



**UNIVERSIDAD DE OVIEDO**

**CENTRO INTERNACIONAL DE POSTGRADO**

# **MÁSTER EN INGENIERÍA MECATRÓNICA**

**TRABAJO FIN DE MÁSTER**

**DESARROLLO DE UN SISTEMA DE MANTENIMIENTO PREDICTIVO  
INTEGRADO DENTRO DEL PROTOCOLO OPCUA**

**07/2019**

**Jakub Krzysztof Jedrszczak**

**Ignacio Álvarez García**

**[Firma]**

**[Firma]**



## **AGRADECIMIENTOS**

A mis tutores Ignacio Álvarez García y Rubén Pérez Chust, por el acompañamiento y la posibilidad de la realización de este trabajo.

A mis profesores José Manuel Sierra Velasco, Alberto García Martínez, Juan Diaz González y Miguel Ángel José Prieto, por los consejos y el permiso de la utilización del material universitario para la realización de pruebas, ensayos y construcción del prototipo.

Al departamento de mecatrónica de ArcelorMittal Eloy Fidalgo Blanco, Jacobo Fernández-Jordón Rodríguez, Andrés García Corbato, Carlos González Díaz y Juan Lopera Templado.

Y sobre todo a todos mis seres queridos y cercanos, que siempre estuvieron allí para apoyarme.



## **RESUMEN**

El objetivo principal de este trabajo fin de máster es el desarrollo de un sistema de mantenimiento predictivo integrado dentro del protocolo OPCUA, para la empresa ArcelorMittal.

El sistema estará compuesto por:

- Sensores de medición y envío de datos.
- Servidor destinado a la recogida e interpretación de esta información.
- Cliente, un interfaz gráfico con el cual podrá interactuar el usuario.

Para conseguirlo, se dividió el trabajo en diferentes apartados que se desarrollaron de forma progresiva.

Estado de arte sobre las plataformas lowcost para desarrollo de sensores para el mantenimiento predictivo e integración del protocolo OPCUA.

Análisis teórico de una maqueta de sistema de elevación por cable con un motor trifásico asíncrono. Búsqueda y determinación de fallos en rodamientos y engranajes dentro del espectro de las frecuencias utilizando un micrófono direccional.

Diseño de una carcasa, para el sensor acústico en el programa SolidWorks (software CAD para el modelado 3D). Impresión de esta mediante una impresora 3D y generación de sus planos.

Diseño y fabricación de una placa PCB para la integración de un módulo xBee destinado a la comunicación UART.

Programación de un medidor de nivel de ruido a determinadas frecuencias, comunicación UART, servidor OPCUA, programas para el mantenimiento predictivo en rodamientos y engranajes.

Integración del protocolo OPC-UA y ModBus conectado a sensores de presión y temperatura a través de RS485.

El presupuesto, conclusiones y trabajos futuros.

## **PALABRAS CLAVE**

OPCUA – Mantenimiento predictivo – FFT (Fast Fourier Transform)- Ordenador de placa reducida - ModBus – UART



## ÍNDICE GENERAL

AGRADECIMIENTOS .....	I
RESUMEN.....	III
PALABRAS CLAVE .....	III
ÍNDICE GENERAL .....	V
ÍNDICE MEMORIA.....	VII
LISTA DE FIGURAS .....	IX
LISTA DE TABLAS .....	XI
MEMORIA.....	XIII
PRESUPUESTO .....	XV
CONCLUSIONES Y FUTUROS TRABAJOS.....	XVII
BIBLIOGRAFÍA.....	XIX
ANEXO I: AUTOSTART DE UN PROGRAMA EN LA BEAGLEBONE BLACK.....	XXI
ANEXO II: PLANOS ELECTRÓNICOS: XBEE - BEAGLEBONE BLACK .....	XXIII
ANEXO III: PLANOS ELECTRÓNICOS: RS485 – ARDUINO .....	XXV
ANEXO IV: EXPLOSIONADO Y PLANOS DE LA CARCASA.....	XXVII





## ÍNDICE MEMORIA

<b>1.</b>	<b>ANTECEDENTES Y OBJETIVOS .....</b>	<b>15</b>
1.1.	OBJETIVOS .....	15
1.2.	ESTADO DE ARTE: MANTENIMIENTO PREDICTIVO .....	15
1.3.	ESTADO DEL ARTE: PLATAFORMAS LOW COST PARA EL DESARROLLO DE UN SISTEMA DE MANTENIMIENTO PREDICTIVO .....	16
1.3.1.	<i>Aplicación del dispositivo</i> .....	16
1.3.2.	<i>Comparación de dispositivos para la aplicación del mantenimiento predictivo</i> .....	17
1.4.	ESTADO DEL ARTE: EL PROTOCOLO OPCUA .....	18
1.4.1.	<i>Introducción</i> .....	18
1.4.2.	<i>OPCUA (Open Protocol Communication Unified Architecture)</i> .....	18
<b>2.</b>	<b>ESPECIFICACIONES DEL SISTEMA.....</b>	<b>21</b>
<b>3.</b>	<b>DISEÑO DEL SISTEMA.....</b>	<b>23</b>
3.1.	ESQUEMA DEL SISTEMA .....	23
3.2.	DESCRIPCIÓN DE DISPOSITIVOS UTILIZADOS .....	24
<b>4.</b>	<b>ESTUDIO DE LAS VIBRACIONES.....</b>	<b>27</b>
4.1.	INTRODUCCIÓN Y DESCRIPCIÓN DEL EQUIPO UTILIZADO .....	27
4.2.	ANÁLISIS DE DEFECTOS EN LOS ENGRANAJES .....	29
4.2.1.	<i>Cálculo de las frecuencias de desgaste</i> .....	30
4.2.2.	<i>Búsqueda de defectos en el espectro de frecuencias</i> .....	30
4.2.3.	<i>Eje desalineado</i> .....	31
4.2.3.1.	Síntomas del eje desalineado .....	31
4.2.3.2.	Búsqueda de defectos: .....	32
4.2.3.3.	Búsqueda de la velocidad del eje: .....	32
4.2.3.4.	Estudio de los armónicos de GMF y sus bandas laterales .....	33
4.2.4.	<i>Desgaste del diámetro primitivo o diente roto:</i> .....	35
4.2.4.1.	Síntomas .....	35
4.2.4.2.	Búsqueda del defecto .....	36
4.2.5.	<i>Engranaje excéntrico o eje doblado</i> .....	37
4.2.5.1.	Síntomas .....	37
4.2.5.2.	Búsqueda de defectos .....	38
4.3.	RODAMIENTOS: .....	38
4.3.1.	<i>Hipótesis de partida y el alcance</i> .....	38
4.3.2.	<i>Frecuencias de fallo en los rodamientos</i> .....	39
4.3.3.	<i>Cálculo de las frecuencias de deterioro de rodamientos:</i> .....	40
4.3.4.	<i>Detección de defectos de tipo BPF1</i> .....	41
4.3.4.1.	Curtosis .....	43
4.3.4.2.	El curtograma .....	44
4.3.5.	<i>Detección de defectos de tipo BPF0</i> .....	48
4.3.5.1.	Síntomas .....	48
4.3.5.2.	Búsqueda de defectos .....	48
<b>5.</b>	<b>DISEÑO ELECTRÓNICO Y LA CONFIGURACIÓN DE LA COMUNICACIÓN.....</b>	<b>51</b>
5.1.	PCB .....	51
5.2.	CONFIGURACIÓN DE LOS MÓDULOS XBEE.....	52
5.3.	CONEXIÓN DE SENSORES DE TEMPERATURA Y PRESIÓN .....	53
<b>6.</b>	<b>DISEÑO DE LA CARCASA PARA EL SENSOR ACÚSTICO .....</b>	<b>55</b>

<b>7.</b>	<b>PROGRAMACIÓN .....</b>	<b>57</b>
7.1.	ANÁLISIS DE VIBRACIONES EN MATLAB .....	57
7.1.1.	<i>Defectos en engranajes</i> .....	57
7.1.2.	<i>Defectos en los rodamientos</i> .....	58
7.1.2.1.	BPFI.....	58
7.1.2.2.	BPFO.....	59
7.2.	PROGRAMACIÓN EN PYTHON.....	59
7.2.1.1.	Programa de análisis de vibraciones .....	59
7.2.2.	<i>Medidor de nivel de ruido integrado en la BeagleBone Black</i> .....	60
7.2.3.	<i>Servidores OPCUA</i> .....	62
7.2.3.1.	OPCUA Micrófono.....	63
7.2.4.	<i>OPCUA Modbus</i> .....	64
<b>8.</b>	<b>RESULTADOS.....</b>	<b>65</b>
8.1.	ANÁLISIS DE VIBRACIONES .....	65
8.2.	MEDICIÓN DE RUIDO INTEGRADO EN EL OPCUA .....	66
8.3.	INTEGRACIÓN DE MODBUS DENTRO DEL OPCUA .....	68

## LISTA DE FIGURAS

FIGURA 1. 1. TRANSFORMADA DE FOURIER, LA SEÑAL ROJA Y LA AZUL REPRESENTAN DOMINIOS DE TIEMPO Y FRECUENCIA RESPECTIVAMENTE .....	17
FIGURA 1. 2. RASPBERRY PI B, BEAGLEBONE BLACK, ODROID-XU4, DE10-NANO .....	17
FIGURA 1. 3. ESQUEMA DE FUNCIONAMIENTO DE OPCUA.....	18
FIGURA 1. 4. UAEXPERT: INTERFAZ GRATUITO DEL CLIENTE DE OPCUA .....	19
FIGURA 1. 5. SERVIDOR OPCUA FUNCIONANDO .....	19
FIGURA 3. 1. ESQUEMA GENÉRICO DEL FUNCIONAMIENTO DEL SISTEMA .....	23
FIGURA 3. 2. MICRÓFONO DIRECCIONAL.....	24
FIGURA 3. 3. MPU 6050 .....	24
FIGURA 3. 4. USB DSD TECH MODEL:SH-U10, UN MÓDULO MAX485 Y ARDUINO MEGA .....	24
FIGURA 3. 5. XBEE PRO S1.....	25
FIGURA 3. 6. TARJETA DE SONIDO, CABLE, TRIPODE .....	25
FIGURA 3. 7. ADAPTADOR DE XBEE/PC.....	25
FIGURA 4. 1. MAQUETA DEL SISTEMA DE ELEVACIÓN.....	27
FIGURA 4. 2. MOTOR ASÍNCRONO SIEMENS/ PLACA DE CARACTERÍSTICAS.....	27
FIGURA 4. 3. DATASHEET DEL FABRICANTE DEL MOTOR.....	28
FIGURA 4. 4. TREN DE ENGRANAJES UTILIZADO EN LOS ENSAYOS.....	28
FIGURA 4. 5. NÚMERO DE DIENTES DE CADA ENGRANAJE.....	28
FIGURA 4. 6. TACÓMETRO UTILIZADO PARA REALIZAR LAS MEDIDAS DE LA VELOCIDAD DE LOS EJES .....	29
FIGURA 4. 7. EJES DESALINEADOS.....	31
FIGURA 4. 8. DESALINEACIÓN DE UN ENGRANAJE, FIGURA DE [1,8,65].....	32
FIGURA 4. 9. GMF DE ENGRANAJE NO DESALINEADO Y DESALINEADO (1X, 2X, 3X) .....	32
FIGURA 4. 10. ENVOLVENTE DE LA SEÑAL .....	32
FIGURA 4. 11. COMPARACIÓN DE ESTUDIO DE ESPECTRO DE LA FRECUENCIA UTILIZANDO SOLAMENTE FFT Y FFT + ENVOLVENTE. FIGURA SACADA DE [1,9,75] .....	33
FIGURA 4. 12. COMPARACIÓN DE AMPLITUDES DE VELOCIDADES EN ENGRANAJE SANO/ DESALINEADO.....	33
FIGURA 4. 13. COMPARACIÓN DE SEÑALES DE ENGRANAJE SANO Y OTRO DEFECTUOSO .....	34
FIGURA 4. 14. 2XGMF DE 792.2 HZ ENGRANAJE SANO Y SUS RESPECTIVAS BANDAS LATERALES .....	34
FIGURA 4. 15. 2XGMF DE 790.2 HZ ENGRANAJE DEFECTUOSO Y RESPECTIVAS BANDAS LATERALES .....	35
FIGURA 4. 16. DIENTE DESGASTADO.....	36
FIGURA 4. 17. GMF ENGRANE DEL EJE 2 IGUAL A 82.6 HZ (1X, 2X, 3X) .....	36
FIGURA 4. 18. GMF EJE 2 ENGRANAJE SANO/GMF EJE 1 ENGRANAJE DEFECTUOSO.....	36
FIGURA 4. 19. GMF EJE 2 ENGRANAJE SANO/GMF EJE 1 ENGRANAJE DEFECTUOSO.....	37
FIGURA 4. 20. GMF EJE 2 ENGRANAJE SANO/GMF EJE 1 ENGRANAJE DEFECTUOSO.....	37
FIGURA 4. 21. EXCENTRICIDAD/ EJE DOBLADO.....	38
FIGURA 4. 22. COMPARACIÓN ENTRE EL ENGRANAJE SANO Y ENGRANAJE EXCÉNTRICO O CON EL EJE DOBLADO..	38
FIGURA 4. 23. AVERÍAS DENTRO DE LOS MOTORES ASÍNCRONOS .....	39
FIGURA 4. 24. PARTES DE RODAMIENTOS .....	39
FIGURA 4. 25. FRECUENCIAS DE FALLO DE UN RODAMIENTO, FIGURA SACADA DE [1,9,69].....	39
FIGURA 4. 26. RODAMIENTOS INSERTABLES AY12-XL-NPP-B .....	40
FIGURA 4. 27. DEFECTOS DE TIPO BPGI, BPGO, FTF, FIGURAS SACADAS DE [10,1,26-27]. .....	41
FIGURA 4. 28. FFT DE LA SEÑAL .....	42
FIGURA 4. 29. ENVOLVENTE DE LA SEÑAL .....	42
FIGURA 4. 30. CURTOSIS DE DOS SEÑALES, FIGURA SACADA DE [8] .....	43
FIGURA 4. 31. FLUJOGRAMA DEL INICIO DE ANÁLISIS DE DATOS PARA ENCONTRAR BPGI.....	43
FIGURA 4. 32. FLUJOGRAMA DE PROCESAMIENTO DE DATOS PARA BPGI .....	44
FIGURA 4. 33. CURTOGRAMA DE BPGI .....	44
FIGURA 4. 34. SEÑAL SIN FILTRAR/ SEÑAL FILTRADA.....	45
FIGURA 4. 35. BPGI ENCONTRADOS .....	45
FIGURA 4. 36. RODAMIENTO SIN FALLO .....	46
FIGURA 4. 37. DIAGNÓSTICO DE BPGI, FIGURA SACADA DE [1,9,72] .....	46
FIGURA 4. 38. BANDAS LATERALES DE 122 HZ +-FRECUENCIA EJE X1, X2, X3 .....	47
FIGURA 4. 39. BANDAS LATERALES DE 122 HZ +-FRECUENCIA EJE X1, X2, X3 .....	47

FIGURA 4. 40. BANDAS LATERALES DE SEGUNDO ARMÓNICO DE BPGI 244 HZ +-FRECUENCIA EJE X1, X2, X3.....	47
FIGURA 5. 1. ESQUEMÁTICO DE CONEXIONES DE LA BEAGLEBONE BLACK Y LA XBEE PRO S1 .....	51
FIGURA 5. 2. PINOUT BEAGLEBONE BLACK [11] Y XBEE PRO S1 [6,1,7] .....	51
FIGURA 5. 3. ADAPTADOR USB (LA OREJA) Y EL XBEE PRO S1.....	52
FIGURA 5. 4. XCTU .....	52
FIGURA 5. 5. USB DSD TECH MODEL:SH-U10 Y UN MÓDULO MAX485.....	53
FIGURA 5. 6. CONEXIÓN DE LOS DISPOSITIVOS.....	53
FIGURA 5. 7. ESQUEMÁTICO DE CONEXIONES DE RS485 .....	53
FIGURA 6. 1. CARCASA PARA LA SUJECIÓN DEL SENSOR ACÚSTICO .....	55
FIGURA 6. 2. MODELO 3D DE LA CARCASA, PERSPECTIVAS LATERALES DE LA CARCASA Y UN CORTE .....	55
FIGURA 6. 3. VISTA INFERIOR DE LA CARCASA .....	55
FIGURA 6. 4. EXPLOSIONADO .....	56
FIGURA 7. 1. AUDACITY Y MUESTRA DE GRABACIÓN.....	57
FIGURA 7. 2. BÚSQUEDA DE DEFECTOS EN LOS ENGRANAJES, FLUJOGRAMA.....	58
FIGURA 7. 3. FLUJOGRAMA DE BPGI .....	58
FIGURA 7. 4. FLUJOGRAMA DE BPGO.....	59
FIGURA 7. 5. PROGRAMA PRINCIPAL DE ANÁLISIS DE VIBRACIONES “MAIN_BUCLE.PY” .....	59
FIGURA 7. 6. ANÁLISIS DE BPGI “ANALISIS.PY” .....	60
FIGURA 7. 7. INICIO DE PROGRAMA “SOUND_LEVEL_TODO.PY” .....	61
FIGURA 7. 8. BUCLE PRINCIPAL.....	61
FIGURA 7. 9. HILO RX UART .....	62
FIGURA 7. 10. FLUJOGRAMA DEL SERVIDOR OPCUA DEL MICRÓFONO.....	63
FIGURA 7. 11. QMODBUSMASTER Y CONFIGURACIÓN RTU.....	64
FIGURA 8. 1. MAIN_BUCLE.PY .....	65
FIGURA 8. 2. MAIN_PROG.PY .....	65
FIGURA 8. 3. ANÁLISIS DE TIPO BPGI EN LA BEAGLEBONE BLACK.....	66
FIGURA 8. 4. TIEMPO DE EJECUCIÓN DE PROGRAMA .....	66
FIGURA 8. 5. SIMPLE TONE GENERATOR [22] .....	66
FIGURA 8. 6. FUNCIONAMIENTO DEL SISTEMA CON LA FRECUENCIA POR DEFECTO 440HZ.....	67
FIGURA 8. 7. MICRÓFONO APAGADO O SIN BATERÍA (<25 DB) .....	67
FIGURA 8. 8. CAMBIO DE FRECUENCIA DE REFERENCIA A 700 HZ, SE ACTIVA A LA ALARMA, Y A 15 KHZ SIN ANOMALÍA .....	67
FIGURA 8. 9. VALORES POR DEFECTO PROGRAMADOS EN ARDUINO.....	68
FIGURA 8. 10. SENSOR ENCENDIDO.....	68
FIGURA 8. 11. SENSOR APAGADO.....	68

## LISTA DE TABLAS

TABLA 1. 1. COMPARATIVA DE DISTINTOS PRODUCTOS LOW COST .....	17
TABLA 4. 1. VELOCIDADES TEÓRICAS DE LOS EJES .....	29
TABLA 4. 2. MEDIA DE VELOCIDADES MEDIDAS CON EL TACÓMETRO EN EL EJE 1 Y EL EJE 2.....	29
TABLA 4. 3. BANDAS ESPECTRALES PARA LA DETECCIÓN DE PROBLEMAS EN ENGRANAJES [1,8,63].....	31
TABLA 4. 4. BANDAS LATERALES DE 1ER GMF .....	34
TABLA 4. 5. BANDAS LATERALES DE 2XGMF .....	35
TABLA 4. 6. GMFS DEL EJE 2.....	36
TABLA 4. 7. BANDAS LATERALES DE SEGUNDO ARMÓNICO DE BPFI 244 HZ +-FRECUENCIA EJE X1, X2, X3 .....	47
TABLA 4. 8. ARMÓNICOS DE LA VELOCIDAD DEL EJE .....	48
TABLA 4. 9. FALLOS EN PISTA INTERIOR BPFO .....	48
TABLA 4. 10. RODAMIENTO SANO SOLAMENTE APARECEN LOS ARMÓNICOS DE LA VELOCIDAD DEL EJE .....	49
TABLA 4. 11. BPFO Y VARIOS DE SUS ARMÓNICOS.....	49
TABLA 7. 1. DEFINICIÓN DEL SERVIDOR OPC-UA .....	62



# MEMORIA





## 1. ANTECEDENTES Y OBJETIVOS

En este capítulo se van a describir tanto los objetivos como la razón del desarrollo de este trabajo. El porque puede interesar a una empresa invertir su tiempo y dinero en el mantenimiento predictivo y el desarrollo de un sistema con el protocolo OPCUA integrado.

### 1.1. Objetivos

El objetivo principal de este trabajo es el desarrollo de un sistema de mantenimiento predictivo, lowcost integrado dentro del protocolo OPCUA. Es importante asegurar que sea un sistema robusto capaz de integrar varios sensores. Ya que según el “Manual Análisis de las vibraciones” [1,3,26-32], un número mayor de fuentes de datos puede asegurar mediciones mucho más precisas y fiables (siempre y cuando se utilizan de forma adecuada).

Inicialmente el reto propuesto por la empresa ArceorMittal consistía en la monitorización de nivel de ruido de una frecuencia conocida utilizando un micrófono direccional de las frecuencias audibles y en el caso de aparición de alguna anomalía envió de una alarma.

Se ha contemplado un objetivo adicional que consiste en la búsqueda de posibles fallos utilizando esta tecnología en un banco de ensayos de un sistema de elevación, más concretamente en engranajes y rodamientos. También una posible mejora del sistema incorporando un acelerómetro e integración de protocolo ModBus dentro del OPCUA para mostrar su robustez.

### 1.2. Estado de arte: Mantenimiento predictivo

El mantenimiento predictivo es una serie de técnicas y acciones aplicadas a la maquinaria, con el fin de detectar los fallos y los defectos en sus etapas iniciales de la aparición. Muchas veces va acompañado por el mantenimiento proactivo, que supone una mejora ya que estudia la causa de la aparición de una avería.

Este tipo de estudios mejora considerablemente la economía de la empresa, ya que avisa cuando y donde se van a producir los fallos. A diferencia del mantenimiento preventivo que ocasiona parones de emergencia para revisiones rutinarias o el mantenimiento correctivo, que espera hasta que se produzca la avería, originando costes de reparación (mano de obra, piezas de repuesto, etc.) y costes derivados de las paradas de producción. [1,1,6]

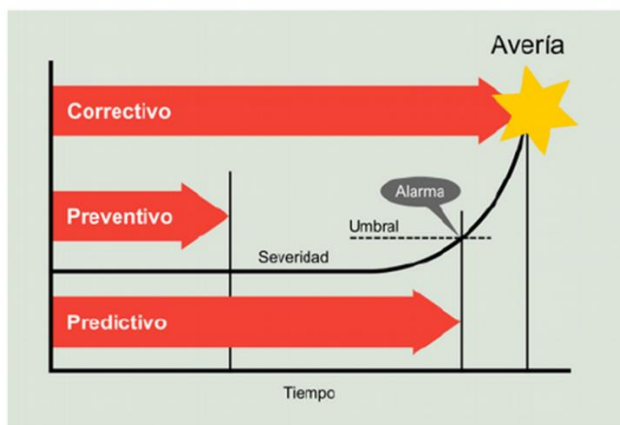


Figura 1.1. Distintos tipos del mantenimiento dentro de la industria, figura copia del [1,1,6]

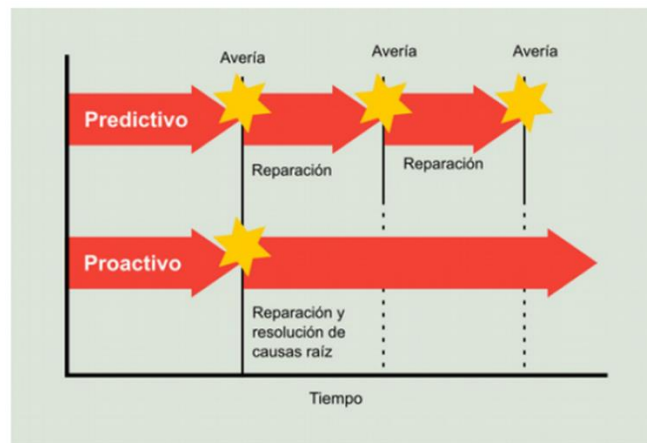


Figura 1.2. Mejora que supone el mantenimiento proactivo frente al predictivo [1,1,6]

Parece ser una forma de mantenimiento idónea, pero tiene sus desventajas importantes. La principal es el elevado precio de sus equipos, ya que necesitan una alta capacidad de cálculo y de almacenamiento.

El mantenimiento predictivo se puede realizar con dos tipos de equipos diferentes, que pueden ser:

- Portátiles, tomados por un operario de forma periódica. Suelen ser más baratos, pero tienen un mayor coste de explotación e intervalos de toma de datos más grandes. Por lo tanto, detectan más tarde las anomalías.
- Continuos u On-line. Son sistemas automatizados que se caracterizan por ofrecer mayores beneficios al reducir de forma drástica los intervalos de tiempo de la toma de los datos, permitiendo detectar de manera inmediata cualquier tipo de anomalía dentro del equipo. Poseen menor coste de explotación y mayor calidad al medir siempre en el mismo punto. Su mayor desventaja es que sus equipos son más caros y suelen ocupar más espacio debido a su tamaño y/o el cableado.

### **1.3. Estado del arte: Plataformas low cost para el desarrollo de un sistema de mantenimiento predictivo**

Gracias al continuo desarrollo de la tecnología de los últimos años, comenzaron a aparecer cada vez más dispositivos de tipo Open Source en el mercado. Marcas como Arduino o Raspberry Pi se han popularizado mucho, gracias a su: sencillez, precio económico y gran cantidad de aplicaciones no solamente en los ámbitos aficionados sino también en los profesionales. Por lo tanto, parecen ser opciones perfectas para este trabajo.

#### **1.3.1. APLICACIÓN DEL DISPOSITIVO**

En el apartado anterior se ha comentado que el mantenimiento predictivo necesita una alta capacidad de cálculo y almacenamiento.

Una de las operaciones matemáticas más utilizadas en este trabajo es la Transformada Rápida de Fourier o la FFT (Fast Fourier Transform) [2]. Es una herramienta clave, ya que permite descomponer una señal en senoides y llevarla desde el dominio de tiempo al de la frecuencia.

