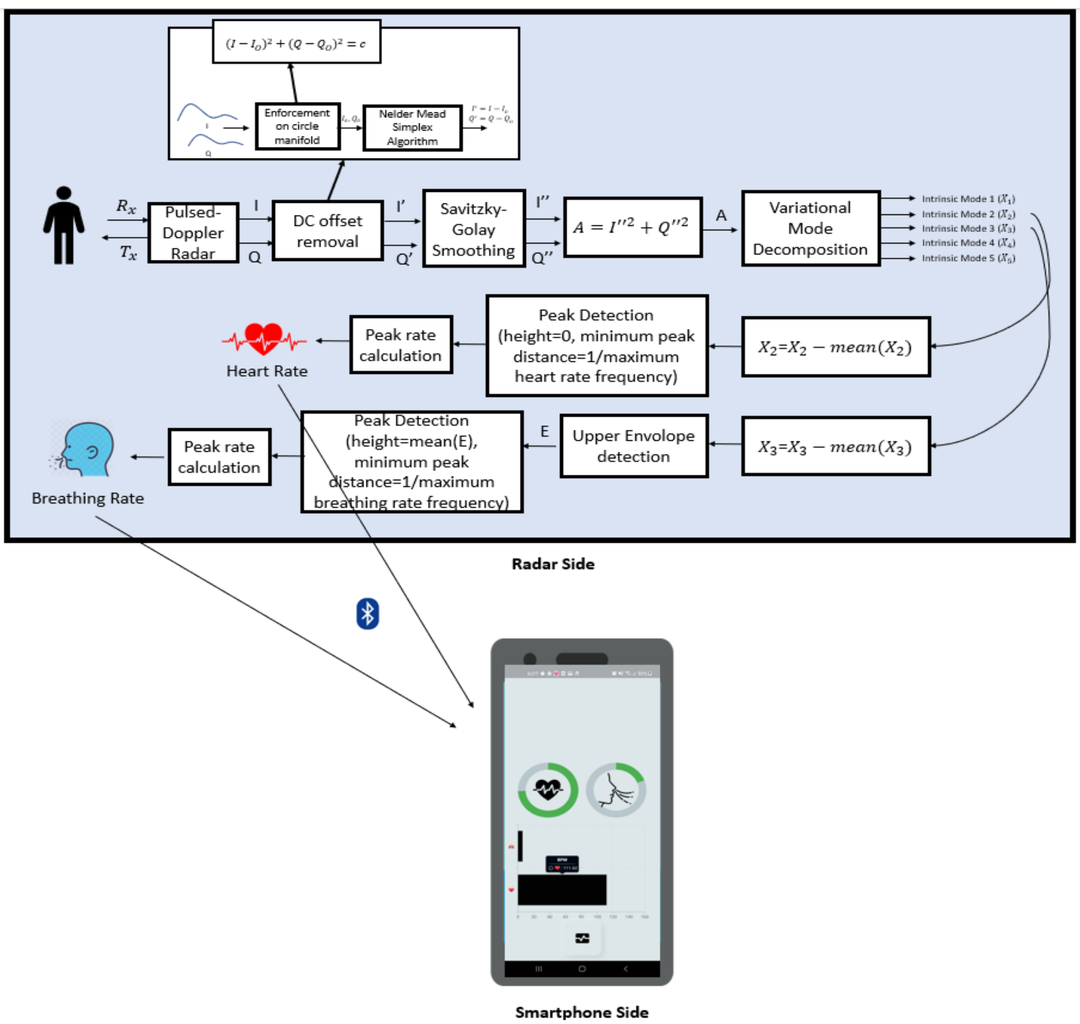


Figure 11 Architecture flow

3.2.2. Radar Side

3.2.2.1. Software Requirements

- Arduino IDE (C/C++ language): Sense2Gol Board
- Thonny IDE (Python language): Raspberry Pi 4

