



GRADO EN INGENIERÍA EN TECNOLOGÍAS DE TELECOMUNICACIÓN

TRABAJO FIN DE GRADO IMPLEMENTACIÓN DE RADAR DE 120GHZ PARA LA DETECCIÓN DEL RITMO CARDIACO EN PERSONAS

Autor: Miguel Torres Valls

Director: Sebastian Scherer

Madrid

Declaro, bajo mi responsabilidad, que el Proyecto presentado con el título
Implementación de radar de 120GHz para la detección del ritmo cardiaco en personas
en la ETS de Ingeniería - ICAI de la Universidad Pontificia Comillas en el
curso académico 2023/2024 es de mi autoría, original e inédito y
no ha sido presentado con anterioridad a otros efectos.

El Proyecto no es plagio de otro, ni total ni parcialmente y la información que ha sido
tomada de otros documentos está debidamente referenciada.



Fdo.: Miguel Torres Valls

Fecha: 11/06/2024

Autorizada la entrega del proyecto
EL DIRECTOR DEL PROYECTO



Fdo.: Sebastian Scherer

Fecha: 11/06/2024



GRADO EN INGENIERÍA EN TECNOLOGÍAS DE TELECOMUNICACIÓN

TRABAJO FIN DE GRADO

IMPLEMENTACIÓN DE RADAR DE 120GHZ PARA LA DETECCIÓN DEL RITMO CARDIACO EN PERSONAS

Autor: Miguel Torres Valls

Director: Sebastian Scherer

Madrid

Agradecimientos

Quisiera agradecer la oportunidad brindada a la Universidad Carnegie Mellon por acogerme y abrirme las puertas como alumno de intercambio. En segundo lugar, al club de robótica, liderado por el profesor Sebastian Scherer, los cuales han creído en mi en todo momento y me han dado la oportunidad de trabajar en un proyecto tan novedoso, codo con codo con alumnos de máster, doctorado e investigadores profesionales de todos los rincones del mundo. También quisiera agradecerle a la Universidad Pontificia Comillas la oportunidad de poder viajar y estudiar en una universidad tan buena como la Carnegie Mellon. Por último y más importante, a mi familia, que me ha brindado los recursos para poder estudiar aquello que me gusta y ha estado siempre dispuesta a ayudarme y apoyarme.

IMPLEMENTACIÓN DE RADAR DE 120 GHZ PARA LA DETECCIÓN DEL RITMO CARDIACO EN PERSONAS

Autor: Torres Valls, Miguel.

Director: Scherer, Sebastian.

Entidad Colaboradora: The Robotics Institute – Carnegie Mellon University

RESUMEN DEL PROYECTO

Este proyecto consiste en la implementación de un radar de 120GHz FMCW¹ para la detección de distancia, fase y velocidad. Este radar está montado sobre un robot y debe coexistir con muchos otros sensores, comunicándose entre sí en un entorno ROS2² Iron³ y cuya misión global es la de ofrecer apoyo sanitario en situaciones extremas donde el contacto físico es limitado.

El radar ha sido programado para ser capaz de detectar las pequeñas variaciones en la piel producidas por los latidos del corazón y los movimientos de exhalar o inhalar, con el fin de detectar el ritmo cardiaco de personas sin ningún contacto físico o asistencia humana.

Palabras clave: Radar, FMCW, ROS2, Distancia, Fase, Velocidad, Procesamiento señal

1. Introducción

Las transmisiones de señales por radiofrecuencia con el fin de detectar y medir el mundo que nos rodea es una técnica que aparece por primera vez a principios del siglo XX y no ha parado de evolucionar. Actualmente, tiene diversas aplicaciones en el mundo de la aviación, astronomía, medicina e incluso belleza [1]. A lo largo de los últimos años, el avance tecnológico ha permitido que estas mediciones sean de mayor precisión, capaces de detectar cuerpos cada vez más pequeños y movimientos mucho más precisos, dando lugar a sistemas inteligentes o sistemas IoT.

Hoy en día, tomar medidas del ritmo cardiaco de una persona puede resultar un proceso complejo y costoso, siendo necesario el contacto físico con la persona y necesitando múltiples aparatos para tomar varias mediciones de manera simultánea. Estas limitaciones se acentúan en situaciones hostiles de guerra o catástrofes naturales, donde el contacto físico es peligroso y la rapidez de mediciones se vuelve imprescindible.

Este trabajo de fin de grado busca explorar nuevas e innovadoras maneras de obtener el ritmo cardiaco en personas, sin ningún contacto físico y asistencia humana. Para ello, se ha implementado un radar de 120GHz para la detección de distancia, fase y velocidad. Estas medidas tienen el objetivo final (el cual está fuera del alcance de este trabajo) de detectar el ritmo cardiaco de varias personas a partir del movimiento realizado por las personas al respirar

¹ Del inglés: Frequency Modulated Continuous Wave. Onda continua modulada en frecuencia

² Del inglés. Robot Operating System: framework para el desarrollo de aplicaciones robóticas

³ Versión de ROS2

como inhalar o exhalar y pequeñas vibraciones en la piel que surgen como consecuencia de los latidos del corazón.

2. Definición del Proyecto

Este proyecto forma parte del Team Chiron⁴, propuesta de la Universidad Carnegie Mellon para el DARPA Triage Challenge. El DTC⁵ se trata de una competición organizada por la Agencia de Proyectos de Investigación Avanzados de Defensa (DARPA) de los Estados Unidos de América. Su objetivo es la innovación tecnológica para situaciones de emergencia sanitarias y consiste en el desarrollo de sistemas que puedan evaluar y priorizar a los pacientes en función de la gravedad de sus lesiones o condiciones médicas en escenarios militares [2].

La propuesta del Team Chiron consiste en una serie de sensores IoT montados sobre un robot cuadrúpedo que debe de ser capaz de dar apoyo sanitario. A lo largo de este trabajo se ha trabajado en uno de estos sensores, concretamente en un radar de onda continua modulado en frecuencia de 120GHz. Todos estos sensores deben de ser capaces de comunicarse utilizando la red WiFi del robot, por lo que se ha optado por utilizar el entorno ROS2 Iron.

El propósito de este trabajo es la detección de distancia, fase y velocidad en el campo de acción del robot para su posterior análisis y la extracción del ritmo cardiaco en personas. Este análisis posterior no entra dentro de los objetivos de este proyecto y será realizado por un equipo independiente, pero forma una parte importante de este trabajo al limitar los parámetros de procesamiento y configuración del radar. La lista completa de objetivos es la siguiente:

- Comprender parámetros y procesamiento de un radar FMCW
- Comprender funcionamiento del entorno ROS2 Iron
- Protocolo de funcionamiento y configuración del radar EasyRad r4
- Detectar la distancia, fase y velocidad de múltiples objetivos
- Transmisión de las mediciones de distancia, fase y velocidad por la red WiFi

A lo largo de esta memoria se define el procesamiento de un radar FMCW y cómo a partir del eco producido se puede extraer la distancia, fase y velocidad de múltiples objetos. También se explicarán y, si es posible, se hará frente a las numerosas limitaciones que existen en la tecnología radar. Estas limitaciones pueden ser físicas (por la densidad de potencia recibida y las pérdidas en el espacio libre), de hardware (por la frecuencia de muestreo del conversor analógico digital) o de procesamiento (varían en función de configuración radar elegida). Además, se hará una breve introducción al resto de sensores que forman el robot y como el radar se comunica con ellos. Finalmente, se explicarán los resultados y conclusiones obtenidos, así como futuras direcciones de investigación.

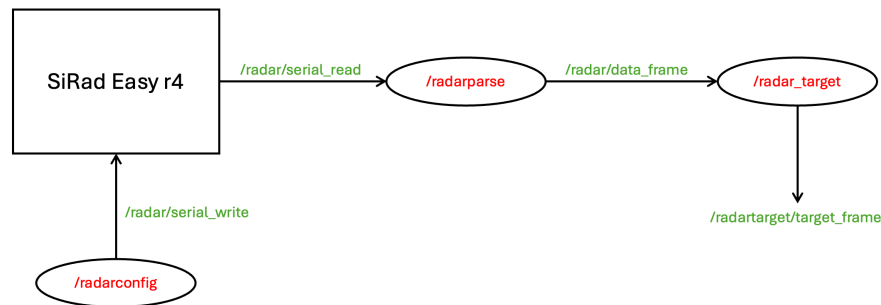
⁴ Equipo representante de la Universidad Carnegie Mellon para el DARPA Triage Challenge

⁵ Del inglés. DARPA Triage Challenge

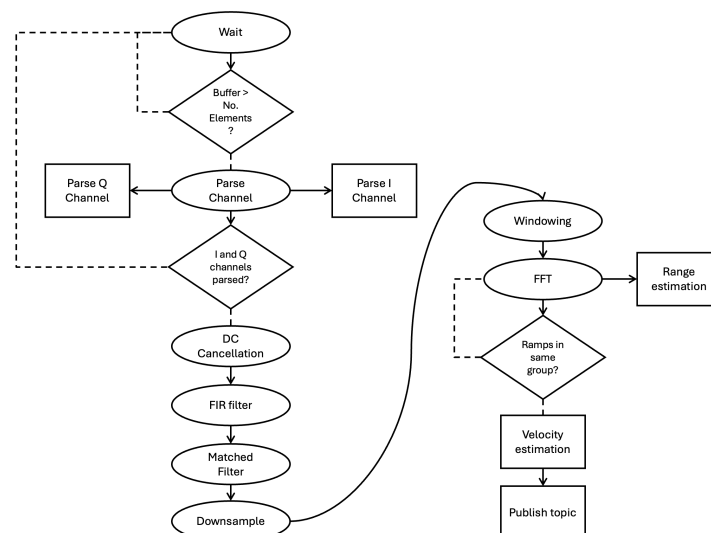
3. Descripción del modelo/sistema/herramienta

Podemos representar el sistema desarrollado bajo el siguiente diagrama de nodos y tópicos del entorno ROS2 Iron. ROS se basa en la comunicación entre nodos mediante tópicos que son transmitidos por la red. Cada sensor del robot está representado por uno o más nodos y cada nodo realiza una tarea específica. Estos nodos pueden publicar información en la red a través de tópicos, y recibir información suscribiéndose a otros tópicos, facilitando así la comunicación entre componentes [3].

El siguiente diagrama muestra la comunicación entre tópicos y nodos que se ha creado para la implementación del radar. Los tópicos aparecen en verde mientras que los nodos en rojo.



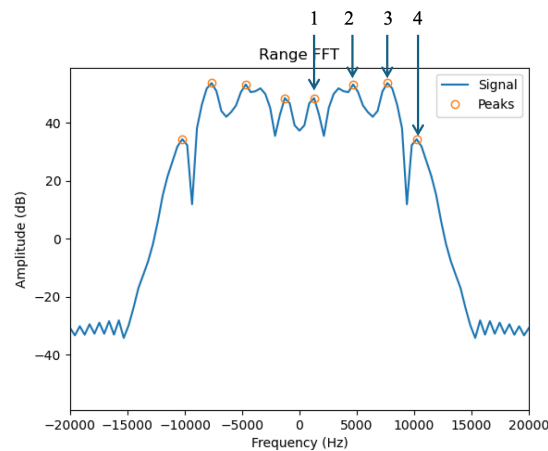
- El nodo **/radardconfig** se encarga de codificar los comandos de configuración del radar y publicarlos en el tópico **/radar/serial_write**
- El nodo **/radarparse** se encarga de decodificar los datagramas recibidos por el radar. Para ello se suscribe al tópico **/radar/serial_read** del radar y publica los mensajes decodificados en **/radar/data_frame**
- El nodo **/radar_target** es el más complejo de todos y se encarga del procesamiento digital de la señal para la detección de distancia, fase y velocidad. Para ello, se suscribe al tópico **/radar/data_frame** y publica la información correspondiente a cada objeto detectado en **/radartarget/target_frame**. El flujo de funcionamiento del nodo **/radar_target** es el siguiente.



4. Resultados

Para la evaluación de resultados, se ha operado con el radar totalmente aislado del resto de componentes del robot, ya que muchos de ellos se encontraban incompletos. Se ha conseguido desarrollar una serie de algoritmos para la detección de distancia, fase y velocidad de múltiples objetos en el rango de acción del radar y se ha sido capaz de integrar estos algoritmos bajo la estructura ROS2 del robot principal al que pertenece. De esta manera, se están tomando y procesando medidas de manera continua en el tiempo. Una vez se extrae la posición de los diferentes cuerpos, se transmiten por la red para que otras funcionalidades del robot pueden hacer uso de ellas.

Además, se ha conseguido optimizar el algoritmo de detección de picos en frecuencia, consiguiendo una detección del 100% de objetos en el campo de visión del radar, como se puede apreciar en la figura de abajo. Cada pico corresponde a un objeto.



Toda medición hecha por el radar es transmitida inmediatamente después por la red mediante la publicación de un tópicos cuyos mensajes tienen la siguiente estructura.

```
--  
frame_counter: 13446  
target_number: 4  
distance:  
- 0.4989304542541504  
- 1.8294117450714111  
- 2.9935827255249023  
- 3.991443634033203  
magnitude:  
- 48.50715637207031  
- 53.2181510925293  
- 53.689598083496094  
- 34.30365753173828  
phase:  
- -15.881953239440918  
- -173.68804931640625  
- 11.53531265258789  
- -77.21569061279297  
velocity:  
- 0.014796042814850807  
- -0.4202495813369751  
- 0.01291094720363617  
- 0.0004970852169208229  
--
```


Para el cálculo de la velocidad se ha optado por utilizar las variaciones de la fase de la señal recibida, en lugar de las variaciones de la distancia. Esto se debe a que las variaciones de fase tienen una resolución muchísimo mayor, siendo capaz de seguir movimientos del orden de milímetros. Por otro lado, las variaciones en distancia son únicamente perceptibles cuando son del orden de metros.

5. Conclusiones

Una vez finalizado el proyecto, se puede afirmar que se ha conseguido desarrollar un radar FMCW con las características y parámetros necesarios para la detección del ritmo cardiaco en personas. Además, se ha integrado satisfactoriamente este radar en el robot del que forma parte, habilitando la comunicación entre los múltiples sensores y la CPU del robot con el radar.

Los resultados han demostrado que el radar es capaz de seguir movimientos muy precisos, resultando un sensor óptimo para la detección del ritmo cardiaco en personas.

El impacto de este trabajo y en general, la tecnología radar de onda continua modulada en frecuencia no tiene límites. Esto se debe al hecho revolucionario de poder monitorizar los movimientos realizados al respirar sin ningún contacto físico, lo que promete ser una verdadera revolución en el sector sanitario. Especialmente en situaciones críticas o de guerra, pero también en el sector doméstico, con el fin de monitorizar a personas mayores o con problemas cardiovasculares y así poder detectar y avisar a profesionales médicos mucho más rápido de anomalías que puedan padecer.

6. Referencias

- [1] A.piquerasm, «Radiofrecuencia,» Wikipedia, 15 Abril 2024. [En línea].
- [2] Defense Advanced Research Projects Agency , «DARPA Triage Challenge Year 1,» Octubre 2024. [En línea]. Available: <https://www.darpa.mil/news-events/darpa-triage-challenge-year-1#:~:text=The%20main%20goal%20of%20the,%2C%20and%20mass%2Dcasualty%20settings.>
- [3] Open Robotics, «Begginer: CLI tools,» [En línea]. Available: <https://docs.ros.org/en/iron/Tutorials/Beginner-CLI-Tools.html>.
- [4] Carnegie Mellon University, «Team Chiron Making remote autonomous triage a reality,» [En línea]. Available: <https://teamchiron.ai/#about>.

DEVELOPMENT OF 120GHZ RADAR FOR HEART RATE EXTRACTION

Author: Torres Valls, Miguel.

Supervisor: Scherer, Sebastian.

Collaborating Entity: The Robotics Institute – Carnegie Mellon University

ABSTRACT

This project consists of the implementation of a 120 GHz FMCW⁶ radar for range, phase and velocity detection. This radar is mounted on top of a robot and must coexist with other sensors, working in a ROS2⁷ Iron⁸ environment with the ultimate goal of providing medical assistance in extreme situations where physical contact is limited.

This radar has been programmed to be able to detect small body movements like the ones caused when breathing such as exhaling or inhaling and small vibration in the skin that indicate heart rate. This makes it possible to obtain accurate heart and breathing rate measurements with no physical contact or human assistance.

Keywords: Radar, FMCW, ROS2, Range, Phase, Velocity, Signal processing

1. Introduction

Localization using radiofrequency waves to measure the world around us is a technique that first appears in the 1900s has not stopped evolving. Nowadays, it has diverse applications, ranging from aviation to astronomy, medicine or even beauty [1]. The technological breakthrough of the past decade has made these measurements much more precise, being able to track smaller objects and movements, creating smart systems.

Taking measurements of a person's heart rate can be an extremely difficult task, needing physical contact and multiple devices to take simultaneous measurements. These limitations are increased in hostile environments such as war zones or natural disasters, where physical contact is not always guaranteed, and the speed of measurements becomes crucial.

This undergraduate project aims to explore new and innovative ways to extract a person's heart rate, without any physical contact or human assistance. To solve this problem, a 120GHz radar has been implemented for range, phase and velocity detection. These measurements have the final objective (which is not part of the scope of this project) of detecting a person's heart rate by analyzing the distance and phase shifts produced when exhaling or inhaling and small body vibrations in the skin that indicate heart rate.

⁶ Frequency Modulated Continuous Wave

⁷ Robot Operating System 2. Framework used to build and program robot components

⁸ ROS2 version

2. Project Definition

This project is part of Team Chiron⁹, Carnegie Mellon's DARPA Triage Challenge team. The DTC¹⁰ is organized by the Defense Advanced Research Projects Agency (DARPA) of the United States Department of Defense, responsible for the development of emerging technologies for use by the military [2].

Team Chiron's proposal consists of a series of IoT sensors mounted on a quadruped robot designed to provide medical support. This project consists of one of these sensors, the 120GHz FMCW radar. All the robot's sensors must be able to communicate with each other using the robot's local network. For this, all sensors' communications take place under ROS2 Iron environment using nodes and topics.

The objective of this project is to detect distance, phase, and speed in the robot's field of action for subsequent analysis and extraction of heart rate. This subsequent analysis is not part of the project objectives and will be conducted by an independent team, but it is a crucial aspect as it limits the radar's processing and configuration parameters. The complete list of objectives is as follows:

- Understand FMCW radar processing
- Understand ROS2 Iron environment and how it operates
- Understand Sirad Easy r4 radar protocol and measurement flow
- Distance, phase and velocity detection
- Transmission of measurements over robot's network

Throughout this document, the processing of a Frequency Modulated Continuous Wave radar is defined, and how the echo produced can be used to extract the distance, phase, and speed of multiple objects. Additionally, the numerous limitations in radar technology will be explained and, if possible, addressed. These limitations can be physical (echo's power density and free-space losses), hardware-related (sampling frequency of the analog-to-digital converter), or processing-related (chosen radar configuration). In addition to this, a brief introduction to the other sensors on the robot and how the radar communicates with them will be provided. Finally, the results and conclusions obtained, as well as future research directions.

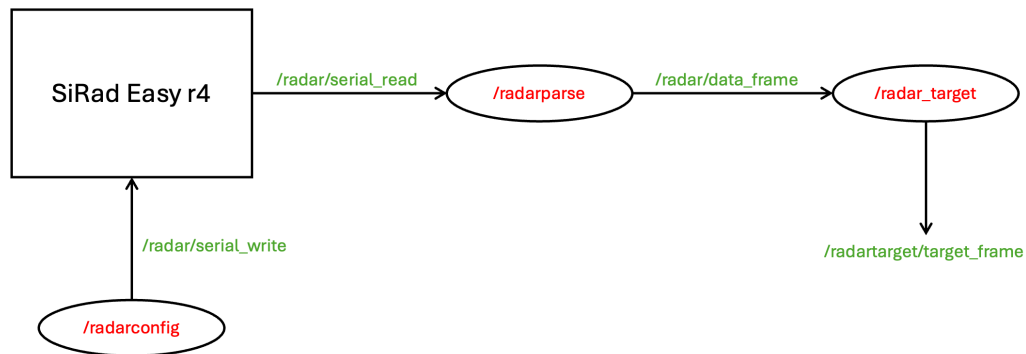
⁹ Carnegie Mellon's team for the Darpa Triage Challenge

¹⁰ Stands for Darpa Triage Challenge

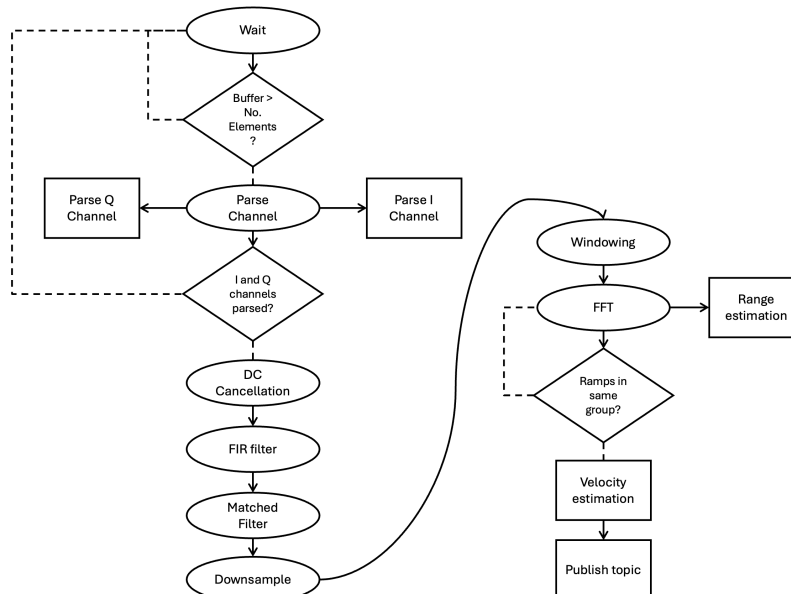
3. Description of model

We can represent the implemented system using a ROS2 diagram that explains the relation between each node used for target extraction. ROS is based on the communication between nodes over the network using topics [3]. Each of the robot's sensor is represented by one or more nodes, each node performing a specific action. Nodes can publish and subscribe to information over the network using topics.

The following diagram is a representation of all nodes and topics used to program the radar. Nodes are color coded in red and topics in green.



- /radarconfig is in charge of parsing configuration commands and sending it to the radar via the /radar/serial_write topic
- /radarparse will receive the radar's dataframes under the topic /radar/serial_read and is in charge of parsing them. It publishes the parsed messages in /radar/data_frame
- /radar_target is the most important node of all. It is in charge of all the digital signal processing used to extract target information. Subscribes to /radar/data_frame, processes the data and publishes the target information in /radartarget/target_frame. It follows the measurement flow described in the graph below.

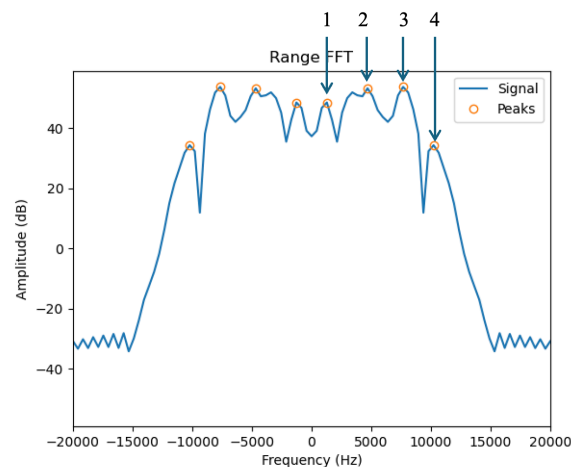


4. Results

All the experiments performed have been conducted without the other sensors, since most of them were incomplete at the time, and the main objective of this project is to test the radar's performance by itself.

Overall, the objectives have been met and the software is capable of obtaining range, phase and velocity measurements of any object in the radar's field of view. These algorithms have also proven to be successfully integrated into the robot's ROS2 environment, working in harmony with the rest of the sensors. The radar takes measurements and transmits target frames over the network so other sensors can use that information for further processing.

The algorithm used for peak detection in the frequency domain has also been optimized, obtaining a 100% detection rate in the radar's field of view. The figure below is a snapshot of a measurement session in which four objects are detected, each of them represented in a frequency peak.



Target frames published by the radar to the network have the following structure, containing (in order from closest to farthest): distance, magnitude, phase and velocity.

```
---
frame_counter: 13446
target_number: 4
distance:
- 0.4989304542541504
- 1.8294117450714111
- 2.9935827255249023
- 3.991443634033203
magnitude:
- 48.50715637207031
- 53.2181510925293
- 53.689598083496094
- 34.30365753173828
phase:
- -15.881953239440918
- -173.68804931640625
- 11.53531265258789
- -77.21569061279297
velocity:
- 0.014796042814850807
- -0.4202495813369751
- 0.01291094720363617
- 0.0004970852169208229
---
```

For the velocity estimation, taking track of phase shifts instead of distance variations has proven to be much more precise, capable of following movement in the order of millimeters, whereas distance in the other hand was only capable of following movements in the order of meters.

5. Conclusions

Once the project is complete, it is safe to say that all the objectives have been met, successfully building a set of algorithms capable of detecting distance, phase and velocity in small body movements like the ones produced when exhaling or inhaling.

In addition to this, the radar is successfully integrated in the robot's network, making it possible for other sensors to leverage the target frames published by the radar into obtaining precise heart and breathing rate measurements.

The impact this project has, and in general, FMCW radar technology has no limits. Being able to track body movements made when breathing without any physical contact or human assistance promises to be a turning point for modern technology and medicine. This is enhanced in warlike situations, where human access is limited, but can also be used in domestic environments. For instance, having an FMCW radar installed at home that takes continuous measurements of everyone's heart rate and triggers an alert in case of anomalies.