Cumple con uno de los objetivos principales de este texto que es: “La monitorización de nivel de ruido de una frecuencia”, pero para llevarla al cabo se necesita un dispositivo con un procesador muy rápido o una FPGA en la que podríamos realizar las sumas y multiplicaciones por hardware.

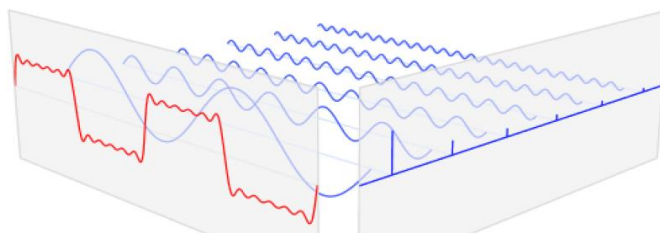


Figura 1. 1. Transformada de Fourier, la señal roja y la azul representan dominios de tiempo y frecuencia respectivamente

### 1.3.2. COMPARACIÓN DE DISPOSITIVOS PARA LA APLICACIÓN DEL MANTENIMIENTO PREDICTIVO

En la siguiente tabla se puede ver una comparativa de distintos productos low cost, ideales para este trabajo.

	Raspberry Pi B	BeagleBone Black	Odroid-XU4	DE10-Nano
CPU	1.2 GHz	1Ghz	A15 2 GHz/ A7 Octa core	800 MHz
USB 2.0	4 puertos	1 puerto	1 puerto	1 puerto
USB 3.0	X	X	2 puerto	X
O.S.	Linux	Linux, Android, Cloud9 IDE,	Linux, Android	Linux
Memoria	512MB SDRAM	512MB	2 GB	1 GB
Alimentación	5.1 V/ 2,5 A	5 V/ 2 A	5V/4A	5V / 2A
Precio (€)	32.08	62.75	74.95	130.0

Tabla 1. 1. Comparativa de distintos productos low cost

Tal y como muestra la Tabla 1.1 la mejor opción para la realización del mantenimiento predictivo sería la utilización de la DE10-Nano o la Odroid-XU4, debido a las rápidas velocidades de sus procesadores y precios económicos. También las FPGAs poseen la gran ventaja de poder realizar los cálculos por hardware sin influir en el software, lo que aceleraría muchísimo el tiempo de funcionamiento del sistema. La DE10-Nano dentro del diseño hardware posee un motor de transformada rápida de Fourier (FFT) de 32 bits.

Debido a la disponibilidad de dispositivos se ha escogido la BeagleBone Black, para programarla de forma genérica y robusta, pudiendo exportar los programas a las demás plataformas.

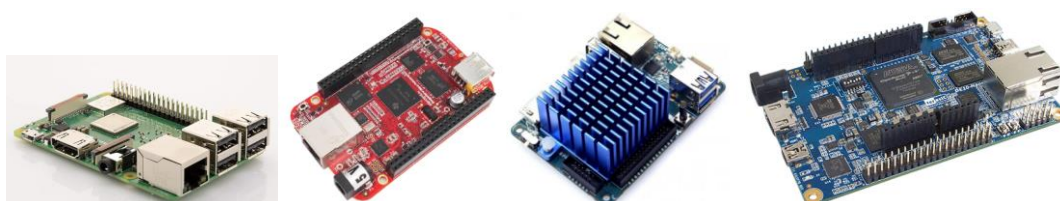


Figura 1. 2. Raspberry Pi B, BeagleBone Black, Odroid-XU4, DE10-Nano

## 1.4. Estado del arte: el protocolo OPCUA

### 1.4.1. INTRODUCCIÓN

Mantener la comunicación en plantas industriales no es una tarea fácil. Alla por los años 80 cada vez que una empresa adquiría una nueva máquina eso suponía la introducción de un nuevo protocolo de comunicación, que tenía que comunicarse con el resto de los equipos, instalar nuevos drivers y mantenerse, junto con el resto de la instalación. Esto generaba grandes problemas para los ingenieros debido a la cantidad de información específica que tenían que manejar, en respuesta a esto se ha creado un protocolo estándar llamado OPC.

OPC es un protocolo de comunicación estándar, abierto e independiente. Supuso toda una revolución en el mantenimiento de los equipos industriales, pero tiene dos defectos muy importantes. El primero es que no es multiplataforma, solamente funciona en Windows y el segundo es la seguridad.

Debido a eso OPC Foundation ha creado el OPCUA un nuevo protocolo que solucionaría estos problemas.

### 1.4.2. OPCUA (OPEN PROTOCOL COMMUNICATION UNIFIED ARQUITECTURE)

OPCUA [3] (Open Protocol Communication Unified Architecture) es un protocolo de comunicación pensado para el intercambio de información entre equipos industriales. Sus mayores ventajas respecto a los demás protocolos son:

- Multiplataforma
- Abierto
- Orientado a servicios,
- Con ricos modelos de información
- Mantenido por la OPC Foundation

Por lo tanto, es idóneo para una aplicación IoT o de Industria 4.0, algo que intentan introducir todas las empresas dentro de su filosofía hoy en día.

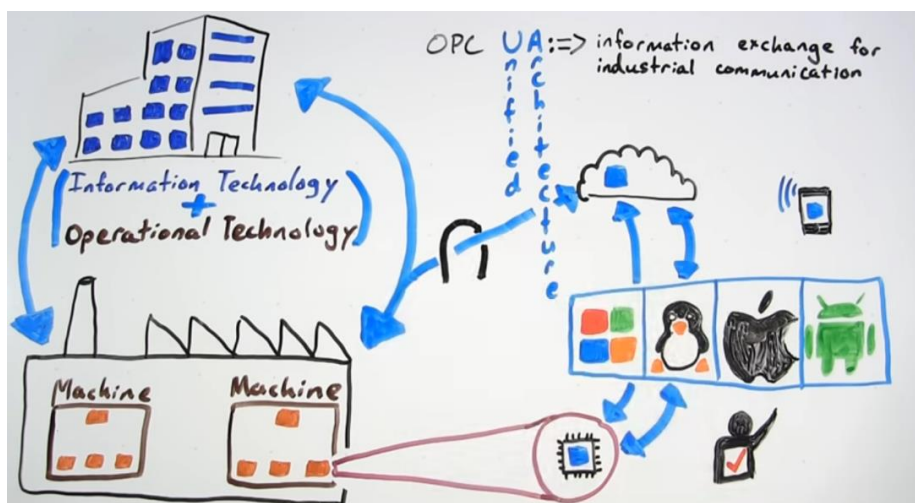


Figura 1. 3. Esquema de funcionamiento de OPCUA

## Desarrollo de un sistema de mantenimiento predictivo integrado dentro del protocolo OPCUA

Es una arquitectura cliente-servidor, lo que significa que tiene dos softwares independientes. Uno para el servidor conectado a la fuente de datos y otro al cliente que será su intérprete. En el caso del presente trabajo el servidor se encargará de recoger la información enviada por los sensores y procesarla. El cliente será un interfaz gráfico, un programa gratuito llamado UaExpert, que permitirá al usuario interpretar la información.

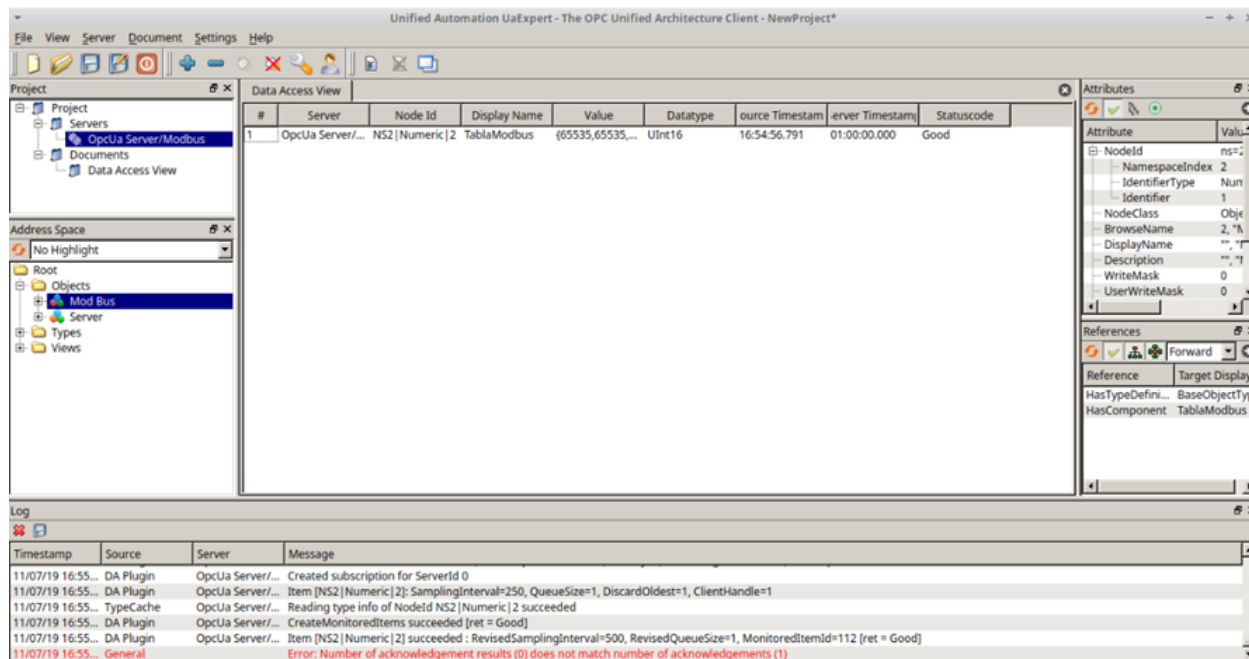


Figura 1. 4. UaExpert: interfaz gratuito del cliente de OPCUA

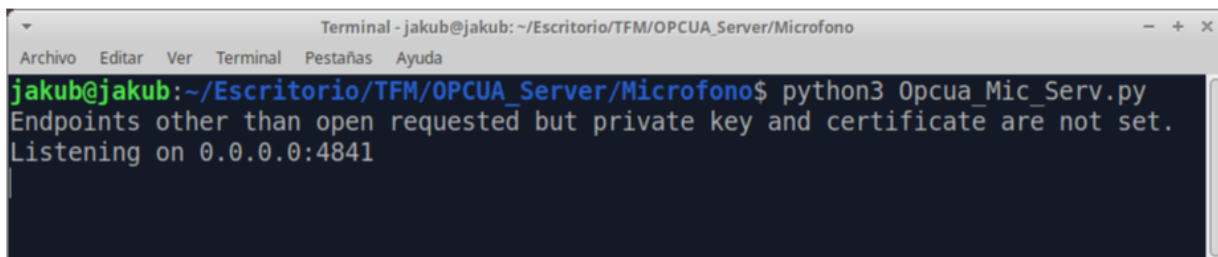


Figura 1. 5. Servidor OPCUA funcionando



## **2. ESPECIFICACIONES DEL SISTEMA**

Las especificaciones del sistema son las siguientes:

- Capaz de realizar el mantenimiento predictivo. Como se ha visto en el capítulo anterior dentro del apartado “1.2 Estado del arte de mantenimiento predictivo”, se necesita un dispositivo con bastante capacidad computacional, para obtener resultados en el tiempo mínimo.
- Robusto, con la posibilidad de incorporación de más sensores.
- Con el protocolo OPCUA integrado. Al ser un protocolo multiplataforma podemos optar por cualquier dispositivo programable.
- Integración del micrófono direccional
- Integración de ModBus, conectado a un sensor de temperatura y presión a través de RS485, dentro del protocolo OPCUA
- Reducido, tiene que ocupar poco espacio y reducción de todo el cableado despreciable.
- Inalámbrico, uso de los módulos xBee
- Propuesta de la integración de un acelerómetro





### 3. DISEÑO DEL SISTEMA

#### 3.1. Esquema del sistema

Teniendo en cuenta los antecedentes, objetivos y las especificaciones del sistema se ha diseñado un esquema para estructurar la forma de funcionamiento del prototipo.

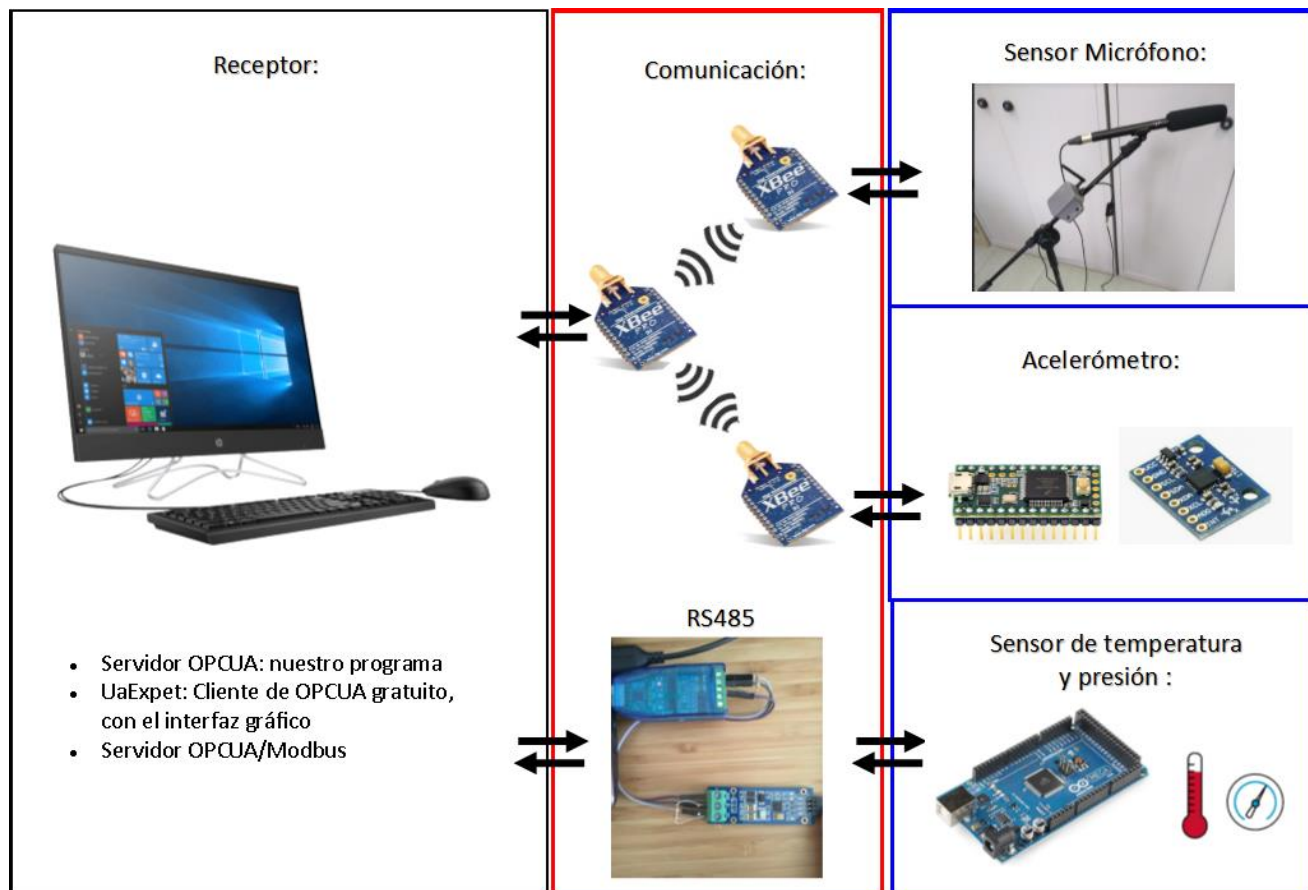


Figura 3. 1. Esquema genérico del funcionamiento del sistema

El sistema está compuesto por 3 bloques:

- **Receptor:** el OPCUA es una arquitectura de tipo cliente-servidor. Para simplificar el trabajo se integrarán los dos dentro del ordenador, pero hay que recordar que son softwares independientes. El cliente es donde se visualizarán los datos mediante un interfaz gráfico llamado UaExpert, es un programa gratuito que servirá del cliente para el OPCUA. El servidor también estará incluido dentro de este bloque y es el programa escrito como parte de este trabajo.
- **Comunicación:** la comunicación con el micrófono y el acelerómetro se realizará mediante los módulos xBee. Para la comunicación con los sensores de presión y temperatura se utilizará el RS485 conectado al receptor mediante el adaptador para el USB DSD TECH model:SH-U10 y a los sensores mediante el módulo MAX485.
- **Sensores:** este es el bloque de los sensores: el micrófono direccional, el acelerómetro, sensor de temperatura y de presión.

### 3.2. Descripción de dispositivos utilizados

El micrófono direccional t.bone EM 9600 [4], se caracteriza por ser sensible en una única dirección y relativamente sordo en las restantes. Su precio económico de 44 euros y la posibilidad de escuchar frecuencias entre 80 Hz y 17 [4] son la razón suficiente para utilizarlo en este trabajo, ya que se pretende analizar las frecuencias audibles entre 20 y 20 kHz.

Este sensor se utilizará principalmente para la medición de nivel ruido a unas determinadas frecuencias, pero también se puede utilizar para detectar fallos en rodamientos o engranajes.



Figura 3. 2. Micrófono direccional

La IMU MPU6050 [5] es un dispositivo barato que sirve para complementar este sistema y poder medir las frecuencias bajas (de 0 a 80Hz), que no podría analizar el micrófono direccional y también para comparar las lecturas de hasta 500 Hz. Su precio es de 3.99 euros y la frecuencia de muestreo máxima de 1000 Hz y acelerómetro en tres ejes.

No se ha podido realizar ensayos fiables con este dispositivo, pero queda como posible mejora para los futuros trabajos.



Figura 3. 3. MPU 6050

**Sensores de temperatura y de presión:** dispositivos conectados al sistema a través de un adaptador USB de RS485 llamado “USB DSD TECH model:SH-U10”. Debido a que los sensores se encontraban en la planta de la empresa, se ha simulado los dos con un Arduino Mega conectado al módulo MAX485.

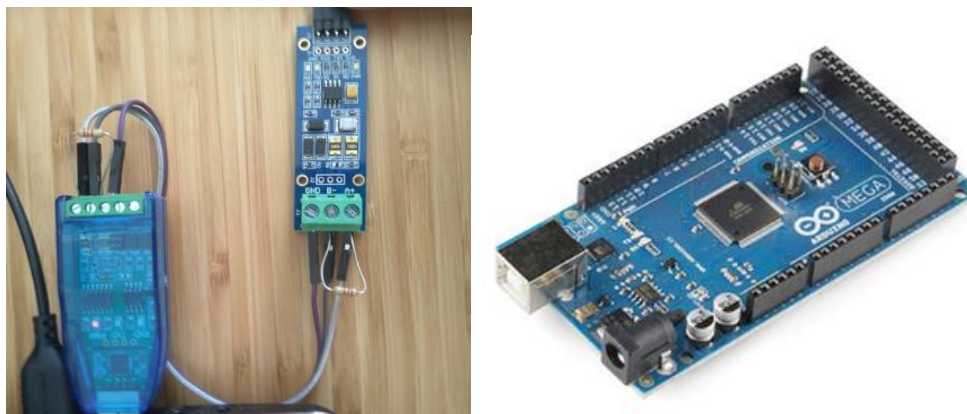


Figura 3. 4. USB DSD TECH model:SH-U10, un módulo MAX485 y Arduino Mega

**Xbee Pro S1 [6]:** son pequeños módulos electrónicos fáciles de configurar y capaces de comunicarse unos con otros de forma inalámbrica. Se utilizarán para el envío de datos. Poseen un protocolo de comunicación estándar IEEE 802.15.4 para crear redes FAST POINT-TO-MULTIPOINT (punto a multipunto) o para redes PEER-TO-PEER (punto a punto), por lo tanto, se suelen utilizar con bastante frecuencia dentro de la industria. Cuestan 29.75 euros.



Figura 3. 5. Xbee Pro S1

Cabe destacar que también se utilizaran:

- Tarjeta de sonido Ugreen, para conectar el micrófono a la BeagleBone
- Pro snake 17562/1,5 Audio Cable
- Trípode
- Adaptador USB de Xbee/PC



Figura 3. 6. Tarjeta de sonido, cable, tripode

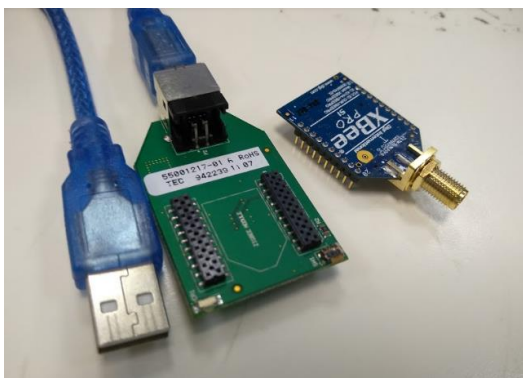


Figura 3. 7. Adaptador de Xbee/PC



## 4. ESTUDIO DE LAS VIBRACIONES

En este capítulo se estudiarán las vibraciones utilizando el micrófono direccional. Se basa en varios artículos científicos con una metodología muy parecida.

Los más destacables son [1] “Manual de Vibraciones”, [7] “Bearing Health Diagnosed with a Mobile Phone (2018)” y [8] “Rolling Element Bearing Fault Diagnosis - MATLAB & Simulink”.

### 4.1. Introducción y descripción del equipo utilizado

El análisis de vibraciones consiste en buscar cambios en los patrones y relacionarlos con los componentes mecánicos de la máquina.

Para llevarlo al cabo se ha utilizado una maqueta de sistema de elevación por cable con un pequeño motor trifásico asíncrono, conectado en estrella a la fuente de 400V. Diseñado para mover la carga de 80 kg a una velocidad constante y una reductora formada por 2 engranajes cónicos acoplados al motor y 6 engranajes de dientes rectos.



Figura 4. 1. Maqueta del sistema de elevación

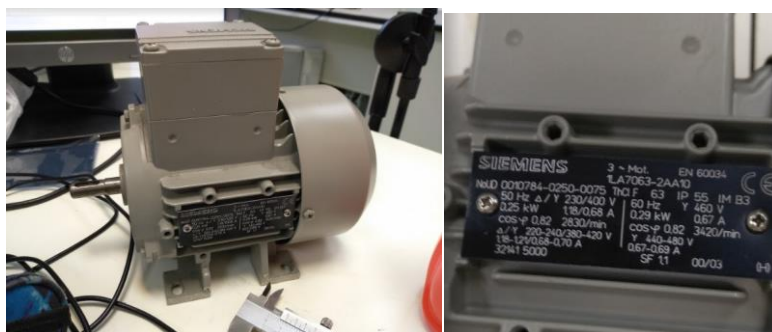


Figura 4. 2. Motor asíncrono Siemens/ Placa de características

**Tablas de selección**

Referencia (alternativa 1LA7 para versiones no disponibles como 1LE1)	Relación entre par de arranque y par nominal $M_A/M_N$	corriente arranque y corriente nominal $I_A/I_N$	par máximo y par nominal $M_e/M_N$	Clase par KL	Momento de inercia $J$ kgm <sup>2</sup>	Ruido		Máxima velocidad mecánica admisible $n_{max}$ min <sup>-1</sup>	Conductores en paralelo necesarios			
						Presión so- nora en sup. de medida 50 Hz $L_{p(A)}$ dB(A)	Potencia sonora 50 Hz $L_{WA}$ dB(A)		230 V	400 V	500 V	690 V
2 polos 3000 min <sup>-1</sup> a 50 Hz 3600 min <sup>-1</sup> a 60 Hz, clase F, utilización B, IP 55									aluminio			
1LA7 050-2AA□□▶	2	3,7	2,3	16	0,00015	41	52	6000				
1LA7 053-2AA□□▶	2,1	3,7	2,4	16	0,00015	41	52	6000				
1LA7 060-2AA□□▶	2	3,7	2,2	16	0,00018	49	60	6000				
1LA7 063-2AA□□▶	2	4	2,2	16	0,00022	49	60	6000				
1LA7 070-2AA□□▶	2,3	3,5	2,3	16	0,00029	52	63	6000				
1LA7 073-2AA□□▶	2,5	4,3	2,6	16	0,00041	52	63	6000				
1LA7 080-2AA□□▶	2,3	5,6	2,4	16	0,00079	56	67	6000				

Figura 4. 3. Datasheet del fabricante del motor

En la placa de características se puede comprobar de que se trata de un motor de 2 polos, con una velocidad de rotor de 2830 rpm. Es una velocidad ensayada en condiciones ideales, pero se verá que se parece mucho a las velocidades a la que girará el motor.

La búsqueda de los defectos en los engranajes y rodamientos se centrará en los dos primeros ejes. Eso es porque poseen la mayor velocidad que los demás, por lo tanto, se percibirá cualquier variación o anomalía con más facilidad.

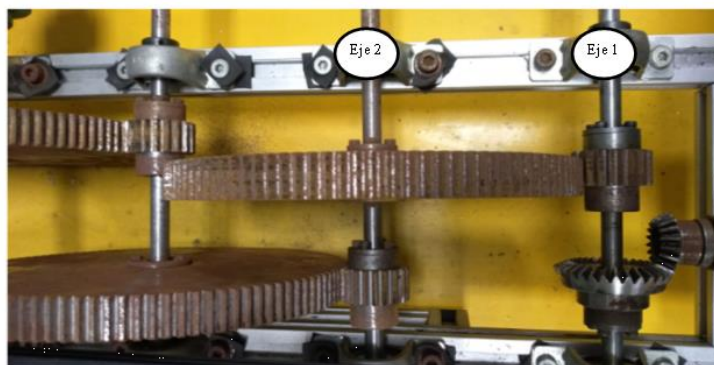


Figura 4. 4. Tren de engranajes utilizado en los ensayos

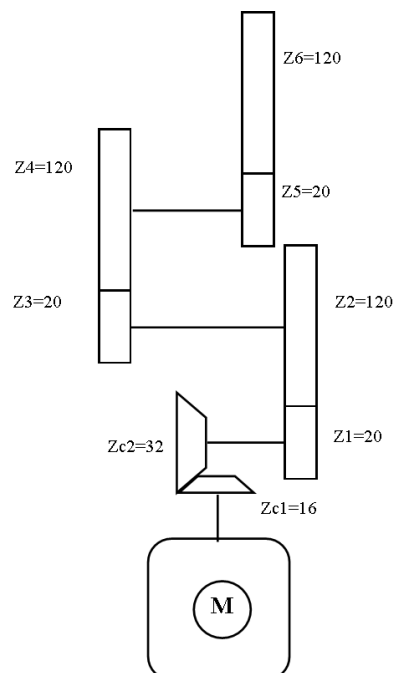


Figura 4. 5. Número de dientes de cada engranaje



Para calcular las velocidades de los ejes se utilizará la siguiente fórmula:

$$N_{salida} = \frac{Z_{entrada}}{Z_{salida}} * N_{entrada} \quad (7.1.)$$

Donde:

- N: es la velocidad
- Z: es el número de dientes

	Z entrada	Z salida	V entrada [RPM]	V salida [RPM]	V salida [Hz]
Eje 1	16	32	2830	1415	23,58
Eje 2	20	120	1415	235,83	3,93

Tabla 4. 1. Velocidades teóricas de los ejes



Figura 4. 6. Tacómetro utilizado para realizar las medidas de la velocidad de los ejes

Para comprobar la validez de los cálculos se ha hecho la media de 3 muestras recogidas en cada eje.