3.2.2.2. Source Code (with Steps)

1. Install Segger-J Link via: https://www.segger.com/downloads/jlink
2. Install the Arduino IDE
3. On Arduino IDE, Select 'Preferences' > 'Additional Boards Manager URLs', add the following URL: https://github.com/Infineon/XMC-for Arduino/releases/latest/download/package_infineon_index.json
4. From the Boards manager: 'Tools > Board > Boards Manager...', search for and install 'Infineon's XMC Microcontroller'
5. Update the Infineon Arduino hardware folder with the latest code from the 24Ghz-radar branch on GitHub: https://github.com/Infineon/XMC-for-Arduino/tree/24Ghz-radar
6. As per the algorithm define in figure 9, the corresponding algorithm code (intended to get I/Q signals) to program the XMC4700 of Sense2Gol board can be gained from following link:
   https://github.com/dianuj1997/vital_signs_instrument_project/tree/main/sense2gol_code
7. Run the "data_collector.ino" file and lad the corresponding hex code onto Sense2Gol Board's XMC microcontroller.
8. Likewise, the algorithm code to program the raspberry pi 4 (intended to perform signal processing from I/Q signal to calculate vital signs) and communicate with smartphone via BLE beacons can be availed from following link:
   https://github.com/dianuj1997/vital_signs_instrument_project/tree/main/raspberryPi_code
9. You can run the code using Thonny IDE, which comes preinstalled on Raspbian OS for Raspberry Pi

3.2.3. Smartphone Side(Android, iOS application)

3.2.3.1. Software Requirements

- Android Studio (Dart language)

3.2.3.2. Source Code

The smartphone application code responsible for getting vital signs from raspberry pi 4, via BLE beacon can be achieved from following link. However, this code is currently configured only for Android. iOS app needs further reconfiguration.
https://github.com/dianuj1997/vital_signs_instrument_project/tree/main/smartphone_code
The above code is a flutter project and can be ran if you follow following tutorial:
https://www.youtube.com/watch?v=kEoyVAF25do

3.2.4. Validation (ECG)

3.2.4.1. Software Requirements

- Arduino IDE (C/C++ Language)
- Visual Studio Code (Python Language)

3.2.4.2. Source Code

Arduino IDE is used to program the NodeMCU ESP8266 to get the ECG signals at 360Hz, and the corresponding source code for it as follows:
https://github.com/dianuj1997/ECG_measurement_system/tree/main/arduino_code

This ECG data is later fed into desktop using USB serial communication, that processes the ECG signals to get heart and breathing rate. The source code for measuring heart rate using ECG can be availed from following link:
https://github.com/dianuj1997/ECG_measurement_system/tree/main/ecg_heart_rate_measurement

Likewise, the source code for ECG based breathing rate measurement is from following link:
https://github.com/dianuj1997/ECG_measurement_system/tree/main/ecg_breathing_rate_measurement

4. Results

4.1. Hardware

4.1.1. Design

When debugging and validating the performance of the radar product in measurement of vital signs, the setup can be witnessed from figure 12.

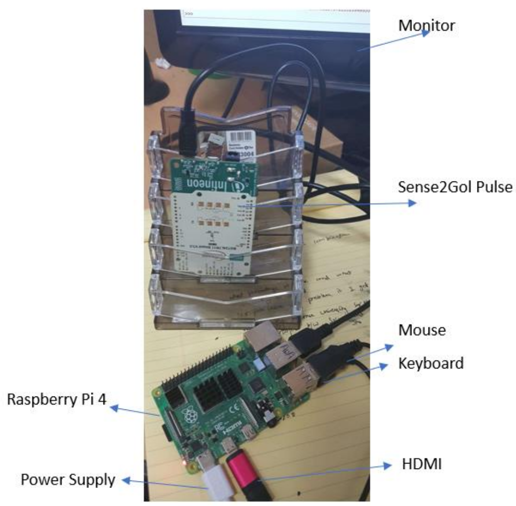


Figure 12 Physical Setup during development phase of radar product

After the validation of performance, the protype takes the following form as version 1, as shown in figure 13.