6. References

- [1] A.piquerasm, «Radiofrecuencia,» Wikipedia, 15 Abril 2024. [En línea].
- [2] Defense Advanced Research Projects Agency , «DARPA Triage Challenge Year 1,» Octubre 2024. [En línea]. Available: <https://www.darpa.mil/news-events/darpa-triage-challenge-year-1#:~:text=The%20main%20goal%20of%20the,%2C%20and%20mass%2Dcasualty%20settings.>
- [3] Open Robotics, «Begginer: CLI tools,» [En línea]. Available: <https://docs.ros.org/en/iron/Tutorials/Beginner-CLI-Tools.html>.
- [4] Carnegie Mellon University, «Team Chiron Making remote autonomous triage a reality,» [En línea]. Available: <https://teamchiron.ai/#about>.

Índice de la memoria

<i>Índice de la memoria</i>	<i>I</i>
<i>Índice de figuras</i>	<i>III</i>
<i>Índice de tablas</i>	<i>VI</i>
<i>Capítulo 1. Introducción</i>	<i>VII</i>
<i>Capítulo 2. Descripción de las tecnologías</i>	<i>IX</i>
2.1. Tecnología RADAR	<i>IX</i>
2.2. Efecto Doppler	<i>XII</i>
2.3. Radar FMCW	<i>XIII</i>
2.3.1. Diagrama de bloques radar FMCW	<i>XIV</i>
2.3.2. Mezclador IQ	<i>XV</i>
2.3.3. Filtro FIR	<i>XVIII</i>
2.3.4. Cálculo de la distancia	<i>XXI</i>
2.3.5. Cálculo de la velocidad	<i>XXIII</i>
2.4. Entorno ROS	<i>XXVII</i>
<i>Capítulo 3. Estado de la cuestión</i>	<i>XXIX</i>
<i>Capítulo 4. Definición del Trabajo</i>	<i>XXXIII</i>
4.1. Justificación	<i>XXXIII</i>
4.2. Objetivos	<i>XXXV</i>
4.2.1. Comprender parámetros y procesamiento de un radar FMCW	<i>XXXV</i>
4.2.2. Comprender entorno ROS2	<i>XXXV</i>
4.2.3. Protocolo de funcionamiento y configuración del radar EasyRad r4	<i>XXXV</i>
4.2.4. Detectar la distancia a la que se encuentran los objetos	<i>XXXV</i>
4.2.5. Detectar la velocidad a la que se mueven los objetos	<i>XXXVI</i>
4.2.6. Transmisión de las mediciones de distancia, fase y velocidad	<i>XXXVI</i>

4.3. Metodología	XXXVI
4.3.1. Estudio teórico.....	XXXVI
4.3.2. Implementación	XXXVII
4.4. Planificación y Estimación Económica.....	XXXVIII
4.4.1. Planificación	XXXVIII
4.4.2. Estimación Económica	XLI
Capítulo 5. Sistema/Modelo Desarrollado	XLIV
5.1. Análisis del Sistema.....	XLIV
5.2. Diseño	XLIX
5.2.1. Flujo de mediciones.....	XLIX
5.2.2. Configuración	LIII
5.3. Implementación	LX
5.3.1. Entorno ROS2	LX
5.3.2. Decodificación de los datagramas del radar	LXIII
5.3.3. Software para la detección de distancia, fase y velocidad.....	LXVII
Capítulo 6. Análisis de Resultados	LXXV
Capítulo 7. Conclusiones y Trabajos Futuros	LXXXII
7.1. Conclusiones	LXXXII
7.1.1. Comprender parámetros y procesamiento de un radar FMCW	LXXXII
7.1.2. Comprender entorno ROS2	LXXXII
7.1.3. Protocolo de funcionamiento y configuración del radar EasyRad r4.....	LXXXIII
7.1.4. Detectar la distancia a la que se encuentran los objetos.....	LXXXIII
7.1.5. Detectar la velocidad a la que se mueven los objetos	LXXXIII
7.1.6. Transmisión de las mediciones de distancia, fase y velocidad.....	LXXXIII
7.2. Trabajos Futuros	LXXXIV
Capítulo 8. Bibliografía	LXXXVII
ANEXO I: ALINEACIÓN DEL PROYECTO CON LOS ODS.....	XC

Índice de figuras

Figura 1. Diagrama tecnología radar	IX
Figura 2. De arriba a abajo: radar de onda continua, radar pulsado, radar de onda continua modulado en frecuencia.....	XII
Figura 3. Rampa de frecuencia radar FMCW	XIII
Figura 4. Representación en dominio tiempo chirp.....	XIII
Figura 5. Diagrama de bloques radar FMCW	XIV
Figura 6. Mezclador IQ	XVI
Figura 7. Ejemplo rama I/Q [1]	XVII
Figura 8. Ejemplo rama I/Q [2]	XVII
Figura 9. Rama I/Q [3]	XVIII
Figura 10. Rama I/Q [4]	XVIII
Figura 11. Diagrama de bloques filtro FIR.....	XIX
Figura 12. Filtro FIR con transitorio	XX
Figura 13. Filtro FIR sin transitorio	XX
Figura 14. Señal IF para un obstáculo	XXI
Figura 15. Señal IF con varios obstáculos.....	XXII
Figura 16. De izquierda a derecha: señales Tx y Rx, señal IF	XXIV
Figura 17. De izquierda a derecha: señales Tx y Rx, señal IF	XXIV
Figura 18. Grupo de chirps.....	XXV
Figura 19. FFT de 2 dimensiones	XXVII
Figura 20. Diagrama de flujo entorno ROS2.....	XXVIII
Figura 21. Diagrama UAVs + radar FMCW.....	XXIX
Figura 22. Radar FMCW en automoción	XXX
Figura 23. Esquema transmisión y recepción estaciones MIMO	XXXI
Figura 24. Exhalar e inhalar contracción pecho	XXXII
Figura 25. OrangePi Zero	XLII

Figura 26. Pisugar 3.....	XLII
Figura 27. SiRad Easy r4.....	XLIII
Figura 28. Esquema general robot Team Chiron.....	XLV
Figura 29. Sensores a bordo del Team Chiron.....	XLVI
Figura 30. Conexión SiRad Easy r4 con OrangePi Zero.....	XLVII
Figura 31. Router a bordo del robot	XLVII
Figura 32. Radar montado con lente.....	XLVIII
Figura 33. Efecto de añadir lente sobre el radar	XLVIII
Figura 34. Flujo de mediciones SiRad Easy r4	XLIX
Figura 35. Efecto matched filter	LI
Figura 36. Parámetros de algoritmo find_peaks de scipy.....	LIII
Figura 37. Ejemplo configuración [1]	LIV
Figura 38. Ejemplo configuración [2]	LIV
Figura 39. Registro configuración del sistema	LV
Figura 40. Valores registro configuración sistema	LV
Figura 41. Registro configuración PLL	LVII
Figura 42. Valores registro configuración PLL	LVII
Figura 43. Registro configuración banda base	LVII
Figura 44. Valores registro configuración banda base.....	LVIII
Figura 45. Raw data en flujo de mediciones	LX
Figura 46. Diagrama entorno ROS2 para el SiRad Easy r4	LXI
Figura 47. Formato mensajes tópico radar/data_frame	LXII
Figura 48. Formato mensajes tópico /radartarget/target_frame.....	LXII
Figura 49. Trama del radar	LXIII
Figura 50. Flujo procesamiento nodo /radar_target.....	LXVII
Figura 51. Respuesta en frecuencia LPF 10KHz.....	LXIX
Figura 52. Respuesta en frecuencia matched filter.....	LXX
Figura 53. Ventana blackman	LXXI

Figura 54. Resultado detección de objetos [1]	LXXII
Figura 55. Tiempo entre grupo de rampas	LXXIII
Figura 56. Tiempo entre rampas mismo grupo	LXXIII
Figura 57. Tópico /radar/target_frame	LXXIV
Figura 58. Resultado detección de objetos [2]	LXXVI
Figura 59. Resultado detección de objetos [3]	LXXVII
Figura 60. Criterio de Rayleigh	LXXVII
Figura 61. Resultado detección de objetos [4]	LXXIX
Figura 62. Resultado detección de objetos [5]	LXXIX
Figura 63. Resultado detección de objetos [6]	LXXX
Figura 64. Resultado detección de objetos [7]	LXXX
Figura 65. Movimiento producido al exhalar/inhalar	LXXXIV
Figura 66. Variación de fase como resultado de respirar	LXXXV
Figura 67. Frecuencia ritmo respiratorio	LXXXV
Figura 68. Efectividad vital-radio	LXXXVI

Índice de tablas

Tabla 1. Planificación del proyecto	XLI
Tabla 2. Costes del proyecto	XLIII
Tabla 3. Parámetros conexión UART	LIV
Tabla 4. Registro configuración sistema.....	LVI
Tabla 5. Registro configuración banda base	LVIII
Tabla 6. Valores frecuencia muestreo	LIX
Tabla 7. Trama del radar	LXIV
Tabla 8. Identificador de trama.....	LXV
Tabla 9. Fuente de datos	LXV
Tabla 10. Campo de datos.....	LXVI
Tabla 11. Análisis de resultados.....	LXXV

Capítulo 1. Introducción

Las transmisiones de señales por radiofrecuencia con el fin de detectar y medir el mundo que nos rodea es una técnica que aparece por primera vez a principios del siglo XX y no ha parado de evolucionar. Actualmente, tiene diversas aplicaciones en el mundo de la aviación, astronomía, medicina e incluso belleza [1]. A lo largo de los últimos años, el avance tecnológico ha permitido que estas mediciones sean de mayor precisión, capaces de detectar cuerpos cada vez más pequeños y movimientos mucho más precisos, dando lugar a sistemas inteligentes o sistemas IoT.

A lo largo de los últimos años, este crecimiento de sistemas IoT ha ido de la mano de la robótica, una tecnología que se basa en la construcción de máquinas autónomas o semiautónomas, capaces de interactuar con su entorno y realizar diferentes acciones de manera eficiente [2]. La integración de sensores avanzados y comunicación por radiofrecuencia ha permitido el desarrollo de robots más complejos y potentes, cuyas actividades están presentes en todos los campos de la ingeniería.

Hoy en día, tomar medidas del ritmo cardíaco de una persona puede resultar un proceso complejo y costoso, siendo necesario el contacto físico con la persona y necesitando múltiples aparatos para tomar varias mediciones de manera simultánea. Estas limitaciones se acentúan en situaciones hostiles de guerra o catástrofes naturales, donde el contacto físico es peligroso y la rapidez de mediciones se vuelve imprescindible.

El DARPA Triage Challenge (DTC) se trata de una competición organizada por la Agencia de Proyectos de Investigación Avanzados de Defensa (DARPA) de los Estados Unidos de América que busca la innovación y el desarrollo del sector biosanitario a través de la tecnología [3]. La propuesta de la Universidad Carnegie Mellon para el DTC, centrada en el Team Chiron, consiste en una serie de sensores IoT montados sobre un robot cuadrúpedo que debe de ser capaz de dar apoyo sanitario en situaciones donde al acceso humano sea limitado [4].

En concreto, durante este trabajo se ha trabajado en el desarrollo e implementación de un radar de 120GHz para la detección de distancia, fase y velocidad de diferentes cuerpos en el rango de acción del robot. Este radar, tiene el objetivo final (el cual está fuera del alcance de este trabajo), de detectar el ritmo cardiaco de varias personas de manera simultánea, sin ningún contacto físico o asistencia humana y de manera totalmente inalámbrica. Para ello, estas medidas serán luego analizadas por un equipo independiente con el objetivo de aislar pequeños movimientos realizados por los seres humanos al respirar como exhalar o inhalar y vibraciones en la piel que se producen como efecto de los latidos del corazón.

Capítulo 2. Descripción de las tecnologías

2.1. Tecnología RADAR

El principio bajo el que opera la tecnología radar consiste en transmitir una señal y analizar su eco para extraer conclusiones acerca de cómo es el medio que nos rodea [5]. A diferencia de otras tecnologías de localización como pueden ser el GPS, donde existen varios equipos transmisores y receptores, la tecnología radar tiene un único transmisor y depende por completo del eco que la señal transmitida produce. Es por esto por lo que la potencia de la señal transmitida debe ser superior a la que emplearíamos en un sistema tradicional con un transmisor y un receptor, ya que debe de ser capaz de volver a la antena transmisora con una relación señal ruido aceptable para su decodificación y posterior procesamiento.

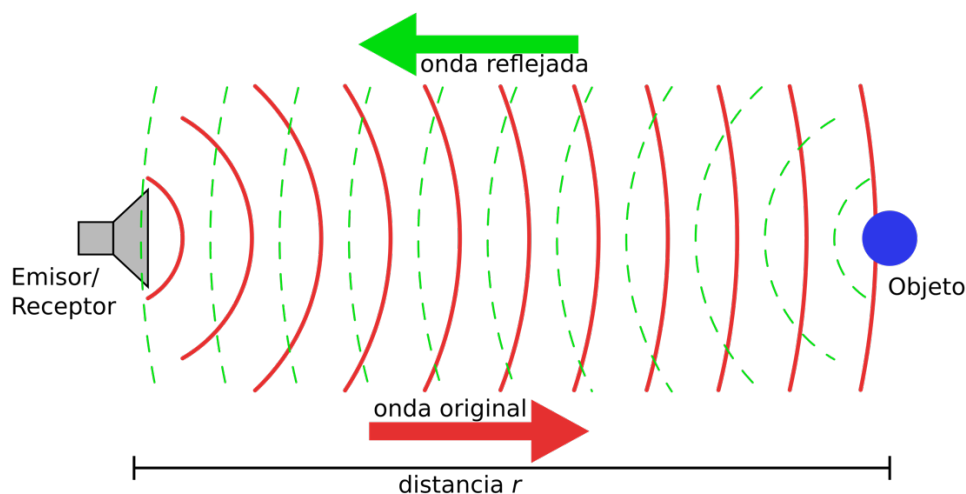


Figura 1. Diagrama tecnología radar (Wikipedia: https://es.wikipedia.org/wiki/Radar#/media/Archivo:Sonar_Principle_ES.svg)

La densidad de potencia transmitida depende en gran medida del tipo de antena utilizada, en la tecnología radar, los dos tipos más comunes son la antena isotrópica y la direccional. La antena isotrópica se caracteriza por radiar la misma densidad de potencia en todas las direcciones mientras que la direccional radia potencia en una cierta dirección. En este contexto, se ha utilizado una antena direccional, cuya densidad de potencia transmitida depende de la potencia transmitida

(P_{Tx}), de la ganancia de la antena transmisora (G_{Tx}) y de la distancia a la que se encuentra el objeto (R) bajo la siguiente expresión [6].

$$S_{Tx} = \frac{P_{Tx} G_{Tx}}{4\pi R^2} \left[\frac{W}{m^2} \right]$$

La potencia reflectada por el objeto se define bajo la siguiente ecuación y depende de la densidad de potencia transmitida por el radar y la sección de cruce del objeto (σ). La sección de cruce se define como una medida que indica cuánto detecta de un objeto el radar y es el correspondiente al área del objeto visto desde la posición del radar [6].

$$P_o = S_{Tx} \sigma = \frac{P_{Tx} G_{Tx}}{4\pi R^2} \sigma \text{ [W]}$$

La densidad de potencia recibida por la antena receptora del radar es la siguiente [6].

$$S_{Rx} = \frac{P_o}{4\pi R^2} = \frac{P_{Tx} G_{Tx}}{4\pi R^2} \frac{\sigma}{4\pi R^2} \left[\frac{W}{m^2} \right]$$

La potencia de la señal recibida es la densidad de potencia recibida multiplicado por la apertura de la antena receptora [6].

$$P_{Rx} = S_{Rx} A_{antena} = \frac{P_{Tx} G_{Tx}}{4\pi R^2} \frac{\sigma}{4\pi R^2} \frac{G_{Rx} \lambda^2}{4\pi} \text{ [W]}$$

La potencia de este eco debe luchar contra todas las fuentes de ruido para conseguir una relación señal ruido aceptable y poder decodificar la señal. Las fuentes de ruido más significativas están compuestas por ruido cósmico, ruido solar, ruido atmosférico, ruido por reflexión con la tierra y ruido por otros sistemas de creación humana. La media de este ruido se puede representar utilizando la constante de Boltzman (k), la temperatura media del sistema (T_s) y el ancho de banda del receptor (B_n) bajo la siguiente expresión [6].

$$N = k T_s B_n$$

La expresión final para la relación señal ruido viene dada por la siguiente expresión, donde N es el ruido total y L las diferentes pérdidas del sistema [6].

$$SNR = \frac{P_{Rx}}{N L}$$

Dado una relación mínima de señal ruido (SNR_{min}), y sustituyendo P_{Rx} en la ecuación anterior, podemos obtener la distancia máxima a la que podemos detectar un obstáculo [6].

$$R_{max} = \left(\frac{P_{Tx} G_{Tx} G_{Rx} \lambda^2 \sigma}{(4\pi)^3 N L SNR_{min}} \right)^{\frac{1}{4}} [m]$$

La tecnología radar ha experimentado numerosos avances a lo largo de su historia, variando en coste, complejidad y alcance. Los radares más antiguos son los llamados radar de onda continua, basados en la transmisión de ondas no moduladas de manera continua en el tiempo, que se caracterizan por tener un bajo consumo y coste. A continuación, surgieron los radares pulsados, los cuales transmiten pulsos de mucha energía en pequeños intervalos de tiempo. A diferencia de los radares de onda continua, estos radares necesitan de una electrónica más pesada y de mayor coste al tener que transmitir señales de mayor potencia. Además, cuentan con una separación mínima a la cual se puede detectar un obstáculo, ya que la onda reflectada debe llegar al transmisor una vez acabe la duración del pulso y no antes. Esto supone varios problemas a la hora de localizar cuerpos a corta distancia y pequeños movimientos. Finalmente, en la actualidad son muy comunes los radares de onda continua modulada en frecuencia (FMCW¹¹ según su acrónimo en inglés). Estos radares funcionan de igual manera que los de onda continua, pero modulando la frecuencia de la señal, lo que nos permite transmitir más información y ser más precisos en las mediciones [7].

¹¹ FMCW: frequency modulated continuous wave

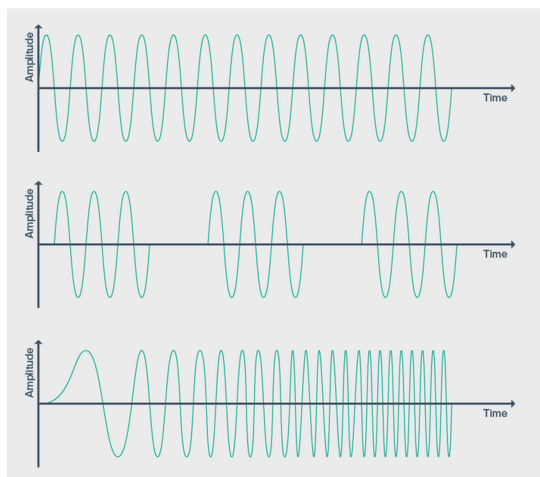


Figura 2. De arriba a abajo: radar de onda continua, radar pulsado, radar de onda continua modulado en frecuencia

2.2. Efecto Doppler

Supongamos que estamos transmitiendo una señal sinusoidal de frecuencia constante desde un transmisor estático en el tiempo. Esa señal rebotará contra otros objetos estáticos y será recibida por el transmisor sin modificación alguna, a excepción de las diferentes fuentes de ruido presentes en el sistema. El efecto Doppler es un fenómeno natural muy común en la transmisión de señales donde el transmisor o receptor están en movimiento con respecto al otro. Supongamos el primer caso, en el que el transmisor está estático y el receptor se está acercando a cierta velocidad. Debido a este movimiento, la señal rebotará cada vez antes en el receptor, aumentando la frecuencia de esta. Si, por el contrario, el receptor se está alejando del transmisor, la señal rebotará cada vez más tarde en el tiempo, disminuyendo la frecuencia de la señal. Estas pequeñas variaciones en la frecuencia de la señal debido al movimiento del sistema se conocen como efecto Doppler y es una característica a tener en cuenta en la localización por radiofrecuencia [8].

2.3. Radar FMCW

Los radares de onda continua modulados en frecuencia se basan en la transmisión de señales denominadas chirp¹² o rampas de frecuencia. Estas son señales sinusoidales cuya frecuencia varía en el tiempo desde una frecuencia inicial o frecuencia base hasta una frecuencia final a lo largo de la duración de la rampa. Siguen la gráfica mostrada en las figuras 3 y 4 [9].

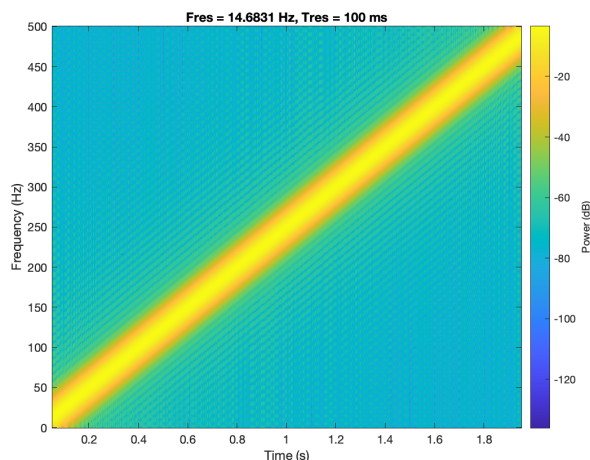


Figura 3. Rampa de frecuencia radar FMCW

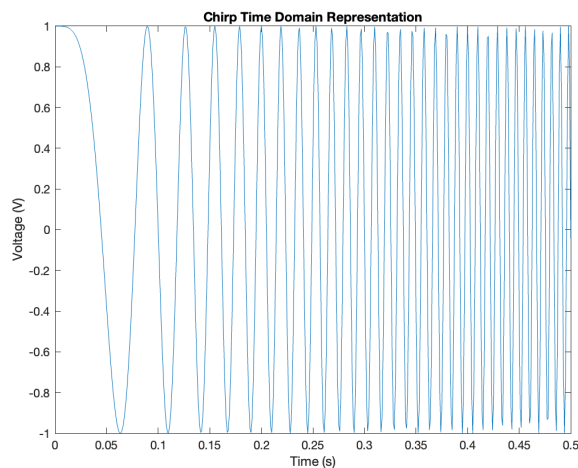


Figura 4. Representación en dominio tiempo chirp

¹² Nombre de la señal transmitida en un radar FMCW. Consiste en variar su frecuencia desde frecuencia base a frecuencia final

Un chirp se caracteriza por la duración del chirp, la frecuencia base, el ancho de banda y la pendiente de la rampa:

- Periodo de chirp (T_c): periodo de la señal transmitida. 2 segundos en la figura de arriba
- Frecuencia base (f_b) o frecuencia portadora: es la frecuencia fundamental del chirp, a la que se inicializa la rampa en el instante de tiempo inicial. $0H_z$ en la figura de arriba
- Ancho de banda (B): ancho de banda del chirp, Es el resultado de la resta entre la frecuencia final y la frecuencia base. $500H_z$ en la figura de arriba
- Pendiente (S): pendiente de la rampa. Siguiendo el ejemplo de arriba: $S = \frac{B}{T_c} = \frac{500}{2} =$

$$250 \frac{Hz}{seg}$$

2.3.1. Diagrama de bloques radar FMCW

El diagrama de bloques de un radar FMCW tiene los siguientes componentes, los cuales son fundamentales en la extracción de distancia, fase y velocidad.

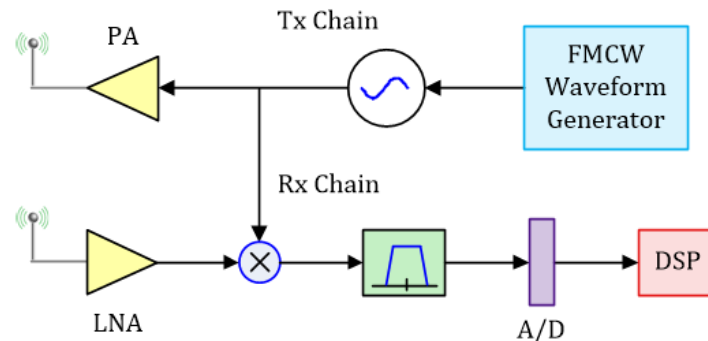


Figura 5. Diagrama de bloques radar FMCW. (Wirelesspi: <https://wirelesspi.com/fmcw-radar-part-1-ranging>)

En primer lugar, se genera la rampa de frecuencia o chirp, modelado bajo el bloque FMCW Waveform Generator¹³. En la práctica, se utilizará un PLL¹⁴. Una vez se ha generado la señal a

¹³ Generador de forma de ondas

¹⁴ Phase-locked loop o lazo de seguimiento de fase. Genera una forma de onda cuya fase está relacionada con la fase de una señal de entrada

transmitir, esta señal pasa sobre un amplificador de alta ganancia para aumentar la potencia de la señal transmitida.

Esta señal se transmite por la antena transmisora, rebota contra uno o varios obstáculos y es recibida de vuelta por la antena receptora. En algunos modelos de radar, se puede usar la misma antena tanto para transmitir como para recibir la señal. Una vez la señal es recibida por la antena receptora, pasa por un mezclador IQ¹⁵ donde las dos entradas son la señal transmitida y la recibida. A la salida del mezclador tendremos la señal de frecuencia intermedia o señal IF, una sinusoidal de frecuencia y fase igual a la diferencia de las señales transmitida y recibida, siguiendo el modelo que se muestra a continuación.

$$x_{Tx} = \sin(w_1 t + \phi_1)$$

$$x_{Rx} = \sin(w_2 t + \phi_2)$$

$$x_{IF} = \sin((w_1 - w_2)t + (\phi_1 - \phi_2))$$

Por último, la señal pasa por un filtro anti-solapamiento para evitar Aliasing¹⁶ con frecuencia de corte igual a la mitad de la frecuencia de muestreo del conversor analógico digital.