Numero de muestras	Eje 1	Eje 2
1	1470.00	248.4
2	1479.00	248.4
3	1468.00	250.7
<b>Velocidad media [RPM]</b>	<b>1472.33</b>	<b>249.16</b>
<b>Velocidad media [Hz]</b>	<b>24.54</b>	<b>4.15</b>

Tabla 4. 2. Media de velocidades medidas con el tacómetro en el Eje 1 y el Eje 2

Se puede comprobar que la velocidad medida con el tacómetro se parece bastante a la teórica. Esto se traduce en que la carga casi no afecta a nuestro sistema. Parece ser algo idóneo, pero no lo es ya que va a dificultar la búsqueda de los defectos en el sistema.

#### 4.2. Análisis de defectos en los engranajes

El análisis de vibraciones, tal y como se ha comentado antes, puede estudiar el estado de los engranajes. Con un tratamiento de señal adecuado es capaz de encontrar: una inapropiada relación entre el número de dientes, excentricidad, montaje en árboles deformados, holguras, desalineación, sobrecarga y la fractura o deterioro de los dientes.

Para poder realizar este estudio se necesitarán los siguientes datos:

- Velocidad de entrada de salida del eje.
- Número de dientes de cada engranaje.

A partir de ellos se pueden calcular las siguientes frecuencias:

- Frecuencia de engrane (GMF) [1,8,60]: aparece siempre en el espectro de frecuencias, independientemente de estado de los engranajes. Se calcula multiplicando el número de dientes por la velocidad de su eje.

$$GMF = Z_{entrada} * N_{entrada} = Z_{salida} * N_{salida} \quad (7.2.)$$

- Bandas laterales de la frecuencia de engrane [1,8,60]: son frecuencias que aparecen en los laterales de la GMF, son muy importantes ya que son el índice de un fallo. Se corresponden con las velocidades de giro de piñón y de la corona.
- Fase de ensamble (NE) [1,8,61]: diferentes modos de engranar que tiene un par de ruedas dentadas. Se puede calcular gráficamente o realizando el máximo común divisor (MCD) del piñón y la corona.
- Frecuencia de repetición de diente (FRD) [1,8,60-61]:

$$FRD = \frac{GMF * NE}{Z_{entrada} * Z_{salida}} \quad (7.3.)$$

- Frecuencia de fase de ensamblaje (FFE) [1,8,61]: consecuencia del desgaste, indica que el espacio entre dientes y su perfil ha cambiado.

$$FFE = GMF * NE \quad (7.4.)$$

#### **4.2.1. CÁLCULO DE LAS FRECUENCIAS DE DESGASTE**

Para estos cálculos utilizaremos las velocidades medidas con el tacómetro 24.54Hz y 4.15 Hz para eje 1 y el eje 2 respectivamente.

$$N_{Eje1} = 24.54 \text{ Hz}; N_{Eje2} = 4.15 \text{ Hz}; Z_{piñon 1} = Z_{piñon 2} = 20 \text{ dientes}; \quad (7.5.)$$

$$Z_{Corona1} = Z_{Corona2} = 120 \text{ dientes} \quad (7.6.)$$

$$GMF_{Eje1} = Z_{piñon1} * N_{Eje1} = 24.54 * 20 = 490.8 \text{ Hz} \quad (7.7.)$$

$$GMF_{Eje2} = Z_{piñon2} * N_{Eje2} = 4.15 * 20 = 83 \text{ Hz} \quad (7.8.)$$

$$NE_{Eje1} = NE_{Eje2} = MCD\{20; 120\} = 20 \quad (7.9.)$$

$$FRD_{Eje1} = \frac{490.8 * 20}{120 * 20} = 4.09 \text{ Hz} \quad (7.10)$$

$$FRD_{Eje2} = \frac{83 * 20}{120 * 20} = 0.692 \text{ Hz} \quad (7.11)$$

$$FFE_{Eje1} = 490.8 * 20 = 9816 \text{ Hz} \quad (7.12)$$

$$FFE_{Eje2} = 83 * 20 = 1660 \text{ Hz} \quad (7.13)$$

Todas estas frecuencias indican alrededor de que valores pueden aparecer los defectos y también el tipo de equipo que hay que utilizar para estudiarlas.

#### **4.2.2. BÚSQUEDA DE DEFECTOS EN EL ESPECTRO DE FRECUENCIAS**

Al realizar lecturas espectrales sobre un engrane hay que asegurarse, que estas frecuencias (GMF) aparecen en el espectro de las frecuencias. Parece ser algo obvio, pero si no se pueden encontrar el estudio no va a ser válido.



La siguiente tabla recoge los anchos de bandas donde suelen aparecer las frecuencias de los defectos. Hay que recordar que los valores calculados antes son aproximaciones teóricas. [1,8,63]

Parámetro espectral	Ancho de banda
Desequilibrio	0,3x a 1,5x RPM
Desalineación	1,5x a 2,5x RPM
Holguras	2,5x a 10,5x RPM
Picos fantasmas y bandas laterales del primer armónico de la frecuencia de engrane.	10x RPM a $GMF - 5x$ RPM
Primer armónico de la frecuencia de engrane.	$GMF - 5x$ RPM a $GMF + 5x$ RPM
Bandas laterales del primer y segundo armónicos de la frecuencia de engrane, frecuencias de rodamiento.	$GMF - 5x$ RPM a $2GMF + 5x$ RPM
Segundo armónico de la frecuencia de engrane.	$2GMF - 5x$ RPM a $2GMF + 5x$ RPM
Estado general del reductor y rodamientos. Lubricación del rodamiento y reductor.	1 kHz a 20 kHz

Tabla 4. 3. Bandas espectrales para la detección de problemas en engranajes [1,8,63]

### 4.2.3. EJE DESALINEADO

#### 4.2.3.1. SÍNTOMAS DEL EJE DESALINEADO

Aparece debido a que las ruedas de los engranajes fueron mal ensambladas, con errores de alineación o cuando sus ejes no están paralelos.

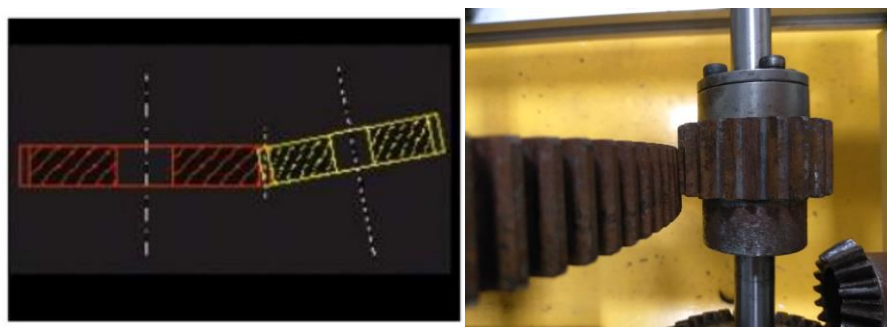


Figura 4. 7. Ejes desalineados

Síntomas [1,8,65]:

- Aumento de la amplitud en  $2xGMF$
- Picos en otros armónicos  $GMF$  ( $1x$ ,  $3x$ , etc.)
- Importantes bandas laterales en  $2x GMF$  y en  $1x$  o incluso  $2x RPM$ .
- Armónicos de la velocidad de giro de cada eje en  $2x$  e incluso  $3x RPM$ .

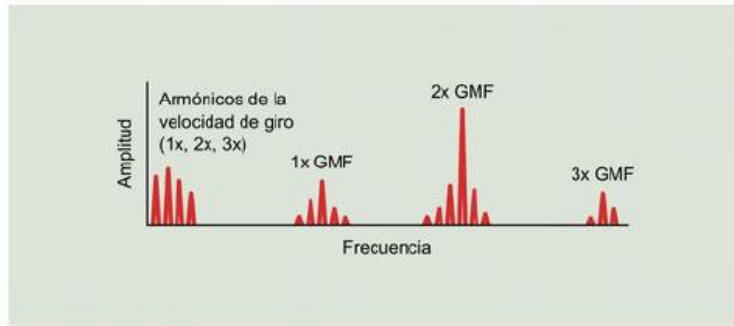


Figura 4. 8. Desalineación de un engranaje, figura de [1,8,65]

#### 4.2.3.2. BÚSQUEDA DE DEFECTOS:

Primero hay que asegurarse que la frecuencia de engrane se encuentra dentro del espectro, Ecuación (4.7).

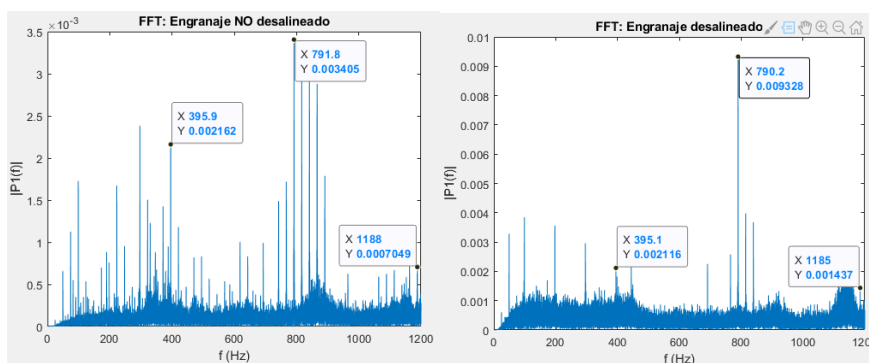


Figura 4. 9. GMF de engranaje NO desalineado y desalineado (1x, 2x, 3x)

Tanto las GMFs de engranaje sano, como las del engranaje desalineado aparecen en el espectro de frecuencias, con un valor de 396 Hz y sus 3 primeros armónicos (x1, x2, x3). El primer armónico está dentro de ancho de banda definido por la tabla 4.3.

$$[GMF - 5x RPM; GMF + 5x RPM] = [367.5; 612.5] \quad (7.14.)$$

#### 4.2.3.3. BÚSQUEDA DE LA VELOCIDAD DEL EJE:

Uno de los índices de la desalineación es el aumento de la amplitud de la velocidad del eje. Para buscarla hace falta utilizar la **envolvente** de la señal o la demodulación.

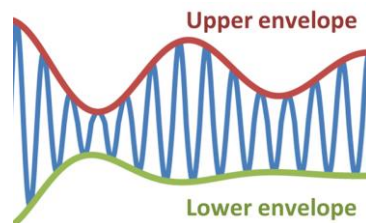


Figura 4. 10. Envolvente de la señal

**Envolvente o la demodulación [1,9,75]:** es una curva suave que describe una señal envolviendo sus picos. Es una excelente componente para mejorar el análisis espectral en la detección de los defectos en rodamientos y cajas de engranajes, porque elimina las frecuencias naturales y otras que aportan información no relevante para el estudio.

En el caso del audio o vibraciones recogidas por el acelerómetro se trata de medidas reales, una forma de calcularla es haciendo el valor absoluto de la transformada de **Hilbert** [8], quitando así la parte imaginaria que no es relevante.

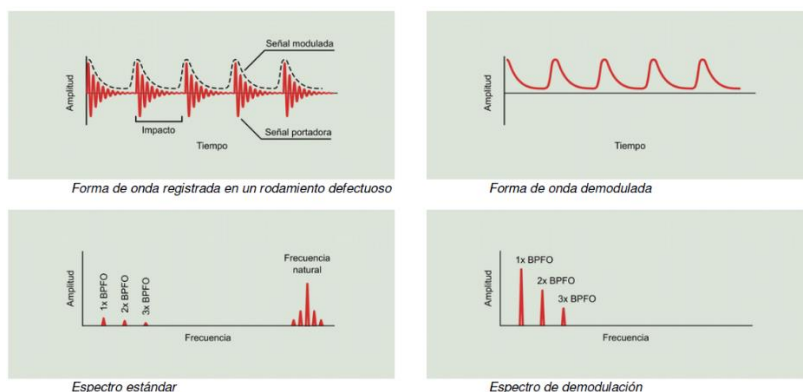


Figura 4. 11. Comparación de estudio de espectro de la frecuencia utilizando solamente FFT y FFT + envolvente. Figura sacada de [1,9,75]

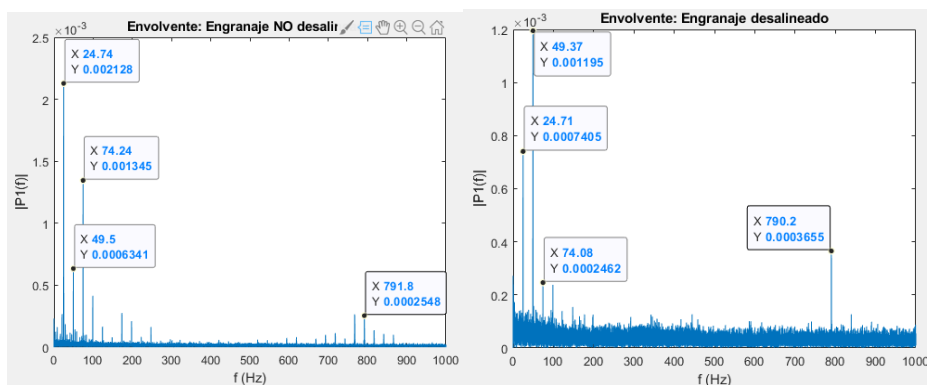


Figura 4. 12. Comparación de amplitudes de velocidades en engranaje sano/ desalineado

Las lecturas de tacómetro recogidas en la tabla 4.2 valía aproximadamente 24.52 Hz, un valor parecido el que muestra nuestro espectro de frecuencias de 24.74 Hz. Se puede comprobar que la amplitud de los armónicos de la velocidad va aumentando. También aparece el segundo armónico de la GMF (790.2 Hz), con bastante amplitud, otro índice del desalineamiento de los engranajes.

#### 4.2.3.4. ESTUDIO DE LOS ARMÓNICOS DE GMF Y SUS BANDAS LATERALES

En la Figura 4.13 se puede apreciar que la señal defectuosa (la de la derecha) es mucho más ruidosa. Eso se debe a que los engranajes con defectos de desalineación chocan con mucha más fuerza lo que produce muchísimo más ruido.

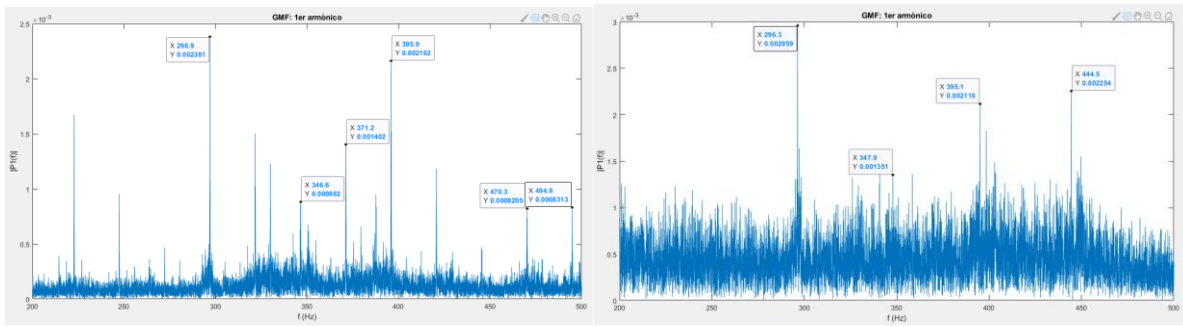


Figura 4. 13. Comparación de señales de engranaje sano y otro defectuoso

Para calcular las bandas laterales de las GMF se utilizan los valores encontrados en la gráfica correspondientes a 395 Hz y de la velocidad del eje 24.5 Hz.

$$\text{Bandas laterales GMF} = \text{GMF} \pm \text{Frecuencia giro del eje} \quad (7.15.)$$

Numero armónico	Banda lateral inferior (Hz)	Banda lateral superior (Hz)
1	370,5	419,6
2	346	444,1
3	321,5	468,6
4	297	493,1

Tabla 4. 4. Bandas laterales de 1er GMF

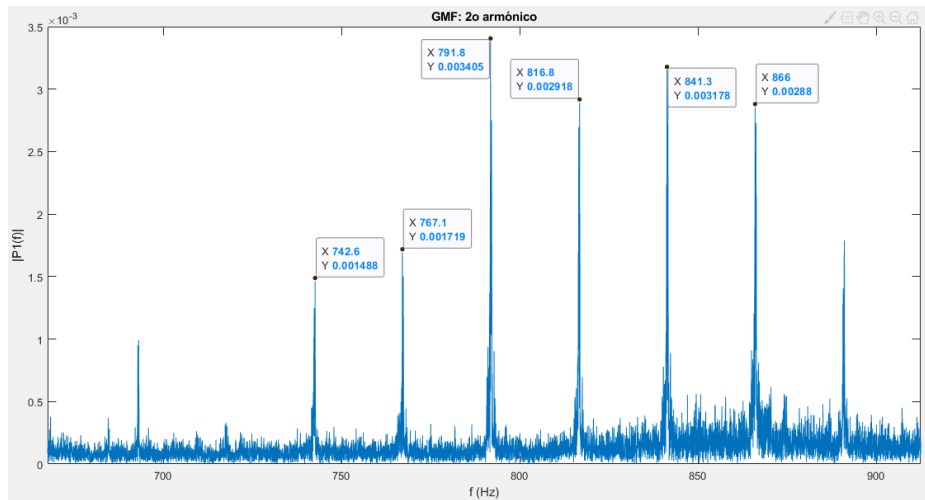


Figura 4. 14. 2xGMF de 792.2 Hz engranaje sano y sus respectivas bandas laterales

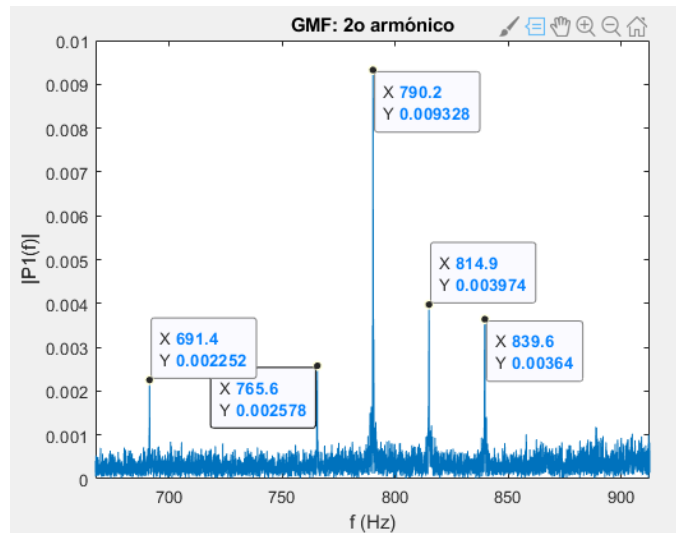


Figura 4. 15. 2xGMF de 790.2 Hz engranaje defectuoso y respectivas bandas laterales

Numero armónico	Banda lateral inferior (Hz)	Banda lateral superior (Hz)
1	767,48	816,92
2	742,76	841,64
3	718,04	866,36

Tabla 4. 5. Bandas laterales de 2xGMF

Se puede comprobar que la magnitud de segundo armónico de GMF ha aumentado considerablemente, pasando de  $1.1 \times 10^{-3}$  (engranaje bien alineado) a  $9.3 \times 10^{-3}$  (engranaje desalineado). También aparecen sus bandas laterales.

Este análisis confirma la existencia de un defecto de desalineación, ya que se cumplen todos sus síntomas: el aumento de 2xGMF, picos de los armónicos 1x y el 3x, importantes bandas laterales alrededor de 1x y 2x y varios armónicos de la velocidad del eje.

#### 4.2.4. DESGASTE DEL DIÁMETRO PRIMITIVO O DIENTE ROTO:

##### 4.2.4.1. SÍNTOMAS

El desgaste de los engranajes o rotura de diente provocan picos importantes en la GMF y sus armónicos. También aparecen frecuencias correspondientes a las frecuencias de resonancia del piñón de la corona, debido al rozamiento entre los engranajes.

Síntomas [1,8,64]:

- Aumento de la amplitud en 1x, 2x y/o 3x GMF.
- Bandas laterales importantes en torno a 1x, 2x y/o 3x GMF a la frecuencia 1x RPM de la rueda desgastada.
- Picos correspondientes a la frecuencia de resonancia del piñón o la corona.
- Bandas laterales a 1x RPM de la rueda desgastada alrededor de la frecuencia de resonancia.

**4.2.4.2. BÚSQUEDA DEL DEFECTO**

Se compararán engranes iguales que girarán a velocidades diferentes, uno con un piñón con el diente roto y otro con uno sano.

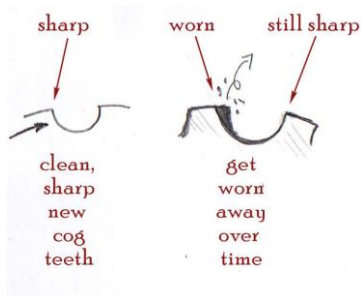


Figura 4. 16. Diente desgastado

Primero hay que comprobar si GMFs aparecen en los espectros, junto con sus armónicos Figura 4.17.

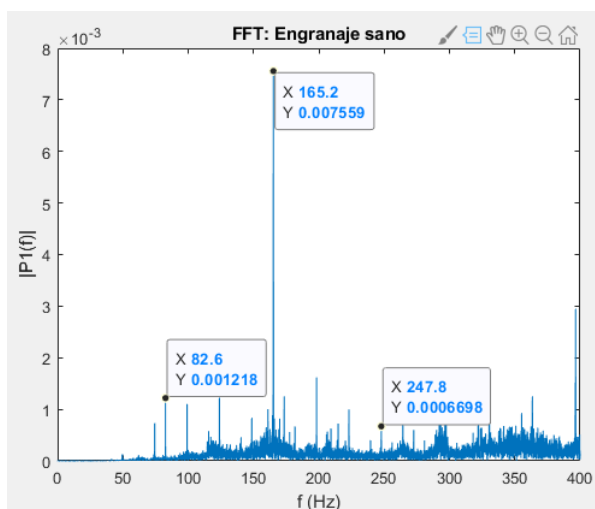


Figura 4. 17. GMF engrane del eje 2 igual a 82.6 Hz (1x, 2x, 3x)

No del armónico	GMFs eje 2
1	83
2	166
3	249

Tabla 4. 6. GMFs del eje 2

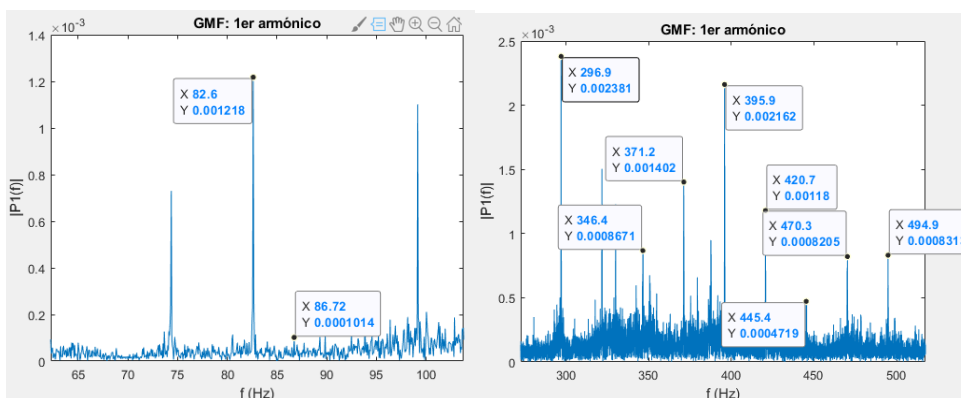


Figura 4. 18. GMF eje 2 engranaje sano/GMF eje 1 engranaje defectuoso

En la siguiente Figura 4.19. se puede comprobar que alrededor del engranaje sano casi no aparecen las bandas laterales ( $82.6\text{Hz} \pm 4.15\text{Hz}$ ), en cambio sí que existen varias alrededor de engranaje con el diente desgastado ( $395.9\text{ Hz} \pm 24.5\text{ Hz}$ ). Lo mismo pasa en los GMFs restantes.