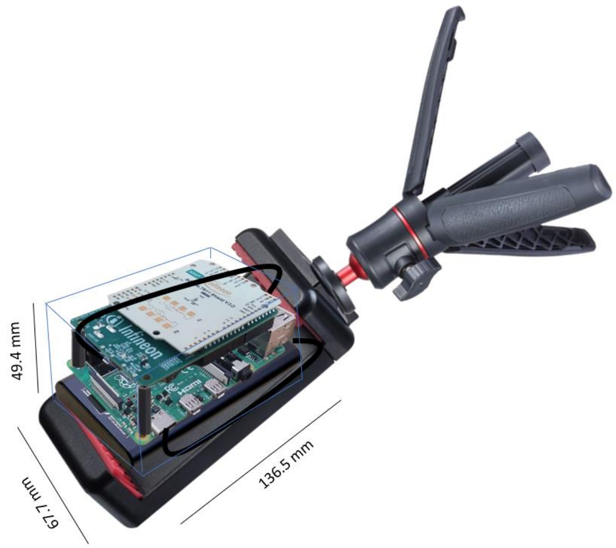


Figure 13 Prototype version 1- Radar Product

4.1.2. Power Efficiency

The power consumption of the radar product is expected to be 45mA and 5W during working mode. On the other hand, figure 14 and 15 shows the energy consumption of the smartphone app during starting phase and measuring phase respectively. Overall, the smartphone energy consumption is efficient due to adoption of Bluetooth low energy as a means to communicate with radar product.

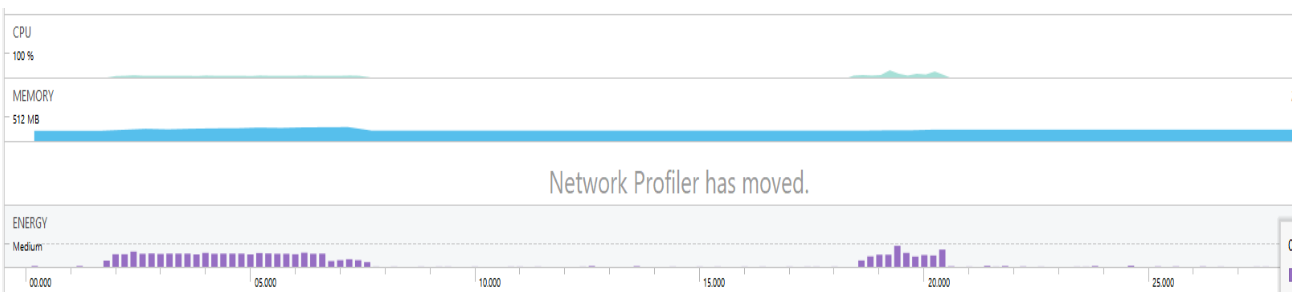


Figure 14 CPU, Memory and Energy profile of smartphone app during starting phase

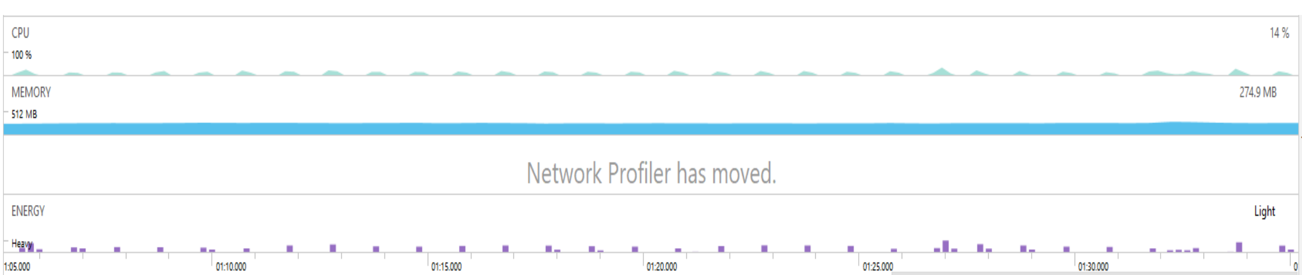


Figure 15 CPU, Memory and Energy profile of smartphone app during measurement phase

4.1.3. Specifications

The design of the radar product should be such that it is easily portable and allows flexibility in applications. Due to lack of involvement of any internet connectivity and server, privacy is expected to be maximum to due to anonymous nature of radar technology. The specifications of the product are expected to be represented by figure 16.