2.3.2. Mezclador IQ

El mezclador IQ es una técnica de modulación de señal muy común en la tecnología radar. Consiste en una única fuente de entrada y dos de salida, como se puede apreciar en la figura 6. La señal de entrada es dividida en dos ramas, una en fase y otra en cuadratura, IQ por su acrónimo en inglés (In-Phase and Quadrature). Ambas ramas son multiplicadas por la misma señal, con la diferencia de que a la rama Q se le añade un desfase de $\frac{\pi}{2}$. De ahí el nombre de la rama en fase (I) y la rama en cuadratura (Q) [10].

¹⁵ Tipo de modulación que divide la señal en dos ramas: una en fase y otra en cuadratura

¹⁶ Efecto producido debido al solapamiento espectral

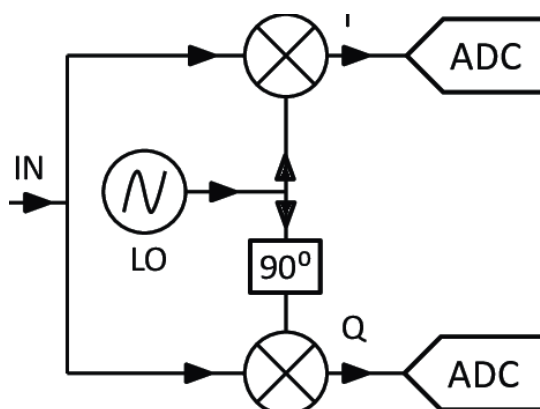


Figura 6. Mezclador IQ. (https://www.researchgate.net/figure/Block-diagrams-of-typical-IQ-detection-schemes-a-analogue-IQ-demodulation-via-a_fig2_357271616)

Esto resulta muy útil debido al efecto Doppler. Supongamos que tenemos un transmisor que transmite una señal con frecuencia 10KHz , debido al efecto Doppler causado por el movimiento del receptor alejándose del transmisor, la señal recibida es de frecuencia 8KHz . En una nueva transmisión el receptor cambia de dirección, pero no de velocidad, por lo que ahora la señal recibida tiene frecuencia 12KHz .

En el primer supuesto, la rama I tendrá componentes de frecuencia en 2KHz y 18KHz . Bajo el segundo supuesto, la rama I tendrá componentes de frecuencia en 2KHz y 22KHz . Debido a que la salida del mezclador es filtrada por un filtro paso bajo para quedarnos únicamente con la diferencia de frecuencias, en ambos escenarios, la salida tendrá únicamente frecuencia 2KHz , como se puede apreciar en las figuras 7 y 8.

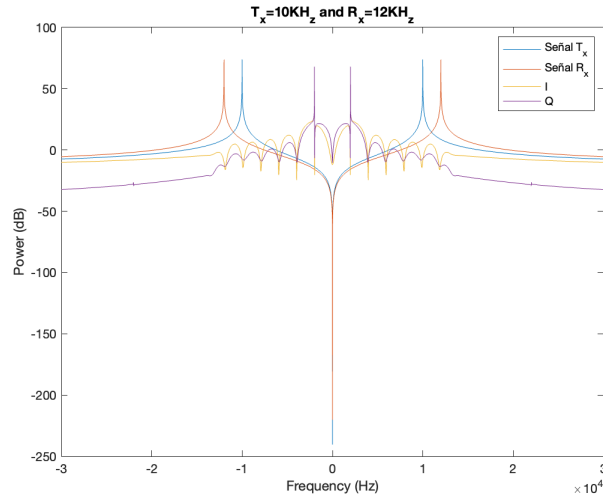


Figura 7. Ejemplo rama I/Q [1]

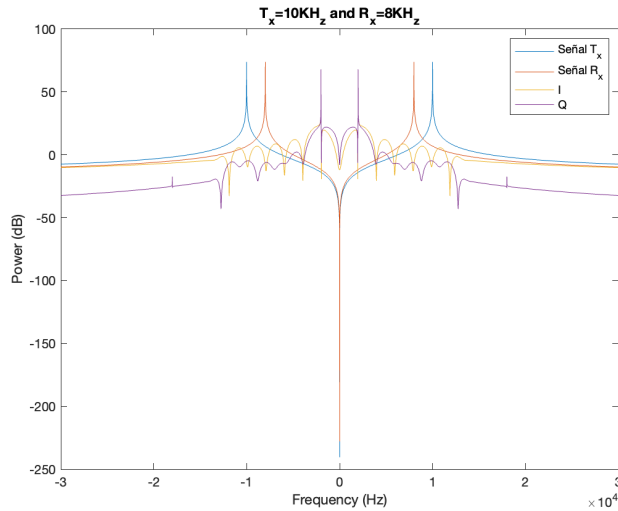


Figura 8. Ejemplo rama I/Q [2]

Gracias al modulador IQ, al añadir la diferencia de fase en la rama Q podemos distinguir el movimiento causado por el receptor. Como se puede observar en las figuras 9 e 10, si el receptor se está moviendo en dirección al transmisor y aumenta la frecuencia de la señal recibida, la rama I irá adelantada a la rama Q por $\frac{\pi}{2}$. Si por el contrario, el receptor se está alejando del transmisor y por tanto, la frecuencia de la señal recibida disminuye, la rama Q será la que vaya $\frac{\pi}{2}$ por delante con respecto a la rama I. Esto resulta muy útil para determinar la dirección del objeto en movimiento.

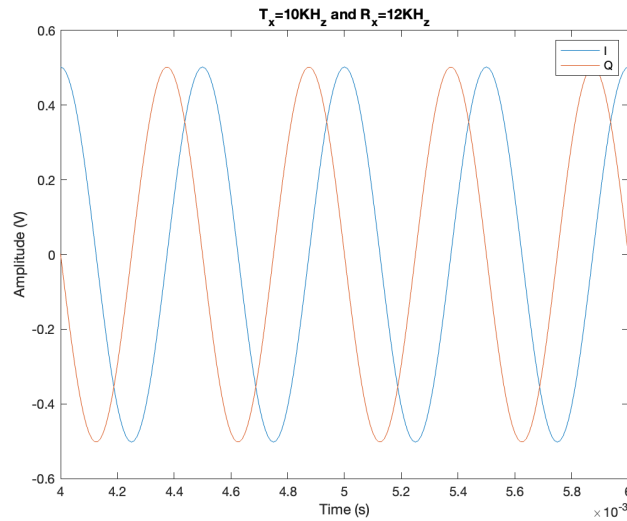


Figura 9. Rama I/Q [3]

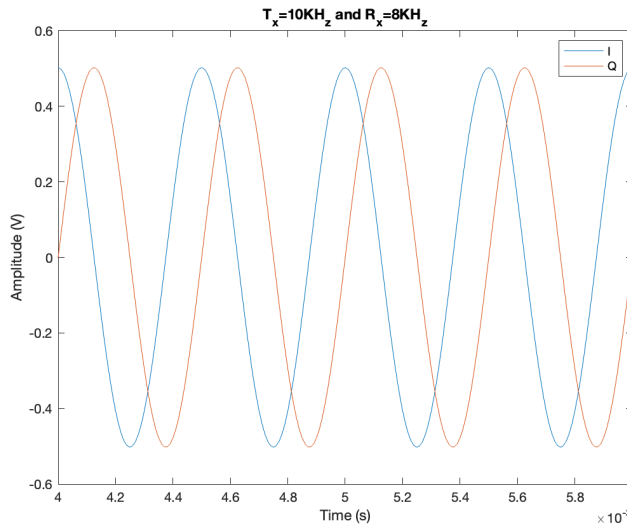


Figura 10. Rama I/Q [4]

2.3.3. Filtro FIR

Los filtros FIR¹⁷ son filtros que tienen una respuesta al impulso finita (FIR por su acrónimo en inglés: finite impulse response). Estos filtros se caracterizan por tener la siguiente respuesta al impulso [11].

$$|y[n]| = \sum_{k=-\infty}^{\infty} h[k]x[n-k]$$

Además, por definición, se trata de filtros estables ya que su respuesta al impulso está acotada.

$$|x[n]| < B_x$$

$$|y[n]| = \left| \sum_{k=-\infty}^{\infty} h[k]x[n-k] \right| \leq \sum_{k=-\infty}^{\infty} |h[k]| |x[n-k]|$$

$$|y[n]| \leq B_x \sum_{k=-\infty}^{\infty} |h[k]|$$

Por último y para que sean realizables, deben de cumplir la condición de causalidad.

$$h[n] = 0 \quad \forall \quad N_2 < n < N_1$$

Se trata de filtros muy comunes en el procesamiento digital de señales, los cuales se pueden representar mediante el siguiente diagrama de bloques.

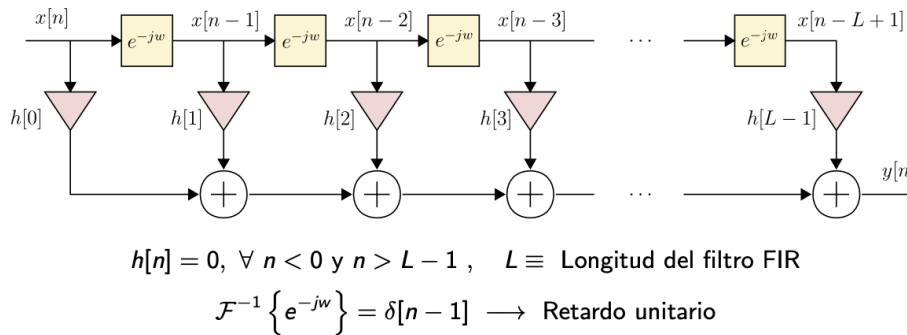


Figura 11. Diagrama de bloques filtro FIR. (Extraída de diapositiva 31 tema 3- Procesamiento Digital de Señales. Carlos García de la Cueva)

Se define como transitorio el periodo de tiempo durante el cual la memoria del filtro no está completa. Este transitorio añade una fuente de ruido al sistema y desplaza la señal de salida.

¹⁷ Filtro cuya respuesta al impulso es finita. Finite Impulse Response por su acrónimo en inglés

Podemos minimizar la fuente de ruido eliminando el transitorio en la señal filtrada, como se puede ver en las figuras 12 y 13.

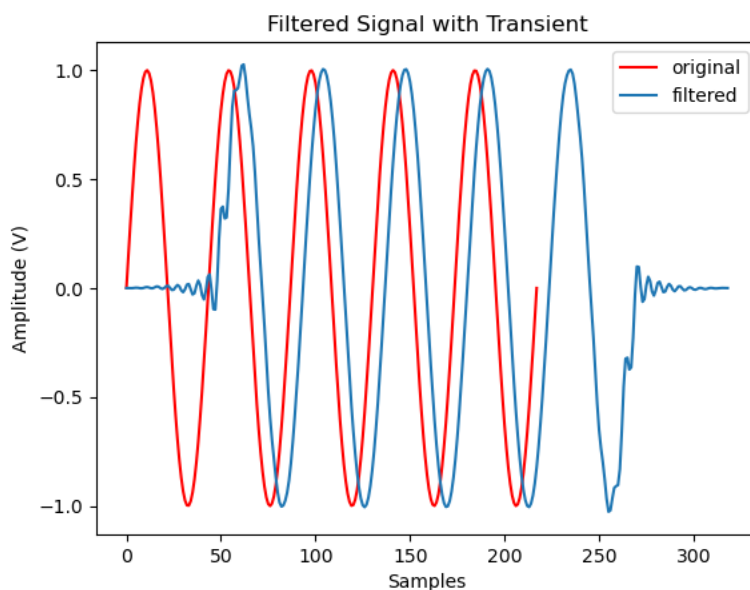


Figura 12. Filtro FIR con transitorio

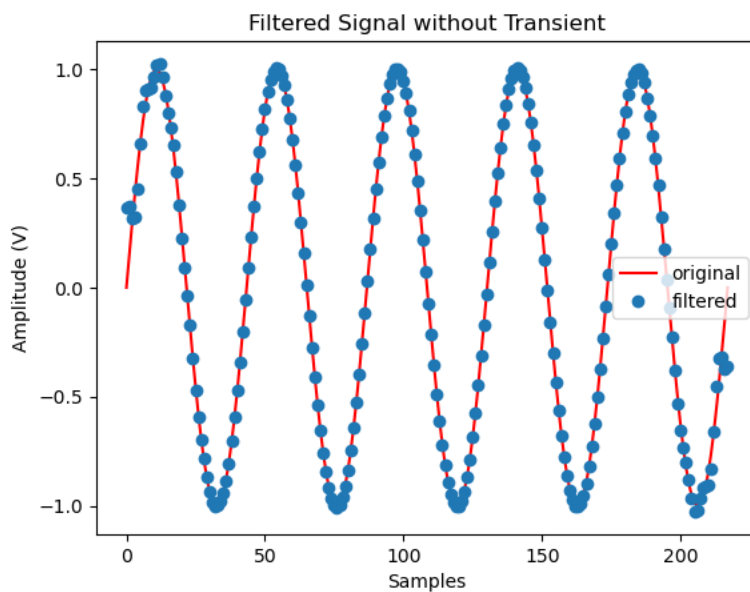


Figura 13. Filtro FIR sin transitorio

2.3.4. Cálculo de la distancia

Para el cálculo de la distancia nos centraremos en la gráfica frecuencia-tiempo de la señal chirp y el mezclador IQ. A la salida del mezclador, tendremos la señal de frecuencia intermedia, cuya frecuencia es la diferencia de las dos señales de entrada. Imaginemos dos sucesos. En primer lugar, la señal transmitida es rebotada por un único objeto y vuelve al transmisor tras un retardo de tiempo que denotaremos τ . Al tener ambas señales la misma pendiente de frecuencia, la señal IF será un tono, como se observa en la figura 14, por lo que un objeto rebotado se traduce en un tono en frecuencia. Pongámonos ahora en el caso de que hay varios objetos en el campo de visión del transmisor, por lo que tendremos varios chirp rebotados. En este caso, la señal IF será una sinusoidal compuesta por varios tonos, como se puede ver en la figura 15. Recordar los parámetros de una señal chirp descritos en el apartado 2.3.1.: $S \equiv$ pendiente de la rampa, $T_c \equiv$ periodo del chirp.

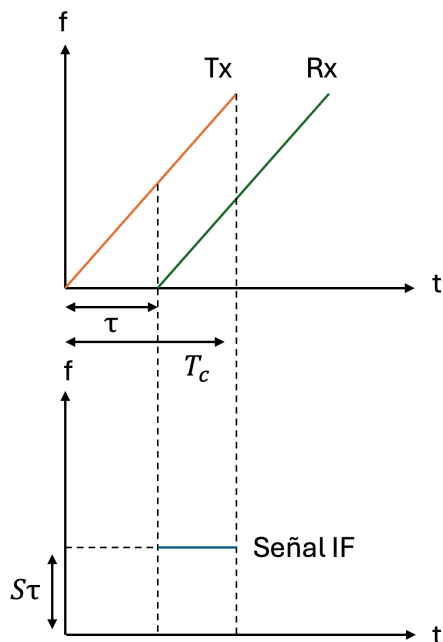


Figura 14. Señal IF para un obstáculo

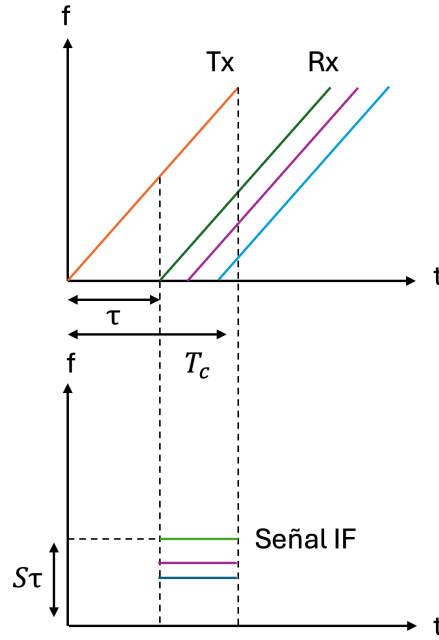


Figura 15. Señal IF con varios obstáculos

La frecuencia del tono producido se puede calcular fácilmente con la pendiente de la rampa y el retardo temporal. Debido a que una onda electromagnética se propaga a la velocidad de la luz, podemos relacionar el retardo temporal τ con la distancia hacia el objeto R de la siguiente manera:

$$c = \frac{R}{\frac{\tau}{2}} \rightarrow \tau = \frac{2R}{c} [s]$$

Si sustituimos por τ , obtenemos una relación directa entre la frecuencia del tono y la distancia al objeto.

$$f_{IF} = S\tau = \frac{S2R}{c} \rightarrow R = \frac{f_{IF}c}{2S} [m]$$

Existen ciertas limitaciones en el cálculo de la distancia. Una de ellas es la resolución del radar, entendiendo por resolución la mínima separación que debe de haber entre dos cuerpos para que el radar los identifique como dos objetos diferentes. Si estos dos cuerpos están demasiado cerca, corre el riesgo de que llegue la misma señal rebotada y no seamos capaces de separarlos.

$$\Delta f = \frac{1}{T_c} \rightarrow \frac{S2\Delta R}{c} > \frac{1}{T_c} \rightarrow \Delta R > \frac{c}{2ST_c} = \frac{c}{2B} [m]$$

$$B = ST_c$$

Por lo que la resolución del radar viene dada por el ancho de banda de este. Un mayor ancho de banda implica una mayor resolución.

El cálculo de la distancia también se ve afectado por la frecuencia de muestreo del conversor analógico digital que se utiliza a continuación para digitalizar la señal. Para que se cumpla el teorema de Nyquist y no haya solapamiento espectral, la frecuencia de muestreo del conversor analógico digital debe de ser al menos 2 veces mayor que la frecuencia máxima de la señal IF. Esto limita la frecuencia máxima detectable por el radar, y por tanto, la distancia máxima.

$$f_s \geq 2f_{max} \rightarrow f_{max} \leq \frac{f_s}{2}$$
$$R_{max} \leq \frac{f_s c}{4S} [m]$$

2.3.5. Cálculo de la velocidad

Para el cálculo de la velocidad estudiaremos las variaciones en la fase de la señal de frecuencia intermedia. Como hemos visto anteriormente, la señal IF será una sinusoidal de frecuencia y fase igual a la diferencia de frecuencia y fase entre la señal transmitida y la señal recibida. Estudiando estas diferencias en fase entre señales IF consecutivas somos capaces de obtener medidas precisas de velocidad. En las figuras 16 y 17 se observa la variación de fase de la señal IF causado por un retardo temporal τ entre la señal transmitida y recibida. Este retardo se debe al movimiento de un obstáculo en el campo de visión del transmisor.

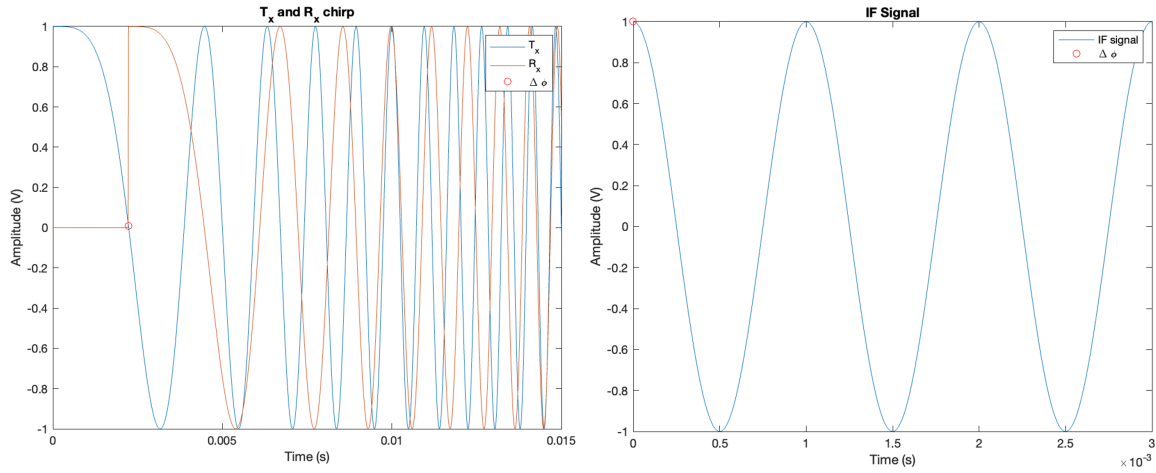


Figura 16. De izquierda a derecha: señales Tx y Rx, señal IF

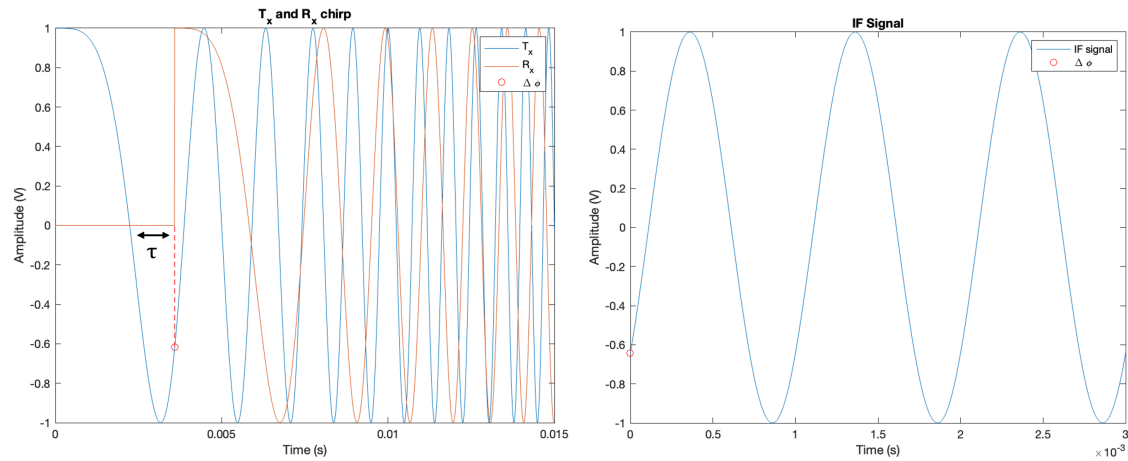


Figura 17. De izquierda a derecha: señales Tx y Rx, señal IF

Podemos calcular la variación de fase de la siguiente manera:

$$\Delta\phi = 2\pi f\Delta\tau = \frac{4\pi\Delta R}{\lambda} = \omega$$

Por lo que tenemos una manera de medir cambios de fase en función de cambios en desplazamiento. Sabiendo el tiempo entre cada desplazamiento (o entre cada chirp) podemos obtener la velocidad del movimiento. Si transmitimos varios chirp consecutivos, de tal manera que uno se transmite justo inmediatamente cuando acaba el anterior, la diferencia temporal entre cada chirp es únicamente de T_c . Esta transmisión de varios chirps de manera consecutiva se conoce

como grupo de chirp. Una vez conocida la diferencia temporal entre cada chirp podemos calcular la velocidad del movimiento con la siguiente expresión.

$$\Delta R = vT_c \rightarrow \omega = \frac{4\pi v T_c}{\lambda} \rightarrow v = \frac{\lambda \omega}{4\pi T_c}$$

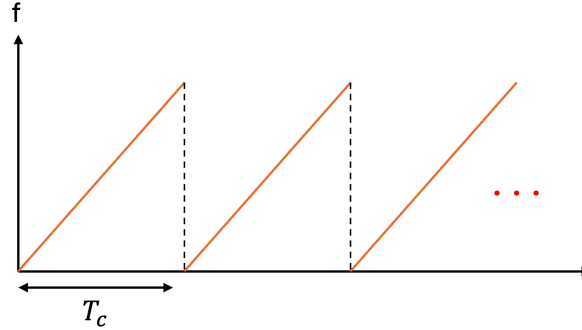


Figura 18. Grupo de chirps

Supongamos que tenemos un radar FMCW de 120 GHz con $T_c = 2.6m_s$, $B = 1GHz$ y un objeto varía su posición $0.6mm$, el cambio en la fase corresponde a $\Delta\phi = \frac{4\pi\Delta R}{\lambda} = \pi$ rad. La variación en frecuencia será únicamente de $\Delta f = \frac{S2\Delta d}{c} = 1.5Hz$. Esta variación es muy complicada de ver bajo el ojo humano, ya que normalmente las frecuencias de la señal IF de un radar están en torno a los kilohercios o megahercios. Es por esto por lo que la velocidad se vuelve una medida de gran precisión que resulta muy útil para seguir pequeños movimientos.

Al igual que con el cálculo de la distancia de un objeto, también existen diferentes limitaciones a las que hacer frente cuando calculamos la velocidad por la fase de la señal. Como norma general, podemos utilizar el signo de la velocidad para ver la dirección del movimiento causado. Si el signo es negativo, el objeto está más próximo al radar. Si, por el contrario, la diferencia de fase es positiva, el objeto se está alejando del radar, como se puede apreciar en la figura 17. Esto conlleva una limitación en cuanto a la medición de velocidades, ya que si la diferencia en fase es mayor que π , no tenemos ninguna manera de determinar si la diferencia ha sido positiva o negativa, debido a la periodicidad de las señales sinusoidales. Es por esto, por lo que toda medición de velocidad está limitada a variaciones en fase menores que π .

A partir de esto, podemos obtener la máxima velocidad medible bajo la siguiente expresión.

$$\frac{4\pi v_{max} T_c}{\lambda} < \pi \rightarrow v_{max} < \frac{\lambda}{4T_c}$$

Como se observa, la máxima velocidad depende de la frecuencia de la señal, a mayor frecuencia, mayor será la velocidad máxima. Esta velocidad también depende de la duración del chirp, lo cual tiene sentido ya que mayor duración de chirp significa mayor equidistancia entre chirps consecutivos, por lo que las variaciones de movimiento y fase serán mayores.