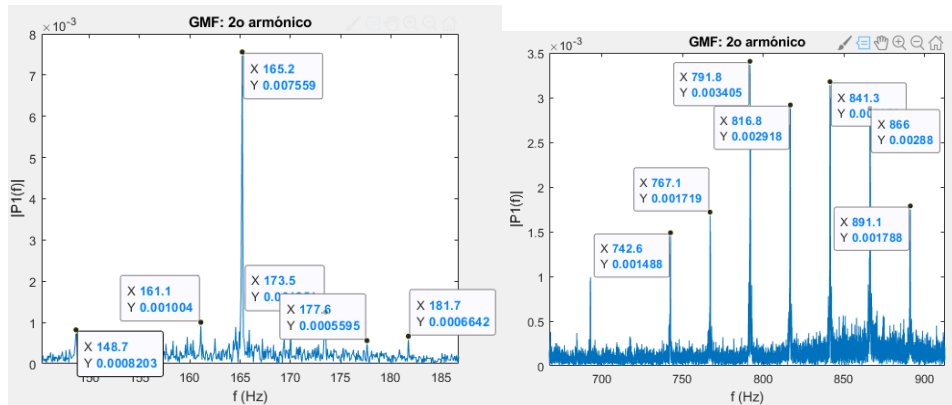


Figura 4. 19. GMF eje 2 engranaje sano/GMF eje 1 engranaje defectuoso

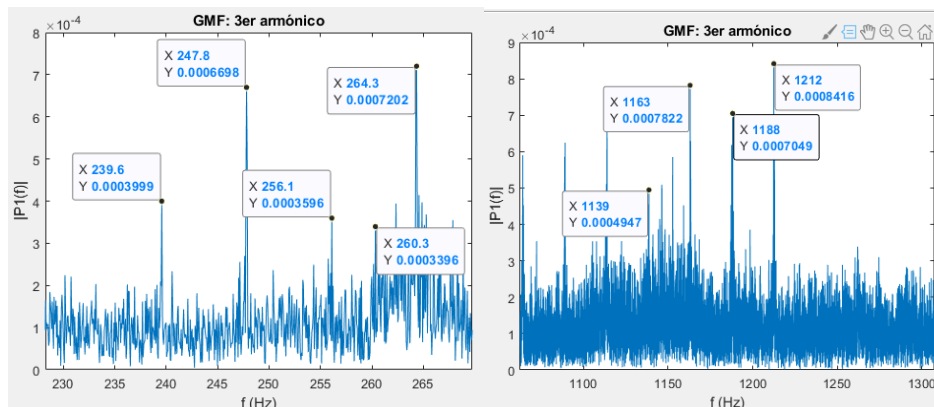


Figura 4. 20. GMF eje 2 engranaje sano/GMF eje 1 engranaje defectuoso

La aparición de bandas laterales indica el de desgaste en los engranajes. De hecho, el engranaje “sano” correspondiente al eje 2 también los presenta. Esto se debe a que también esta desgastado, ya que ninguno de ellos está en perfecto estado.

Las bandas se han calculado con las velocidades de la Tabla 4.2. para eje 1 y el eje 2 respectivamente.

#### 4.2.5. ENGRANAJE EXCÉNTRICO O EJE DOBLADO

##### 4.2.5.1. SÍNTOMAS

El engranaje excéntrico aparece cuando el centro de la simetría no coincide con el centro de rotación.

El eje doblado se debe a la deformación de este debido al peso de los engranajes.



Figura 4. 21. Excentricidad/ Eje doblado

Síntomas [1,8,64]:

- Aumento de la amplitud en 1x, 2x y/o 3x GMF.
- Bandas laterales importantes en torno a 1x, 2x y/o 3x GMF a la frecuencia 1x RPM de la rueda causante del problema.
- Aumento de la amplitud en 1x RPM de la rueda que presenta el problema y, si se trata de un problema grave, aparecerán además armónicos.

#### 4.2.5.2. BÚSQUEDA DE DEFECTOS

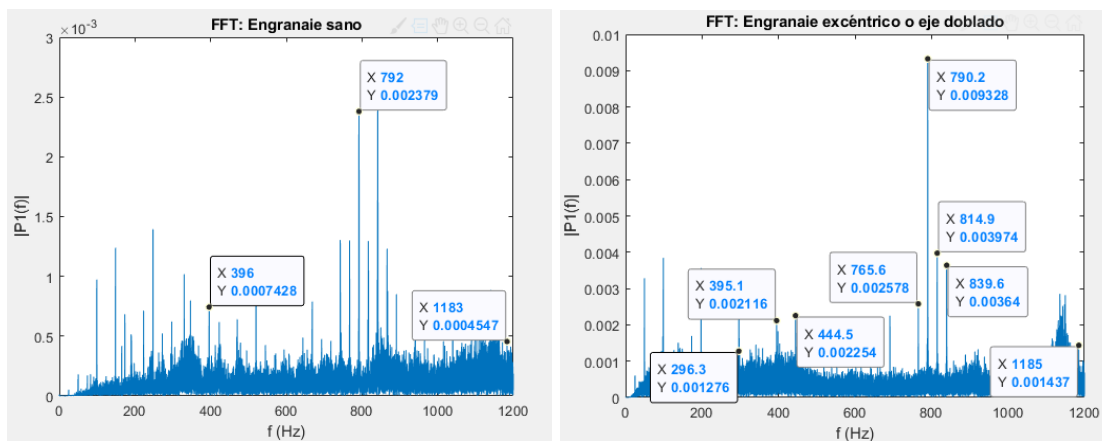


Figura 4. 22. Comparación entre el engranaje sano y engranaje excéntrico o con el eje doblado

Se puede comprobar de que hay un considerable aumento de amplitud en el engranaje defectuoso y aparecen varias bandas laterales con amplitudes altas.

### 4.3. Rodamientos:

#### 4.3.1. HIPÓSTASIS DE PARTIDA Y EL ALCANCE

Dentro de los elementos que suelen fallar con más regularidad dentro de los motores asíncronos son los rodamientos (41% de las veces). Por lo tanto, es de especial interés monitorizar su comportamiento.



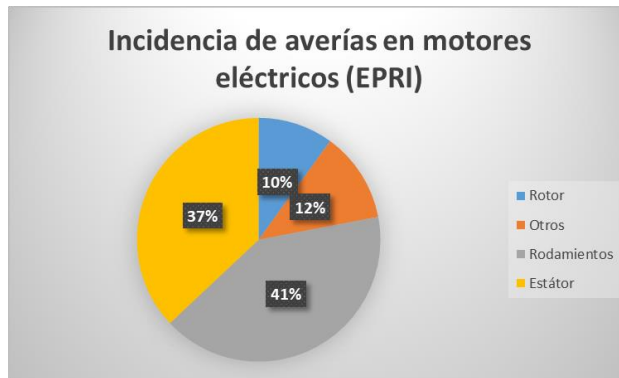


Figura 4. 23. Averías dentro de los motores asíncronos

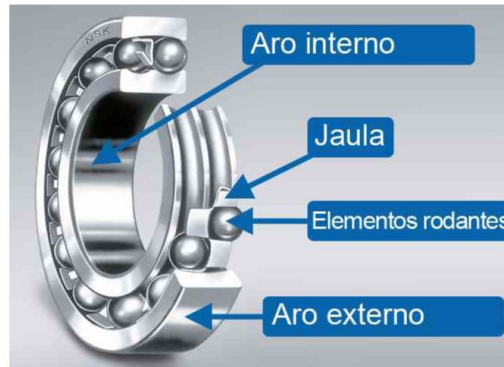


Figura 4. 24. Partes de rodamientos

#### 4.3.2. FRECUENCIAS DE FALLO EN LOS RODAMIENTOS

Existen 4 tipos de fallos en los rodamientos [1,9,69]:

- BPFO (Ball Pass Frequency Outer) o frecuencia de deterioro de la pista exterior
- BRFI (Ball Pass Frequency Inner) o frecuencia de deterioro de la pista interior.
- BSF (Ball Spin Frequency) o frecuencia de deterioro de los elementos rodantes.
- FTF (Fundamental Train Frequency) o frecuencia de deterioro de la jaula.

Para calcularlas se utilizan las siguientes formulas [1,69,9]

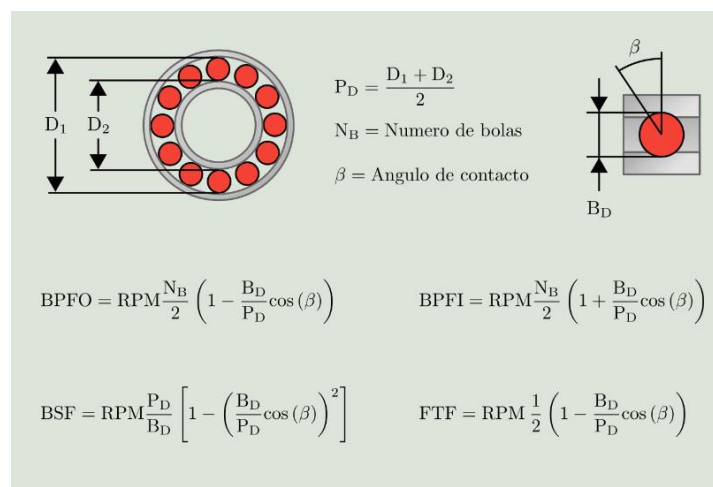


Figura 4. 25. Frecuencias de fallo de un rodamiento, figura sacada de [1,9,69]

En general las frecuencias de fallo de rodamientos serán no síncronas, no coincidirán con la frecuencia del giro del eje.

Esto no quiere decir que serán fáciles de distinguir unas de otras, ya que todas ellas dependen del ángulo de contacto de las bolas que pueden variar debido a: desalineación, dilataciones térmicas, aprietes excesivos, picaduras o desconchamientos en las pistas, etc. Por lo tanto hay que permitir cierto margen de error a la hora de calcularlas, sobre todo si se realizara un programa que los busque de forma automatizada.

#### 4.3.3. CÁLCULO DE LAS FRECUENCIAS DE DETERIORO DE RODAMIENTOS:

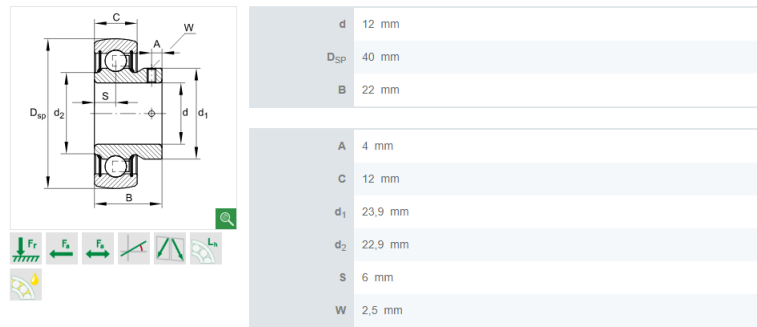


Figura 4. 26. Rodamientos insertables AY12-XL-NPP-B

Los rodamientos utilizados en los ensayos son AY12-XL-NPP-B.

Los datos para calcular las frecuencias de deterioro son: diámetro de las bolas, número de bolas, diámetro de aro interno y externo, ángulo de contacto (alfa). En Figura 4.26

$$d_{Bola} = 6.7 \text{ mm}; N_{OBolas} = 8; D_{Aro\ interno} = d_2 = 22.9 \text{ mm}; \alpha = 0 \text{ grados}; \quad (7.16.)$$

$$D_{Aro\ exteno} = D_{Aro\ interno} + 2 * d_{Bola} = 22.9 + 2 * 6.7 = 36.3 \text{ mm} \quad (7.17.)$$

$$PD = \frac{D_{Aro\ interno} + D_{Aro\ exteno}}{2} = \frac{22.9 + 36.3}{2} = 29.6 \text{ mm} \quad (7.18.)$$

$$F_{Giro\ eje} = 24.5 \text{ Hz} \quad (7.19.)$$

$$BPFO = \frac{N_{OBolas} * F_{Giro\ eje}}{2} * \left(1 - \frac{d_{Bola}}{PD} * \cos(\alpha)\right) = 3.095 * F_{Giro\ eje} = 75.83 \text{ Hz} \quad (7.20.)$$

$$BPFI = \frac{N_{OBolas} * F_{Giro\ eje}}{2} * \left(1 + \frac{d_{Bola}}{PD} * \cos(\alpha)\right) = 4.9054 * F_{Giro\ eje} = 120.16 \text{ Hz} \quad (7.21.)$$

$$BSF = \frac{PD}{d_{Bola}} * F_{Giro\ eje} * \left[1 - \left(\frac{d_{Bola}}{PD} * \cos(\alpha)\right)^2\right] = 4.19156 * F_{Giro\ eje} = 102.69 \text{ Hz} \quad (7.22.)$$

$$FTF = \frac{F_{Giro\ eje}}{2} * \left(1 - \frac{d_{Bola}}{PD} * \cos(\alpha)\right) = 0.3868 * F_{Giro\ eje} = 9.48 \text{ Hz} \quad (7.23.)$$

Igual que con los engranajes, todas estas frecuencias indican alrededor de que valores pueden aparecer los defectos y también el tipo de equipo que hay que utilizar para estudiarlas.

Por ejemplo, para encontrar la FTF dentro del espectro de frecuencias se necesita un equipo capaz de medir frecuencias muy bajas, un acelerómetro, por ejemplo.

Muchas veces no se dispondrá de los datos tan detallados acerca de los rodamientos, como el ángulo de contacto, diámetros de aros internos, etc. Para solucionar este problema existen fórmulas empíricas, que se aproximan bastante a los resultados calculados anteriormente.

$$FTF = 0.4 * F_{Giro\ eje} = 9.8\ Hz \quad (7.24.)$$

$$BPFO = 0.4 * N_{oBolas} * F_{Giro\ eje} = 78.4\ Hz \quad (7.25.)$$

$$BPFI = 0.6 * N_{oBolas} * F_{Giro\ eje} = 117.6\ Hz \quad (7.26.)$$

Al analizar el espectro del sistema hay que tener en cuenta las siguientes consideraciones [1,9,70-71]:

- Los fallos suelen originarse en el siguiente orden: pista exterior, pista interior, elementos rodantes y finalmente en la jaula. Esto ocurre siempre y cuando el rodamiento haya sido correctamente montado.
- Los rodamientos con defecto en jaula exterior suelen tener mayor vida útil que los de pista interior, por lo tanto, es muy importante determinar si tenemos una BPFI.
- Hay que asegurarse de colocar el sensor cerca del rodamiento que queremos analizar ya que las energías de la transmisión debida a los defectos son pequeñas. También que nuestro sensor es capaz de registrar señales que producen los impactos al pasar por la zona de carga, al menos las frecuencias de revolución de eje (en nuestro caso 44100 Hz es más que suficiente).

Para los ensayos se han utilizado rodamientos con fallos inducidos de BPFO, BPFI y FTF, utilizados en el laboratorio de mecánica de José Manuel Sierra Velasco [10,1,26-27].



Figura 4. 27. Defectos de tipo BPFI, BPFO, FTF, figuras sacadas de [10,1,26-27].

#### **4.3.4. DETECCIÓN DE DEFECTOS DE TIPO BPFI**

Los defectos BPFI son con los que hay que tener especial cuidado ya que acortan de forma exponencial la vida útil de los rodamientos y por lo tanto del sistema.

Para buscarlos se va a utilizar la **envolvente** de la señal (**Hilbert [8]**). Porque al igual que en los engranajes estos tipos de señales no solamente contienen la información sobre los defectos, también aparecen las frecuencias naturales de más componentes del sistema que van a

enmascarar las lecturas de interés. Si se trabaja con un sensor como el micrófono direccional podrían aparecer aún más datos como por ejemplo las frecuencias de fluorescentes, aproximadamente 16 kHz. Por lo tanto, es importante filtrarlos y sacar lo más relevante.

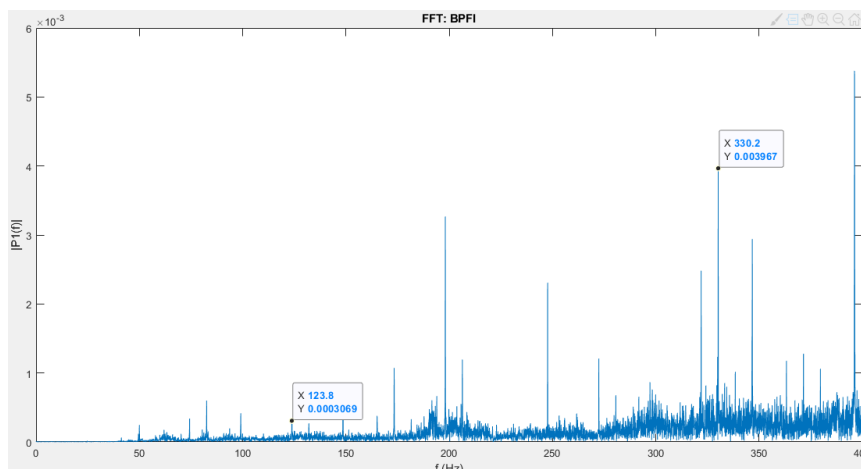


Figura 4. 28. FFT de la señal

En la Figura 4. 27 se ve que resulta difícil identificar con facilidad el defecto correspondiente a BPII utilizando solamente la FFT. Aparecen más señales con mucha mayor magnitud, de hecho, el pico de 123.8 Hz se corresponde a uno de los armónicos de la velocidad del eje y podría ser confundido con facilidad con la del defecto.

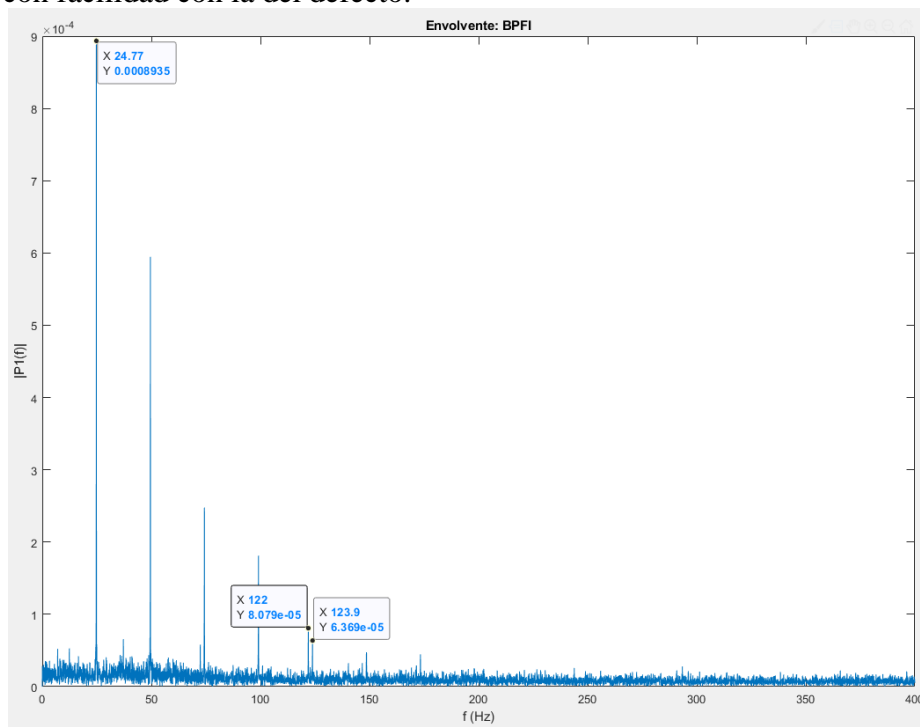


Figura 4. 29. Envoltura de la señal

Utilizando la envoltura de la señal se han filtrado los datos no deseados. Se puede identificar el pico de 122 Hz correspondiente con el valor calculado en la Ecuación 4.21.

Solo se ha identificado un armónico del defecto, por lo tanto, hace falta seguir filtrando la señal hasta dar con el defecto.

#### 4.3.4.1. CURTOSIS

La **curtosis** es una variable estadística, que proporcionara la información sobre la impulsividad de la señal o el peso de una variable aleatoria.

Se va a demostrar su utilidad comparado dos señales de un rodamiento con BPFO y otro sin ningún defecto.

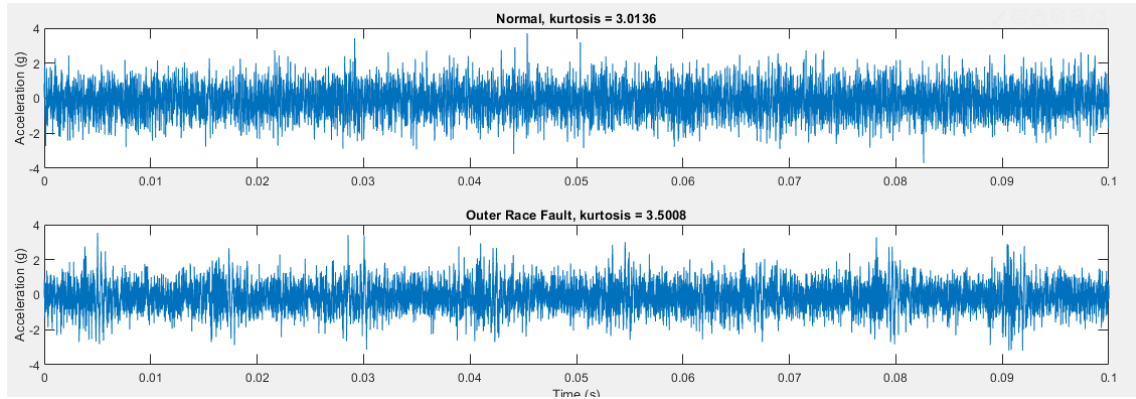


Figura 4. 30. Curtosis de dos señales, figura sacada de [8]

Los valores próximos a 3 siguen una distribución normal (definición de Pearson), en cambio los valores mayores que 3, en este caso 3.5008 indican la impulsividad de la señal y posibilidad de un fallo en alguno de los componentes del rodamiento.

En el caso del presente prototipo, si se realiza la curtosis de una señal recogida con el micrófono:

- En un rodamiento sano, curtosis de: **3.8551**
- En un rodamiento defectuoso, curtosis de: **5.7389**

Las dos indican impulsividad, esto es debido a señales no deseadas que recoge el micrófono. Pero conociendo el valor de curtosis de un rodamiento sano puede servir de umbral para realizar el estudio.

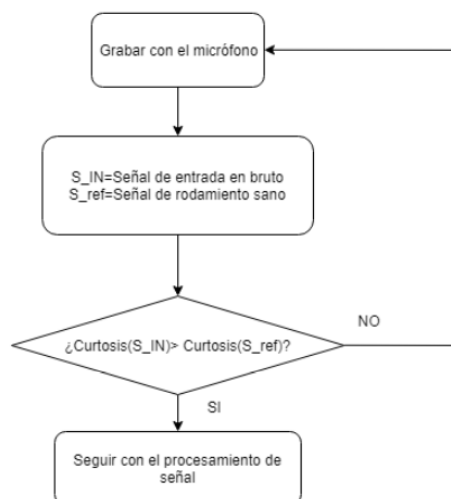


Figura 4. 31. Flujograma del inicio de análisis de datos para encontrar BPFI

Curtosis es un buen índice de la impulsividad, pero se necesita filtrar más información que permita determinar el defecto con certeza. Para esto se utilizará el Curtograma.

**4.3.4.2. EL CURTOGRAMA**

El **Curtograma** es una herramienta muy poderosa, que realiza la computación de la curtosis local y es capaz de determinar el ancho de banda que contiene la impulsividad más alta. En pocas palabras saca la Frecuencia central y el ancho de banda de las señales con más impulsividad.

Con esta información se puede crear un filtro para la señal en bruto.

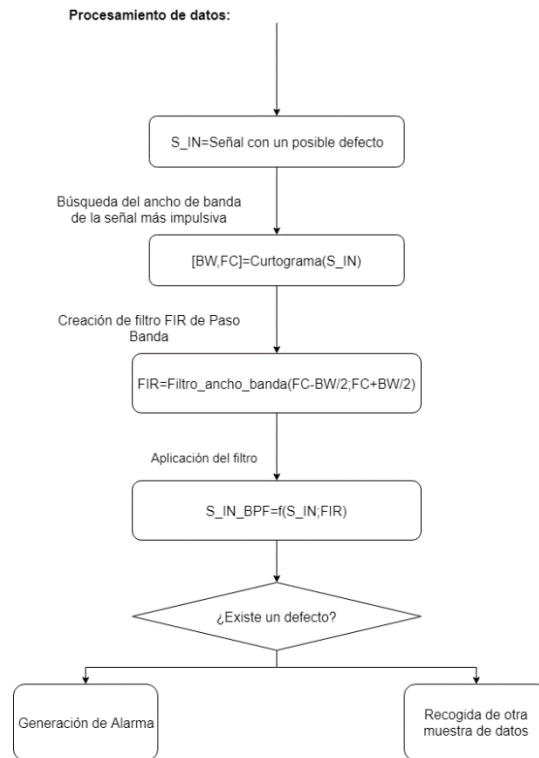


Figura 4. 32. Flujograma de Procesamiento de datos para BPF

A continuación, se ilustra el Curtograma para comprender mejor el concepto.

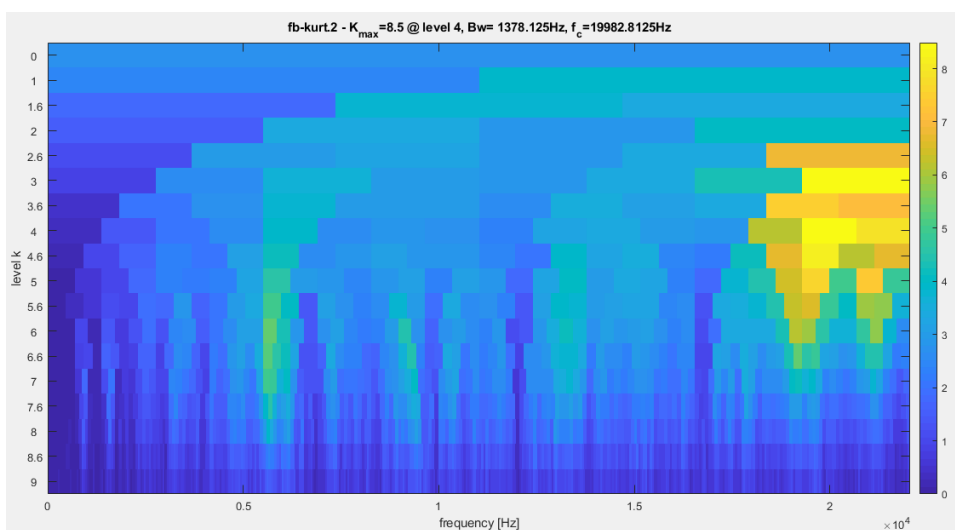


Figura 4. 33. Curtograma de BPF

El curtograma divide la señal en pequeñas ventanas, cuyo tamaño se define con un numero. En este caso particular 9 partes, que corresponde a 1024 bits. Puede darse el caso que la señal sea

demasiado corta y no admita tantas divisiones. La solución sería grabar durante más tiempo o disminuir la división a un nivel más bajo.

En la Figura 4.33 se puede apreciar que el curtograma indica una frecuencia central con máxima impulsividad de  $F_c=19.982$  kHz y un ancho de banda de  $BW=1378$  Hz.

Con estos datos se crea un filtro de tipo FIR, Filtro de Respuesta Finita al Impulso, (los IIR filtran mal las señales impulsivas por lo tanto el BPF no sería visible) y se aplica a la señal en bruto, recogida por el micrófono.