Technology	• Doppler Pulse Radar • Advanced Signal Processing • Bluetooth
Privacy	• Zero identification with radar technology • No third part data collection
Measurement	• Heart Rate (99.30 % accurate) • Breathing Rate (92.62 % accurate)
User Mode	• Office Mode • Gym Mode • Elder Monitor • Patient Monitor Mode • Baby Monitor Mode
Range	• 10cm to 1m
Weight	• ~200 g
Size	• 136.5 mm x 67.7 mm x 48.7 mm

Figure 16 Specifications of the radar product

4.2. Software

4.2.1. Vital Signs Accuracy

4.2.1.1. Comparison with ECG System

In comparison of measurement performance of heart rate and breathing rate from radar and ECG based system, random trials were conducted with reference as Mi-Band 3 for heart rate and manual respiration cycles counting over time interval for breathing rate. It was discovered that the radar system show an error of 2.4 beats per minute from ECG based heart rate measurement. The individual measurements are plotted in figure 17.

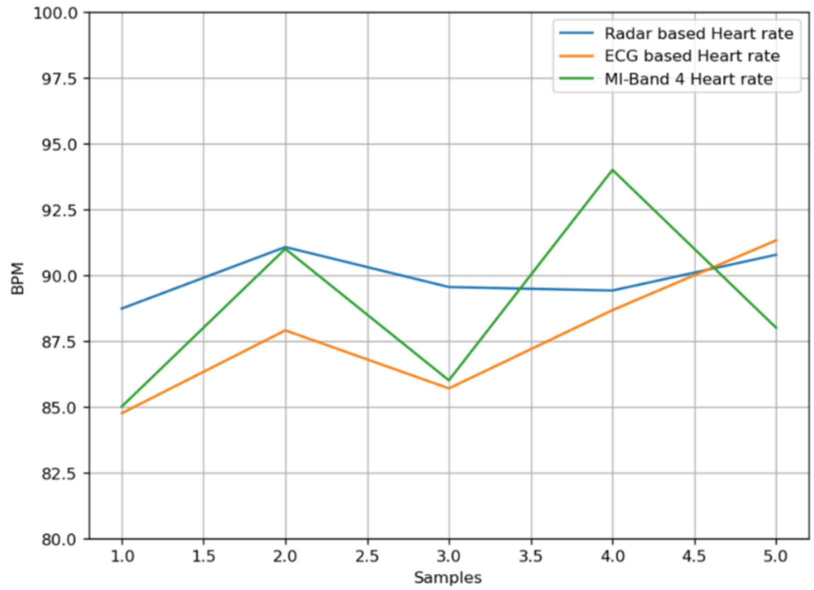


Figure 17 Heart rate measurement comparison between radar and ECG system

On the other hand, for breathing rate measurement , radar system showed an average error of 0.9 beats per minute in comparison to ECG system. The individual predictions are plotted in figure 18