Además, existe la posibilidad de que dos objetos se encuentren a la misma distancia del objeto, pero con distintas velocidades, por lo que en la representación en frecuencia de la señal IF, solo veríamos un pico y nos sería imposible distinguir los objetos. En este caso, analizamos las diferencias de fase de ese pico a lo largo de una ventana temporal y realizamos la transformada de Fourier de esas diferencias de fase, lo que se denomina, como FFT¹⁸ de 2 dimensiones, o transformada de Doppler. Si ese pico en frecuencia es el correspondiente a dos objetos que se mueven con velocidades distintas, tendremos dos picos en el dominio de la frecuencia. Si contrario se trata de un único objeto moviéndose a una velocidad, tendremos un único pico. Finalmente, si ese pico es el correspondiente a dos objetos moviéndose a la misma velocidad, no tenemos manera de separar dichos movimientos y el radar estaría produciendo un error, hasta que uno de esos dos objetos cambie de velocidad.

¹⁸ Algoritmo para la realización de la Transformada de Fourier de manera eficiente

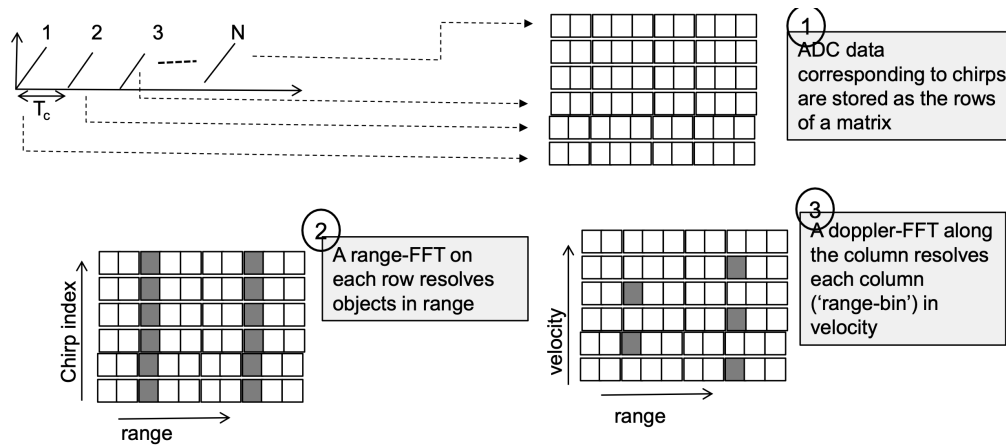


Figura 19. FFT de 2 dimensiones. (*mmWaveSensing: An introduction to FMCW radar technology*)

2.4. Entorno ROS

ROS¹⁹ es una plataforma de software de código abierto diseñada para facilitar el desarrollo de aplicaciones robóticas. Proporciona herramientas y bibliotecas que permiten la creación de sistemas modulares, mejorando la comunicación entre componentes. Además, ROS 2 soporta sistemas operativos en tiempo real, ofrece mejoras en seguridad y escalabilidad, y promueve la interoperabilidad entre diferentes robots y dispositivos, lo que lo convierte en una solución robusta y flexible para la robótica moderna.

La librería ROS2 trata de reducir la complejidad de un sistema robótico, dividiendo sus acciones en nodos. Cada nodo es responsable de realizar una única acción, la cual es modular y varía desde controlar las ruedas de un motor a leer los datos de un sensor. Los nodos se comunican entre sí mediante tópicos, los cuales funcionan como canales de comunicación. Un nodo puede estar suscrito a cero o varios tópicos, por lo que estará escuchando ese tópico y los mensajes que se publican por él. Por el contrario, un nodo puede publicar contenido en ningún o varios tópicos. Todos los mensajes que se publican en un tópico utilizan el protocolo IEEE 802.11²⁰. De esta

¹⁹ Robot Operating System 2. Plataforma software utilizada para el desarrollo de robots

²⁰ Una serie de estándares creado por el Institute of Electrical and Electronics Engineers. Agrupan la tecnología WiFi

manera, los diferentes componentes de un robot se pueden comunicar entre sí con tal de estar operando bajo la misma red.

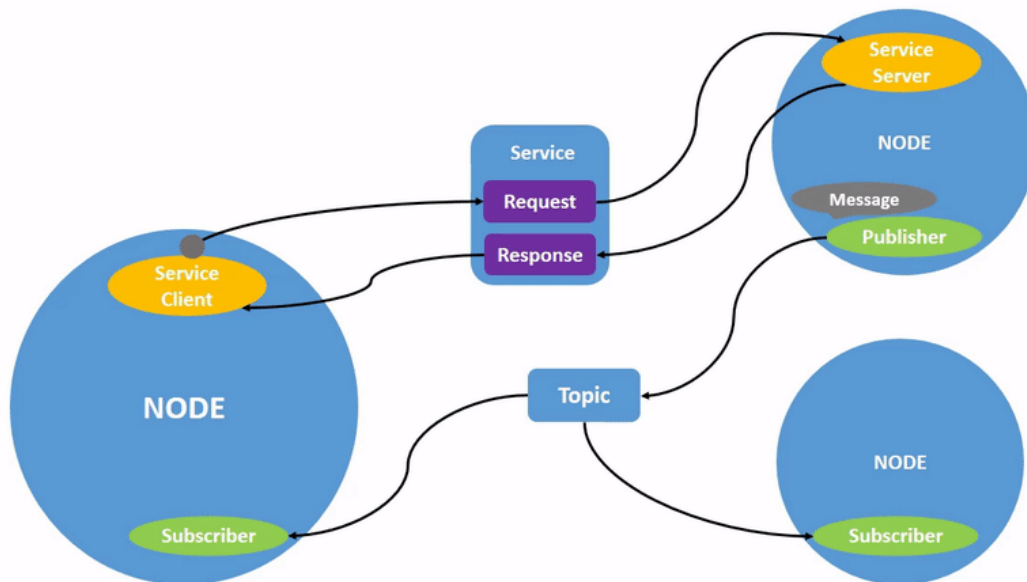


Figura 20. Diagrama de flujo entorno ROS2. (<https://docs.ros.org/en/iron/Tutorials/Beginner-CLI-Tools/Understanding-ROS2-Nodes/Understanding-ROS2-Nodes.html>)

Capítulo 3. Estado de la cuestión

Debido a la gran necesidad de localizar y seguir pequeños o grandes movimientos con gran precisión en tiempo real, el uso de la tecnología radar de onda continua modulada en frecuencia se ha visto incrementado en gran medida en los últimos años. Esta tecnología está presente en los principales campos de la ingeniería: desde la aviación y automoción hasta la salud.

Un claro ejemplo de ellos son los UAVs²¹, los cuales han experimentado un crecimiento significativo en aplicaciones tanto comerciales como militares. Uno de los componentes cruciales para la navegación segura y la detección de obstáculos en UAVs es el radar de onda continua modulada en frecuencia. Estos radares son de vital importancia debido a su capacidad de extraer con gran precisión el rango y velocidad de diferentes obstáculos, lo que los hace extremadamente útiles en aplicaciones de entrega de paquetes, inspección de infraestructura o misiones de búsqueda y rescate. Compañías como DHL, Manna, Google Wing, UPS y Zipline han mostrado gran interés en este tipo de aplicación de los radares FMCW

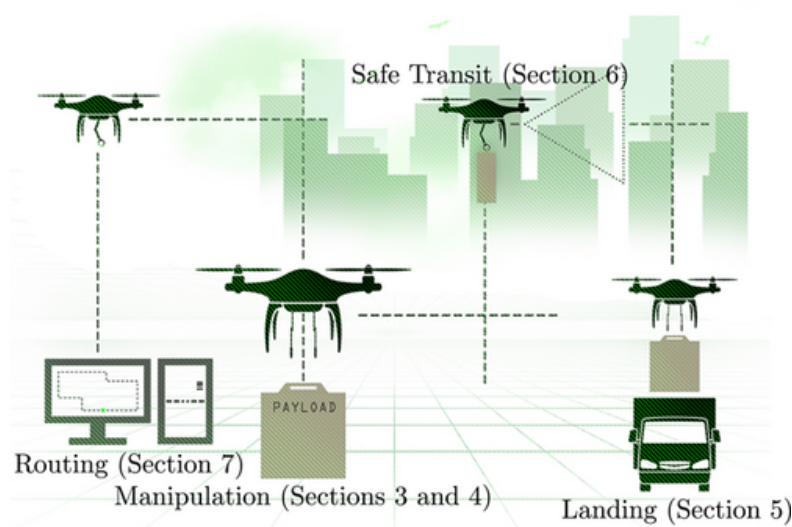


Figura 21. Diagrama UAVs + radar FMCW. (<https://onlinelibrary.wiley.com/doi/10.1002/rob.22231>)

²¹ Del inglés Unmanned Aerial Vehicle. Vehículo aéreo no tripulado

Además, los radares FCMW están ganando una presencia significativa en el mundo de la automoción debido a sus capacidades avanzadas y versatilidad como sensores en sistemas de asistencia al conductor y conducción autónoma. Estos radares son esenciales para aplicaciones como el control de crucero adaptativo (ACC²²), que ajusta la velocidad del vehículo para mantener una distancia segura respecto al vehículo que le precede, y la frenada automática de emergencia (AEB²³), que puede prevenir colisiones o mitigar su impacto. La capacidad de los radares FCMW para medir tanto la distancia como la velocidad de los objetos en movimiento con alta precisión los hace ideales para detectar otros vehículos, peatones y ciclistas, mejorando así la seguridad en diversas condiciones de conducción. Además, los radares FCMW operan de manera efectiva en todas las condiciones meteorológicas, incluyendo lluvia intensa, niebla y nieve, donde otros sensores como las cámaras pueden fallar. Su integración en sistemas de detección de punto ciego, asistencia de mantenimiento de carril y estacionamiento automático subraya su importancia en la creación de un entorno de conducción más seguro y autónomo [13].

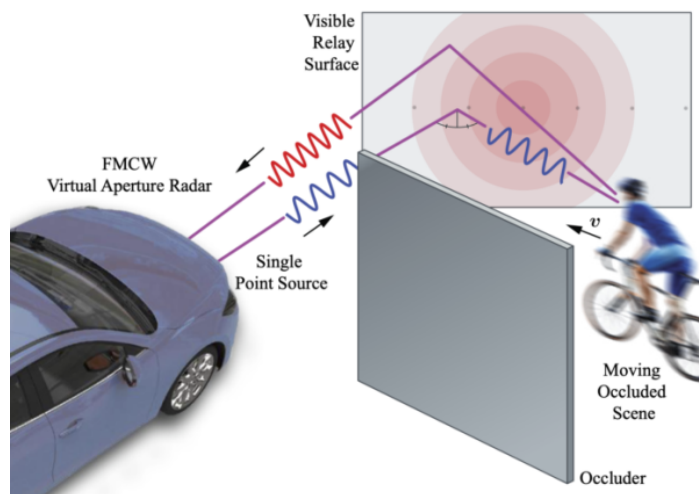


Figura 22. Radar FCMW en automoción. (<https://www.dekra.com/en/sensors-when-the-car-sees-around-the-corner/>)

²² Del inglés Adaptive Cruise Control

²³ Del inglés Automatic Emergency Braking

El uso de radares FMCW también se ha visto incrementado con el avance de la tecnología MIMO²⁴. Esta tecnología consiste en formar un array de antenas transmisores y receptoras. Las señales transmitidas son divididas y codificadas en el número de antenas del array de tal manera que cada antena transmite una porción del mensaje. En recepción, todos los mensajes se decodifican individualmente en cada antena y se combinan para conseguir mayor robustez frente al ruido, mejorando la calidad señal ruido y la capacidad total del canal. En el contexto de la tecnología MIMO, los radares FMCW en automoción pueden ofrecer una resolución espacial y angular aún mayor, permitiendo una mejor detección y clasificación de objetos en el entorno del vehículo. Esta capacidad mejorada es crucial para el desarrollo de vehículos autónomos que puedan navegar de manera segura y eficiente en entornos complejos y dinámicos [14].

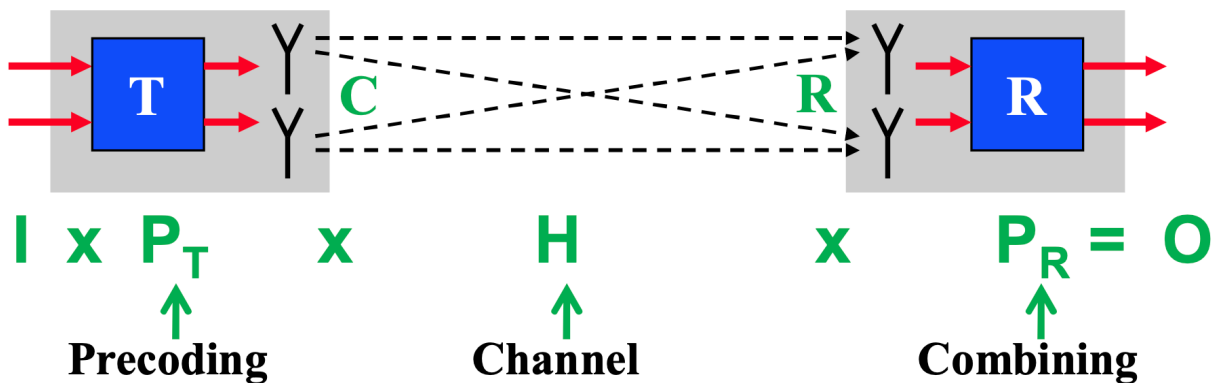


Figura 23. Esquema transmisión y recepción estaciones MIMO. (Proffessor Steenkiste MIMO slides, Wireles Networks CMU)

Por último, la tecnología radar FMCW también ha presentado un gran impacto en la medicina moderna ya que puede utilizar de manera innovadora para monitorizar los movimientos respiratorios y el ritmo cardíaco humano, ofreciendo una solución no invasiva y precisa para el monitoreo de salud. Cuando una persona inhala o exhala, el pecho y el abdomen se mueven, creando cambios en la distancia entre el radar y la superficie del cuerpo. Los radares FMCW son capaces de detectar estos pequeños desplazamientos gracias a su alta resolución y sensibilidad, midiendo con precisión la frecuencia y amplitud de los movimientos respiratorios. Además, el

²⁴ Del inglés Multiple In Multiple Out

radar FMCW puede detectar las diminutas vibraciones en la piel causadas por los latidos del corazón. Estas microvibraciones, aunque apenas perceptibles a simple vista, generan cambios en la distancia y la velocidad que el radar puede captar. Al analizar estas variaciones en las señales de retorno, los algoritmos de procesamiento de señal pueden extraer información detallada sobre el ritmo cardíaco, como la frecuencia de los latidos y la variabilidad de la frecuencia cardíaca (HRV²⁵). Esta tecnología es especialmente útil en entornos donde el contacto físico es limitado, proporcionando un monitoreo continuo y sin contacto. Además, su capacidad para operar en diversas condiciones ambientales hace que los radares FMCW sean ideales para aplicaciones en telemedicina y cuidado remoto, facilitando la supervisión de pacientes con enfermedades crónicas o en recuperación sin necesidad de visitas frecuentes al hospital [15].

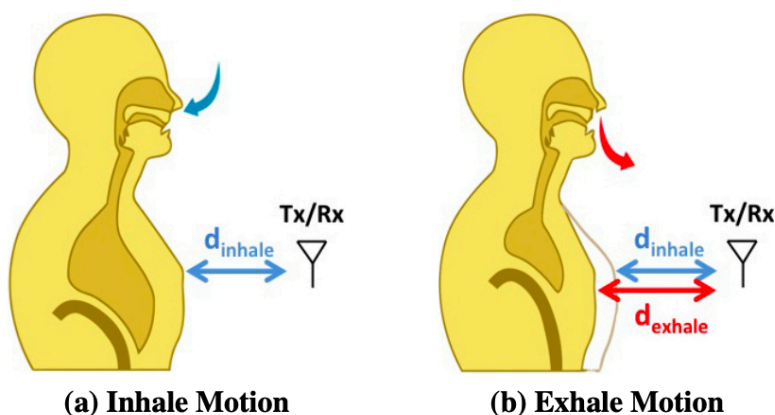


Figura 24. Exhalar e inhalar contracción pecho. (Publicación científica. Smart Homes that Monitor Breathing and Heart Rate)

²⁵ Del inglés: Heart rate variability

Capítulo 4. Definición del Trabajo

4.1. Justificación

La evolución de sistemas de medición ha llevado a la creación de entornos inteligentes, capaces de monitorizar nuestras actividades y ajustarse a las mismas. Estos ajustes pueden ser subir o bajar la calefacción basado en la temperatura corporal del usuario, responder a nuestros gestos o monitorizar a los más mayores. Durante este trabajo de fin de grado, se busca poner en práctica los conocimientos sobre ingeniería de telecomunicaciones aprendidos durante la carrera con el fin de construir un sistema inteligente capaz de salvar vidas y optimizar los recursos sanitarios.

Con la robótica avanzando a pasos agigantados, cada vez es más común emplear robots autónomos o semiautónomos para la exploración y realización de tareas costosas. Esto tiene un especial valor en misiones espaciales de larga duración, al tener un coste mucho menor mandar a un robot al espacio en lugar de una nave capaz de soportar vida humana. Otro escenario donde se pone en práctica es en misiones militares, siendo muy utilizada la práctica de enviar robots de antemano para hacer un reconocimiento de la zona o para ofrecer apoyo sanitario.

Tomar mediciones del corazón puede resultar una tarea compleja y costosa, pues necesita del contacto físico y de al menos un instrumento por paciente, siendo complicado la toma de mediciones simultáneas. Estos problemas se acentúan aún más en entornos de guerra o catástrofes naturales, donde el contacto físico es limitado y la rapidez de tomas se vuelve primordial. Este es el problema planteado en este trabajo de fin de grado: ser capaces de tomar mediciones del ritmo cardíaco de una persona sin contacto físico y sin necesidad de presencia o acción humana. Para ello, se ha seleccionado la tecnología radar de 120GHz que debido a su bajo coste adquisitivo (al tratarse de radares de baja potencia) y a su gran precisión en la toma de mediciones de rango, fase y velocidad se convierte en la solución ideal.

Los radares FMCW ofrecen una precisión excepcional en la detección de pequeños movimientos, lo que es crucial para captar las diminutas vibraciones en la piel causadas por los latidos del corazón y el movimiento que realizan las personas al inhalar o exhalar. La frecuencia de 120GHz permite una resolución aún mayor, mejorando la capacidad del sistema para distinguir entre los micro-movimientos del ritmo cardíaco y otros movimientos corporales. Estos micro-movimientos cuentan con la característica de ser periódicos en el tiempo, por lo que resulta muy fácil aislarlos de otros movimientos utilizando una ventana temporal y comprobando su periodicidad.

El radar FMCW proporciona una solución de monitoreo completamente inalámbrica, lo cual es un avance significativo en comparación con los métodos tradicionales que requieren contacto físico, como los electrocardiogramas (ECG²⁶). La capacidad de medir el ritmo cardíaco sin contacto directo es especialmente útil en situaciones donde el contacto físico es limitado o debe evitarse.

Por último, la tecnología de radar FMCW puede revolucionar la manera en que se monitorea la salud, permitiendo una supervisión continua y remota del ritmo cardíaco de los pacientes. Esto es particularmente relevante para personas con enfermedades crónicas, pacientes en recuperación, y aquellos que requieren monitoreo constante pero no pueden realizar visitas frecuentes al hospital. En el contexto de la telemedicina, esta tecnología facilita un seguimiento preciso y en tiempo real, mejorando la calidad del cuidado sin necesidad de visitas presenciales.

²⁶ Del inglés: electrocardiogram

4.2. *Objetivos*

Los objetivos que se persiguen en este trabajo son los siguientes.

4.2.1. **Comprender parámetros y procesamiento de un radar FMCW**

Se busca comprender el funcionamiento del radar FMCW y el porqué de sus componentes. Como se extrae el rango, la fase y la velocidad de diferentes cuerpos únicamente analizando el rebote causado al transmitir una señal en el tiempo. Además, se quiere optimizar la configuración del radar para conseguir máxima precisión posible en las mediciones.

4.2.2. **Comprender entorno ROS2**

Este proyecto forma parte de un robot con muchos más componentes, por lo que se trabajará en paralelo con otras aplicaciones. Para ello, es de vital importancia el entorno ROS2, el cual permite la fácil conectividad entre sensores a través de diferentes tópicos en la red.

4.2.3. **Protocolo de funcionamiento y configuración del radar EasyRad r4²⁷**

En este proyecto se trabajará con el radar SiRad Easy r4, fabricado por Indie Semiconductors, por lo que supone de vital importancia comprender el flujo de medición de este radar en particular y configurarlo de la manera óptima para perseguir los objetivos descritos.

4.2.4. **Detectar la distancia a la que se encuentran los objetos**

Consiste en la implementación del software necesario para la detección de distancia de los diferentes objetos en el rango de interés. Para ello se debe tener un amplio conocimiento de la tecnología radar de onda continua modulada en frecuencia y del protocolo del radar seleccionado.

²⁷ Radar seleccionado para el desarrollo de este proyecto. Es fabricado por Indie Semiconductors

Además, es crítico minimizar las fuentes de ruido y las posibles limitaciones hardware, físicas, y/o de software.

4.2.5. Detectar la velocidad a la que se mueven los objetos

Implementar la detección de velocidad de los diferentes movimientos en el área de interés del radar. Esta fase conlleva trabajar con la fase de la señal, ya que es lo que utilizaremos para detectar la velocidad. Tiene una importancia vital, al ser lo que se utilizará para medir el ritmo cardiaco

4.2.6. Transmisión de las mediciones de distancia, fase y velocidad

Una vez recopilada la información de distancia, fase y velocidad de los diferentes cuerpos en el rango de interés, estos se deben transmitir por la red haciendo uso de tópicos del framework ROS2. Posteriormente, otro nodo del robot que está fuera del alcance de este proyecto analizará las diferencias en fase y velocidad para extraer el ritmo cardiaco y respiratorio de diferentes personas.

4.3. Metodología

Para el cumplimiento de los objetivos se va a seguir la siguiente metodología de trabajo, la cual podemos dividir en dos grandes fases.

4.3.1. Estudio teórico

La fase del estudio teórico engloba los primeros tres objetivos: comprender parámetros y procesamiento de un radar FMCW, entorno ROS2 y el protocolo del radar seleccionado para el proyecto. Durante esta fase se hará uso de herramientas online, como el curso “*mmWave Radar Sensors*” de Texas Instruments, concretamente, el bloque de “*Introduction to mmWave Sensing: FMCW Radars*”, para completar el objetivo número 1. Este bloque cuenta con cinco capítulos en los que se explica con detalle el procesamiento y la lógica que hay detrás de un radar FMCW. El material es reforzado por los contenidos de propagación de señales en el espacio que se vio en

“Radiación y Propagación” y “Wireless Networks”, cursados a lo largo de la carrera, y el curso “Teoría de la Comunicación”, para comprender el funcionamiento del modulador IQ.

El objetivo número dos se basa en comprender el entorno de trabajo ROS2 Iron. Para ello, se ha utilizado los tutoriales de la documentación oficial. Estos tutoriales consisten en seis capítulos en los que se explica en detalle el funcionamiento y la lógica del framework. Para la realización de este trabajo se han hecho uso únicamente de tres de ellos: “Beginner: CLI tools”, “Beginner: Client Libraries” y “Demos”.

Para el objetivo número tres, se estudiará el manual del fabricante, proporcionado por Indie Semiconductors, el cual explica con detalle el flujo de funcionamiento del radar y sus registros de configuración.

4.3.2. Implementación

La segunda fase consiste en la implementación software de los diferentes componentes del radar y engloba los objetivos restantes. Para el procesamiento de la señal, se han usado dos nodos del entorno ROS2. El primero de ellos tiene la funcionalidad de decodificar los datagramas del radar y transmitir la información decodificada por un tópico llamado /radar/data_frame. A continuación, se ha implementado otro nodo suscrito a ese tópico, que recibe las tramas decodificadas e implementa los algoritmos de detección de distancia, fase y velocidad, transmitiendo dicha información por otro tópico llamado /radar/target_frame. También se ha creado un tercer nodo para la configuración del radar, que publicará en el tópico /radar/serial_write los comandos de configuración que luego serán transmitidos al radar mediante conexión UART. La implementación de los nodos es realizada en Python.

Durante todo este proceso, se ha trabajado de tres maneras diferentes:

- Foxglove²⁸: se trata de un software parte del entorno ROS2 que nos permite visualizar los datos transmitidos por los tópicos en tiempo real. Se ha utilizado principalmente para fases iniciales con el objetivo de comprobar el funcionamiento correcto del radar.
- Ficheros ROS bags: los cuales almacenan mensajes ROS en archivos binarios. Son muy útiles a la hora de grabar información de sensores y tópicos del robot para su posterior reproducción y procesamiento. Se ha utilizado durante la mayor parte del procesamiento de la señal para poder trabajar de manera local y sin conexión a la red en la que opera el robot.
- Con conexión a la red del robot: debido a que el robot publica los tópicos en la red, de esta manera se puede analizar el funcionamiento del radar y ejecutar los algoritmos de detección de objetos en tiempo real. Ha sido fundamental a la hora de analizar los resultados y extraer conclusiones.

4.4. Planificación y Estimación Económica

A lo largo de este apartado, se explicarán en detalle la planificación seguida y la estimación económica del coste total del radar, obviando los demás componentes del robot.

4.4.1. Planificación

La planificación del proyecto está dividida en cinco fases, las cuales se detallan a continuación.