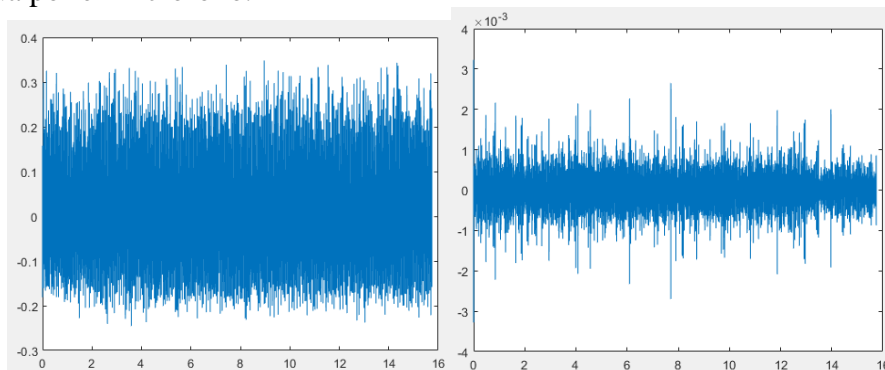


Figura 4. 34. Señal sin filtrar/ señal filtrada

La señal filtrada tiene muchísimo mejor aspecto, lo que ayudara a encontrar los defectos con la FFT de la envolvente de la señal.

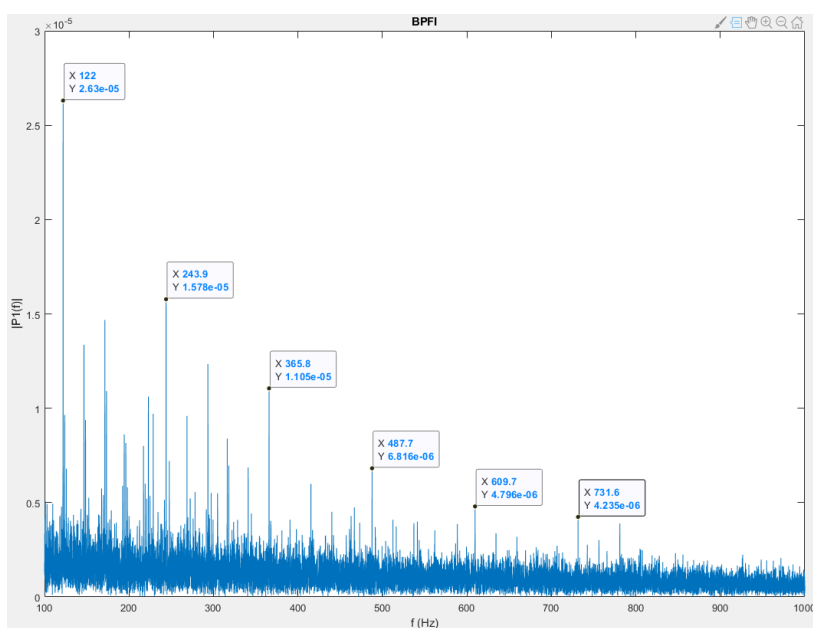


Figura 4. 35. BPF encontrados

En la Figura 4.35 aparecen 6 armónicos correspondientes a la BPF, obtenidos con Filtro paso banda + Envolvente + FFT.

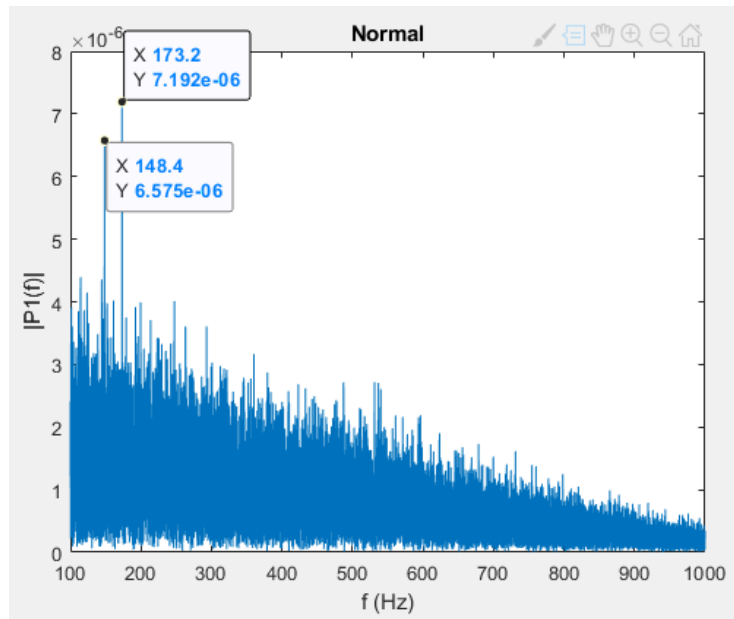


Figura 4. 36. Rodamiento sin fallo

En la Figura 4.36 Rodamiento sin fallo, no aparece ningún armónico de fallo. Solamente las frecuencias correspondientes a la velocidad del eje.

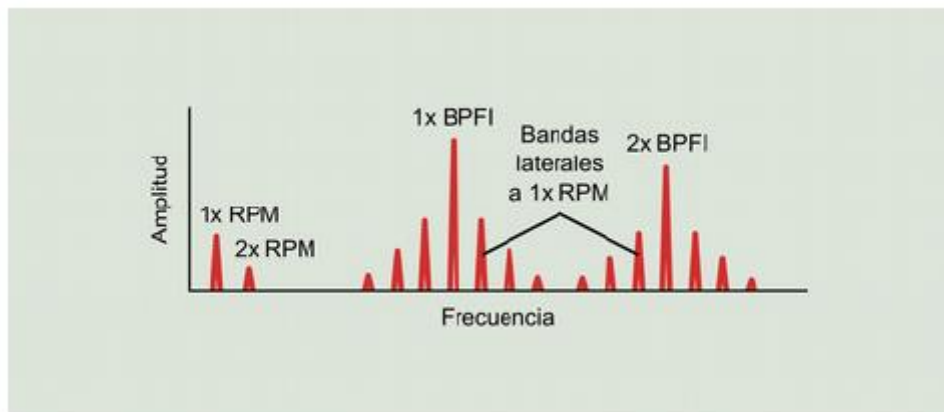


Figura 4. 37. Diagnóstico de BPF, figura sacada de [1,9,72]

Otro índice de un defecto en la pista interior es la aparición de las bandas laterales alrededor de los armónicos de BPF, los mismos que aparecen en nuestro el espectro.



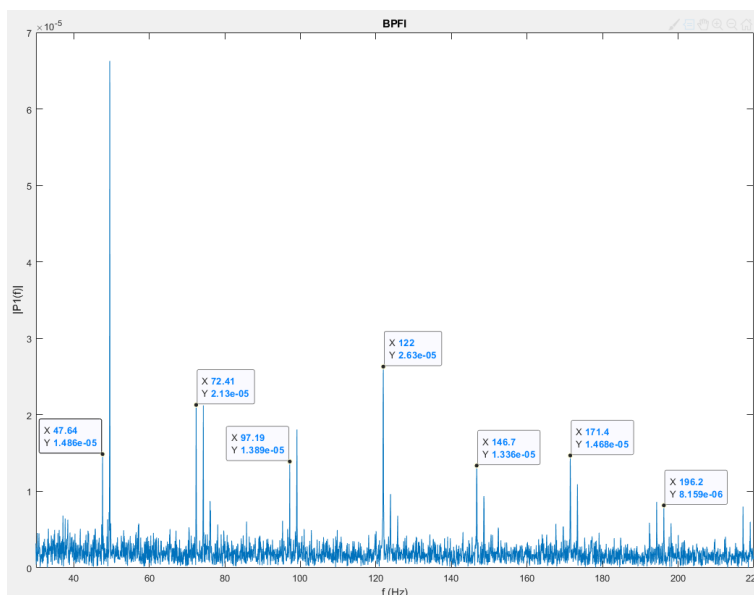


Figura 4. 38. Bandas laterales de 122 Hz +-Frecuencia eje x1, x2, x3

Numero armónico	Banda lateral inferior (Hz)	Banda lateral superior (Hz)
1	97,3	146,7
2	72,6	171,4
3	47,9	196,1

Figura 4. 39. Bandas laterales de 122 Hz +-Frecuencia eje x1, x2, x3

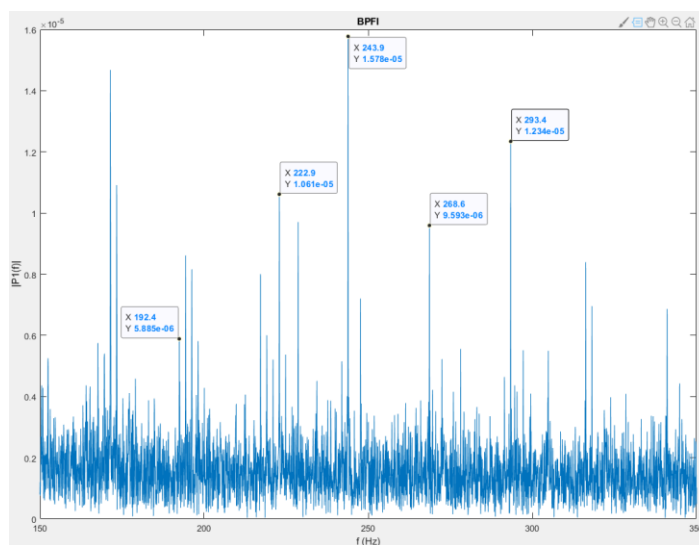


Figura 4. 40. Bandas laterales de segundo armónico de BPF1 244 Hz +-Frecuencia eje x1, x2, x3

Numero armónico	Banda lateral inferior (Hz)	Banda lateral superior (Hz)
1	219,3	268,7
2	194,6	293,4
3	169,9	318,1

Tabla 4. 7. Bandas laterales de segundo armónico de BPF1 244 Hz +-Frecuencia eje x1, x2, x3

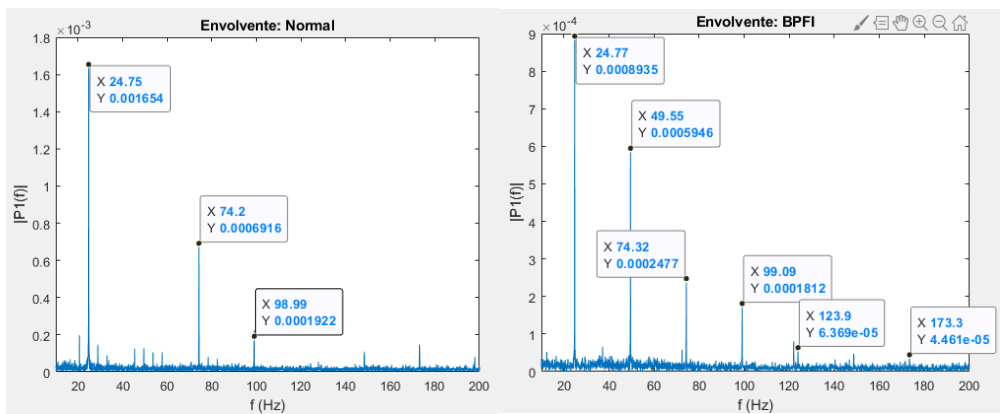


Tabla 4. 8. Armónicos de la velocidad del eje

Otro índice de fallo del rodamiento es la aparición de armónicos de la velocidad del giro del eje. Aparecen más de ellos en la envolvente del BPF1.

Otro posible defecto que podríamos detectar es la **holgura entre rodamiento y la cajera**, ya que en los dos casos se presentan varios armónicos del giro del eje. Pero no se presenta una mayor amplitud en 4xFrecuencia del eje asique tampoco se puede determinar con certeza.

#### 4.3.5. DETECCIÓN DE DEFECTOS DE TIPO BPFO

##### 4.3.5.1. SÍNTOMAS

Defectos en la pista exterior se corresponden físicamente con el número de bolas o rodillos que pasan por un punto de la pista exterior cada vez que el eje realiza un giro completo. Los espectros se caracterizan por presentar picos armónicos de la frecuencia de deterioro de la pista exterior (entre 8 y 10 armónicos de la BPFO).

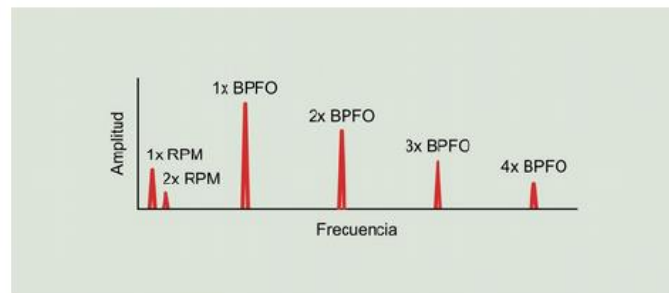


Tabla 4. 9. Fallos en pista interior BPFO

##### 4.3.5.2. BÚSQUEDA DE DEFECTOS

Se aplicará el mismo procedimiento que en BPF1.

$$\text{Curtosis de la señal} = 5.7704 > 3.8551 \quad (7.27.)$$

En este caso solamente se ha utilizado la envolvente, ya que el filtro FIR no mostraba todos los armónicos.

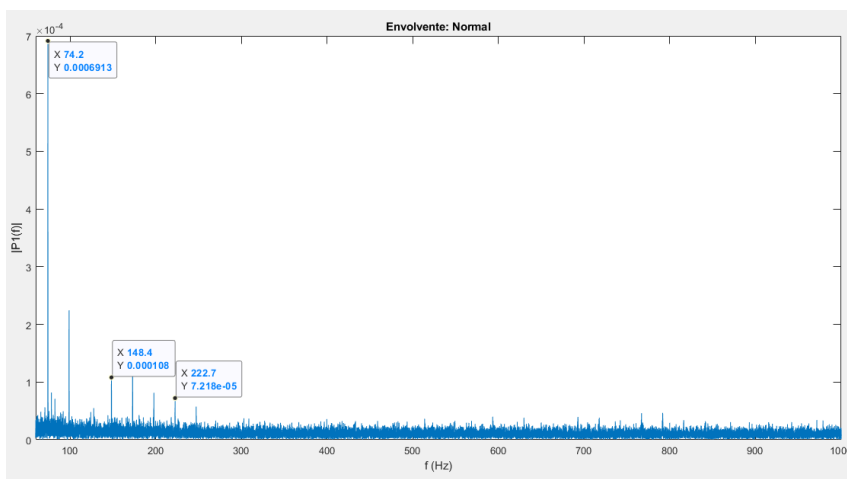


Tabla 4. 10. Rodamiento sano solamente aparecen los armónicos de la velocidad del eje

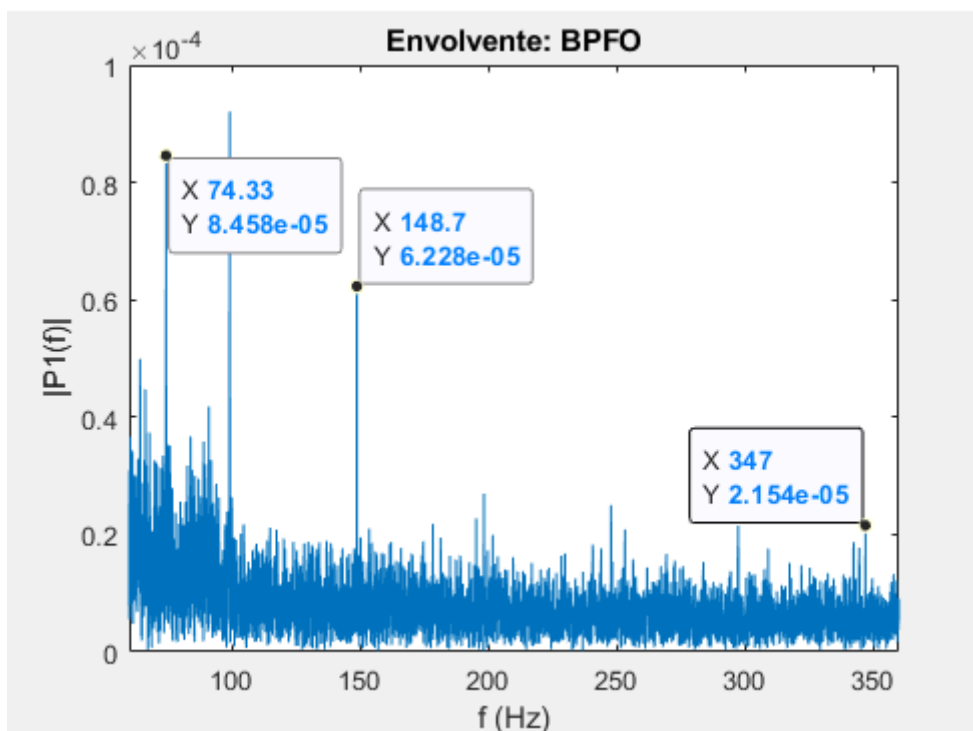


Tabla 4. 11. BPFO y varios de sus armónicos

La señal presenta varios armónicos que se pueden asociar al defecto de tipo BPFO. También aparecen bandas laterales en el armónico  $\times 10$ , otro índice de este tipo fallos.



## 5. DISEÑO ELECTRÓNICO Y LA CONFIGURACIÓN DE LA COMUNICACIÓN

Siguiendo los objetivos, las especificaciones y el diseño, propuestos en los Capítulos 1 y 2, se han diseñado partes electrónicas y configurado los sensores para un correcto funcionamiento del sistema.

### 5.1. PCB

Se ha diseñado una placa PCB, conectando la BeagleBone Black y Xbee Pro S1, siguiendo la filosofía de las shields de Arduino, con el fin de minimizar el espacio utilizado.

En la Figura 5.1 se muestran las conexiones realizadas, basadas en las salidas de los pines definidas en las Figuras 5.2 y 5.3.

El footprint y el esquema eléctrico de la placa se encuentran en el Anexo II.

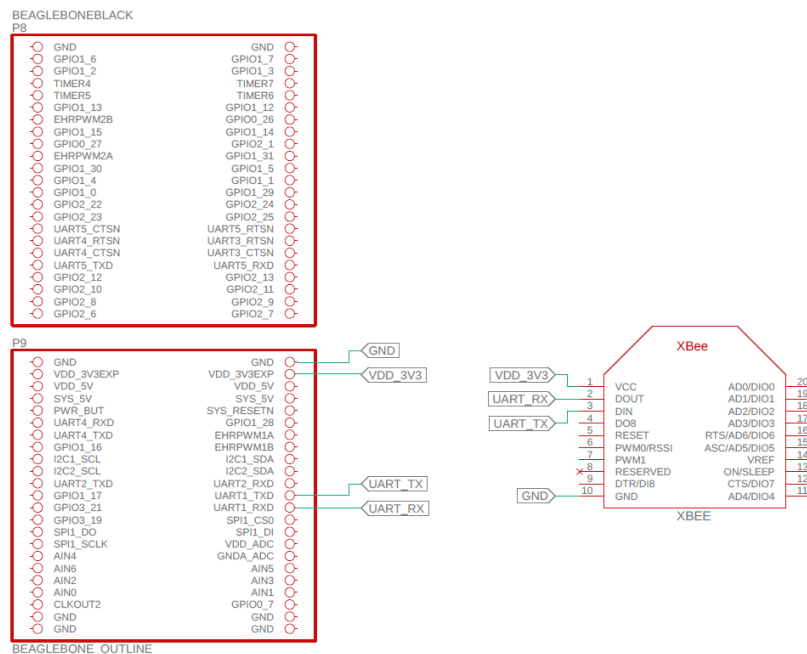


Figura 5. 1. Esquemático de conexiones de la BeagleBone Black y la Xbee Pro S1

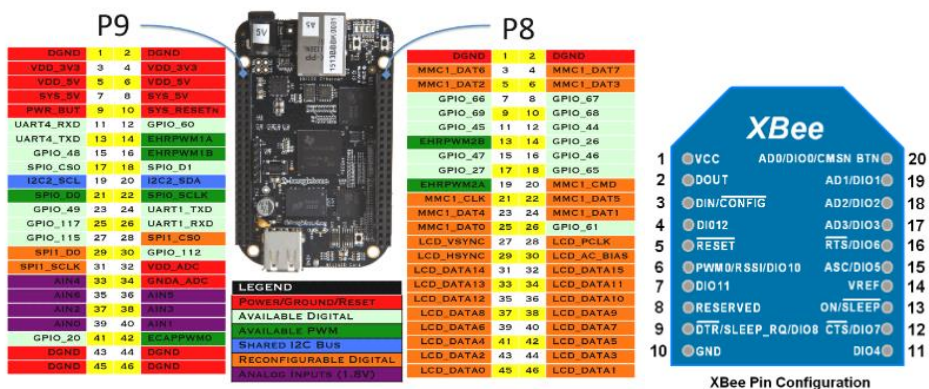


Figura 5. 2. PinOut BeagleBone Black [11] y Xbee Pro S1 [6,1,7]

## 5.2. Configuración de los módulos Xbee

La configuración de los módulos Xbee se ha realizado utilizando el programa gratuito llamado XCTU y un adaptador USB conectado al ordenador.



Figura 5. 3. Adaptador USB (la oreja) y el Xbee Pro S1

Se han configurado los dispositivos Xbee Pro S1 utilizando una comunicación en modo AT con el estándar IEEE 802.15.4, debido a su facilidad para la programación posterior.

Se definen siguientes parámetros:

- Canal (CN): C
- ID: 2311
- Velocidad de la comunicación (BD): 112500 baudios, típico valor utilizado en la industria.

Los valores restantes son los definidos por defecto por el fabricante. Es importante asegurarse de que todos los dispositivos estén configurados de la misma forma, sino serán incapaces de comunicarse.

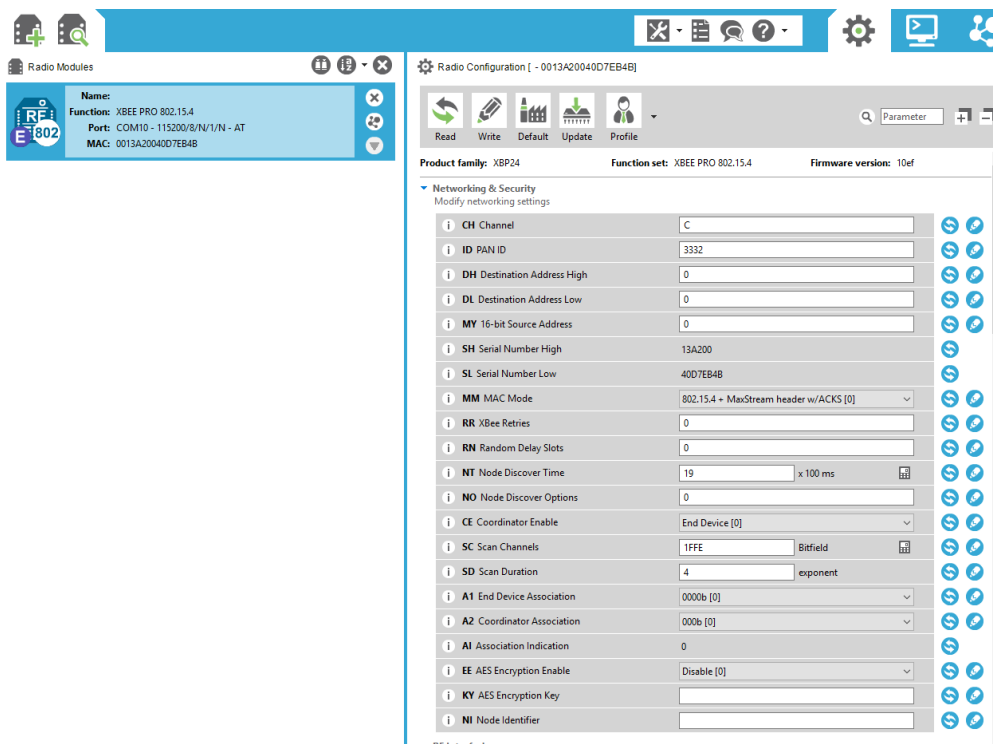


Figura 5. 4. XCTU

### 5.3. Conexión de sensores de temperatura y presión

Uno de los objetivos de este trabajo es la conexión de los sensores de presión y temperatura a través de RS485 al nuestro sistema.

Los valores son tomados utilizando el protocolo Modbus integrado en el OPCUA.

En el Capítulo 2 se ha comentado, que debido a falta de estos sensores se han simulado utilizando un Arduino.

Para conectar el RS485 al ordenador se ha utilizado un adaptador USB llamado “USB DSD TECH model:SH-U10” y un módulo MAX485 al Arduino.

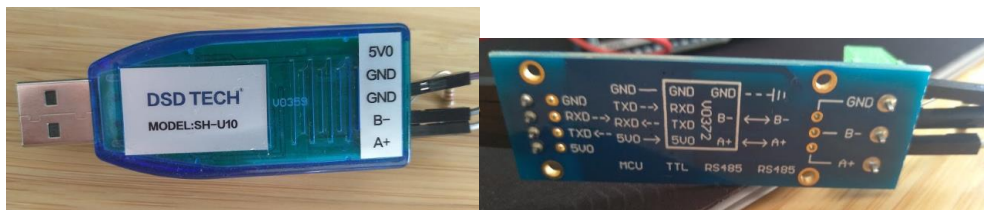


Figura 5. 5. USB DSD TECH model:SH-U10 y un módulo MAX485

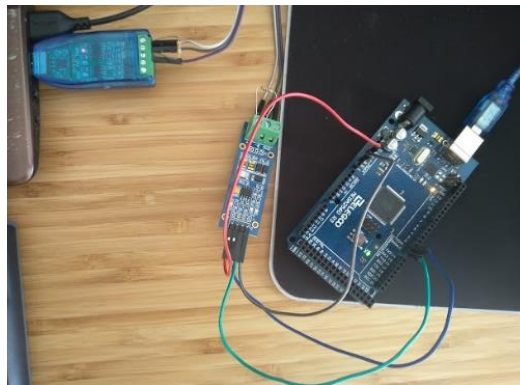


Figura 5. 6. Conexión de los dispositivos

Se han conectado resistencias de 120 ohms para garantizar impedancia al principio y al final de las líneas RS485, garantizando una buena comunicación sin errores.

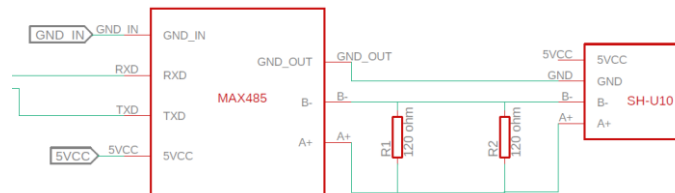


Figura 5. 7. Esquemático de conexiones de RS485

El esquemático de conexiones completo se encuentra en el anexo III.