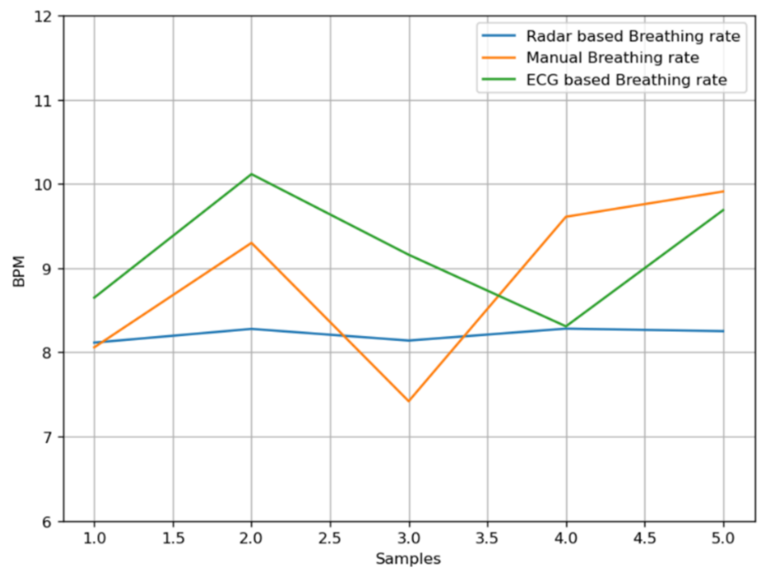


Figure 18 Breathing rate comparison between radar and ECG system

4.2.1.2. Comparison with Mi-Band 3 and Manual measurement

In final comparison, on random trials, the accuracy of vital signs achieved for the case of heart rate was 99.3% and breathing rate was 92.6%. The reference chosen for heart rate was Mi-Band 3, and for breathing rate, manual respirate cycles over time intervals were measure. The individual measurements by radar product and references are shown in figure 19 and 20.