1. Tecnología radar de onda continua modulada en frecuencia. La primera fase del proyecto consiste en estudiar la tecnología radar y en concreto, la tecnología radar de onda continua modulada en frecuencia. Esta fase dura 6 semanas, dedicando una semana a cada uno de los elementos que componen la tecnología radar FMCW. En la semana uno se analizará la señal chirp y de frecuencia intermedia, en la semana dos, el cálculo de la distancia, durante

²⁸ Software de código libre para la visualización de tópicos en el entorno ROS2

la semana tres la fase y la sensibilidad de la señal, la semana cuatro está dedicada a el estudio de la velocidad de los objetos. Por último, en las últimas dos semanas se estudiarán los parámetros de diseño de un radar FMCW y la relación señal ruido necesaria para decodificar la señal. Una vez completada la primera fase del proyecto, se debe tener un amplio conocimiento teórico acerca de la tecnología de radar de onda continua modulada en frecuencia.

2. ROS2 Iron. Durante esta segunda fase se planea estudiar y entender el funcionamiento del framework ROS2 Iron. Esta fase consta de dos semanas. Durante la primera semana se estudiarán los diferentes componentes de un entorno ROS y como se comunican desde una perspectiva teórica. Será durante la segunda semana cuando se empiece a programar los diferentes nodos y tópicos que serán necesarios para la implementación del radar.
3. Protocolo del SiRad Easy r4. La fase tres del proyecto tiene el objetivo final de tener el radar en funcionamiento y tomando medidas de manera continua. Para ello, se dedicará tres semanas al estudio del protocolo del radar elegido, el SiRad Easy r4 de Indie Semiconductors. Durante la primera semana, se estudiará el flujo de medición del radar y el diagrama de bloques de los diferentes componentes. A continuación, se estudiará los registros de configuración del radar y sus posibles valores. Por último, se dedicará una semana para configurar el radar con los valores óptimos elegidos. Para ello, se hará uso de un microcontrolador, el cual a través de la UART y una comunicación en serie se comunicará con el radar, escribiendo sus diferentes registros de configuración. La conexión UART con el microcontrolador no forma parte de este proyecto y ha sido realizada por una persona externa al mismo.
4. Software de procesamiento de la señal. A lo largo de la fase número cuatro del proyecto se busca la obtención de las medidas finales de distancia, fase y velocidad del radar. Para ello, se utilizará poco más de dos meses, en concreto, nueve semanas. Las dos primeras semanas

están dedicadas a la programación de un nodo en el entorno ROS2 que decodifique los datagramas que transmite el radar por el puerto serie. Este nodo, además deberá publicar dicha información decodificada en un tópico. A continuación, se dedicarán dos semanas a la detección de la distancia de los diferentes cuerpos. Esto supone la implementación de un nodo que esté suscrito al tópico mencionado previamente y cuya función es implementar el procesamiento digital de la señal del radar. Una vez obtenidas las mediciones de distancia, se añadirá la funcionalidad de detectar la velocidad de los diferentes objetos a través de la fase y el retardo temporal de la señal. Por último, se dedicará la última semana para la programación de un tópico en el que se publique toda la información relativa a los objetos detectados: número de objetos, distancia a cada uno, magnitud de potencia recibida de cada uno, fase y velocidad. Al terminar la fase cuatro del proyecto, se debe tener el trabajo completado en toda su totalidad, faltando la documentación de la memoria, tarea que se realizará durante la quinta fase.

5. Documentación del proyecto. La quinta y última fase del proyecto está dedicada a documentar el trabajo realizado. Esta fase se realiza de manera paralela a la fase número dos, una vez se tenga conocimiento del funcionamiento de la tecnología FMCW y una vez terminada la fase cuatro, cuando el trabajo esté completo.

A continuación, se muestra una tabla detallando la duración de cada actividad. En la tabla, cada fase tiene un color específico para facilitar su lectura, y cada cuadrado representa una semana en el tiempo, comenzando la semana del 10 de diciembre de 2023 y acabando la semana del 10 de junio de 2024.

DEVELOPMENT OF 120GHz FMCW RADAR FOR HEART AND RESPIRATORY RATE PROJECT TIMELINE																													
ID	Task Name	10-12	17-12	24-12	31-12	7-1	14-1	21-1	28-1	4-2	11-2	18-2	25-2	4-3	11-3	18-3	25-3	1-4	8-4	15-4	22-4	29-4	6-5	13-5	20-5	27-5	3-6	10-6	
1	Phase 1: FMCW Radar Technology																												
1.1	Study Chirp																												
1.2	Study Range Resolution																												
1.3	Study Phase and Sensitivity																												
1.4	Study Velocity Detection																												
1.5	Study System Design Topics																												
1.6	Study Sound Noise Relation																												
MILESTONE #1: COMPLETELY UNDERSTAND HOW FMCW RADAR WORKS																													
2	Phase 2: ROS2 Iron																												
2.1	Understand Nodes and Topics																												
2.2	Create Node and Topic for Target Extraction																												
3	Phase 3: Protocol Specifications for SiRad Easy r4 Radar																												
3.1	Understand Measurement Flow																												
3.2	Understand Radar Configuration Registers																												
3.3	Configure Radar																												
MILESTONE #2: RADAR RUNNING AND OUTPUTTING MEASUREMENTS																													
4	Phase 4: Signal Processing Software																												
4.1	Parse Radar Output																												
4.2	Develop Range Resolution Software																												
4.3	Develop Phase Detection Software																												
4.5	Develop Velocity Detection Software																												
4.5	Determine Radar's Maximum Range																												
MILESTONE #3: SIGNAL PROCESSING SOFTWARE CAPABLE OF DETECTING RANGE, VELOCITY AND PHASE OF INCOMING OBJECTS																													
5	Phase 5: Document Project																												
5.1	Write Project Documentation																												
MILESTONE #4: PROJECT DOCUMENTATION FINISHED																													

Tabla 1. Planificación del proyecto

4.4.2. Estimación Económica

Debido a que este proyecto forma parte de un robot mucho más grande, en este apartado solo se han documentado los gastos necesarios para la realización de este proyecto, obviando los demás componentes del robot. Es importante destacar que la elección de estos componentes ha sido realizada por el director de este proyecto, el profesor Sebastian Scherer. Podemos desglosar los gastos económicos en tres grandes categorías en función de su finalidad.

1. Computacional. Los gastos computacionales son aquellos necesarios para la programación del entorno ROS2 Iron y la comunicación entre los diferentes componentes del robot. En este caso, se ha optado por utilizar el procesador OrangePi Zero de 2W. Se trata de un procesador de la familia Cortex-A53 con velocidad de reloj de hasta $1.5GHz$. Entre sus características, aparte de tener los clásicos componentes de un microcontrolador como la UART, un ADC o un módulo PWM, destaca su conectividad WiFi, muy importante para la transmisión de tópicos del entorno ROS2



Figura 25. OrangePi Zero. (<http://www.orangepi.org/html/hardWare/computerAndMicrocontrollers/details/Orange-Pi-Zero-2W.html>)

2. Alimentación. Se trata de aquellos productos utilizados para alimentar el sistema. En este caso se ha optado por utilizar una Pisugar 3 de 1200 mAh. La Pisugar 3 es una batería portátil hecha específicamente para la familia de Raspberry Pi. Cuenta con conexión USB-C para carga, interruptor de encendido y apagado, un reloj RTC (Real-Time-Counter) para programar encendidos y apagados y la opción de consultar el porcentaje de batería restante a través de una API o interfaz gráfica.



Figura 26. Pisugar 3 (<https://www.amazon.es/bater%C3%ADas-port%C3%A1tiles-Raspberry-Accessories-handhold/dp/B09QS12NIW>)

3. Radar. Para la realización de este proyecto se ha utilizado el Sirad Easy r4 de Indie Semiconductors, una placa de evaluación radar que soporta las frecuencias de $24GHz$ y $120GHz$. Permite diferentes configuraciones, las cuales se explicarán en detalle más adelante.



Figura 27. SiRad Easy r4. (<https://siliconradar.com/evalkits/>)

La siguiente tabla muestra en detalle el coste total del proyecto.

Name	Purpose	Quantity	Price	Total
OrangePi Zero 2W	Compute	1	\$ 52.99	\$ 52.99
Pisugar 3 1200 mAh	Power	1	\$ 33.99	\$ 33.99
Starter Kit Sirad Easy r4	Radar	1	\$ 779.00	\$ 779.00
TOTAL				\$ 865.98

Tabla 2. Costes del proyecto

Capítulo 5. Sistema/Modelo Desarrollado

5.1. *Análisis del Sistema*

El sistema desarrollado forma parte del Team Chiron²⁹, propuesta de la Universidad Carnegie Mellon para el DARPA³⁰ Triage Challenge. El DTC³¹ es una competición organizada por la Agencia de Proyectos de Investigación Avanzados de Defensa de los Estados Unidos de América y su objetivo es la innovación tecnológica en situaciones de emergencia sanitaria. Consiste en desarrollar sistemas que puedan evaluar y priorizar a los pacientes en función de la gravedad de sus lesiones o condiciones médicas [3].

La propuesta del Team Chiron es un robot cuadrúpedo equipado con un gran número de sensores cuya función es asistir y evaluar a pacientes en zonas de guerra, catástrofes naturales o donde el acceso humano sea limitado. Este robot, desarrollado por Boston Dynamics³², debe de ser capaz de moverse de manera autónoma y hacer rápidos diagnósticos de la situación individual de cada paciente. Cuenta con una batería de Litio con autonomía para 1h de actividad, peso de 40Kg, dimensiones (ancho x largo x alto) de 0,55 x 1.10 x 0.61 [m], velocidad de $2 \frac{m}{s}$, WiFi a bordo y comunicación robot-base, robot-robot integrada con una potencia de transmisión de 30dBm.

A continuación, se hará una breve descripción de los diferentes componentes del robot y de cómo están integrados. Destacar que durante este proyecto se ha trabajado únicamente y de manera exclusiva con el radar FMCW, en su programación para la detección de objetos y en la incorporación con el entorno ROS2 del robot.

²⁹ Equipo representante de la Universidad Carnegie Mellon para el DARPA Triage Challenge

³⁰ Del inglés: Defense Advanced Research Projects Agency

³¹ Del acrónimo en inglés: DARPA Triage Challenge

³² Empresa estadounidense de ingeniería y robótica que se especializa en la construcción de robots

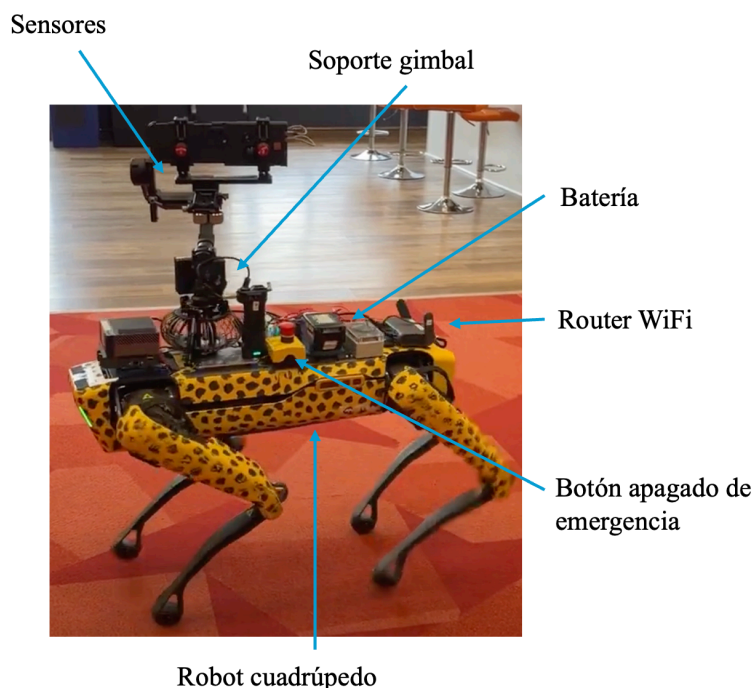


Figura 28. Esquema general robot Team Chiron

El coste total del robot asciende a más de 100.000 dólares y está formado por la base cuadrúpeda de Boston Dynamics y un grupo de sensores los cuales deben estar en constante comunicación con la CPU del robot:

1. Cámara RGB: Apple iPhone 15 Pro Max con resolución 1920x1080 y rango de visión horizontal de 110° con 0,5x de zoom y 12,5° con 5x de zoom. Su función se basa en el movimiento del robot para la localización de pacientes, guiado del soporte gimbal y dar apoyo al diagnóstico médico.
2. Micrófono: micrófono integrado del Apple iPhone 15 Pro Max.
3. Cámara infrarroja: modelo FLIR Boson+ con resolución 640x480 y 90° de visión horizontal. Su función es medir la temperatura corporal de los diferentes pacientes y dar apoyo para al diagnóstico médico.
4. Radar FMCW: Silicon Radar SiRad Easy r4 de Indie Semiconductors y objetivo de este proyecto. Su función se basa en medir la distancia, fase y velocidad de diferentes cuerpos en el rango de acción del robot y transmitirlo por la red.

5. LiDAR: Velodyne, Puck LITE con 16 líneas de escaneo y alcance máximo de 100 metros. Su función consiste en ayudar al movimiento del robot, haciendo un mapeado de la superficie.
6. Soporte Gimbal: soporte que permite que un objeto montado se incline libremente en cualquier dirección, manteniendo su estabilidad. El soporte gimbal utiliza algoritmos de inteligencia artificial para el seguimiento, dirigiendo al radar y el resto de sensores para que apunten a las respectivas zonas de interés.

A continuación, se muestra una imagen detallada de los sensores del robot.

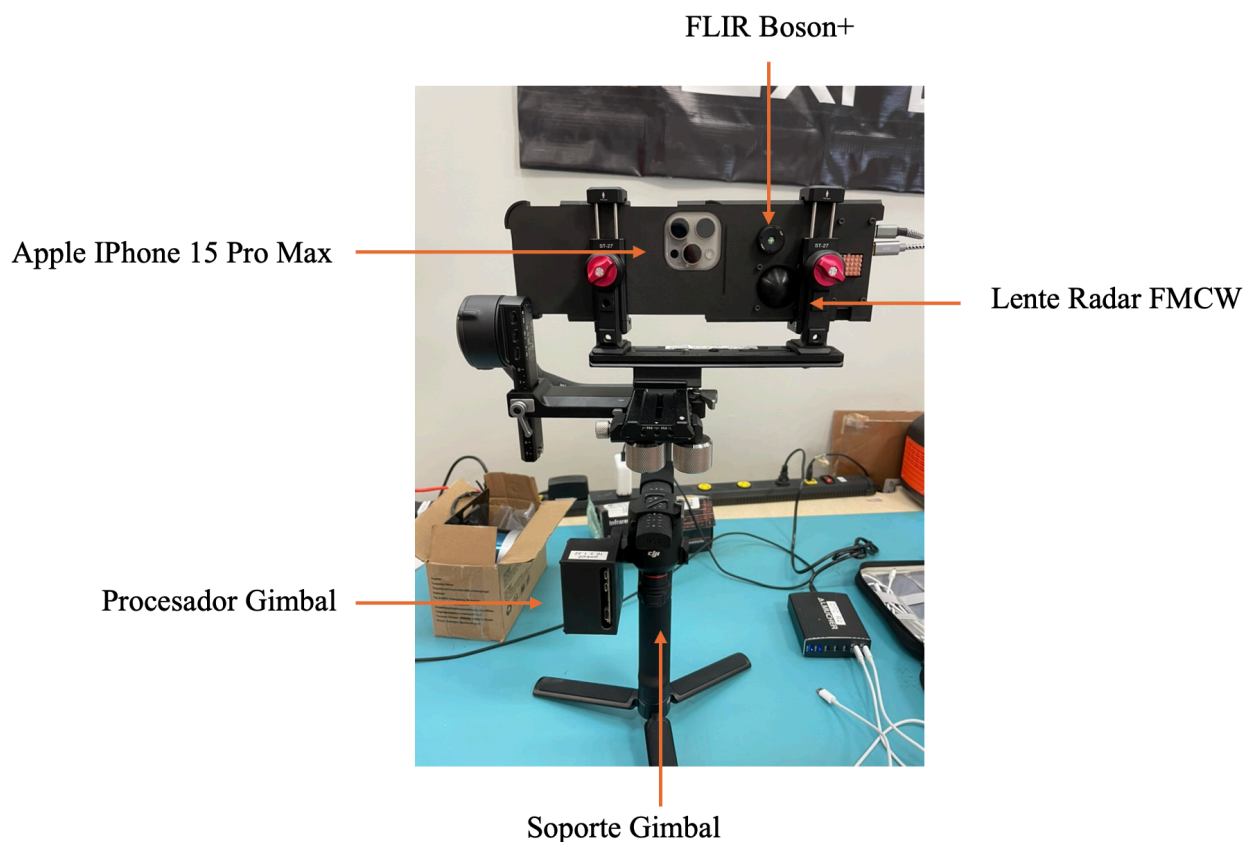


Figura 29. Sensores a bordo del Team Chiron

Como se puede observar en la figura de arriba, tanto el radar como la cámara térmica y el iPhone 15 se encuentran en el interior de una carcasa negra la cual ha sido impresa en una impresora 3D. Internamente, el radar se encuentra conectado a un microcontrolador que actuará como la CPU del

robot mediante una conexión UART de la siguiente manera. Esta conexión UART no forma parte de este proyecto y ha sido realizada por personas externas.

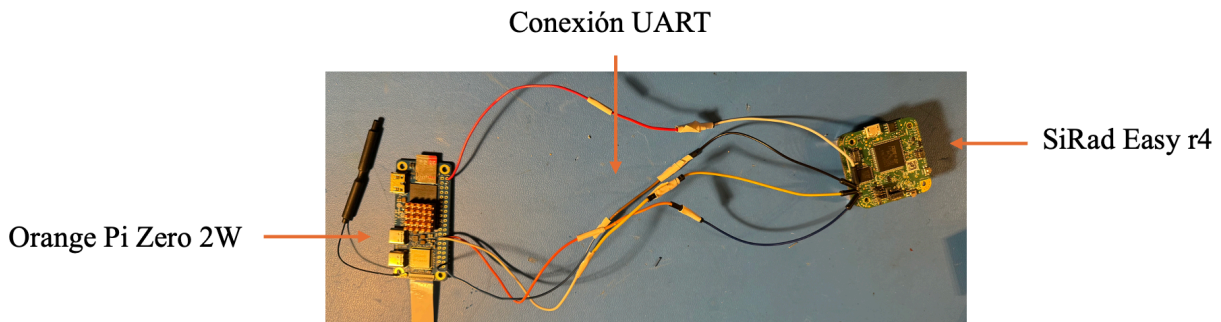


Figura 30. Conexión SiRad Easy r4 con OrangePi Zero

Además, el robot cuenta con cobertura WiFi para la publicación y suscripción de tópicos entre los sensores y el microcontrolador. Para ello, se ha utilizado el Slate AX (GL-AXT1800) de GL Inet, el cual dispone de 3 puertos ethernet, un puerto USB-A y un puerto adicional USB-C para alimentación. Este router el cual se encuentra montado sobre el robot y conectado a una batería externa, es el gateway de la red 10.3.2.0/24.

Para la configuración de la red se ha utilizado un servidor DHCP que asigna una IP estática a la CPU del robot (10.3.2.20) y una IP dinámica a cualquier otro equipo que se conecte.



Figura 31. Router a bordo del robot. (<https://www.gl-inet.com/products/gl-axt1800/>)

Para mejorar el campo de visión del radar, se le ha añadido una lente para direccionar la energía transmitida y aumentar el alcance. Sin embargo, la lente también reduce la visión angular del radar.



Figura 32. Radar montado con lente. (https://siliconradar.com/datasheets/User_Guide_Easy_Simple_V2.5.pdf)

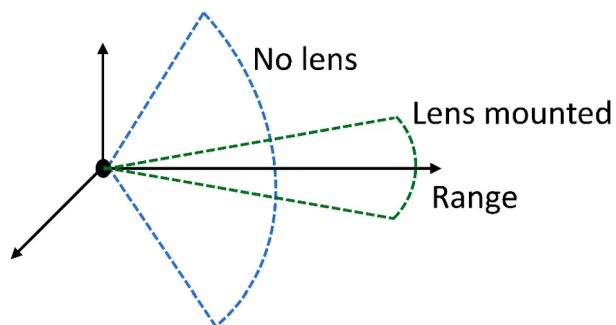


Figura 33. Efecto de añadir lente sobre el radar

5.2. Diseño

A lo largo de este apartado, se explicará el flujo de funcionamiento del radar SiRead Easy r4 y las diferentes configuraciones utilizadas.

5.2.1. Flujo de mediciones

El flujo de mediciones del radar sigue el esquema que aparece en la figura 34 y cuenta con varias fases:

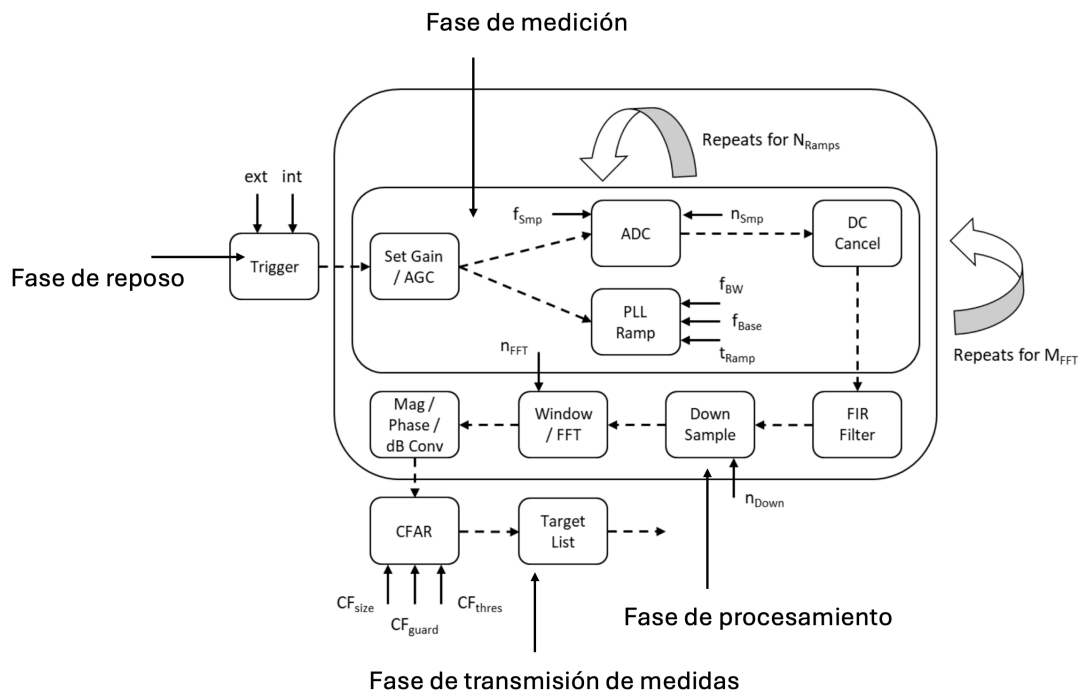


Figura 34. Flujo de mediciones SiRad Easy r4. (https://siliconradar.com/datasheets/Protocol_Description_Easy_r4_V1.2.pdf)

1. Fase de reposo: durante esta fase y con el objetivo de ahorrar energía, el radar estará en reposo hasta que reciba una interrupción. Esta interrupción puede ser de dos formas.
 - a. Interrupción interna: utilizando el reloj interno del radar, salta una interrupción cada vez que un contador interno llega a 0, posteriormente, se restaura el valor inicial del contador.

- b. Interrupción externa: en esta función, el radar estará en reposo hasta que reciba una interrupción externa. Una vez recibe la interrupción, se transmite por el puerto serie los datos obtenidos y vuelve al estado de reposo. Esta opción resulta atractiva en situaciones donde la frecuencia de medición no es necesario que sea alta, y permite ahorrar batería del radar.
2. Fase de medición: comienza una vez se recibe la interrupción. En ella, el PLL se inicializa y genera una rampa de frecuencia desde la frecuencia base hasta la frecuencia base + ancho de banda del radar para cada una de las rampas del grupo de rampas. Estas rampas son transmitidas individualmente y una vez se recibe su eco, pasan por el modulador IQ interno del radar para obtener la señal IF. Finalmente, se digitalizan a través el conversor analógico digital y se elimina la componente de continua. Este proceso es repetido el número de rampas que haya en cada grupo, el cual se puede modificar por software en los registros de configuración del radar, que se explicarán con detalle más adelante.

Si está activado el modo de Control de Auto Ganancia (AGC³³), antes de comenzar la medición, el front-end³⁴ del radar transmitirá dos rampas al comienzo de cada grupo de rampas con el objetivo de determinar la mejor ganancia posible. Si el modo de Control de Auto Ganancia está desactivado, la ganancia del radar vendrá dada por un registro de configuración y no se transmitirán estas dos rampas extras al inicio de cada grupo.

- -
 3. Fase de procesamiento de la señal: a esta fase le llega la señal digitalizada sin componente de continua, y engloba todo el procesamiento de señal que se realiza para la obtención de picos en frecuencia y diferencias en fase con los que determinar distancia y velocidad de los diferentes objetos localizados. Esta fase cuenta con varios bloques:

³³ Del inglés. Auto Gain Control

³⁴ Termino que se utiliza para referirse a toda la circuitería que forma parte de un equipo transmisor

- Filtro FIR: se utiliza un filtro con respuesta al impulso infinita (FIR) de varias etapas. En la primera de ellas, se trata de un filtro paso bajo que busca la eliminación de aquellas frecuencias que correspondan a objetos fuera del campo de interés. En el caso de este proyecto, se han eliminado frecuencias correspondientes a objetos a más de 4 metros de distancia, el rango de acción del robot. La segunda etapa consiste en un filtro cuyos coeficientes son el conjugado en orden inverso de una chirp de frecuencia base igual a cero y ancho de banda igual a la frecuencia de corte del filtro FIR previamente realizado. Este tipo de filtro se denomina filtro coincidente o matched filter. Consiste en la eliminación de ruido al convolucionar la señal por lo que se espera que sea. A la salida de esta segunda etapa de filtrado, los picos en frecuencia correspondientes a cada objeto son mucho más pronunciados. La figura 35 muestra el efecto de un matched filter en una señal con mucho ruido.