## 6. DISEÑO DE LA CARCASA PARA EL SENSOR ACÚSTICO

Se ha diseñado una carcasa, para la sujeción del sensor acústico utilizando el software CAD Solidwoks. Generando un modelo 3D y a partir de él los planos.

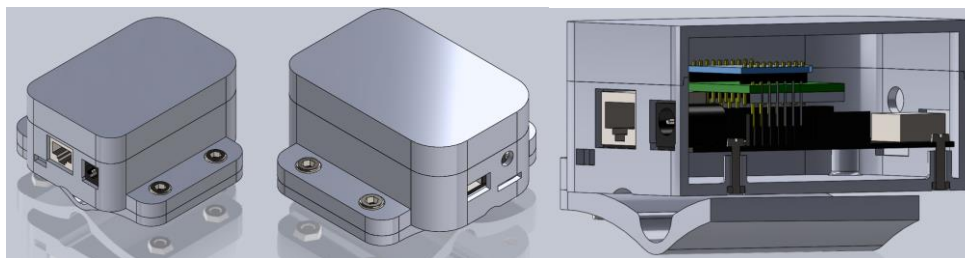
El prototipo está pensado para atornillar la carcasa al trípode dejándola fija, con una salida para la antena y múltiples entradas, en el caso de que se necesitare utilizar alguna y asegurando así la robustez del sensor.



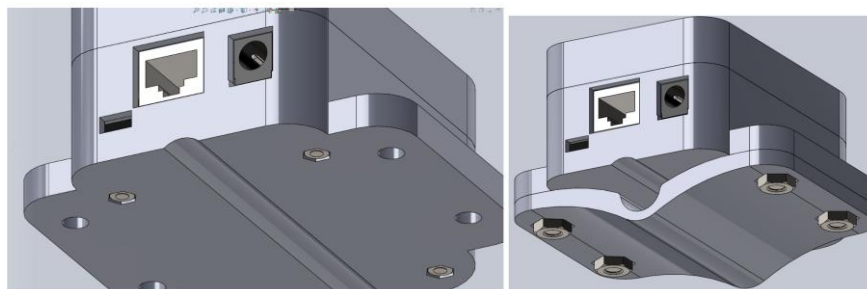
*Figura 6. 1. Carcasa para la sujeción del sensor acústico*

Se han tomado las medidas del diámetro del trípode, del conector de la antena y la altura del sistema electrónico (BeagleBone + PCB + Xbee).

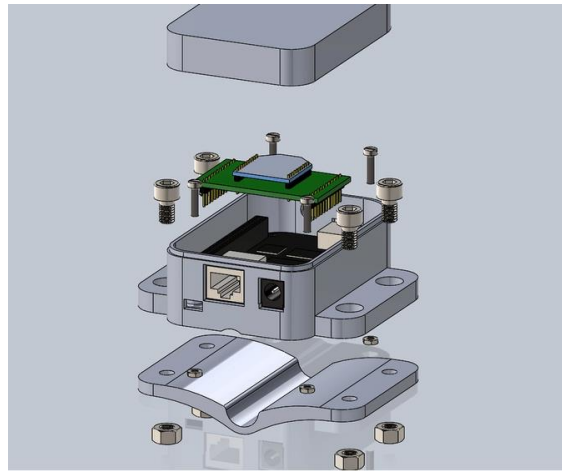
Para agilizar el modelado del conjunto se han utilizado modelos 3D descargables del repositorio GrabCad: la BeagleBone Black [12], la Xbee [13] y los pines Dupont 12x1 tipo macho [14].



*Figura 6. 2. Modelo 3D de la carcasa, perspectivas laterales de la carcasa y un corte*



*Figura 6. 3. Vista inferior de la carcasa*



*Figura 6. 4. Explosionado*

Las paredes de la carcasa son de 3 mm de espesor para asegurar suficiente rigidez del conjunto.

Las tuercas están incrustadas en la carcasa dentro de unas ranuras con forma hexagonal, para poder atornillarlas con facilidad. Tal y como se muestra en la Figura 6.3.

Los planos de la carcasa junto con su lista de materiales se encuentran dentro del Anexo IV.

## 7. PROGRAMACIÓN

La programación de este trabajo se divide en dos partes:

- Análisis de vibraciones en Matlab
- Creación de un servidor OPCUA, comunicación UART, integración de Modbus en el sistema y adaptación de los programas creados en Matlab, todos ellos escritos en Python.

Uno de los objetivos de este texto es estudiar frecuencias audibles (de 0 a 20 kHz), por lo tanto, se utilizará una frecuencia de muestreo de 44100 Hz basándose en el teorema de Nyquist-Shannon [16]. El formato de datos serán 16 bits por muestra, debido a las limitaciones de la BeagleBone Black.

En los programas de análisis de vibraciones se utiliza una señal pregrabada con rodamientos sanos y los engranajes en buen estado, en función de la cual se realiza la toma de decisiones.

En Anexo I, se encuentra la explicación de cómo inicializar los programas en la BeagleBone de forma automática.

### 7.1. Análisis de vibraciones en Matlab

La programación en Matlab se ha escrito siguiendo la metodología descrita en el Capítulo 4, basándose en el Manual de vibraciones [1] y un ejemplo de MATLAB & Simulink [8].

Las muestras de audio están grabadas con el programa Audacity [17], una aplicación informática multiplataforma libre, en el formato “.wav” debido a que no realiza ningún tipo de conversión.

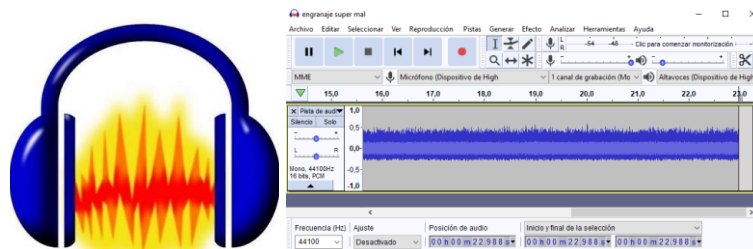


Figura 7. 1. Audacity y muestra de grabación

#### 7.1.1. DEFECTOS EN ENGRANAJES

Se han creado tres programas para análisis de defectos en engranajes “Gear\_eje\_desalineado.m”, “Gear\_diente\_roto.m” y “Gear\_eje\_doblado.m”. Correspondientes subapartados 4.2.3, 4.2.4 y 4.2.5 respectivamente. En la Figura 7.2. se muestra el flujograma de todos estos programas.

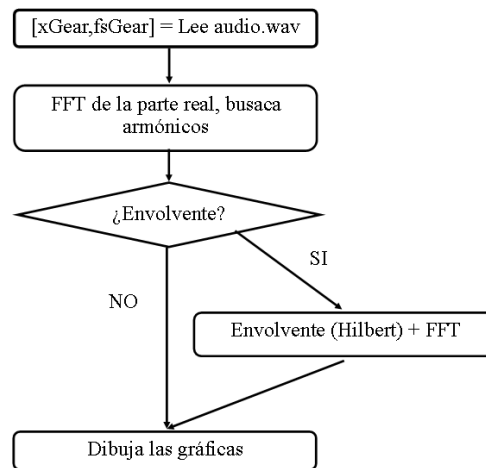


Figura 7. 2. Búsqueda de defectos en los engranajes, flujograma

Los programas comienzan leyendo los archivos de audio, realizan la FFT sacando su parte real, la envolvente junto con la FFT si es necesario y finalmente muestran los resultados en las gráficas, en función de las cuales se realiza la toma de decisiones.

### 7.1.2. DEFECTOS EN LOS RODAMIENTOS

Los “BFPI.m” y “BPFO.m” se corresponden con el análisis realizado en los subapartados 4.3.4 y 4.3.5.

#### 7.1.2.1. BPFI

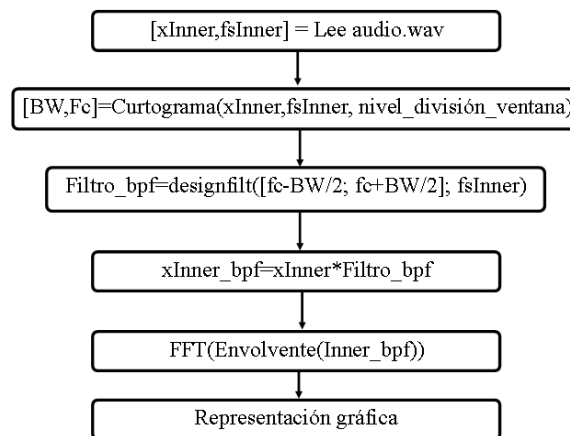


Figura 7. 3. Flujograma de BPFI

El programa empieza leyendo los archivos de audio. Computa el curtograma, programa de Jerome Antoni [15] transcrito, para sacar solamente la información de la posición de la frecuencia central (Fc) y el ancho de banda (BW). Se aplica el filtro de paso banda en función a la información proporcionada a la señal de audio en bruto. Una vez hecho esto se hace la envolvente de la señal filtrada, su FFT y finalmente se muestran los resultados en la gráfica.

**7.1.2.2. BPFO**

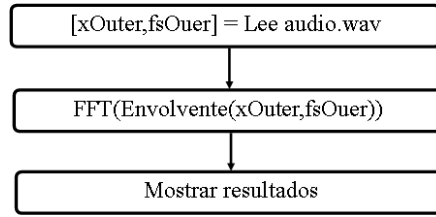


Figura 7. 4. Flujograma de BPFO

**7.2. Programación en Python**

La programación del prototipo está basada en gran parte en los programas escritos anteriormente en Matlab. Para ello se ha utilizado el lenguaje de programación Python, debido a:

- Sus librerías como por ejemplo numphy, pyaudio, Adafruit\_BBIO, etc., que contienen muchas de las funciones que se han utilizado en el análisis de vibración
- Lenguaje multiplataforma
- Posibilidad de programación del cliente y del servidor OPC-ua
- Librerías de Modbus RTU

Y se han comprobado los resultados en la BeagleBone Black.

Todos los scripts se pueden ejecutar de forma independiente para comprobar su correcto funcionamiento.

**7.2.1.1. PROGRAMA DE ANÁLISIS DE VIBRACIONES**

En este parte se ha hecho la adaptación de código de Matlab a Python.

Solamente se ha implantado una metodología para el análisis de defectos de tipo BPFi como ejemplo, pero están todas las funciones necesarias para realizar la búsqueda de defectos.

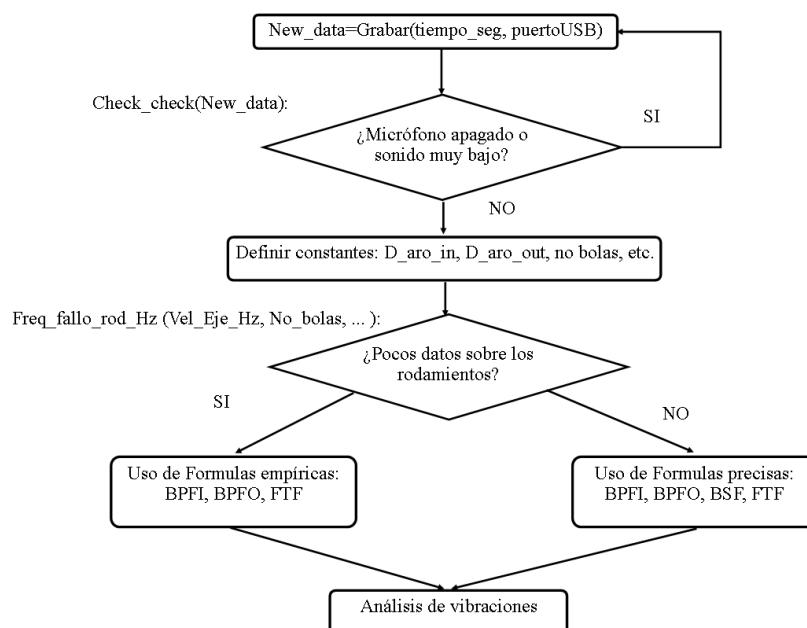


Figura 7. 5. Programa Principal de análisis de vibraciones “Main\_bucle.py”

El programa comienza grabando una muestra y comprueba si cabe la posibilidad de que el micrófono este apagado, sin batería o hay que acercar el micrófono a la máquina.

Se definen las constantes para calcular las frecuencias de deterioro de los rodamientos subapartado 4.3.2, si no se dispone de suficientes datos se utilizarán los datos empíricos.

Una vez hecho esto se pasa al análisis de BPFÍ

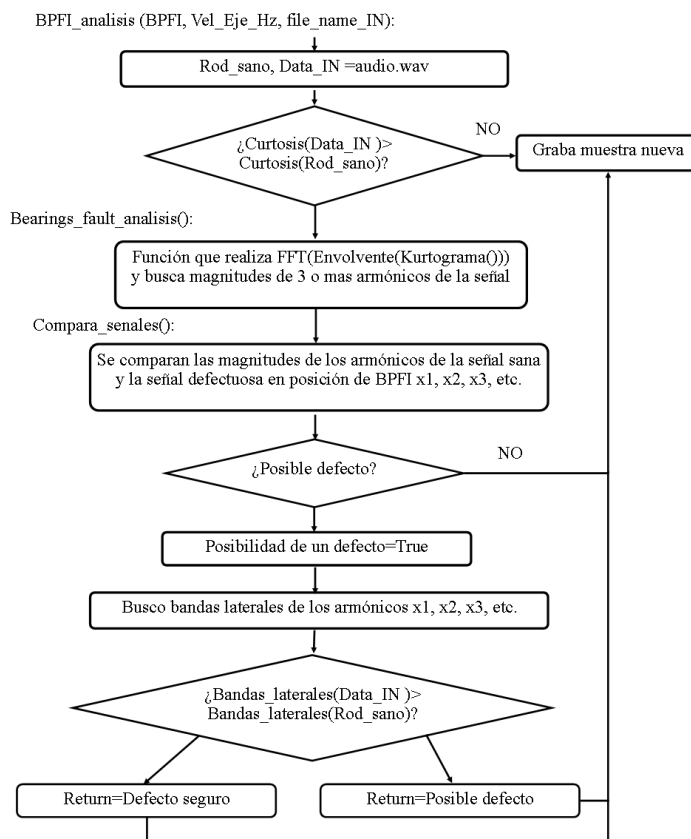


Figura 7. 6. Análisis de BPFÍ "Analisis.py"

Los pasos siguientes son los mismos explicados en el subapartado 4.3.4.

Todas las funciones creadas son genéricas por lo tanto se puede especificar el número de armónicos deseados a analizar o cambiar el puerto utilizado para la grabación por defecto.

### 7.2.2. MEDIDOR DE NIVEL DE RUIDO INTEGRADO EN LA BEAGLEBONE BLACK

Este programa se ejecuta en la BeagleBone Black de forma automática al encenderla.

Monitoriza un rango de frecuencias alrededor de una central definida por el usuario, por defecto 440Hz y alrededor de en un rango de +-10Hz. La frecuencia de referencia puede ser cambiada por el usuario utilizando Rx del Xbee. Para ello realiza la FFT, hace una transformación de la señal a pascles (Pa) y mide el nivel de sonido del entorno en decibelios (dB), utilizando el valor de la sensibilidad del micrófono -41.6 [dB re V/Pa]. Fórmulas definidas en la página web [18].

Para los calcular el nivel de sonido en decibelios se utiliza “A-weighting function (IEC 61672-1)”, definida en la página web [18]:

$$W_A(f) = \frac{12194^2 * f^4}{(f^2 + 20.6^2) * (f^2 + 12194^2) * \sqrt{(f^2 + 107.7^2)(f^2 + 737.9^2)}} \quad (7.1.)$$

$$W_{dBA} = 20 * \log_{10}(W_A(f)) + 2.0 \quad (7.2.)$$

Finalmente envía los resultados por el puerto TX del xBee, utilizando el siguiente formato:

- “encabezado: mensaje: fin del mensaje”
- “XD: Frecuencia máxima detectada Hz; Presión [Pa]; Ruido del entorno [dB] :\n” en el caso este programa

El cambio de referencia se realiza a través del UART y un hilo, para no frenar el programa principal. Si al dispositivo le llega un mensaje nuevo, es decodificado, comprobado si esta en formato correcto y en el caso de cumplirse estos requisitos definido como una nueva frecuencia a monitorizar.

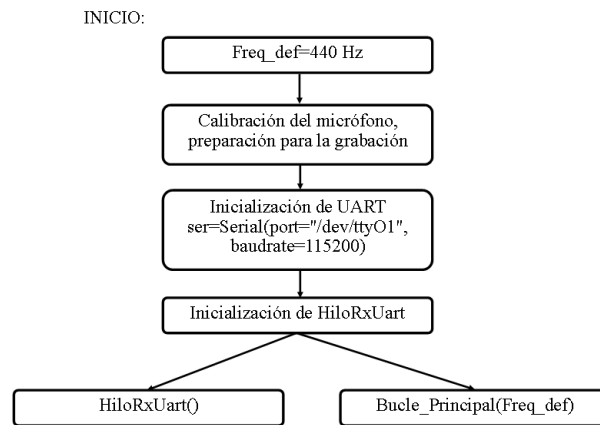


Figura 7. 7. Inicio de programa “Sound\_level\_todo.py”

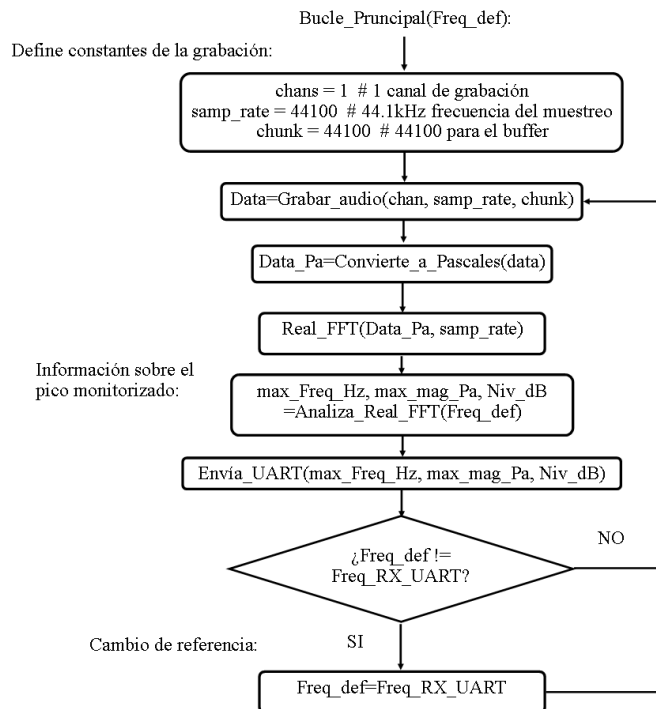


Figura 7. 8. Bucle Principal

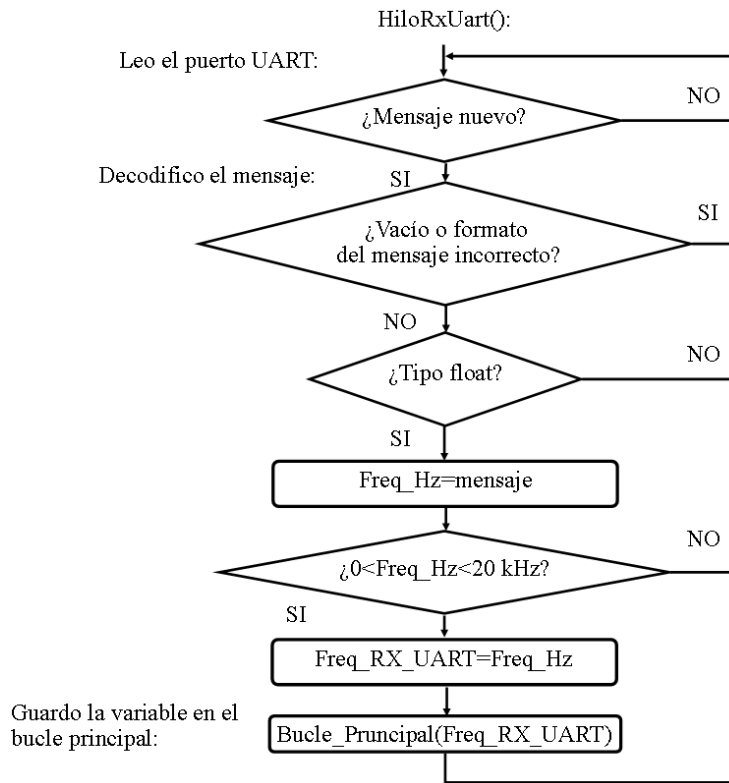


Figura 7. 9. Hilo rx UART

Los pines UART1 [11] de la BeagleBone Black se definen gracias a la librería Adafruit\_BBIO.UART.

### 7.2.3. SERVIDORES OPCUA

Se han creado dos servidores distintos de OPCUA basados en los ejemplos de la librería de Python-Opcua [19].

Se comienza creando el objeto de la clase Servidor y definiendo:

Servidores Opc-Ua:	Micrófono:	Modbus-RS485:
Dirección física de la red	"opc.tcp://0.0.0.0:4841/Microfono"	opc.tcp://0.0.0.0:4840/opcuamodbus
Nombre del servidor	OpcUa-Microfono	OpcUa Server/Modbus

Tabla 7. 1. Definición del servidor OPC-ua

Una vez hecho esto se definen las variables que serán visibles por el cliente y los permisos de su escritura.



7.2.3.1. OPCUA MICRÓFONO

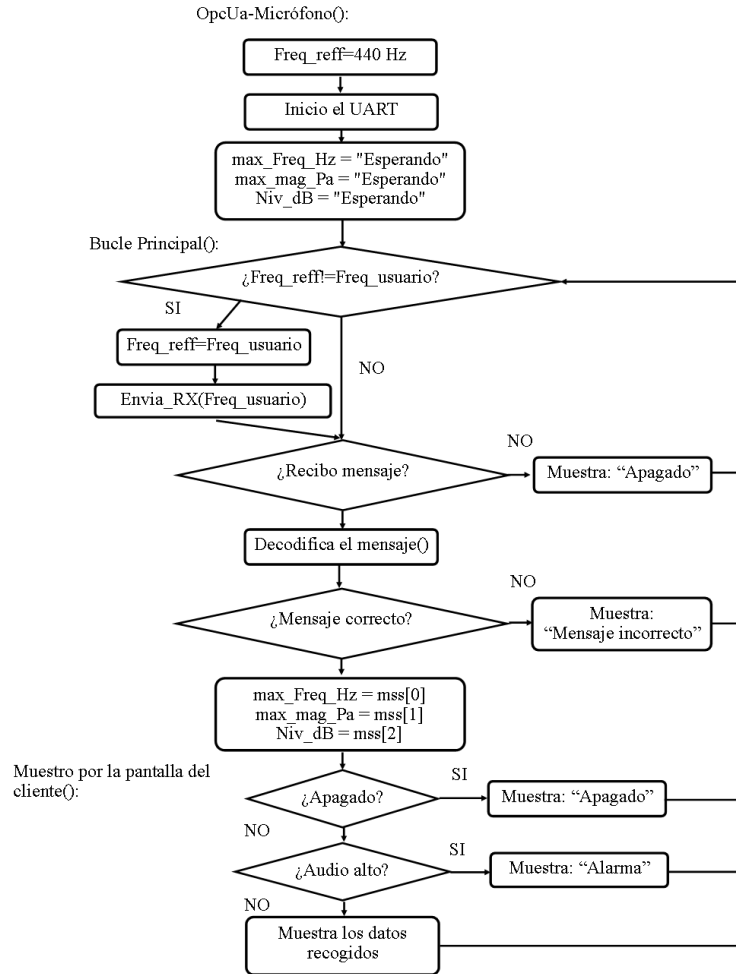


Figura 7. 10. Flujograma del Servidor OpCUa del Micrófono

Este programa es encargado de recibir los datos por el UART, procesarlos y mostrar los resultados al Cliente.

Para asegurarse de que el servidor no se quede parado por culpa de la espera del mensaje de forma infinita, se ha definido un tiempo de 2 segundos, después de esto se mostrara el mensaje de “Conexión perdida”.

En caso de recibir un mensaje pasa a ser decodificado, si esta en formato incorrecto se mostrara “Mensaje incorrecto”.

Finalmente, si el mensaje es correcto se comprueba la posibilidad de que el micrófono este apagado (“Micrófono apagado o batería baja”), de que se ha detectado una anomalía de incremento de pico de frecuencia (“Alarma: Valore\_de\_Frecuencia [Hz]”) o en caso de no detectar ninguna se mostraran los valores recibidos: Frecuencia con pico máximo Hz, Amplitud de la frecuencia [Pa] y el nivel de sonido del entorno [dB].

#### 7.2.4. OPCUA MODBUS

El servidor OpcUa de Modbus está basado en los ejemplos de la librería Modbus-tk [20]. Para la realización de las pruebas se ha utilizado el programa QModbusMaster, comprobando así el buen funcionamiento del hardware y del software.

El programa muestra una tabla de 12 posiciones, 2 primeras son valores de temperatura y los 2 siguientes de presión. Los 8 restantes son de escritura para la calibración.

Si el sensor se desconecta muestra [65535, 65535, 65535, 65535].