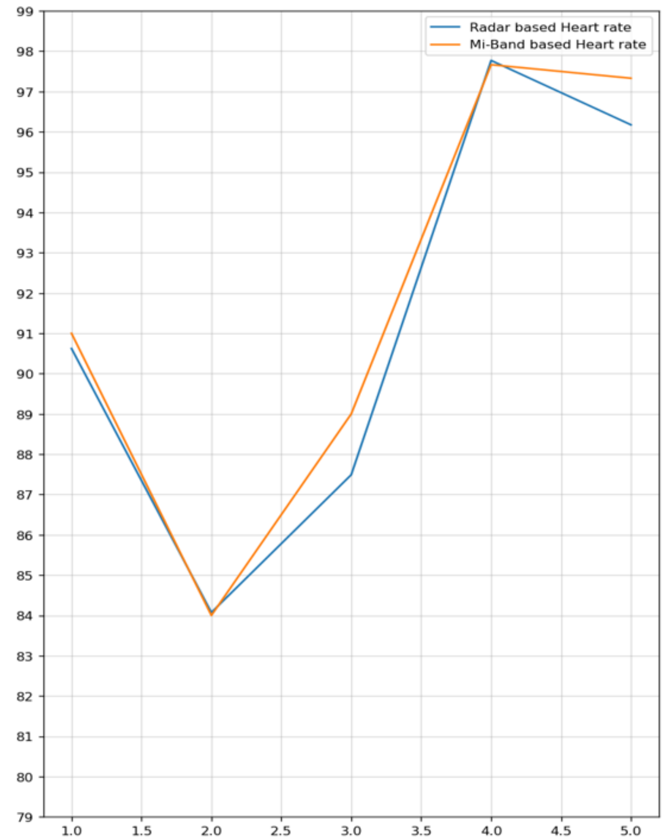


Figure 19 Comparison of heart rate measurements by radar product and Mi-Band 3

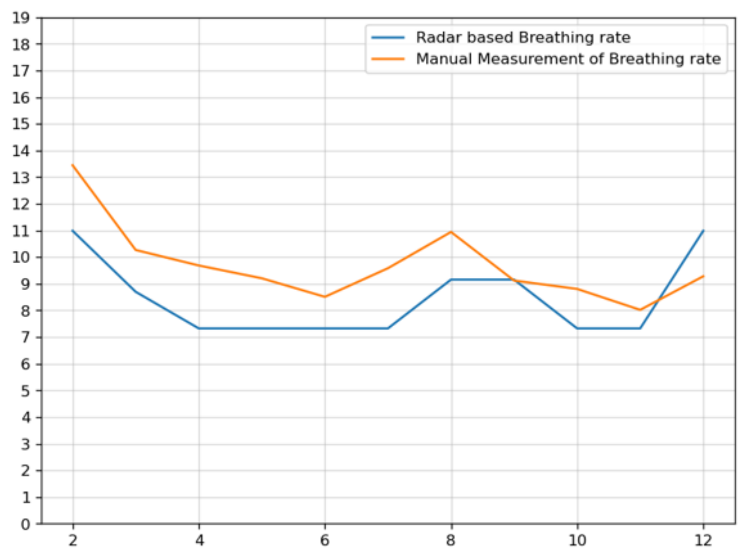
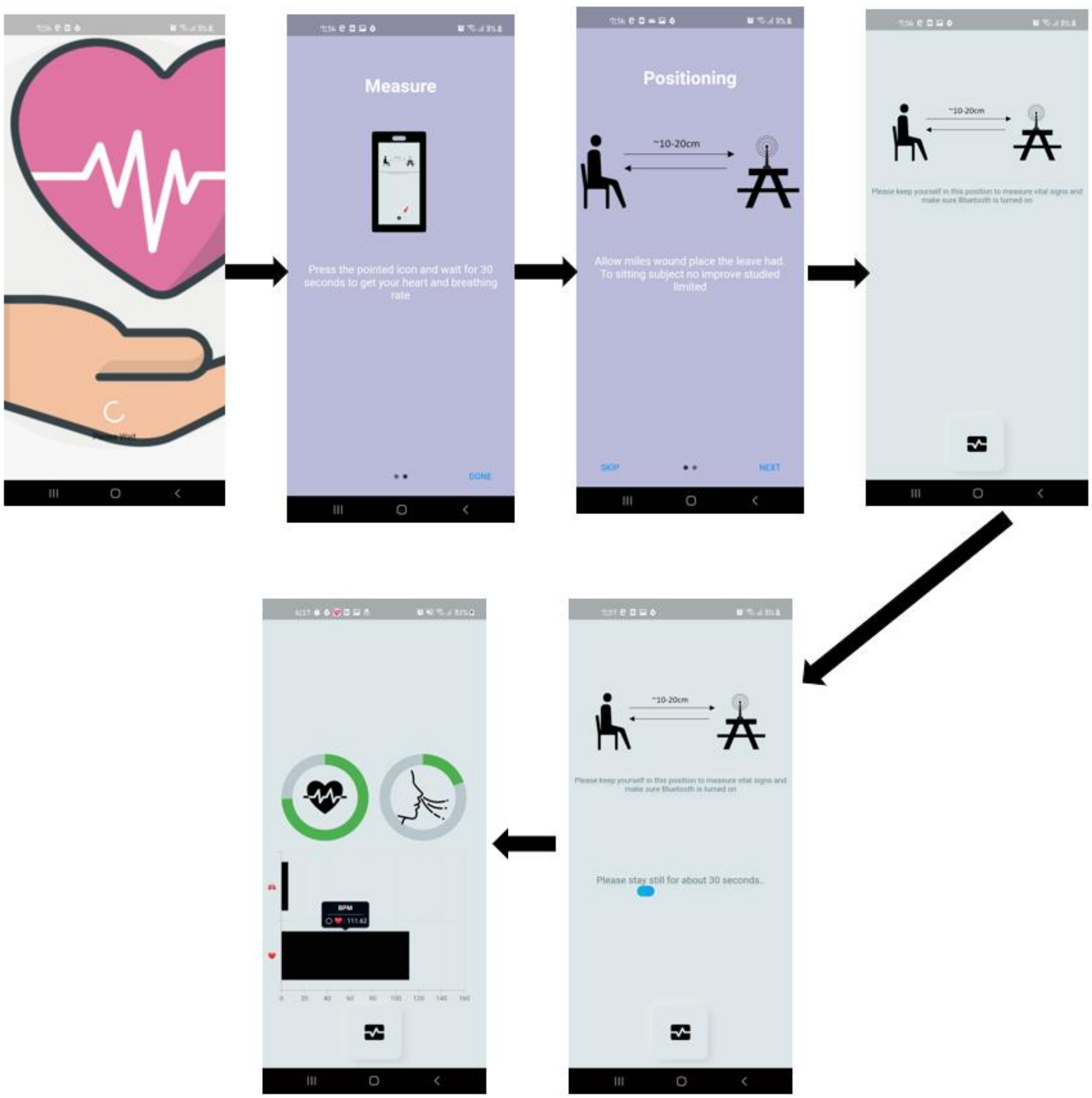


Figure 20 Comparison of breathing rate measurements by radar product and manual cycles measurement

4.2.2. Smartphone GUI

The smartphone GUI on Android or iOS operating system, can be operated in the sequence represented by figure 21. The app begins with a starting screen, followed by a guide. Then, the main screen appears, which is the fourth screen in figure 21, and the user has to press the button on the bottom to measure his/her vital signs, as long as he/she is in front of the radar product. The results appear as two types of graphs. The top graph shows the intensity of heart and breathing rate, while lower graph is an interactable bar graph, that shows exact heart and breathing rate.