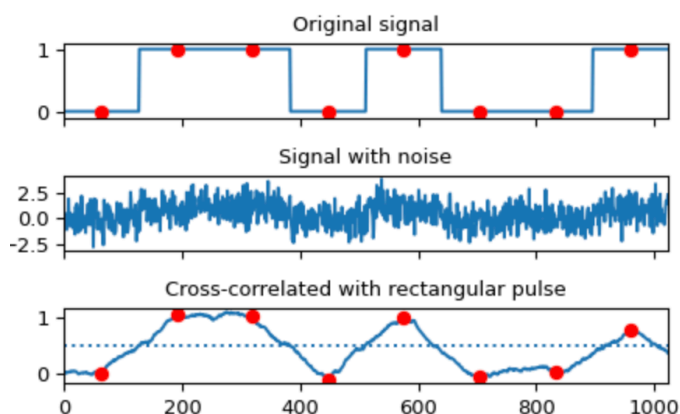


Figura 35. Efecto matched filter. (<https://docs.scipy.org/doc/scipy/reference/generated/scipy.signal.correlate.html>)

- Diezmado: este bloque consiste en diezmar por un factor D la señal a la salida de la etapa de filtrado. Debido a que, al diezmar, se ensancha el espectro un factor D, se aumenta la precisión para detectar los picos en frecuencia.
- Enventanado: consiste en aplicar técnicas de enventanado a la señal con el fin de reducir la fuga espectral que se produce al hacer la transformada de Fourier de una señal discreta, mejorando la representación y precisión del dominio espectral.

- Transformada de Fourier: una vez se ha filtrado y diezmado la señal con los parámetros apropiados para la detección de objetos, se procede a realizar la Transformada de Fourier de una señal discreta, para ello, se hará uso del algoritmo FFT (Fast Fourier Transform), y se extraerá la potencia en decibelios de cada pico en frecuencia. Como se ha demostrado previamente, cada pico en el espectro equivale a un objeto detectado, por lo que analizando la potencia recibida en cada frecuencia podemos aislar picos, y comparando la fase de cada pico, podemos obtener la velocidad.
4. Fase de detección de objetos: la cuarta y última fase consiste en el análisis de la señal en el dominio de la frecuencia para la obtención de los picos correspondientes a objetos. Para la detección de picos se ha usado el algoritmo `find_peaks` de la librería `scipy.signal` de Python. Este algoritmo define un pico como cualquier elemento de mayor amplitud que sus dos vecinos directos. La altura del pico viene dada por el número de vecinos directos de menor amplitud y la anchura por la diferencia en el eje x de los dos puntos limítrofes del pico. Los siguientes gráficos muestran el funcionamiento del algoritmo de detección de picos en función de unos parámetros clave: `distance`³⁵, `prominence`³⁶, `width`³⁷ y `threshold`³⁸.

³⁵ Del inglés. Significa la distancia a picos vecinos

³⁶ Del inglés. Significa la altura de un pico

³⁷ Del inglés. Significa la anchura de un pico

³⁸ Del inglés. Significa la altura de un pico respecto a sus vecinos

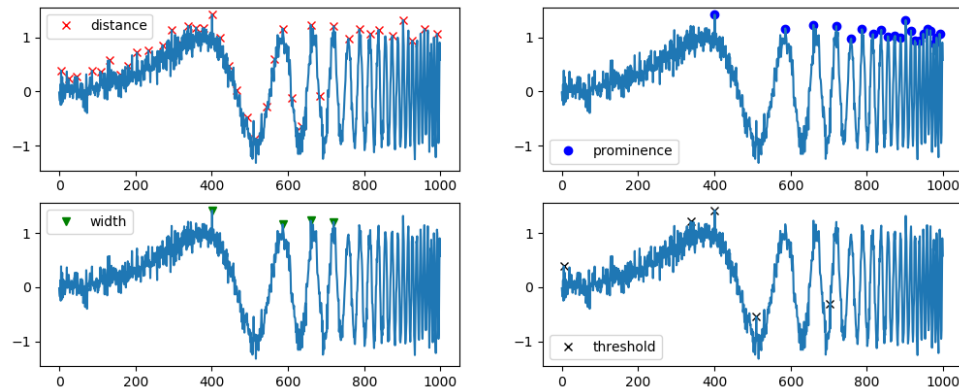


Figura 36. Parámetros de algoritmo `find_peaks` de `scipy`. (<https://stackoverflow.com/questions/1713335/peak-finding-algorithm-for-python-scipy>)

- Distance: únicamente selecciona aquellos picos con una distancia mínima entre ellos
- Prominence: únicamente selecciona picos con una cierta altura respecto a la base del pico
- Width: únicamente selecciona picos con una anchura mínima
- Threshold: únicamente selecciona picos con una cierta altura respecto a sus vecinos directos

Al combinar estos parámetros, extraemos con precisión los diferentes picos de nuestro sistema.

5.2.2. Configuración

El SiRead Easy r4 cuenta con varios registros de configuración que nos permite cambiar la configuración del sistema, configuración del PLL que genera la rampa de frecuencia y la configuración de los diferentes parámetros del procesamiento de la señal. Todos estos registros de configuración se acceden mediante comandos con una conexión UART que tiene los siguientes parámetros.

Campo	Valor
BaudRate	230400
Data bits	8

Number of start bits	1
Number of stop bits	1
Parity	None
Flow control	None

Tabla 3. Parámetros conexión UART

El radar utiliza un microcontrolador basado en un sistema de 32 bits, y los comandos son la palabra de 32 bits en hexadecimal, precedida por dos letras que indican el registro al que se quiere acceder. Siguen el siguiente esquema.

	Bit	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1		
SYS_CONFIG		SelfTrigDelay			reserved			LED		reserved								RAW		res	AGC	Gain		SER2	SER1	EXT	ST	TL	P	C	R	DC	res	SLF	PRE
Binary		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	1	1	0	1	1	1	0	1	0		

Figura 37. Ejemplo configuración [1]

	Bit	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1			
SYS_CONFIG		SelfTrigDelay				reserved				LED	reserved							RAW	res	AGC	Gain	SER2	SER1	EXT	ST	TL	P	C	R	DC	res	SLF	PRE			
HEX		0				0					0				0					4				9					B				A			

Figura 38. Ejemplo configuración [2]

En este ejemplo, el comando resultante sería: !S000049BA\r\n, donde el !S hacen referencia al registro de configuración global del radar y el \r\n a un salto de carro. Los comandos se transmiten al radar con la comunicación UART previamente descrita.

Existen 3 registros de configuración. El primero de ellos es el registro de configuración del sistema, el cual se encarga de la configuración de funciones básicas como puede ser el trigger elegido, el funcionamiento de los LEDs, el protocolo utilizado y la ganancia del sistema. La siguiente figura muestra una descripción más detallada de los posibles valores de configuración.

Bit	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
SYS_CONFIG	SelfTrigDelay	CL	LOG	FMT	LED	reserved				Protocol	AGC	Gain	SER2	SER1	ERR	ST	TL	C	R	P	CPL	RAW	res	res	SLF	PRE						
SelfTrigDelay				LED				AGC				SER2				SLF																
0	0	0	0 ms		0	0	off		0	auto gain control off			0	output on SER2 off			0	external trig mode														
0	0	1	2 ms		0	1	1st target		1	auto gain control on			1	output on SER2 on			1	standard mode														
0	1	0	4 ms		1	0	reserved		Protocol				SER1				PRE															
0	1	1	8 ms		1	1	reserved		0	0	WebGUI		0	output on SER1 off			0	standard mode														
1	0	0	16 ms		FMT				0	1	TSV output		1	output on SER1 on			1	use pre-trigger														
1	0	1	32 ms		0	TL mm distance			1	0	BIN output		x	x	reserved		Gain				data frames				off	on						
1	1	0	64 ms		1	TL cm distance											0	0	0	0	RAW	raw ADC	0	1								
1	1	1	128 ms														0	0	1	1	CPL	cmplx FFT	0	1								
LOG				CL												0	1	0	2	P	phase	0	1									
0	log MAG			0	DC Coupling											0	1	1	3	R	magnitude	0	1									
1	linear MAG			1	AC Coupling											0	1	0	4	C	CFAR	0	1									
																1	0	0	4	TL	target list	0	1									
																1	0	1	5	ST	status	0	1									
																				ERR	error	0	1									

Figura 39. Registro configuración del sistema

Para nuestro sistema, se ha elegido la siguiente configuración.

SelfTrigDelay			CL	LOG	FMT	LED		Reserved				Protocol		AGC	Gain			Ser2	Ser1	ERR	ST	TL	C	R	P	CPL	RAW	res	res	SLF	PRE
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	1	0

Figura 40. Valores registro configuración sistema

Campo	Valor	Significado
SelfTrigDelay	0	Retardo de 0 ms entre triggers internos
CL	0	Acoplamiento del amplificador en banda base en continua
LOG	0	Magnitud en dB
FMT	0	Distancia a objeto en mm
LED	1	El Led cambiará de color en función de la distancia hacia el primer objetivo
Protocol	2	Binary Output Mode es el protocolo seleccionado
AGC	1	El modo de Auto Ganancia Controlada está activo
Gain	0	Ganancia 0, es controlada por el AGC
Ser2	1	El output del radar es transmitido por el cable serie 2
Ser1	0	El puerto serie 1 del radar se encuentra apagado
ERR	0	Los datagramas de error no se transmiten

ST	0	Los datagramas de status no se transmiten
TL	0	Los datagramas de detección de objetivos no se transmiten
C	0	Los datagramas de CFAR no se transmiten
R	0	Los datagramas de magnitud de picos detectados no se transmiten
P	0	Los datagramas de fase de objetos detectados no se transmiten
CPL	0	Los datagramas de la transformada de Fourier compleja no se transmiten
RAW	1	Se transmiten los datagramas del conversor analógico digital
SLF	1	Modo de funcionamiento para el trigger es interno
PRE	0	No se utiliza un pre trigger. Modo funcionamiento es estándar

Tabla 4. Registro configuración sistema

El radar tiene un modo de funcionamiento en el que él mismo calcula automáticamente la distancia, fase, velocidad y magnitud de los diferentes cuerpos en el rango que se le indique, y transmite dicha información en el datagrama correspondiente. Estas tramas se indican en los campos ST, TL, C, R, P y CPL, pero al estar desactivados con nuestra configuración, únicamente recibiremos los valores que salen del conversor analógico digital.

Finalmente, el comando utilizado para la configuración global del sistema es el !S010A2012, donde el !S hacen referencia al registro de configuración del sistema y el resto es el valor hexadecimal de los 32 bits descritos con anterioridad.

El registro de configuración del PLL selecciona el ancho de banda del front-end del radar. El ancho de banda es configurado en saltos de $2MHz$. Valores negativos de ancho banda resultarán en una rampa con pendiente negativa mientras que valores positivos darán como resultado una rampa en frecuencia con pendiente positiva. A lo largo de este proyecto, se ha trabajado en el ancho de banda alocado a las bandas ISM, las cuales tienen fines industriales, científicos y médicos. Debido a esta limitación, el ancho de banda no puede exceder los $123GHz$, ya que estaríamos entrando en otra

banda de frecuencias, incumpliendo la normativa vigente [16]. Es por esto, por lo que el ancho de banda del radar se ha limitado a $1GHz_z$, aunque teóricamente podría haber sido mucho mayor.

	Bit	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
PLL_CONFIG		reserved																Bandwidth [MHz] (16 Bits)															
Bandwidth [MHz] (16 Bits)																																	
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	-2	MHz															
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	-4	MHz															
...																	...																
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-65536	MHz															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	MHz															
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	MHz															
...																	...																
0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	+65534	MHz															

Figura 41. Registro configuración PLL

Nuestra configuración seleccionada es la siguiente, con un ancho de banda de $1000MHz_z = 1GHz_z$. El comando de configuración es: !P000001F4, donde la !P indican el registro de configuración del PLL y el resto el valor de ancho de banda en MHz_z .

Reserved																Bandwidth [MHz]															
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	1	0

Figura 42. Valores registro configuración PLL

Por último, tenemos el registro de configuración del procesamiento en banda base del radar, en el cual se incluye toda la información relativa con el procesamiento digital de la señal y tiene la siguiente estructura.

	Bit	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1				
BB_CONFIG		WIN	FIR	DC	CFAR	CFAR Threshold [dB]				CFAR Size				CFAR Grd	Average n	FFT Size		Downsampling	# Ramps	# Samples		# Ramps		# Samples	ADC ClkDiv		MS/s										
WIN						FFT Size				Downsampling						# Ramps		# Samples				# Ramps		# Samples	ADC ClkDiv		MS/s										
0	windowing off				0	0	0	32	0	0	0	0	0	0	0	0	0	1	0	0	0	32	0	0	0	1.800											
1	windowing on				0	0	1	64	0	0	1	1	0	0	1	2	0	0	1	2	0	0	1	64	0	0	1	1.000									
					0	1	0	128	0	1	0	2	0	1	0	4	0	1	0	4	0	1	0	128	0	1	0	0.675									
					0	1	1	256	0	1	1	4	0	1	1	8	0	1	1	8	0	1	1	256	0	1	1	0.397									
0	FIR filter off				1	0	0	512	1	0	0	8	1	0	0	16	1	0	0	16	1	0	0	512	1	0	0	0.28125									
1	FIR filter on				1	0	1	1024	1	0	1	16	1	0	1	32	1	0	1	32	1	0	1	1024	1	0	1	0.218									
					1	1	0	2048	1	1	0	32	1	1	0	64	1	1	0	64	1	1	0	2048	1	1	0	0.173									
					1	1	1	reserved	1	1	1	64	1	1	1	128	1	1	1	128	1	1	1	reserved	1	1	1	0.055									
0	DC cancellation off																																				
1	DC cancellation on				CFAR				CFAR Threshold [dB]				dB	CFAR Size				CFAR Grd		Average n																	
					0	0	CA-CFAR		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
					0	1	GO-CFAR		0	0	0	1	2	0	0	0	1	1	0	0	1	1	0	1	1	0	1	1	0	1	1	0	1				
					1	0	SO-CFAR						...					...					1	0	2	1	0	2	1	0	2	1	0	2	1	0	2
					1	1	reserved		1	1	1	1	30	1	1	1	1	15	1	1	1	1	3	1	1	3	1	1	3	1	1	3	1	1	3		

Figura 43. Registro configuración banda base

Para nuestro sistema, se ha elegido la siguiente configuración:

WIN		FIR		DC	CFAR		CFAR Threshold (dB)				CFAR Size				CFAR Grd		Average n		FFT Size			Downsampling			# Ramps			# Samples			ADC ClkDiv		
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
1	0	1	0	0	1	0	0	0	1	0	1	0	0	1	0	1	1	0	0	0	0	0	1	0	0	1	0	0	1	0	1		

Figura 44. Valores registro configuración banda base

Campo	Valor	Significado
WIN	1	Enventanado de la señal activado
FIR	0	Filtro FIR activado
DC	1	Cancelación de componente de continua activada
CFAR	0	Modo CFAR es CA-CFAR
CFAR	8	Intervalo de confianza del CFAR es 8dB
CFAR Size	10	Tamaño de ventana CFAR es 10
CFAR Grd	1	Valor de CFAR Grd es 1
Average n	1	No se realiza la media de rampas
FFT Size	4	El tamaño de la transformada de Fourier es 512 muestras
Downsampling	0	Factor de Diezmado es 0
# Ramps	4	Cada grupo de rampas está formado por 16 rampas
# Samples	4	El número de muestras que toma el ADC es 512
Adc ClkDiv	5	Frecuencia de muestreo del ADC es 218KHz

Tabla 5. Registro configuración banda base

El periodo de la rampa o tiempo de rampa se puede calcular mediante la siguiente expresión, donde t_{smp} es el periodo de muestro y se mide en ciclos de reloj. n_{smp} es el número de muestras, 55 es un encabezado fijo que tienen todas las tramas y $27MHz$ la frecuencia de muestreo del conversor analógico digital.

$$T_c = \frac{t_{smp} \cdot (n_{smp} + 55)}{27MHz}$$

ADC ClkDiv	ADC periodo muestreo [ciclos reloj]	Frecuencia muestreo [MHz]
0	15	1,8

1	27	1
2	40	0,675
3	68	0,397
4	96	0,28125
5	124	0,218
6	156	0,173
7	492	0,055

Tabla 6. Valores frecuencia muestreo

En nuestro caso y con los valores seleccionados en los registros de configuración:

$$T_c = \frac{124 \cdot (512 + 55)}{27MHz} = 2,604 [m_s]$$

Como se ha explicado en el apartado de configuración del sistema, el radar únicamente transmitirá las tramas que contengan los datos que salen del conversor analógico digital, por lo que todos los parámetros de procesamiento de la señal como DC, CFAR, Average n, FIR o WIN no tienen ningún efecto. Volviendo al diagrama original del flujo de mediciones, la información con la que se ha trabajado es la que se indica como Raw Data.

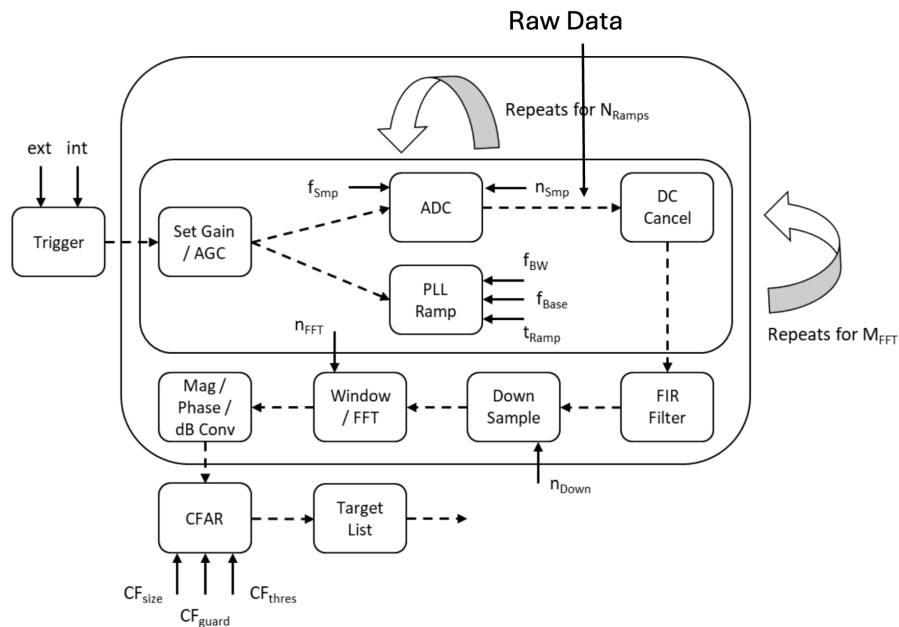


Figura 45. Raw data en flujo de mediciones

5.3. Implementación

La implementación de este proyecto se puede dividir en tres grandes bloques, en primer lugar, el entorno ROS2 sobre el que se ha trabajado y su estructura de tópicos y nodos. En segundo lugar, la decodificación de los frames recibidos por el radar y por último, el procesamiento digital de señal para la detección de distancia y fase de los diferentes cuerpos en el rango de acción del radar.

5.3.1. Entorno ROS2

Debido a que el radar forma parte de una red de sensores montados sobre el robot, resulta de vital importancia su integración en la red y la capacidad para comunicarse con la CPU y el resto de los componentes. Para ello, se ha optado por utilizar el entorno ROS2 Iron³⁹. ROS se basa en la comunicación entre nodos mediante tópicos que son transmitidos por la red [17]. El flujo de funcionamiento es el siguiente: cada sensor del robot está representado por uno o más nodos y

³⁹ Versión de ROS2 utilizada

cada nodo realiza una tarea específica. Estos nodos pueden publicar información en la red a través de tópicos, y recibir información suscribiéndose a otros tópicos, facilitando así la comunicación entre componentes. Todo sistema que esté conectado a la red del robot y con la versión correspondiente de ROS instalada, será capaz de suscribirse a cualquier tópico.

El siguiente diagrama muestra la comunicación entre tópicos y nodos que se ha utilizado para la implementación del radar. Los tópicos aparecen en verde mientras que los nodos en rojo.

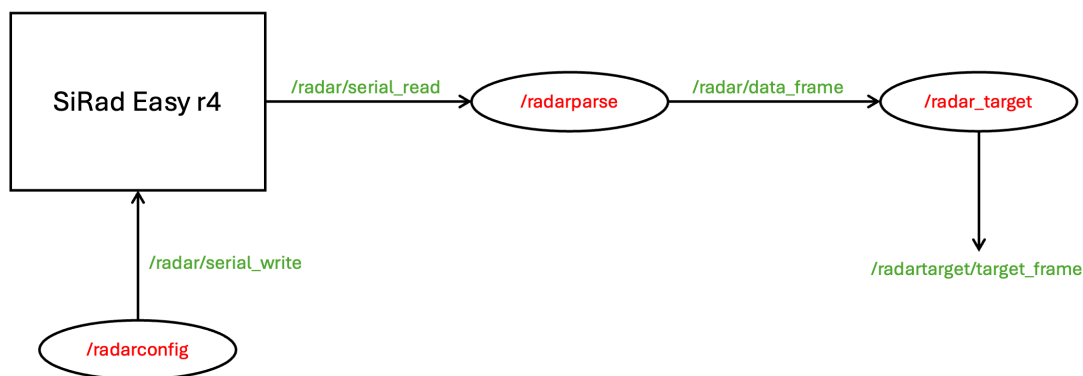


Figura 46. Diagrama entorno ROS2 para el SiRad Easy r4

Como se puede observar, el radar funciona bajo tres nodos, los cuales publican y suscriben tópicos.

1. Nodo `/radardconfig`: se encarga de la configuración del radar mediante los comandos previamente descritos. Este nodo, codifica los comandos para su transmisión por la UART y lo publica en la red mediante el tópico `/radar/serial_write`. Posteriormente, el radar recibe los comandos y modifica los registros de configuración.
2. Nodo `/radarparse`: se encarga de decodificar los datagramas recibidos por el radar y está suscrito al tópico `/radar/serial_read`, en el cual se transmite los datos del conversor analógico digital. Este nodo decodifica las tramas del ADC y lo publica en un tópico denominado `/radar/data_frame` utilizando la siguiente estructura.

```
---  
frame_counter: 3654  
frame_identifier: 68  
frame_length: 516  
tx_id: 1  
rx_id: 1  
data_source: 2  
gain: 16  
measurement_cnt: 3226  
slowtime_cnt: 0  
update_rate: 4700  
data_type: 1  
variable_type: 1  
no_elements: 1024  
data:  
- 0  
- 0  
- 0
```

Figura 47. Formato mensajes tópico radar/data_frame

3. Nodo /radar_target: este es el nodo más complejo de todos y engloba la funcionalidad del procesamiento digital de la señal y la detección de distancia, fase y velocidad. Está suscrito al tópico /radar/data_frame para recibir los datagramas decodificados y utiliza estos mensajes para reconstruir la señal, aplicar filtros para mejorar la calidad señal ruido y finalmente hacer la transformada de Fourier para la detección de picos y fase. A su vez, una vez identificados los objetos, publicará en la red el número total de objetos, la distancia a cada uno de ellos, la magnitud de potencia recibida, la fase del objeto, y la velocidad a la que se mueve a través del tópico /radartarget/target_frame.

```
---  
frame_counter: 26822  
target_number: 1  
distance:  
- 0.0  
magnitude:  
- 39.805580139160156  
phase:  
- 180.0  
velocity:  
- 0.0  
---
```

Figura 48. Formato mensajes tópico /radartarget/target_frame.

A continuación, se explica el contenido, codificación y los posibles valores que pueden tomar cada campo, esencial para su posterior decodificación.

Campo	Tamaño	Contenido	Codificación	Posibles valores
Header	4 bytes	Indica comienzo de la trama	Fijo	AAAABBCC
Frame counter	2 bytes	Contador de trama	Unsigned Int	0 - 65535
Frame Id	2 bytes	Identificador de trama	Fijo	Tabla 8
Frame Length	2 bytes	Longitud total de la trama	Unsigned Int	0 - 65535
Tx Id	1 byte	Id del transmisor	Unsigned Int	1 – 2
Rx Id	1 byte	Id del receptor	Unsigned Int	1 – 4
Data source	1 byte	Fuente de datos	Unsigned Int	Tabla 9
Gain	1 byte	Ganancia del sistema	Unsigned Int	0 – 256
Measurement Cnt	2 bytes	Contador de medidas	Unsigned Int	0 - 65535
Slowtime Cnt	2 bytes	Contador slowtime	0	0
Update Rate	2 bytes	Ratio de transmisión de datagramas	Unsigned Int	0 - 65535
Data type	1 byte	Tipo de dato en Data Payload	Unsigned Int	0 - 65535
Variable Type	1 byte	Formato de datos muestreados	Unsigned Int	0 - 65535
No. Elements	2 bytes	Número de muestras en Data Payload	Unsigned Int	0 - 65535
Data Payload	N bytes	Datos del radar	Variable	
CRC-32	4 bytes	Checksum del datagrama	Unsigned Int	0 - 4294967295
Stop Mark	2 bytes	Stop del datagrama	Unsigned Int	\r\n

Tabla 7. Trama del radar

La información el sensor viene dada en formato I/Q, por lo que para identificar cada rama, usaremos el campo Data Source, con sus posibles valores descritos en la tabla 8. Como únicamente nos interesa las ramas I y Q, filtraremos por los valores 1 y 2, omitiendo el resto.

Data Source	Valor
I Channel	1
Q Channel	2
Interleaved Channel	3
Summed Channel	4

Tabla 8. Identificador de trama

El identificador de trama viene dado por la tabla 8. Durante este proyecto y debido a la configuración del sistema utilizada, solo se recibirán datagramas con información del sensor, por lo que en nuestro caso, el único valor posible del campo Frame Identifier es una D. Este también será otro parámetro por el que filtrar.