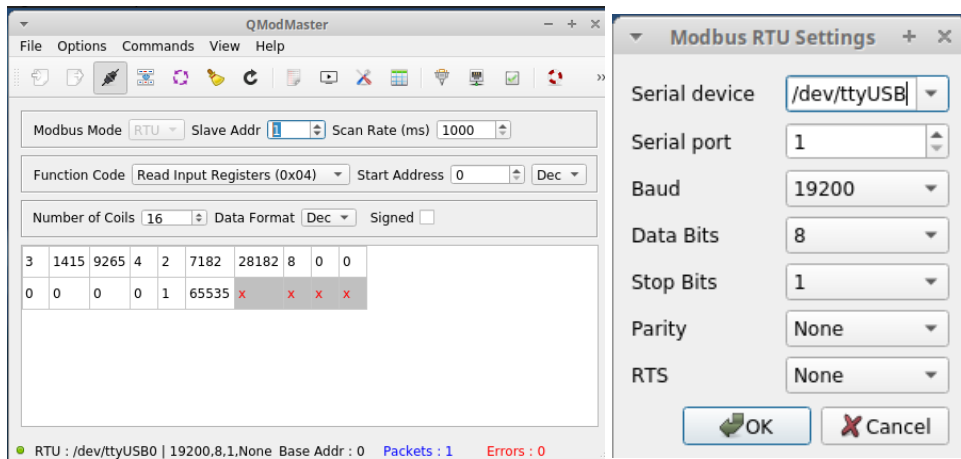


Figura 7. 11. QModbusMaster y configuración RTU

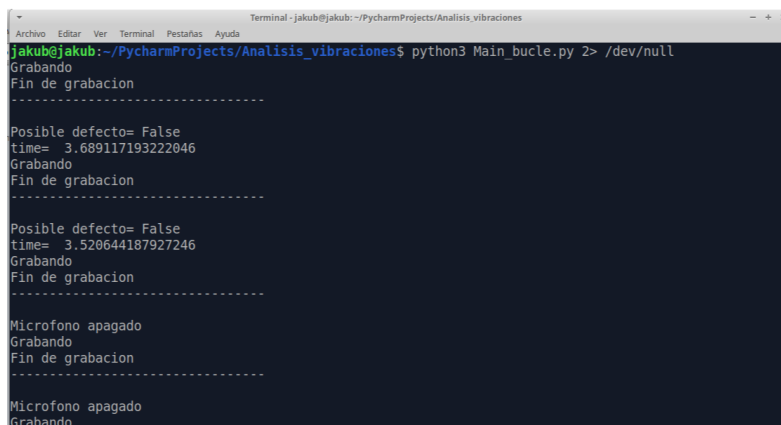
Es importante asegurarse que los parámetros de la comunicación estén bien configurados, Figura 7.11.

## 8. RESULTADOS

En este capítulo se describen los resultados obtenidos.

### 8.1. Análisis de vibraciones

Se ha comprobado que es posible realizar el mantenimiento predictivo con el micrófono direccional no solamente para la medición del nivel de ruido, sino también para la búsqueda de defectos en los engranajes y los rodamientos. También la posibilidad de su la integración en una plataforma low cost BeagleBone Black junto con el protocolo OpcUa.



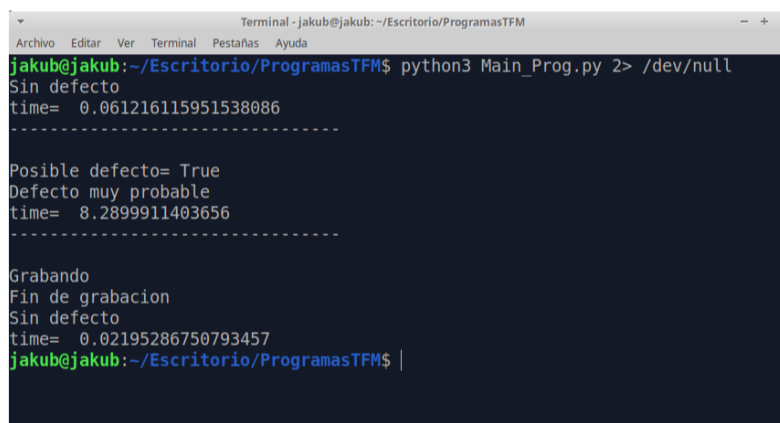
```

Terminal - jakub@jakub: ~/PycharmProjects/Analisis_vibraciones
jakub@jakub:~/PycharmProjects/Analisis_vibraciones$ python3 Main_bucle.py 2> /dev/null
Grabando
Fin de grabacion
-----
Posible defecto= False
time= 3.689117193222046
Grabando
Fin de grabacion
-----
Posible defecto= False
time= 3.520644187927246
Grabando
Fin de grabacion
-----
Microfono apagado
Grabando
Fin de grabacion
-----
Microfono apagado
Grabando

```

Figura 8. 1. Main\_bucle.py

Este programa realiza el análisis de vibraciones de forma continua y comprueba si se ha detectado algún fallo o no, Figura 8.1.



```

Terminal - jakub@jakub: ~/Escritorio/ProgramasTFM
jakub@jakub:~/Escritorio/ProgramasTFM$ python3 Main_Prog.py 2> /dev/null
Sin defecto
time= 0.061216115951538086
-----
Posible defecto= True
Defecto muy probable
time= 8.2899911403656
-----
Grabando
Fin de grabacion
Sin defecto
time= 0.02195286750793457
jakub@jakub:~/Escritorio/ProgramasTFM$

```

Figura 8. 2. Main\_Prog.py

En la Figura 8.2. se muestra el mismo programa, pero sin el bucle infinito, que realiza comparaciones entre:

- Dos muestras iguales, la curtosis es igual por lo tanto ni siquiera realiza el análisis
- Una muestra de referencia y otra con BPF1, encuentra el defecto
- Muestra de referencia y otra grabada al instante, la curtosis es más grande por lo tanto realiza el análisis, pero no encuentra ningún tipo de defecto.

```

Terminal - debian@beaglebone: ~/ProgramasTFM
Archivo Editar Ver Terminal Pestañas Ayuda
debian@beaglebone:~/ProgramasTFM$ python3 ./Analisis.py
Posible defecto= True
Defecto muy probable
104.78364396095276
debian@beaglebone:~/ProgramasTFM$
    
```

Figura 8. 3. Análisis de tipo BPF1 en la BeagleBone Black

En la Figura 8.3 se comprueba que el análisis de los defectos de tipo BPF1 no es factible en la BeagleBone Black. Debido a que tarda mucho tiempo en realizarse, casi dos minutos.

Cabe destacar que se han comparado dos muestras de más de 10 segundos por eso lleva tanto tiempo, disminuyendo el tiempo a 2 segundos se pueden obtener resultados mucho más rápidos, pero con más cantidad de ruido. Por lo tanto, para obtener los resultados óptimos es preferible usar una Odroid-XU4 o DE10-Nano.

## 8.2. Medición de ruido integrado en el OPCUA

La BeagleBone Black es una buena opción para convertirla en un sensor capaz de minorizar picos de frecuencias en un determinado rango, para el mantenimiento predictivo.

```

Terminal - debian@beaglebone: ~/Tests/Sound_Level
Archivo Editar Ver Terminal Pestañas Ayuda
jakub@jakub:~$ ssh debian@beaglebone.local
debian@beaglebone.local's password:
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Jul 3 14:42:46 2019 from 192.168.6.1
debian@beaglebone:~$ cd Tests/Sound_Level/
debian@beaglebone:~/Tests/Sound_Level$ ls
Nivel_de_sonido.py Prog_principal.py __pycache__ Sound_level_todo.py
debian@beaglebone:~/Tests/Sound_Level$ python3 Nivel_de_sonido.py 2> /dev/null
time: 0.7831969999999999
Maximo de Frecuencia en: 437.0 Magnitud= 1.8007672931e-07 [Pa]
dB: 25.1340501957
debian@beaglebone:~/Tests/Sound_Level$ |
    
```

Figura 8. 4. Tiempo de ejecución de programa

El tiempo de ejecución es de unos 800 ms, por lo tanto, es aceptable.

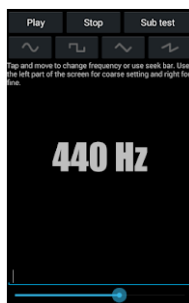


Figura 8. 5. Simple tone generator [22]

Se ha probado el funcionamiento del sistema utilizando “Simple tone generator” [22], una aplicación gratuita para el móvil capaz de generar tonos a frecuencias predefinidas.

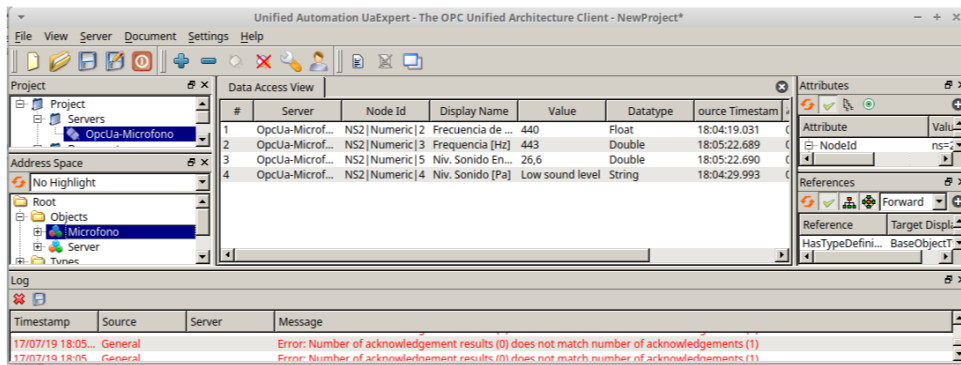


Figura 8. 6. Funcionamiento del sistema con la frecuencia por defecto 440Hz

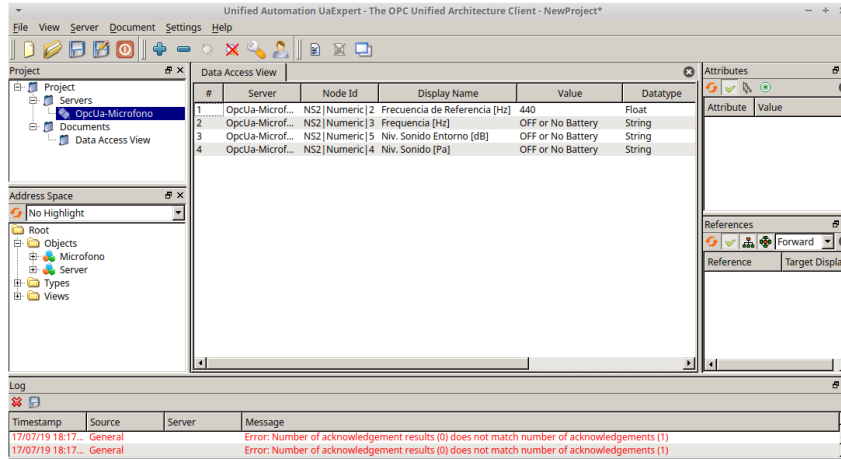


Figura 8. 7. Micrófono apagado o sin batería (<25 dB)

#	Server	Node Id	Display Name	Value	Datatype	#	Server	Node Id	Display Name	Value	Datatype
1	OpcUa-Microf...	NS2 Numeric 2	Frecuencia de Referencia [Hz]	777	Double	1	OpcUa-Microf...	NS2 Numeric 2	Frecuencia de Referencia [Hz]	15000	Double
2	OpcUa-Microf...	NS2 Numeric 3	Frecuencia [Hz]	ALARMA 776.0	String	2	OpcUa-Microf...	NS2 Numeric 3	Frecuencia [Hz]	15001	Double
3	OpcUa-Microf...	NS2 Numeric 5	Niv. Sonido Entorno [dB]	ALARMA 54.8	String	3	OpcUa-Microf...	NS2 Numeric 5	Niv. Sonido Entorno [dB]	32,8	Double
4	OpcUa-Microf...	NS2 Numeric 4	Niv. Sonido [Pa]	ALARMA 0.00203	String	4	OpcUa-Microf...	NS2 Numeric 4	Niv. Sonido [Pa]	4,22e-05	Double

Figura 8. 8. Cambio de frecuencia de referencia a 700 Hz, se activa a la alarma, y a 15 kHz sin anomalía

Tal y como se puede comprobar en las figuras 8.6, 8.7 y 8.8 el programa funciona correctamente. Busca máximos alrededor de frecuencias de referencia, avisa sobre incrementos de magnitud con una alarma y sobre la posibilidad de que el micrófono se quede sin pilas o este apagado.

### 8.3. Integración de Modbus dentro del OPCUA

Se ha programado el Arduino utilizando un ejemplo de la librería Modbus\_RTU [23] “simple\_slave.ino” con una velocidad de la comunicación de 19200 baudios.

```
#include <ModbusRtu.h>

// data array for modbus network sharing
uint16_t aul6data[16] = {
    3, 1415, 9265, 4, 2, 7182, 28182, 8, 0, 0, 0, 0, 0, 1, -1 };

```

Figura 8. 9. Valores por defecto programados en Arduino

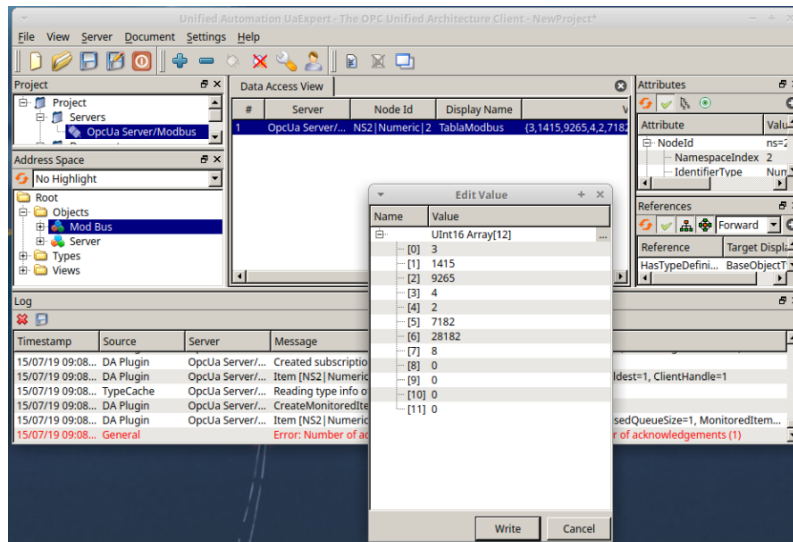


Figura 8. 10. Sensor encendido

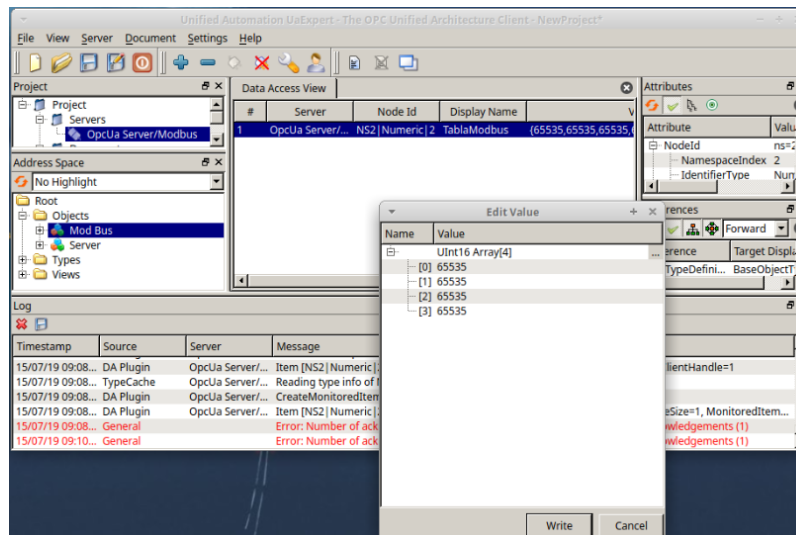


Figura 8. 11. Sensor apagado

En la Figura 8.10 y 8.11 se han obtenido los resultados esperados. Si el sensor esta funcionando muestra los valores esperados y si está desenchufado se muestran los valores indicando el de error.

**PRESUPUESTO**

<b>MÁSTER DE INGENIERÍA MECATRÓNICA</b>		Pag. xv MEDICIONES Y PRESUPUESTO				
Nº ORDEN	CONCEPTOS		Nº UNIDADES	FABRICANTE	PRECIO UNITARIO MATERIAL	PRECIO UNITARIO TOTAL
1	The t.bone EM 9600	Micrófono direccional	x1	Thomann	44,00 €	44,00 €
2	Pro snake Camera Cable 1,5	Cable de micrófono	x1	Thomann	8,30 €	8,30 €
3	Millenium MS 2005	Trípode	x1	Thomann	9,90 €	9,90 €
4	UGREEN USB	Tarjeta de Sonido USB externa	x1	Ugreen	9,99 €	9,99 €
5	Beaglebone Black Industrial	Ordenador de placa única	x1	Texas Instruments	62,75 €	62,75 €
6	Xbee	Modulo Xbee Pro S1	x2	Digi International	59,50 €	59,50 €
7	Xbee FT232RL USB	Xbee USB Shield, adaptador PC	x1	Digi International	4,83 €	4,83 €
8	ANT-2.4-LCW- ccc	Antena para el Xbee	x1	Linx Technologies	2,47 €	2,47 €
9	RP SMA male female	Adaptador de RP SMA macho hembra	x1	Linx Technologies	2,25 €	2,25 €
10	USB DSD TECH model:SH-U10	Adaptador USB- RS485	x1	DSD TECH	12,49 €	12,49 €
11	MAX485	Adaptador RS485/Arduino	x1	Arduino	1,91 €	1,91 €
12	Arduino Mega	Mega 2560 R3 + Cable USB	x1	Arduino	13,29 €	13,29 €
13	ISO - 4034 - M6 - S	Tuerca M6	x4	SHERCO	0,05 €	0,20 €
14	DIN EN ISO 1207 - M3 x 12 - 12C	Tornillo M3 x 12	x4	AccuScrews	0,30 €	1,20 €
15	ISO - 4035 - M3 - C	Tuerca M3	x4	SHERCO	0,05 €	0,20 €
16	8 DIN 912 M6 x 10 --- 10S	Tornillo M6 x 10	x4	AccuScrews	0,27 €	1,08 €
					Total	234,36 €

Mano de obra ingeniero:	30 €/h
Tiempo de montaje:	2h
Total:	60 €

Precio total + mano de obra:	294,36 €
IVA:	21%
Total:	356,18 €



## CONCLUSIONES Y FUTUROS TRABAJOS

Este trabajo fin de máster estudia la posibilidad de la realización del diagnóstico de fallos en máquinas utilizando un sistema compuesto por una Beaglebone Black y un micrófono direccional.

Se han estudiado sus posibles aplicaciones, basadas en múltiples ensayos en un banco de trabajo, y también sus limitaciones.

Se han comprobado las posibilidades que ofrece el protocolo Opcua y se ha prototipado un sistema robusto capaz de incluir varios sensores y otros protocolos de comunicación.

Para los futuros trabajos se propone:

- Utilización de una placa más potente como, por ejemplo: DE10-Nano o la Odroid-XU4. Si se intenta realizar mantenimiento proactivo, identificando el defecto (Capítulo 1.2).
- Incorporación de más sensores como por ejemplo un acelerómetro. Para obtener lecturas de múltiples fuentes grabadas al mismo tiempo, con el fin de compararlas.
- Mejora de envío de mensajes utilizando los xBee utilizando un protocolo de comunicación, que codifique el mensaje sí que se confunda con la cabecera.



## BIBLIOGRAFÍA

- [1] “Manual Análisis de las vibraciones” Power-MI cloud conditioning monitoring.
- [2] Transformada Rápida de Fourier o la FFT (Fast Fourier Transform), Acceso el 19/07/2019, [https://es.wikipedia.org/wiki/Transformada\\_de\\_Fourier](https://es.wikipedia.org/wiki/Transformada_de_Fourier)
- [3] “¿Qué es el OPCUA?”, Acceso el 19/07/2019, <https://bit.ly/2JJaVea>
- [4] Micrófono Direccional the t.bone EM 9600, Ficha técnica, Acceso el 19/07/2019, <https://bit.ly/2JDVm7I>
- [5] MPU6050, Ficha técnica, Acceso el 19/07/2019, <https://bit.ly/28IQJaX>
- [6] Xbee Pro S1, Ficha técnica, Acceso el 19/07/2019, <https://bit.ly/2Y21gJa>
- [7] Pawel Rzeszucinski, Maciej Orman, Cajetan T. Pinto, Agnieszka Tkaczyk, and Maciej Sulowicz , “*Bearing Health Diagnosed with a Mobile Phone*”, 2018
- [8] MATLAB & Simulink example, *Rolling Element Bearing Fault Diagnosis*, Acceso el (19/07/2019), <https://bit.ly/2LYKbsS>
- [9] Transformada de Hilbert, Acceso el (19/07/2019), <https://bit.ly/2XQLyfM>
- [10] José Manuel Sierra Velasco, “Método del trabajo”
- [11] BeagleBone Black Pin Out, Acceso el (19/07/2019), <https://bit.ly/2XFgAvT>
- [12] Modelo 3D de la BeagleBone Black, Acceso el (19/07/2019), <https://bit.ly/2LILFy4>
- [13] Modelo 3D del Xbee, Acceso el (19/07/2019), <https://bit.ly/2NVNQu8>
- [14] Modelo 3D de Pines 12x1 tipo DuPont, Acceso el (19/07/2019), <https://bit.ly/2LnVZ8A>
- [15] Jerome Antoni, “Fast Kurtogram”, Acceso el (19/07/2019), <https://bit.ly/2SjbNtT>
- [16] Teorema Nyquist-Shannon, Acceso el (19/07/2019), <https://bit.ly/2MnnpeU>
- [17] Audacity, Acceso el (19/07/2019), <https://www.audacityteam.org/>
- [18] Audio Processing in Python Part II: Exploring Windowing, Sound Pressure Levels, and A-Weighting Using an iPhone X, Acceso el (19/07/2019), <https://bit.ly/2SjqDAD>
- [19] Python-Opcua, Acceso el (19/07/2019), <https://bit.ly/2Gcut9R>
- [20] ModbusTk, Acceso el (19/07/2019), <https://bit.ly/32xzMKz>
- [21] Five Ways To Run a Program On Your Raspberry Pi At Startup, Acceso el (19/07/2019), <https://bit.ly/2knS5g4>
- [22] Simple tone generator apk, Acceso el (19/07/2019), <https://bit.ly/2Y5Ho7L>
- [23] ModbusRTU, Acceso el (19/07/2019), <https://bit.ly/2xTtE1b>



## ANEXO I: AUTOSTART DE UN PROGRAMA EN LA BEAGLEBONE BLACK

En este anexo se explica cómo inicializar un programa de forma automática, asegurándose de que todos los procesos que son necesarios para grabar estén ejecutándose. Basado en Five Ways To Run a Program On Your Raspberry Pi At Startup [21].

Dentro de la BeagleBone Black, en el directorio:

```
$cd /lib/systemd/system/sample.service
```

Se crea un documento con el siguiente programa:

```
[Unit]
Description=Este programa se inicializara de forma automatica
After=multi-user.target

[Service]
Type=idle #Este comando se asegura que esta funcionando todo lo demas
ExecStart=/usr/bin/python3 /bin/Sound_level_todo.py & #ruta python; ruta del programa; "&" bucle infinito

[Install]
WantedBy=multi-user.target
```

### *Anexo I. 1. El programa de autostart*

Este programa se ejecutará, esperando a que todos los procesos de inicio se inicien.