\

Figure 21 Sequence of operating the smartphone app to interact with radar product

As an example, it can be shown in figure 22, that the measurement on smartphone greatly correlate with the measurement from the Mi-Band 3.

Figure 22 Comparison of measurement gotten on smartphone and Mi-Band 3

5. Conclusion

The proposed product is based on radar technology to provide a contactless and privacy preserving method to measure heart rate and breathing rate, using radar based technology. This product is provided with a smartphone app to provide ease of access to measure the vital signs at user’s comfort and provides flexibility in its applications.

6. Future Works

Due to innate flexibility of the setup of the product, it can be extended to diverse applications as follows:

- Office Mode
- Gym Mode
- Elder Monitor
- Patient Monitor Mode
- Baby Monitor Mode

7. Further References

These are the references I used for additional information on the project:

7.1. My Collective Work on ECG and Radar Based Heart Rate and Breathing Rate measurement
https://github.com/dianuj1997/HR_BR_Radar_ECG
https://github.com/dianuj1997/ECG_BreathingRate_Methods
https://github.com/dianuj1997/ECG_HeartRate_Methods
https://github.com/dianuj1997/ECG_measurement_system

7.2. Raspberry Pi 4 Beacon Implementation
https://github.com/mrin/domoticz-bt-presence
http://ianharvey.github.io/bluepy-doc/scanner.html
https://pypi.org/project/beacontools/
https://github.com/bowdentheo/BLE-Beacon-Scanner
https://www.instructables.com/iBeacon-Entry-System-with-the-Raspberry-Pi-and-Azu/
https://github.com/Gerzer/PiBeacon
https://chowdera.com/2021/06/20210606230541220j.html
https://chowdera.com/2021/06/20210606230541220j.html
https://bluezero.readthedocs.io/en/stable/examples.html
https://github.com/dipghoshraj/pibeacon/blob/master/broadcasting/beacon.py
https://medium.com/@sudhirsb2003/making-an-ibeacon-from-raspberry-pi-117b08ef6776
https://github.com/Gerzer/PiBeacon
https://github.com/CosminDanielSolomon/eddystone-beacon-broadcast
https://github.com/CosminDanielSolomon/eddystone-beacon-broadcast

7.3. Flutter Beacon Implementation
https://pub.dev/packages/beacon_broadcast
https://stackoverflow.com/questions/67231737/flutter-eddystone-beacon-detection
https://github.com/michaellee8/flutter_blue_beacon/blob/master/example/lib/main.dart

7.4. Portable Raspberry Pi 4 setup
https://www.youtube.com/watch?v=clbt12upuaA
https://www.youtube.com/watch?v=AFjYKEf7j2M

7.5. Flutter UI
https://flutterui.design/
https://pub.dev/packages/flutter_neumorphic

7.6. Sense2Gol Board Data Collection and Projects
https://www.edgeimpulse.com/blog/infineon-sense2gol-pulse
https://github.com/Infineon/24GHz-Radar-for-Arduino/blob/master/docs/pages/develop.md
https://www.hackster.io/Infineon_Team/blind-spot-detector-6bd8a5?fbclid=IwAR1AnAWMjprpJHef4QvnwU0Kh2WsicJobntP_51muubKQayGj3Hje2dfk6s

7.7. Sense2Gol Pin Configuration
https://github.com/Infineon/XMC-for-Arduino/wiki/DEMO-Radar-BB-XMC4700

7.8. Sense2Gol Arduino Setup
https://www.infineon.com/cms/en/product/promopages/makeradar/makeradar-school/programming-tutorial/