Identificador de trama	Valor	Contenido
Data Frame	D	Información del sensor
Error Frame	E	Información de error
Info Frame	I	Información hardware y firmware
Parameter Frame	P	Parámetros del radar

Tabla 9. Fuente de datos

Por último, la siguiente gráfica describe la estructura del campo Data Payload, el más importante y el que contiene la información del radar. Una vez más, debido a la configuración seleccionada por el registro de configuración global del sistema, solo recibiremos información proveniente del conversor analógico digital, la llamada raw data⁴¹. Como se puede observar, la información del sensor viene dada en números de 16 bits con signo.

Formato de campo	Tamaño de	Contenido	Codificación	Posibles valores
Data Payload	No. Samples	Magnitud/Alcance/CFAR	Signed integer	-140 a 0 (dB)
	No. Samples	Fase	Signed integer	$10000 * (-\pi \text{ a } \pi)$ rad
	No. Samples	Raw Data	Unsigned integer	-32768 a +32767

⁴¹ Tipo de dato que sale del conversor analógico digital, sin ningún procesamiento previo

Tabla 10. Campo de datos

Las tramas se decodifican byte a byte, siguiendo el orden establecido en la figura 49. En primer lugar, el programa espera hasta recibir la trama por completo y una vez recibida empieza a decodificarla. Primero comprueba que la cabecera sea la correcta y si lo es, pasa a los campos de contador e identificador de trama. Debido a la configuración seleccionada, solo recibiremos tramas con el identificador “D”.

Una vez tenemos localizado el tipo de trama a decodificar resulta muy sencillo continuar. Vamos leyendo byte a byte la información y formando el mensaje ROS a transmitir por la red. Si algún campo tiene más de un byte, se agrupan siguiendo el formato del protocolo, en este caso, Little endian⁴² por lo que el primer byte leído será el menos significativo y así sucesivamente. Con cada trama decodificada, se publica el siguiente mensaje en la red haciendo uso del tópico radar/data_frame. Estos mensajes tienen la siguiente estructura.

```
#straps_msgs

# Parsed Binary data frames from radar

uint16 frame_counter
uint16 frame_identifier
uint16 frame_length
uint8 tx_id
uint8 rx_id
uint8 data_source
uint8 gain
uint16 measurement_cnt
uint16 slowtime_cnt
uint16 update_rate
uint8 data_type
uint8 variable_type
uint16 no_elements
uint8[] data
```

⁴² Modo de agrupar posiciones de memoria, comenzando por las más bajas

5.3.3. Software para la detección de distancia, fase y velocidad

Como se ha explicado anteriormente, el software de procesamiento digital de la señal se ha realizado en un nodo aparte siguiendo el siguiente flujo de funcionamiento descrito en la figura 50.

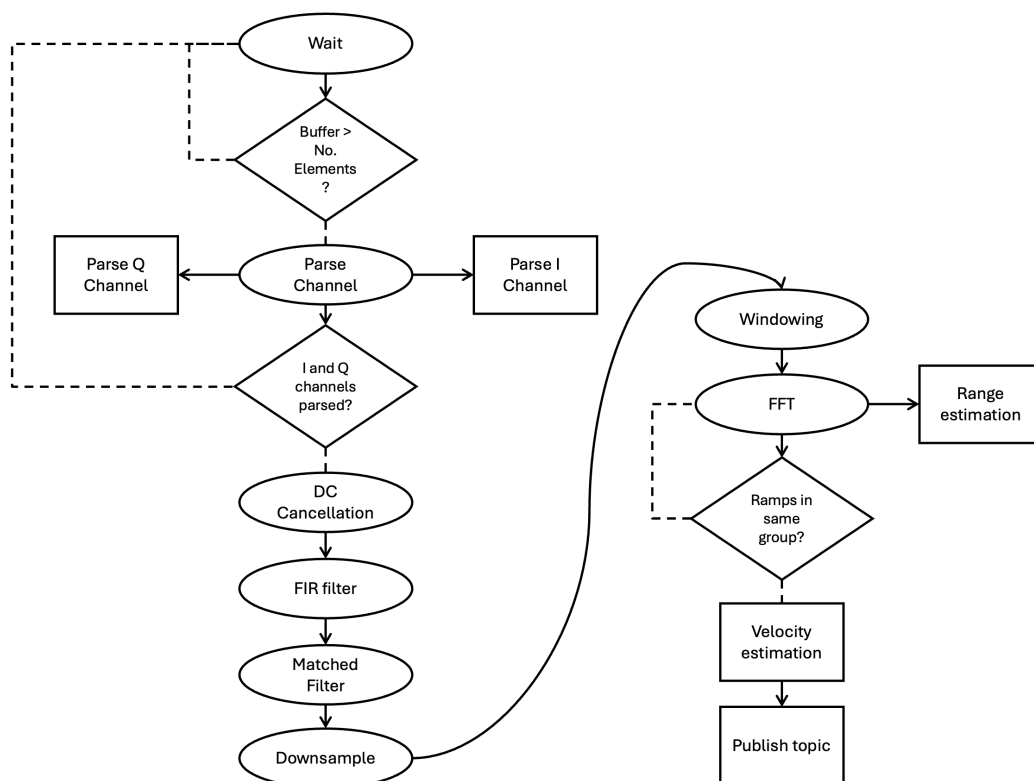


Figura 50. Flujo procesamiento nodo /radar_target

En primer lugar, el nodo se suscribe al tópico radar/data_frame, y escucha todos los mensajes publicados. De estos mensajes, solo nos interesan los siguientes campos:

- data: contiene la información del sensor, es el campo correspondiente a Data Payload de la trama original y el que contiene la forma de onda de la señal de frecuencia intermedia.
- data_source: contiene la información del canal de la trama actual. Nos interesa para poder diferenciar entre canal I y canal Q y saber cuándo una medida está completa
- no_elements: contiene la longitud total del campo data

- `frame_counter`: contador que se incrementa con cada trama transmitida. Resulta muy útil para ayudarnos a saber cuándo una medida está completa, ya que cada medida completa consta de dos tramas
- `update_rate`: contiene la frecuencia de envío de datagramas. Se utiliza para la representación de gráficos

En primer lugar, el programa se encuentra esperando y guardando en un buffer interno la información del campo data. Una vez la longitud de este buffer sea mayor o igual al `no_elements`, sabremos que ha llegado la trama por completo y podemos empezar a procesarla.

Para ello, primero comprobamos si se trata del canal I o el canal Q utilizando el campo `data_source`, y decodificamos el canal correspondiente, guardándolo en un array. El campo data es un array de bytes, pero la información del sensor viene dada en signed shorts, por lo que para decodificarlo se van leyendo los bytes de dos en dos, agrupándolos en un short y usando el formato Little-endian del protocolo para obtener el valor real. Esto se ha realizado a través de la librería struct de Python.

A continuación, comprobamos si la medida actual es un factor de dos y si los arrays de la rama Q y la rama I están completos. En caso de que se cumplan todas las condiciones, sabremos que se ha decodificado completamente las dos ramas. Para evitar errores, y como la rama Q siempre se transmite inmediatamente después de la rama I, antes de decodificar la rama I, se pone a cero la rama Q. De esta manera, se evita que el anterior condicional se cumpla utilizando los valores de la rama Q y la rama I de mediciones diferentes.

Una vez tenemos las dos ramas decodificadas correctamente, da lugar al procesamiento digital de la señal. Como se ha explicado en el capítulo 2.3.2, la rama Q y la rama I son idénticas pero con un desfase de $\frac{\pi}{2}$. Debido a que nuestro objetivo es ver picos en frecuencia y diferencias en fase, nos da igual trabajar con una que con la otra, en este caso, trabajaremos con la rama I.

El procesamiento de la señal comienza con la cancelación de la componente de continua de la señal, lo cual resulta muy sencillo de hacer simplemente restando la media. Posteriormente comienza las dos etapas de filtrado, para ello se debe seleccionar una distancia máxima para el campo de acción del radar. Debido a que el radar va montado sobre un robot, el campo de acción del radar se ve limitado por los demás sensores del robot (de que nos sirve poder detectar objetos a 20 metros si luego la cámara infrarrojos o el LiDar no ve más allá de 2). Es por esto, por lo que la distancia máxima a la que detectar objetos se ha fijado a 3-4 metros aproximadamente, lo que equivale a una frecuencia de corte 10KHz . El filtro se ha realizado utilizando la herramienta `fdatool`⁴³ de Matlab, utilizando el método de Constrained Equiripple⁴⁴, con 100 puntos, frecuencia de muestreo 218KHz , frecuencia de corte 10KHz y ganancia 0dB . Los coeficientes del filtro se exportan para utilizarse en Python bajo el nodo `radar_target`.

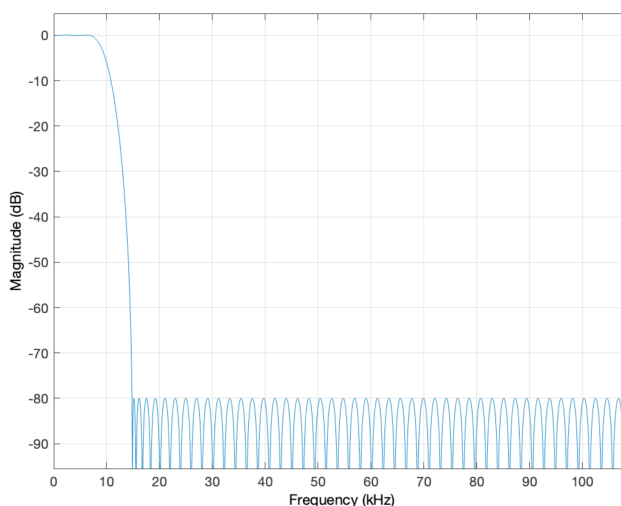


Figura 51. Respuesta en frecuencia LPF 10KHz

Después de realizar el filtrado por el filtro paso bajo, se realiza el matched filter⁴⁵, donde los coeficientes del filtro son el conjugado en orden inverso de un chirp con frecuencia base igual a cero y ancho de banda igual al campo de acción del radar, en este caso, 10KHz .

⁴³ Programa Matlab para el diseño de filtros

⁴⁴ Método para diseño de filtros que se utiliza para minimizar la desviación máxima de la banda de paso y de corte

⁴⁵ Del inglés. Significa filtro coincidente

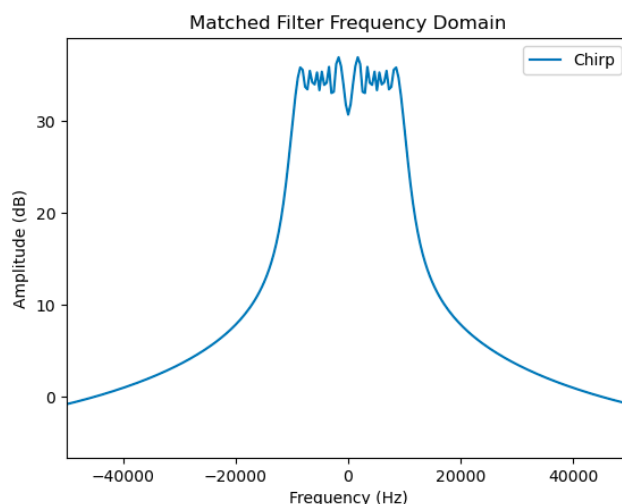


Figura 52. Respuesta en frecuencia matched filter

A continuación, podemos diezmar la señal para aumentar la precisión en la detección de picos, en este caso, se ha optado por dejar la señal como está ya que los resultados han demostrado ser lo suficientemente precisos. Por último, se aplica un enventanado⁴⁶ con el objetivo de minimizar las fuentes de ruido. Se ha optado por utilizar una ventana Blackman, la cual se describe bajo la siguiente ecuación.

$$w(n) = 0.42 - 0.5\cos\left(\frac{2\pi n}{L-1}\right) + 0.08\cos\left(\frac{4\pi n}{L-1}\right)$$

⁴⁶ Técnica que se realiza en el procesamiento digital de señales con el fin de reducir la fuga espectral y el nivel de ruido

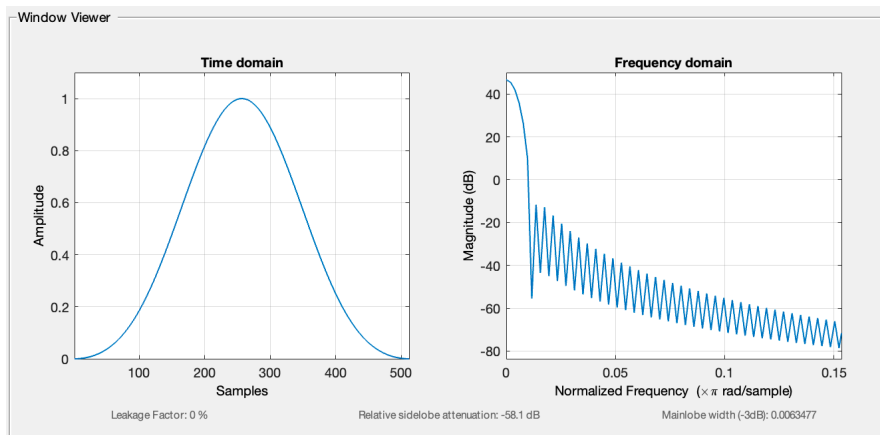


Figura 53. Ventana blackman

Una vez llegado a este punto, se tiene una representación precisa de la señal de frecuencia intermedia y se puede proceder a la detección de picos. Para ello, se ha usado el algoritmo `find_peaks` de `scipy` descrito previamente. Los parámetros por los que se ha filtrado han sido `prominence=10`, por lo que únicamente detectaremos los picos que tengan un alto (desde la base del pico) de 10dB. La configuración de estos parámetros varía en función de la precisión y lo riguroso que se quiere ser. Se puede conseguir más rigurosidad, eliminando fuentes de ruido y probabilidad de error al aumentar la altura del pico o añadiendo una anchura de pico mínima. Por el contrario, si el objetivo es la detección de cualquier movimiento, se puede rebajar los filtros para que sean de más fácil cumplimiento y así detectar más objetos, aumentando también la probabilidad de error. Como muchos problemas en la ingeniería, no hay una solución perfecta.

El ejemplo de abajo muestra la detección de un objeto a muy corta distancia. Observamos, que hay unos glóbulos en torno a la frecuencia de 10KHz y que tal y como está el algoritmo de detección de picos configurado no se han detectado. Posiblemente, estos glóbulos no tienen mayor potencia debido a que la señal rebota de refilón o parcialmente en el cuerpo, por lo que esa frecuencia no es amplificada tanto en comparación con la principal.

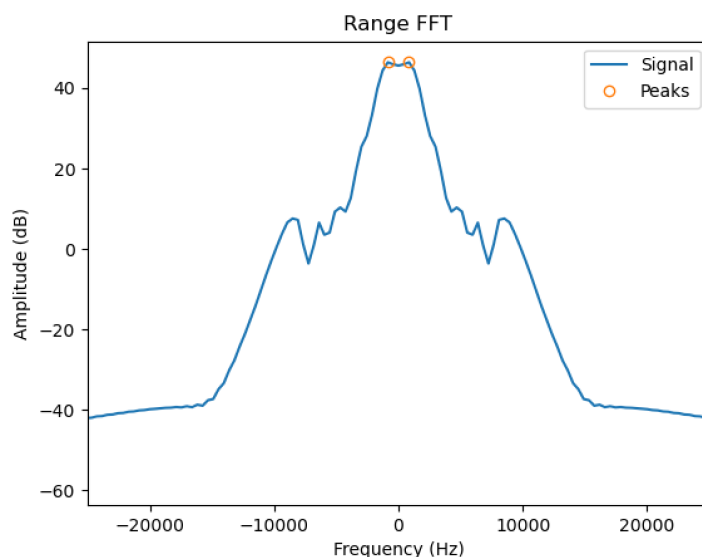


Figura 54. Resultado detección de objetos [1]

Por último y siguiendo el diagrama de flujo planteado al principio de esta sección, da lugar la detección de velocidad. Para ello, se calcula las diferencias de fase entre señales sucesivas. Para que la estimación de velocidad sea precisa, la diferencia de tiempos entre cada chirp transmitido debe de serlo. Como se puede observar en la figura 55, proporcionada por el fabricante, la diferencia temporal entre cada rampa transmitida es prácticamente nula, pero sí que existe una diferencia considerable entre rampas de diferentes grupos. Para solucionar este problema, se ha optado por tomar como referencia las diferencias de fase de únicamente rampas pertenecientes al mismo grupo.

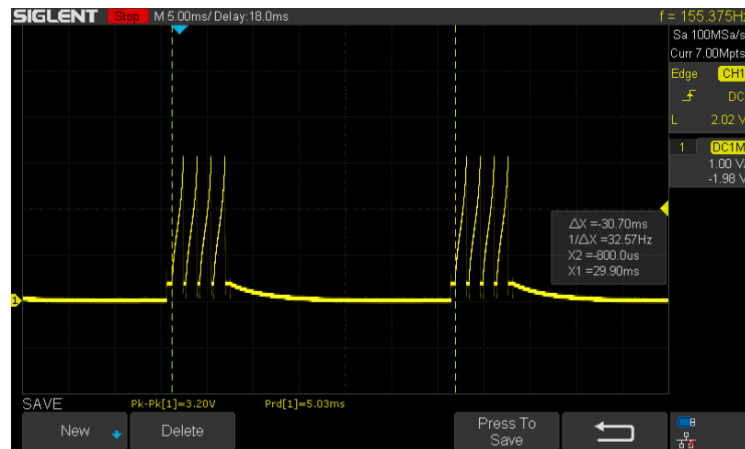


Figura 55. Tiempo entre grupo de rampas

Además, si hacemos zoom en la imagen superior, observamos que el tiempo entre cada rampa del mismo grupo no es exactamente el tiempo de rampa, sino que hay un retardo extra. Como se puede observar, esta diferencia temporal es de aproximadamente 260 μ s, y también se ha tenido en cuenta para el cálculo de la velocidad.

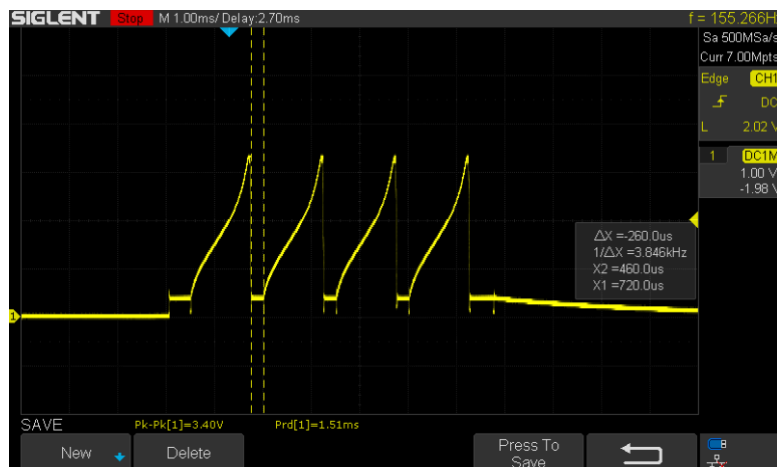


Figura 56. Tiempo entre rampas mismo grupo

Sin embargo, el protocolo nos explica que este retardo temporal se debe al tiempo que necesita el procesador para generar la nueva rampa, calcular la auto ganancia y la transmisión de datos y no es una cantidad fija. Es por esto por lo que las mediciones de velocidad tienen un margen de error aproximadamente del 10%. Podemos reducir el margen de error aumentando el tiempo de rampa,

de tal manera que este retardo tenga menor impacto sobre la diferencia total entre una rampa y la siguiente.

Una vez se tienen las medidas de distancia y velocidad, se transmiten por la red utilizando el tópico radar/target_frame. La estructura de los mensajes transmitidos es la siguiente.

```
#straps_msgs

# Radar Target information

uint16 frame_counter
uint8 target_number
float32[] distance # in m
float32[] magnitude # in dB
float32[] phase # in radians
float32[] velocity # in m/s
```

```
---
frame_counter: 25302
target_number: 1
distance:
- 0.3326203227043152
magnitude:
- 46.418495178222656
phase:
- 48.48763656616211
velocity:
- -3.9380783078968307e-08
---
```

Figura 57. Tópico /radar/target_frame

Capítulo 6. Análisis de Resultados

Para el análisis de resultados se han realizado diferentes experimentos para la detección de distancia, fase y velocidad de diferentes objetos en el campo de visión del radar. Estos experimentos se han realizado con el radar aislado del resto de componentes del robot, puesto que algunas de ellas no estaban todavía completamente implementadas.

En primer lugar, se han colocado múltiples objetos a diferentes distancias en el campo de visión del radar y se ha medido la precisión obtenida en la distancia detectada. Los objetos estaban colocados siguiendo la tabla descrita

Id	Distancia real (m)	Distancia obtenida (m)	Error (cm)
1	1.20	1.33	13
2	3.00	2.99	1
3	4	3.82	18

Tabla 11. Análisis de resultados

Como se puede apreciar, existen diferencias de errores significativas. Estas diferencias las podemos explicar a través de la representación en frecuencia de la señal. Como se puede observar en la figura 58, el pico más pronunciado y con mayor altura y anchura es el correspondiente al glóbulo del objeto número 2, lo que explica que sea el objeto con menor error de los tres. Tanto los objetos número 1 como número 3 presentan errores significativos en comparación con el objeto número 2. Sin embargo, estos errores tienen diferente fondo. En primer lugar, el objeto número 1 se encuentre en un rango de alcance óptimo para la detección, pues está más cerca que el objeto 2, pero no presenta un glóbulo tan bien definido. Esto se debe a que, aunque la distancia al radar sea óptima, no lo es su colocación, pues se encuentra en el límite del campo de visión angular del radar. Por el contrario, el objeto número tres se encuentra más alejado y por ello su glóbulo tiene menor amplitud. Al estar colocado dentro del campo de visión del radar, este glóbulo sí que tiene una anchura correcta y está bien definido. Como se ha mencionado en capítulos anteriores, al

añadir la lente al transmisor del radar, se aumenta el alcance máximo, pero se reduce el campo de visión angular.

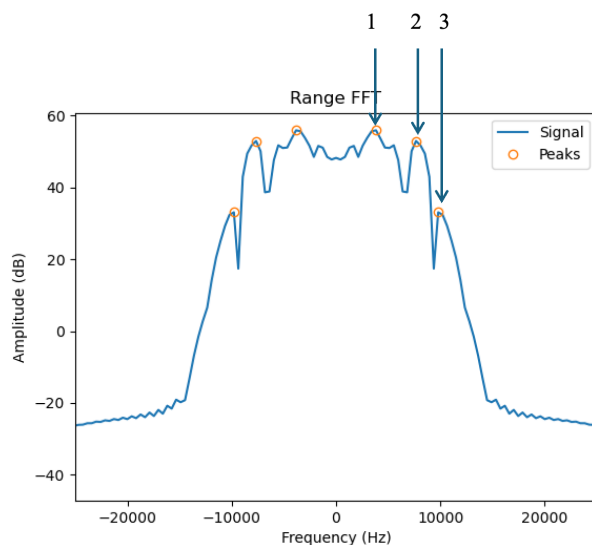


Figura 58. Resultado detección de objetos [2]

A continuación, se muestra la información transmitida bajo el tópico radar/target_frame correspondiente. Esta información viene dada en listas ordenadas desde el elemento más próximo al más lejano. La información correspondiente al primer objeto se encuentra en la posición número 1, la del segundo objeto en el número dos y así sucesivamente. Observamos que tanto el objeto número 1 como el número 2 tienen una amplitud muy parecida, lo que se debe a su proximidad al radar. En cuanto a las velocidades, ambos tienen velocidades muy parecidas y se podrían considerar estáticos.

```
frame_counter: 17674
target_number: 3
distance:
- 1.3304812908172607
- 2.9935827255249023
- 3.8251335620880127
magnitude:
- 55.425010681152344
- 53.212158203125
- 33.078556060791016
phase:
- -152.9485626220703
- 8.082763671875
- 62.1777458190918
velocity:
- -0.008188014850020409
- -0.0035444360692054033
- -0.0010352523531764746
---
```

Figura 59. Resultado detección de objetos [3]

Como hemos visto en el anterior experimento, la precisión en la detección de distancia, fase y velocidad se ve limitada por numerosos eventos, entre las que destacan las siguientes. En primer lugar, por la lente colocada en el radar, la cual aumenta la distancia máxima, pero reduce el campo de visión angular. Sin embargo, debido a que el radar va montado sobre una estructura gimbal la cual haciendo uso de algoritmos de inteligencia artificial se encarga de seguir a los seres humanos y dirigir el radar para que apunte hacia el pecho, esta limitación no resulta un problema para el proyecto planteado ya que ninguna detección quedará limitada por la reducción de campo de visión producido por añadir la lente.

Otras de las limitaciones presentes se dan cuando dos objetos están tan próximos que no hay resolución suficiente para resolver los dos objetos, o cuando están al límite de poder diferenciarlos, zona llamada Criterio de Rayleigh [18].