Una vez hecho esto se guarda el archivo como, por ejemplo: “autostart”.

```
$sudo systemctl daemon-reload
$sudo systemctl enable sample.service
```

Se habilita el servicio y reinicia el sistema con:

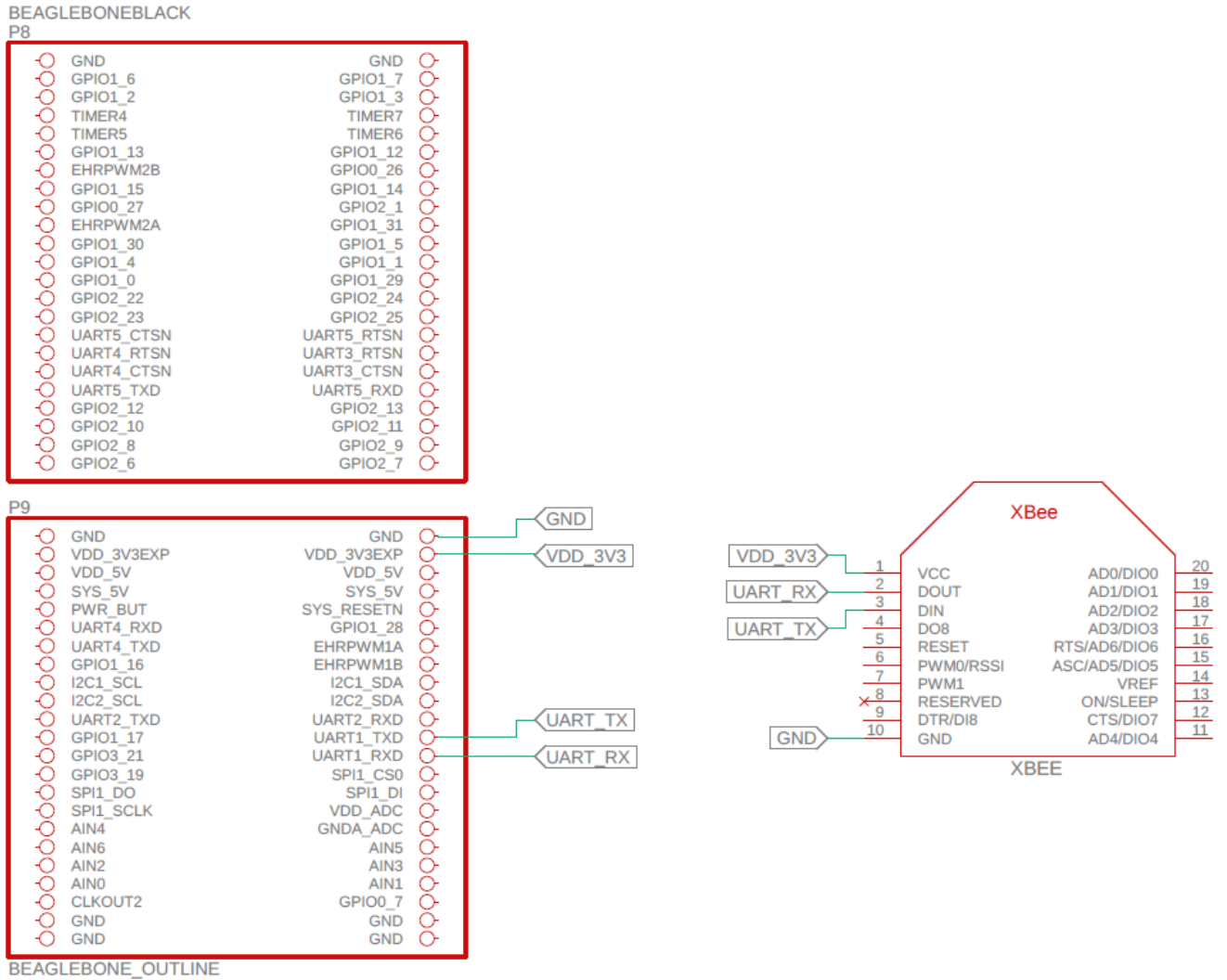
```
$sudo reboot
```

Si se quiere deshabilitar el servicio, basta con utilizar:

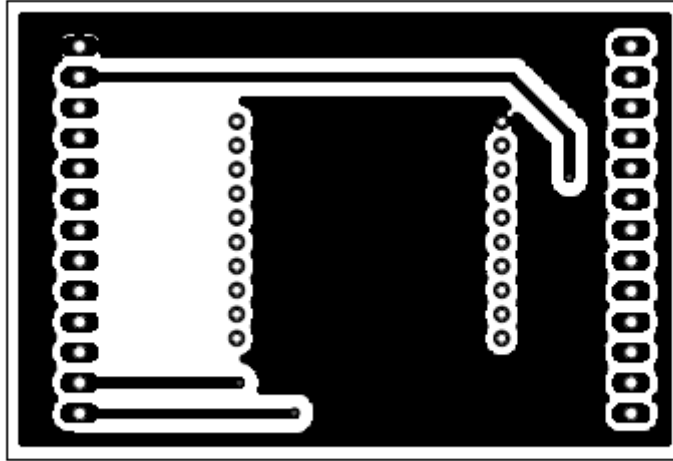
```
$sudo systemctl disable sample.service
```



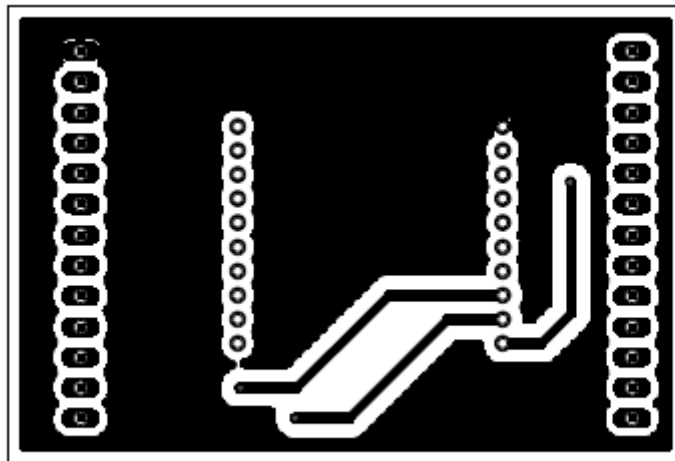
## ANEXO II: PLANOS ELECTRÓNICOS: XBEE - BEAGLEBONE BLACK



Anexo II. 1. Esquemático de conexión Xbee - BeagleBone Black



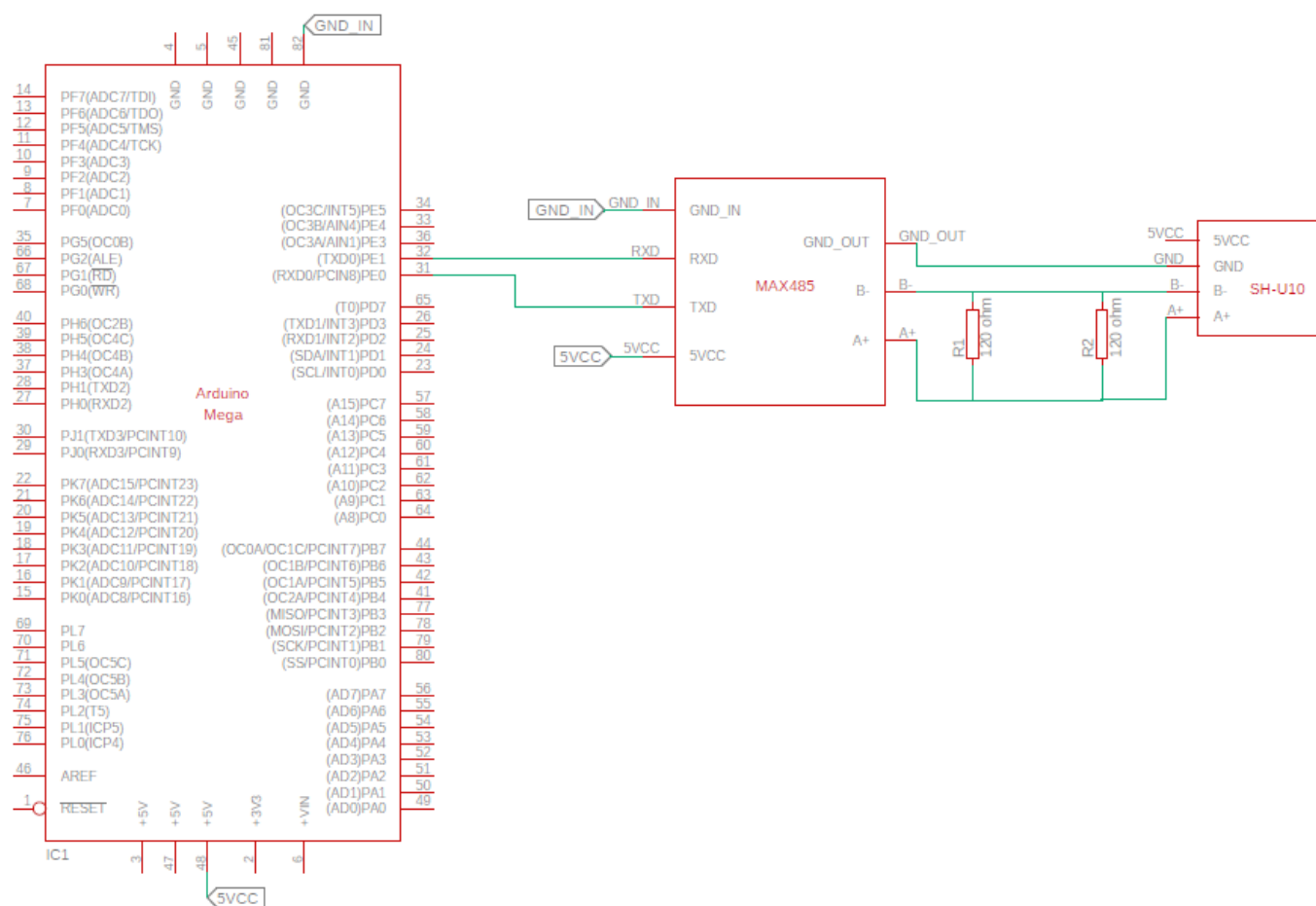
*Anexo II. 2. TOP*



*Anexo II. 3. Bottom*



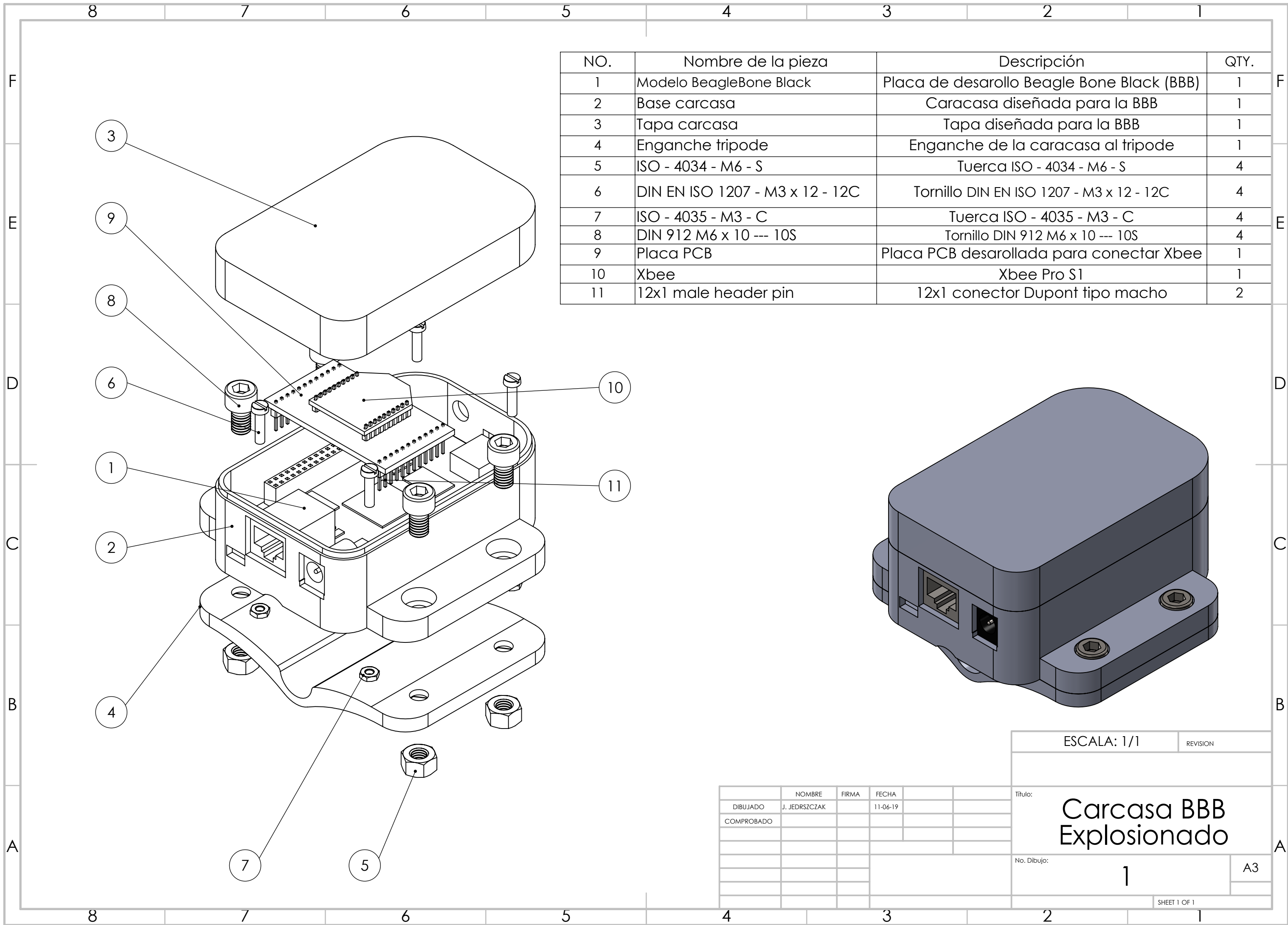
### ANEXO III: PLANOS ELECTRÓNICOS: RS485 – ARDUINO



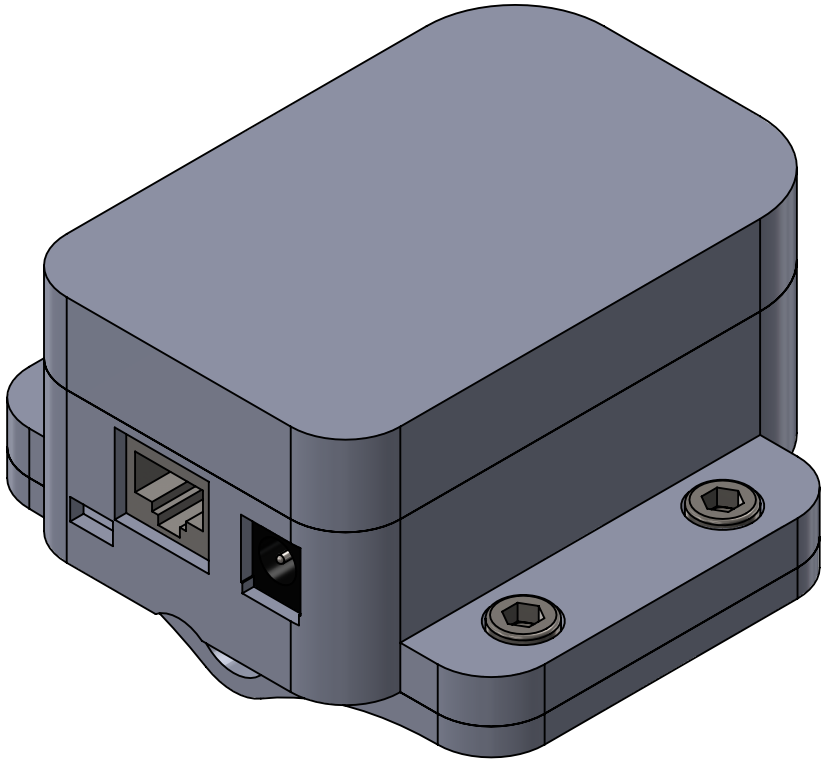
Anexo III. 1. Planos electrónicos: rs485 – Arduino



# **ANEXO IV: EXPLOSIONADO Y PLANOS DE LA CARCASA**

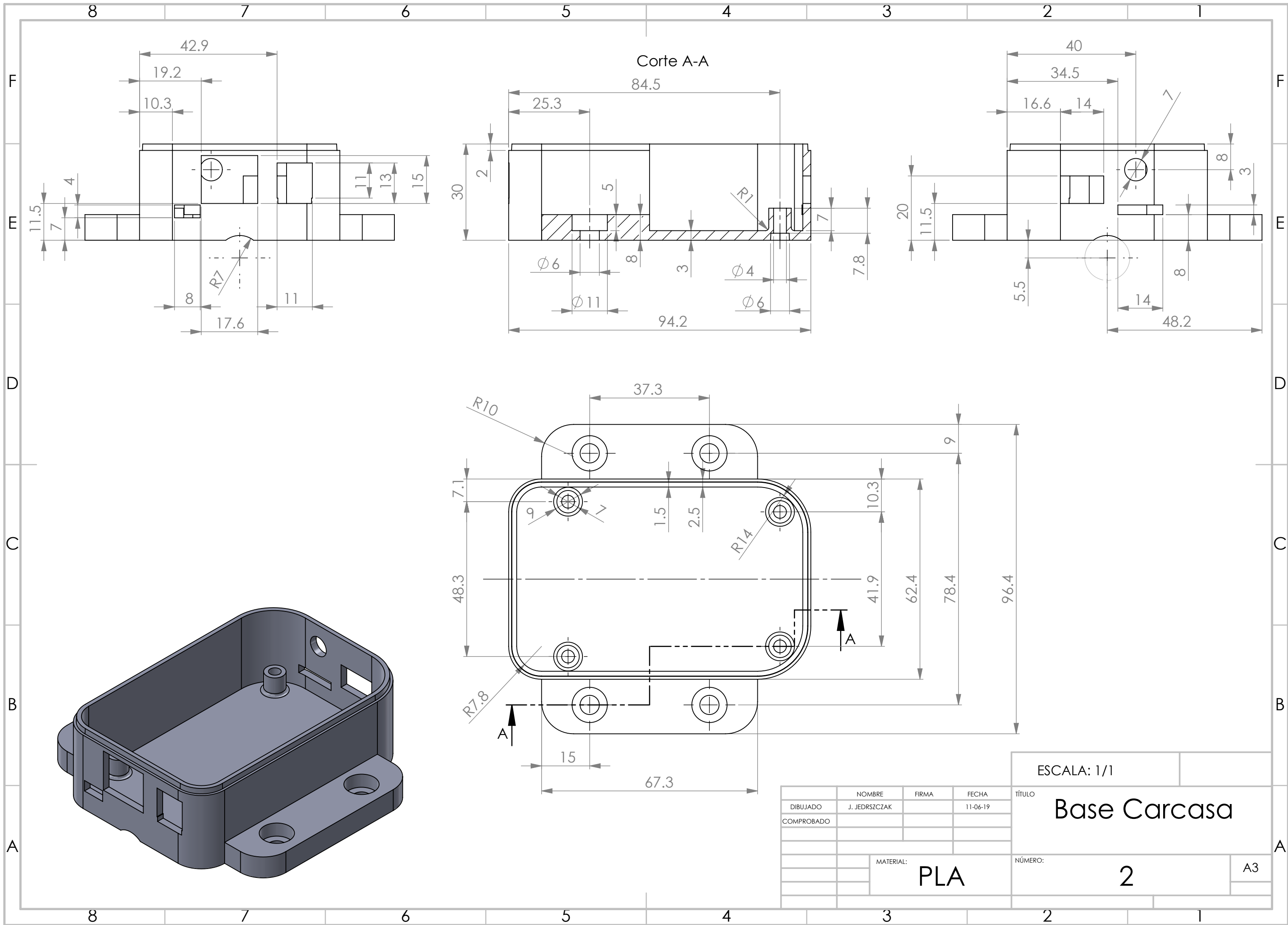


NO.	Nombre de la pieza	Descripción	QTY.
1	Modelo BeagleBone Black	Placa de desarrollo Beagle Bone Black (BBB)	1
2	Base carcasa	Caracasa diseñada para la BBB	1
3	Tapa carcasa	Tapa diseñada para la BBB	1
4	Enganche tripode	Enganche de la carcasa al tripode	1
5	ISO - 4034 - M6 - S	Tuerca ISO - 4034 - M6 - S	4
6	DIN EN ISO 1207 - M3 x 12 - 12C	Tornillo DIN EN ISO 1207 - M3 x 12 - 12C	4
7	ISO - 4035 - M3 - C	Tuerca ISO - 4035 - M3 - C	4
8	DIN 912 M6 x 10 --- 10S	Tornillo DIN 912 M6 x 10 --- 10S	4
9	Placa PCB	Placa PCB desarrollada para conectar Xbee	1
10	Xbee	Xbee Pro S1	1
11	12x1 male header pin	12x1 conector Dupont tipo macho	2



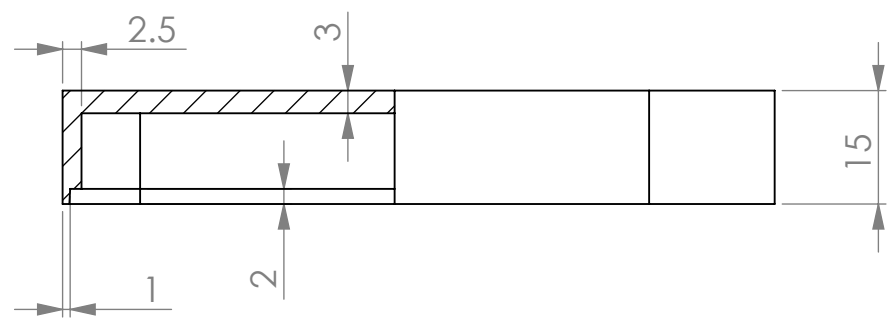
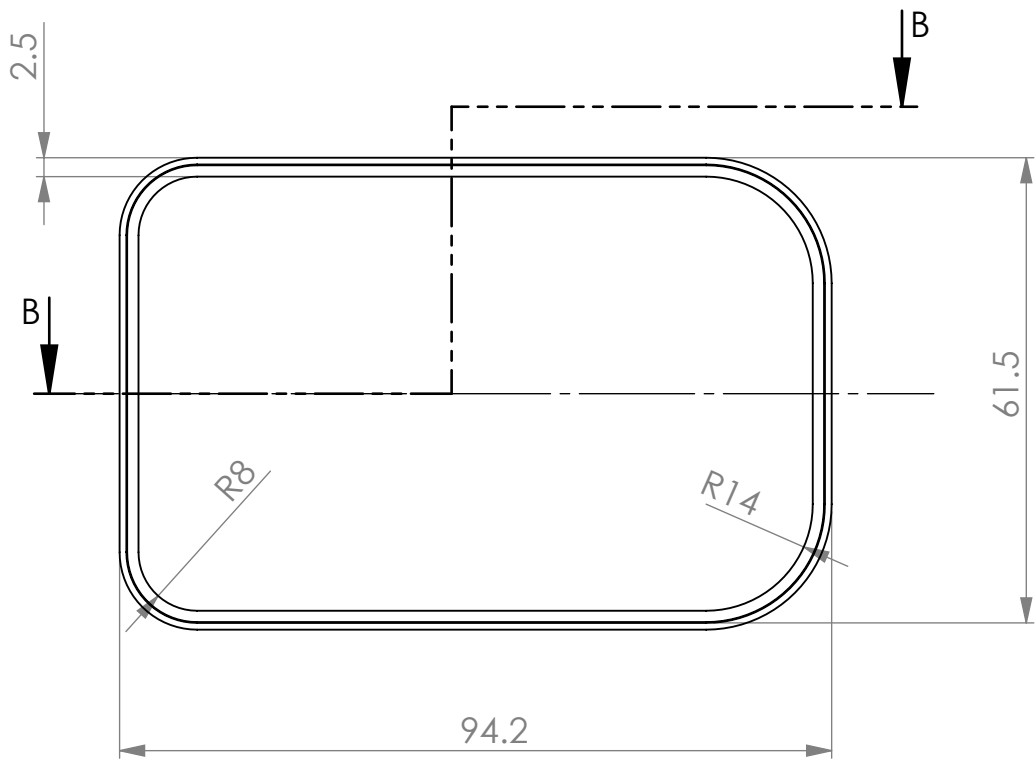
	NOMBRE	FIRMA	FECHA
DIBUJADO	J. JEDRSZCZAK		11-06-19
COMPROBADO			

ESCALA: 1/1	REVISION
Título: <b>Carcasa BBB Explosionado</b>	
No. Dibujo: <b>1</b>	A3
SHEET 1 OF 1	

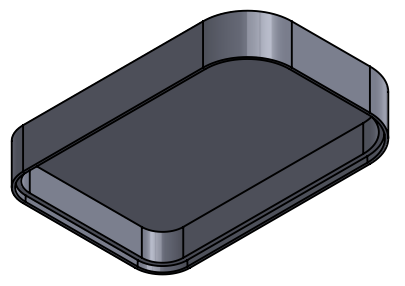
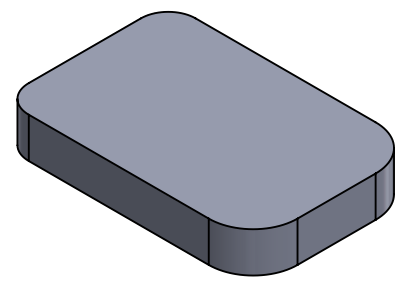


NOMBRE	FIRMA	FECHA
J. JEDRSZCZAK		11-06-19
COMPROBADO		
MATERIAL: <b>PLA</b>		

ESCALA: 1/1		TÍTULO
		<b>Base Carcasa</b>
NÚMERO:		A3
<b>2</b>		

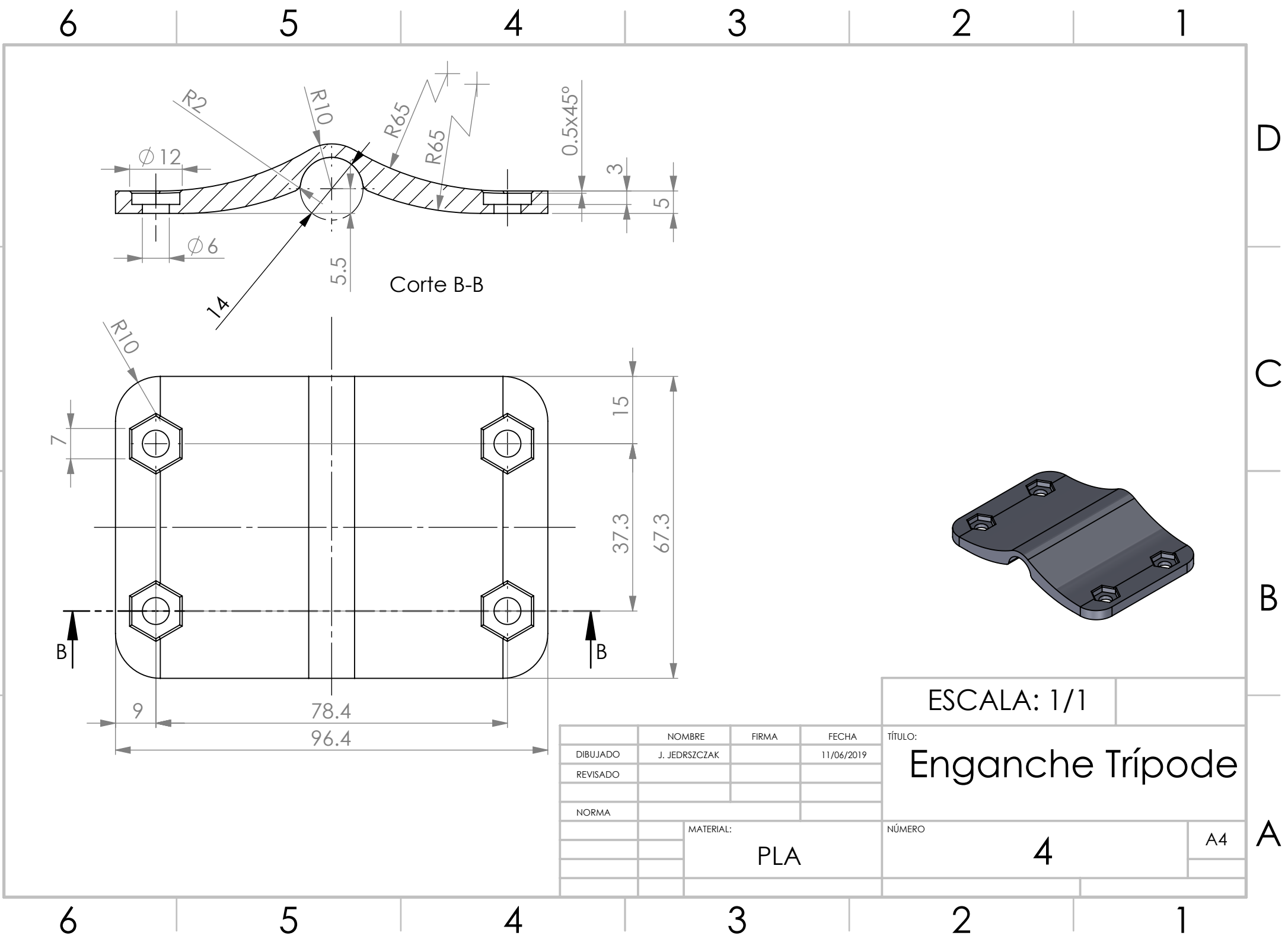


Corte B-B



ESCALA: 1/1		TÍTULO:	
		Tapa Carcasa	
NÚMERO:		A4	
3			

	NOMBRE	FIRMA	FECHA
DIBUJADO	J. JEDRSCZAK		11/06/2019
REVISADO			
NORMA			
MATERIAL:		PLA	



Corte B-B

ESCALA: 1/1

TÍTULO:  
Enganche Trípode

	NOMBRE	FIRMA	FECHA
DIBUJADO	J. JEDRSZCZAK		11/06/2019
REVISADO			
NORMA			
	MATERIAL:		
	PLA		

NÚMERO	4	A4
--------	---	----