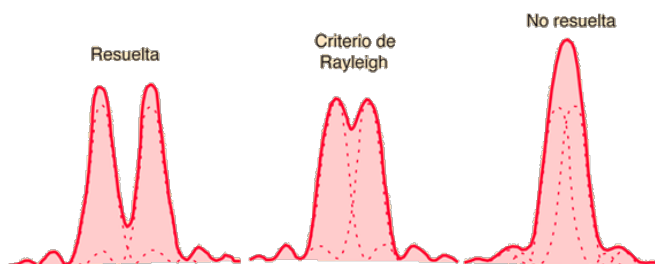


Figura 60. Criterio de Rayleigh. (<http://hyperphysics.phy-astr.gsu.edu/hbasees/phyopt/Raylei.html>)

Como se ha visto en el capítulo 2.3.4, la resolución de un radar viene dada por la siguiente ecuación, la cual depende del ancho de banda. Para mayor resolución, mayor ancho de banda. Se entiende por resolución la separación mínima que debe haber entre dos objetos para poder resolverlos en el dominio de la frecuencia.

$$\Delta f = \frac{1}{T_c} \rightarrow \frac{S2\Delta R}{c} > \frac{1}{T_c} \rightarrow \Delta R > \frac{c}{2ST_c} = \frac{c}{2B}$$

Sin embargo, esta es una resolución teórica, ya que en la práctica es mucho menor y depende en gran medida del algoritmo de detección de picos empleado. Podemos mejorar la resolución diezmando la señal, para expandir el dominio de la frecuencia y que resulte más fácil resolver picos bajo el Criterio de Rayleigh.

El problema de la resolución de objetos o del Criterio de Rayleigh se ve minimizado por el uso de la estructura gimbal que guía al radar. De esta manera, se intenta evitar que dos cuerpos a identificar queden tan cerca que no sea posible resolverlos.

La detección de objetos también depende en gran medida del algoritmo utilizado y los parámetros que se le pasa. Para ello, se han realizado una gran variedad de mediciones para tratar de optimizar el proceso. Filtrar por la altura no es una buena decisión, ya que la amplitud de la señal recibida es muy variable en función del nivel del ruido del sistema y de las diferentes ganancias de la antena. Filtrar por la anchura del pico tampoco es recomendable ya que, si se da la situación que hay varios objetos muy próximos, sus picos estarán muy próximos, dando lugar a picos con menor anchura de lo normal que no pasan el filtro, como se puede ver en la figura 61. En este caso, los glóbulos en torno a la frecuencia de 10KHz no son detectados al haber filtrado por la anchura del pico.

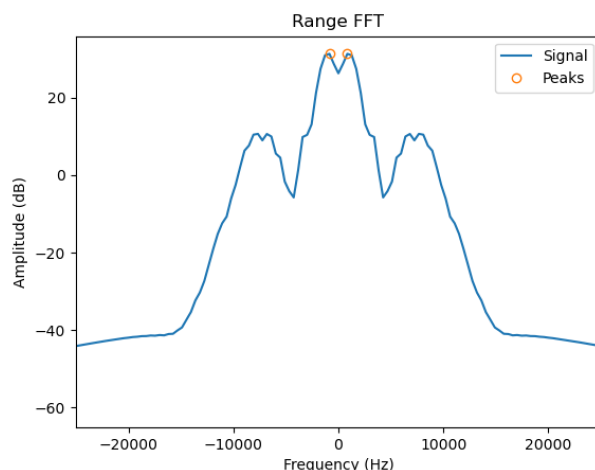


Figura 61. Resultado detección de objetos [4]

```
--  
frame_counter: 5600  
target_number: 1  
distance:  
- 0.3326203227043152  
magnitude:  
- 31.341812133789062  
phase:  
- 145.55783081054688  
velocity:  
- -0.00023573987709823996  
--
```

Figura 62. Resultado detección de objetos [5]

Lo óptimo ha resultado ser filtrar por la altura individual del pico, entendiendo como altura la distancia desde el pico hasta la base de menor altura. Siguiendo esté filtrado, se consigue la mayor detección de picos posibles, como se puede ver en la figura 63.

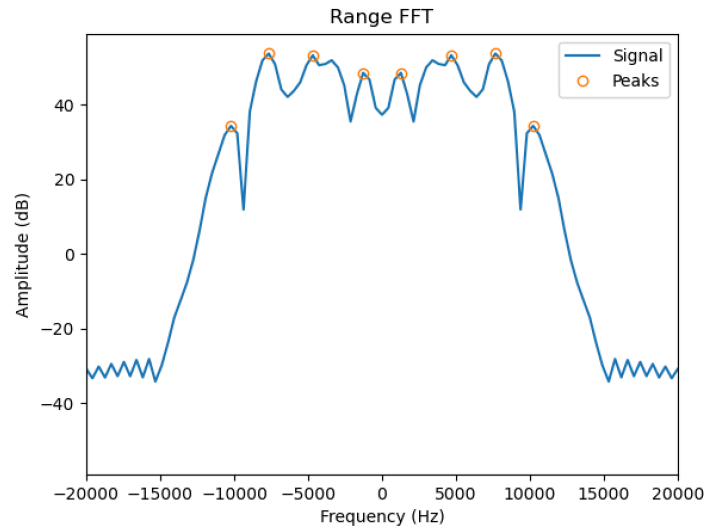


Figura 63. Resultado detección de objetos [6]

```

---
frame_counter: 13446
target_number: 4
distance:
- 0.4989304542541504
- 1.8294117450714111
- 2.9935827255249023
- 3.991443634033203
magnitude:
- 48.50715637207031
- 53.2181510925293
- 53.689598083496094
- 34.30365753173828
phase:
- -15.881953239440918
- -173.68804931640625
- 11.53531265258789
- -77.21569061279297
velocity:
- 0.014796042814850807
- -0.4202495813369751
- 0.01291094720363617
- 0.0004970852169208229
---
  
```

Figura 64. Resultado detección de objetos [7]

A la hora de seguir movimientos pequeños, resulta mucho más preciso utilizar la velocidad a partir de las variaciones de fase que a partir de las variaciones de distancia. Por ejemplo, dada la configuración actual del radar, si el objeto se mueve $0.6mm$, el cambio en frecuencia será de apenas $1.5H_z$.

$$\Delta f = \frac{2S\Delta R}{c} = \frac{2 \cdot 3.84 \cdot 10^{11} \cdot 0.6mm}{3 \cdot 10^8} = 1.5H_z$$

Esta variación es prácticamente imposible de detectar en el espectro de la señal, ya que la señal de frecuencia intermedia es típicamente del orden de kilohercios o megahercios. Si miramos el cambio de fase correspondiente a ese movimiento, observamos que se trata de π .

$$\Delta\phi = \frac{4\pi\Delta R}{\lambda} = \frac{4\pi \cdot 0.6mm}{0.0025} = \pi$$

Por lo que el cambio en la fase de la señal de frecuencia intermedia es muy sensible a variaciones en la distancia y por ello resulta mucho más eficaz para el seguimiento de pequeños movimientos.

Debido a que la ventana temporal entre chirps transmitidos es muy pequeña ($2.6\mu_s$), es difícil ver una gran variación de movimiento en ese periodo por lo que todas las velocidades con las que se trabaja son pequeñas. Esto no supone ningún problema ya que lo que se busca son variaciones de velocidad periódicas en el tiempo, ya que esas variaciones son las producidas por los movimientos corporales que este radar trata de seguir.

Capítulo 7. Conclusiones y Trabajos Futuros

A lo largo de este capítulo se van a exponer las conclusiones y posibles trabajos futuros que se pueden perseguir en el ámbito de la tecnología radar de onda continua modulada en frecuencia.

7.1. Conclusiones

Se puede afirmar que se ha complicado satisfactoriamente el objetivo principal de este trabajo: la detección de distancia, fase y velocidad de numerosos objetos en el campo de visión del radar. Durante este subapartado, se definirá las conclusiones sacadas de cada uno de los objetivos planteados en el capítulo 4.2.

7.1.1. Comprender parámetros y procesamiento de un radar FMCW

Este objetivo ha sido la base sobre el que se han apoyado todos los demás y el más importante de todos. Por ello ha sido al que más tiempo se le ha dedicado. Finalmente, se puede afirmar un completo conocimiento del funcionamiento de un radar de onda continua modulada en frecuencia. Esto engloba las diferentes configuraciones posibles y más importante, las limitaciones a las que hace frente la tecnología radar FMCW y como poder solventarlas. Entre estas limitaciones se encuentran la mínima potencia transmitida para recibir el eco con una buena relación señal a ruido, la resolución del radar, el Criterio de Rayleigh, la frecuencia de muestreo del ADC o el retardo temporal entre rampas transmitidas.

7.1.2. Comprender entorno ROS2

Esta era uno de los objetivos primordiales de este trabajo, puesto que no es un radar cualquiera, sino que es un radar integrado en un robot con muchos más componentes y que debe de funcionar en armonía con el resto. Se puede afirmar que se ha cumplido satisfactoriamente el objetivo propuesto al haber conseguido integrar el radar en el entorno ROS2 Iron mediante la creación de tres nodos y cuatro tópicos únicamente dedicados a la programación del radar.

7.1.3. Protocolo de funcionamiento y configuración del radar EasyRad r4

Este objetivo ha resultado, sorprendentemente, de los más complicados, quizás por la escasa experiencia trabajando con radares. Para su consecución se ha hecho referencia al manual del fabricante, el cual ha resultado en muchas ocasiones contradictorio y no demasiado claro. Sin embargo, también se puede afirmar su consecución al entender el flujo de mediciones y las diferentes configuraciones posibles del radar. Ha sido clave para el desarrollo del software de detección de objetos.

7.1.4. Detectar la distancia a la que se encuentran los objetos

Este objetivo ha sido el más importante de todos, y el objetivo primordial de este trabajo. Engloba los tres anteriores ya que requiere del conocimiento de la tecnología FMCW, del protocolo del radar EasyRad r4 y del entorno ROS sobre el que trabajar y programar el radar. Una vez más, podemos afirmar que las mediciones obtenidas son precisas y que se ha cumplido el objetivo.

7.1.5. Detectar la velocidad a la que se mueven los objetos

Este objetivo va de la mano con el número cuatro, ya que se apoya en los tres primeros. Como se ha explicado en el capítulo 6, para la detección de velocidad resulta imprescindible conocer el retardo temporal entre cada rampa, lo cual no hay manera de saber con precisión, lo que conlleva un pequeño error en la medición. Sin embargo, esta es una limitación hardware del fabricante a la que no se puede hacer frente y que resulta en un error pequeño por lo que también se puede afirmar haber cumplidos el objetivo.

7.1.6. Transmisión de las mediciones de distancia, fase y velocidad

El último objetivo de todos resulta, quizás el más sencillo. Consiste en la transmisión de las medidas obtenidas por la red haciendo uso de tópicos del entorno ROS2. Ha resultado de fácil

cumplimiento ya que, una vez llegado a este punto, se cuenta con un gran conocimiento del entorno ROS2 y un procesamiento digital de señales completo.

Quisiera además hacer hincapié en la oportunidad brindada a la hora de poder trabajar con el Instituto de Robótica de la Universidad Carnegie Mellon, posicionada como número uno en el campo de la robótica tanto en el ranking estadounidense, seguida por el Massachusetts Institute of Technology y Stanford, como en el mundial [19]. El poder trabajar junto a alumnos de máster, doctorado o investigadores profesionales en el desarrollo del Team Chiron ha resultado ser una experiencia muy enriquecedora de la que he aprendido una gran cantidad de cosas y me siento muy afortunado de poder vivir.

7.2. Trabajos Futuros

Existe un gran número de trabajos futuros que se pueden perseguir en el mundo de la robótica aplicada a sistemas IoT y a la medición por radiofrecuencia. En este apartado, se explicarán un par de alternativas que se pueden perseguir utilizando la misma tecnología radar FMCW.

En primer lugar, la extracción del ritmo cardiaco a través de radar FMCW. Esta técnica se basa en la detección de movimientos periódicos que realizan los seres humanos al respirar como puede ser inhalar o exhalar y pequeñas vibraciones en la piel que se producen como efecto de los latidos del corazón. Este proceso cuenta con tres etapas las cuales se describen a continuación.

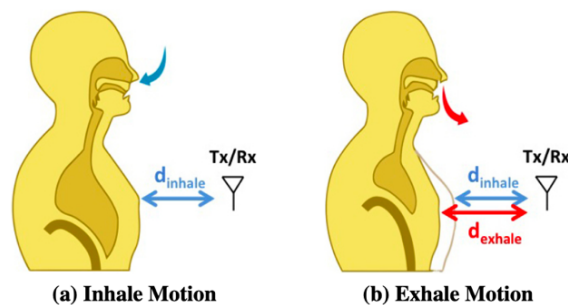


Figura 65. Movimiento producido al exhalar/inhalar. (Publicación científica: Smart Homes that Monitor Breathing and Heart Rate)

1. Eliminar mediciones de elementos estáticos. Debido a que los movimientos producidos por los seres humanos al respirar son periódicos, la primera técnica de filtrado consiste en analizar variaciones de fase (o velocidad) a lo largo de una ventana temporal. Toda variación que se mantenga constante se descarta.
2. Identificar movimientos relacionados con el ritmo respiratorio y cardiaco. Consiste en centrarse únicamente en aquellos movimientos que sean periódicos. Por ejemplo, el movimiento repentino de un brazo o una rodilla no es periódico en el tiempo, se descarta. Sin embargo, el movimiento producido al inhalar o exhalar sí lo es.

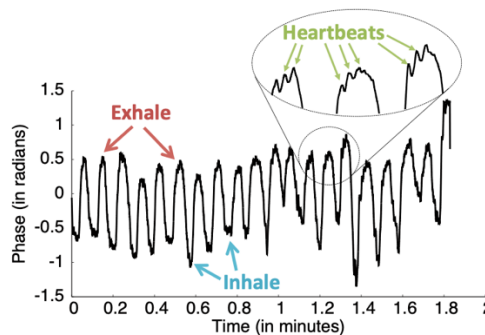


Figura 66. Variación de fase como resultado de respirar. (Publicación científica: Smart Homes that Monitor Breathing and Heart Rate)

3. Una vez tenemos aislados los movimientos que indiquen actividad respiratoria o latidos del corazón, se procede a realizar la transformada de Fourier mediante el algoritmo FFT para extraer la frecuencia respiratoria.

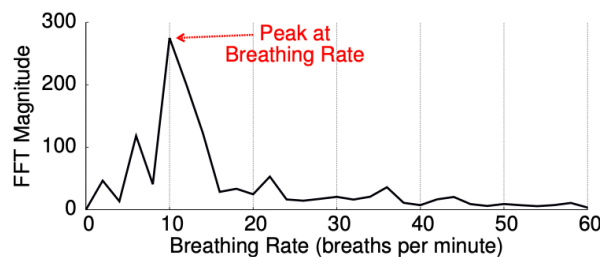


Figura 67. Frecuencia ritmo respiratorio. (Publicación científica: Smart Homes that Monitor Breathing and Heart Rate)

Otro proyecto y futura dirección de investigación es el llamado vital-radio, planteado por el Massachusetts Institute of Technology. Consiste en la utilización de la tecnología radar de onda continua modulada en frecuencia para la detección del ritmo cardiaco en varias personas, de

manera simultánea y en entornos cerrados, concretamente en casas u oficinas. De esta manera se crea un entorno inteligente, que puede estar conectado con médicos o servicios externos, proporcionando información del ritmo cardiaco de manera continua, sin ningún aparato de por medio y mientras los sujetos están realizando actividades cotidianas como puede ser ver la televisión, dormir o trabajar [15].

Este estudio muestra una alta eficacia en la monitorización del ritmo cardiaco en varios sujetos, en un rango total de 8 metros, como se puede ver en el gráfico inferior.

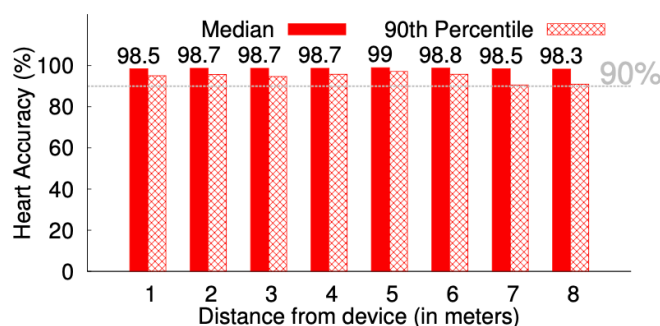


Figura 68. Efectividad vital-radio. (Publicación científica: Smart Homes that Monitor Breathing and Heart Rate)

Sin embargo, también muestra numerosas limitaciones que se deben de tener en cuenta. El sujeto o sujetos expuestos a vital-radio deben de estar semi quietos para que las mediciones sean precisas. Si, por ejemplo, el sujeto está andando, o haciendo deporte resulta imposible aislar las mediciones causadas al respirar o como efecto de los latidos del corazón. En segundo lugar, este dispositivo no limita el rango de acción a humanos, por lo que, si un perro o un gato estuvieran dentro del alcance del radar, los confundiría como humanos e interpretaría su ritmo cardiaco como el de cualquier persona [15]. Este problema se puede solucionar añadiendo un pequeño chaleco a las mascotas hecho de materiales RAM (Radar Absorbing Material). Este material se caracteriza por la absorción de ondas electromagnéticas, reduciendo casi por completo los ecos producidos. Actualmente, se utilizan en entornos militares con el fin de evitar ser detectados por el enemigo y en entornos de laboratorio donde se busca crear ecosistemas perfectos para la medición de señales por radiofrecuencia. [20]

Capítulo 8. Bibliografía

- [1] A.piquerasm, «Radiofrecuencia,» Wikipedia, 15 Abril 2024. [En línea].
- [2] Defense Advanced Research Projects Agency , «DARPA Triage Challenge Year 1,» Octubre 2024. [En línea]. Available: <https://www.darpa.mil/news-events/darpa-triage-challenge-year-1#:~:text=The%20main%20goal%20of%20the,%2C%20and%20mass%2Dcasualty%20settings.>
- [3] Open Robotics, «Begginer: CLI tools,» [En línea]. Available: <https://docs.ros.org/en/iron/Tutorials/Beginner-CLI-Tools.html>.
- [4] Ferrovial, «Robótica,» Ferrovial, [En línea]. Available: <https://www.ferrovial.com/es-es/innovacion/tecnologias/robotica/>.
- [5] Carnegie Mellon University, «Team Chiron Making remote autonomous triage a reality,» [En línea]. Available: <https://teamchiron.ai/#about>.
- [6] D. R. M. O'Donnell, «Introduction to Radar Systems,» de Radar: Introduction to Radar Systems — Online Course.
- [7] D. R. M. O'Donnell, «The Radar Equation,» de Radar: Introduction to Radar Systems — Online Course, 2002.
- [8] tutorialspoint, «Radar Systems - Types of Radars,» tutorialspoint, [En línea]. Available: https://www.tutorialspoint.com/radar_systems/radar_systems_types_of_radars.html.
- [9] «The Doppler Effect,» Physics Classroom - Home, 2023. [En línea]. Available: <https://www.physicsclassroom.com/class/waves/Lesson-3/The-Doppler-Effect>.
- [10] T. I. Sandeep Rao, «mmWave Radar Sensors,» 28 Abril 2017. [En línea]. Available: <https://www.ti.com/video/series/mmwave-training-series.html>. [Último acceso: 2024].
- [11] «Understanding I/Q Signals and Quadrature Modulation,» All About Circuits, [En línea]. Available: <https://www.allaboutcircuits.com/textbook/radio-frequency-analysis-design/radio-frequency-demodulation/understanding-i-q-signals-and-quadrature-modulation/>.
- [12] C. G. d. I. Cueva, «Tema 3: Sistemas LTI en tiempo discreto. Filtros FIR.,» de Procesamiento Digital de Señales.

- [13] J. Saunders, «Autonomous aerial robotics for package delivery: A technical review,» Wiley Online Library, [En línea]. Available: <https://onlinelibrary.wiley.com/doi/10.1002/rob.22231>.
- [14] M. Gaul, «Sensors: When the Car Sees Around the Corner,» Dekra, [En línea]. Available: <https://www.dekra.com/en/sensors-when-the-car-sees-around-the-corner/>.
- [15] P. Steenkiste, «18-452 / 18-750 Wireless Networks and Applications,» 2024. [En línea]. Available: <https://www.cs.cmu.edu/~prs/wirelessS24/>.
- [16] H. M. Z. K. D. K. R. C. M. Fadel Adib, «Smart Homes that Monitor Breathing and Heart Rate,» Massachusetts Institute of Technology, Massachusetts.
- [17] Federal Communications Commission, «FCC ONLINE TABLE OF FREQUENCY ALLOCATIONS,» Federal Register.
- [18] M. O. R. Nave, «El Criterio de Rayleigh,» [En línea]. Available: <http://hyperphysics.phy-astr.gsu.edu/hbasees/phyopt/Raylei.html>.
- [19] EduRank, «100 Best colleges for Robotics in the United States,» 29 Febrero 2024. [En línea]. Available: <https://edurank.org/engineering/robotics/us/>.
- [20] AFT, «THE SCIENCE BEHIND RADAR ABSORPTION MATERIALS,» AFT, 17 Abril 2024. [En línea]. Available: <https://aft.systems/the-science-behind-radar-absorption-materials/#:~:text=Materials%20Used%20in%20Radar%20Absorption,painted%20with%20fire%20Dresistant%20paint.>
- [21] educo, «Qué son los 17 Objetivos de Desarrollo Sostenible y la Agenda 2030,» educo, 23 Septiembre 2019. [En línea]. Available: https://www.educo.org/blog/que-son-los-17-objetivos-de-desarrollo-sostenible?utm_source=google&utm_medium=cpc&utm_campaign=educo_brand_dsa&utm_term=kw&utm_content=text&tc_alt=64115&n_o_pst=n_o_pst&n_okw=__c_103267085027&gad_source=1&gclid=CjwKCAjw1K-zBhB.
- [22] Q. Chaudhari, «Wireless Pi,» [En línea]. Available: <https://wirelesspi.com/fmcw-radar-part-1-ranging/>.

- [23] P. O. González, Radiación y propagación Tema 4: Ondas esféricas, Madrid : comillas.edu, 2023.
- [24] Q. Chaudhari, «Wireless Pi,» [En línea]. Available: <https://wirelesspi.com/fmcw-radar-part-2-velocity-angle-and-radar-data-cube/#:~:text=A%20CW%20radar%20measures%20the,frequency%20to%20yield%20the%20velocity.>
- [25] Open Robotics, «Beginner: Client libraries,» [En línea]. Available: <https://docs.ros.org/en/iron/Tutorials/Beginner-Client-Libraries.html>.
- [26] S. team, «strapsai github association,» [En línea]. Available: <https://github.com/strapsai>.
- [27] C. D. T. C. T. Chiron, Dirección, DTC Qualitifaction Spot Short video. [Película]. United States of America: Youtube, 2024.

ANEXO I: ALINEACIÓN DEL PROYECTO CON LOS ODS

Los Objetivos de Desarrollo Sostenible (ODS) son un conjunto de 17 metas globales adoptadas por todos los Estados miembros de las Naciones Unidas en 2015, como parte de la Agenda 2030 para el Desarrollo Sostenible. Estos objetivos buscan abordar los mayores desafíos que enfrenta el mundo actual, entre los que destaca la erradicación de la pobreza, la lucha contra el hambre, la promoción de la salud y el bienestar, la educación de calidad, la igualdad de género, y la protección del medio ambiente, entre otros [21].



Mediante este radar, se busca la innovación tecnológica en busca del apoyo sanitario en situaciones críticas donde el acceso humano es limitado. El radar implementado nos permite detectar el ritmo cardíaco de una persona sin ningún contacto físico, asistencia humana y de manera totalmente inalámbrica, lo cual tiene el potencial de salvar muchas vidas.

Esta solución promete grandes avances biosanitarios en el mundo militar y civil. Puede ser integrada en un robot con más componentes con el fin de proporcionar apoyo sanitario y realizar diagnósticos rápidos sin necesidad de presencia humana, como se ha planteado en este trabajo. Además, también puede estar presente en entornos domésticos, monitorizando el ritmo cardíaco de personas mayores o con algún tipo de problema cardiovascular, con el fin de detectar rápidamente bajadas en el ritmo cardíaco, o anomalías y así avisar a los profesionales médicos con la mayor brevedad posible.

Analizando cada uno de los 17 objetivos de desarrollo sostenible, los que más se alinean con este trabajo son los siguientes.

Salud y bienestar. La tecnología planteada a lo largo de este trabajo tiene un impacto directo en el monitoreo de la salud, permitiendo la medición de manera continua del ritmo cardíaco en personas. Esta técnica puede ser utilizada para monitorizar pacientes de riesgo como puede ser una persona mayor, alguien con problemas cardiovasculares, o incluso bebés. De esta manera, se es capaz de detectar anomalías y bajadas de tensión mucho más rápido, permitiendo una intervención a tiempo y llegando a salvar muchas vidas.

Además, al no requerir contacto físico o asistencia humana, se puede utilizar de manera paralela con profesionales médicos. Particularmente, en situaciones críticas donde el número de heridos es elevado y el personal sanitario limitado. En estos casos, esta tecnología nos permite en una sola pasada, un escaneo y diagnóstico rápido de la situación, facilitando la intervención a los más necesitados.

Industria, innovación e infraestructura. El desarrollo y la implementación de tecnologías avanzadas como el radar FMCW para la detección del ritmo cardiaco fomenta la innovación tecnológica. Esto impulsa la industria de dispositivos médicos, mejorando las infraestructuras de salud y contribuyendo al avance tecnológico en la medicina. Un ejemplo es el proyecto comentado con anterioridad en la sección 7.2, donde un equipo del MIT⁴⁷ se ha hecho eco de esta tecnología para el desarrollo de vital-radio, un sistema con el cual monitorizar el ritmo cardiaco de personas en entornos domésticos.

Ciudades y comunidades sostenibles. En los objetivos anteriores se ha comentado como se puede aplicar esta tecnología a situaciones críticas o entornos domésticos, pero también se puede aplicar en gran escala a entornos urbanos y comunitarios, proporcionando servicios sanitarios accesibles a todo el público. Por ejemplo, se puede instalar radares FMCW en lugares concurridos para el monitoreo del ritmo cardiaco de todas las personas que pasen por allí, agrandando así el campo de acción y la repercusión de esta tecnología.

⁴⁷ Del inglés: Massachussets Institute of Technology