



Universidad de Oviedo

Memoria del Trabajo Fin de Máster realizado por

JAVIER DÍEZ FERNÁNDEZ

para la obtención del título de

Máster en Ingeniería de Automatización e Informática Industrial

**SISTEMA DE MONITORIZACIÓN INTELIGENTE
DE MAQUINARIA INDUSTRIAL**

Mayo de 2018

Resumen

Las nuevas tecnologías siguen abriéndose camino día a día en el sector industrial, permitiendo alcanzar puntos casi inimaginables y mejoras que favorecen no solo a la productividad, sino también a la eficiencia¹, eficacia² y calidad de los productos desarrollados. El problema que presenta mucha de la maquinaria de hoy en día, es la desactualización en lo que a sistemas autónomos se refiere, es decir, aunque siguen estando operativas y realizan los trabajos adecuadamente, los nuevos equipos incorporan sistemas inteligentes que permiten lograr grandes mejoras en la producción.

En este proyecto se va a proceder a la mejora, mediante tecnología “*Low Cost*”, de un centro de mecanizado industrial aún útil y que puede seguir trabajando de forma que, en función de parámetros como el consumo eléctrico, el sonido y las vibraciones o aceleraciones de las máquinas, se pueda desarrollar un sistema predictivo de la rotura de las herramientas, evitando de esta forma gastos innecesarios en la producción de piezas. De la misma forma, toda la información estará disponible en una base de datos que permitirá realizar consultas y análisis en tiempo real.

Con este proyecto se pretende introducir la maquinaria en la nueva era del “Internet de las Cosas” (*IoT*), la cual permite la conectividad con los equipos y poder visualizar todos los datos anteriormente mencionados.

¹ Capacidad de cumplimiento de los objetivos con el menor número de recursos posible.

² Capacidad de cumplimiento de los objetivos planificados.

Agradecimientos

Este proyecto ha supuesto un gran trabajo y dedicación que ha sido completado gracias a la ayuda de todo un equipo que me ha aconsejado y enseñado en todo momento lo necesario para lograr finalizarlo con éxito.

En primer lugar, dar las gracias a Fernando Jesús (mi tutor en la Fundación Pro dintec) que me ha ayudado en los momentos de más complicación y siempre aconsejando y enseñando nuevos conocimientos, lo cuales serán de gran utilidad en el futuro.

En segundo lugar, agradecer a todo el resto de compañeros sus consejos e integrarme desde el primer día como uno más del equipo.

Finalmente, agradecer a José María Enguita (mi tutor académico) su esfuerzo y ayuda a la hora de organizar y desarrollar este documento.

Índice de figuras

Figura 1. Diagrama del alcance del proyecto	20
Figura 2. Diagrama de bloques del Sistema Global	21
Figura 3. Diagrama de bloques del Análisis de Datos.....	23
Figura 4. Centro de mecanizado instalado	24
Figura 5. Industria 4.0	27
Figura 6. Representación de dispositivos conectados	28
Figura 7. Estructura de un sistema conectado a una Base de Datos.....	30
Figura 8. Concepto gráfico de sistema de análisis de datos	32
Figura 9. Concepto de Big Data	34
Figura 10. Diagrama de una Señal Eléctrica Alterna	35
Figura 11. Elementos para Rectificación e Inversión de la señal.....	36
Figura 12. Señal Eléctrica Rectificada e Invertida	36
Figura 13. Diagrama de Bloques del Sistema de Medición de Consumo Eléctrico	37
Figura 14. Esquema eléctrico general.....	38
Figura 15. Esquema Eléctrico del Regulador de Tensión	38
Figura 16. Máquina de fabricación de PCB's	39
Figura 17. PCB fabricada	39
Figura 18. PCB conectada a Arduino	40
Figura 19. Módulo wifi ESP-05	40
Figura 20. Dispositivo instalado en máquina	41
Figura 21. Barra de herramientas Altium	42
Figura 22. Configuración de parámetros. Parte 1	43
Figura 23. Configuración de parámetros. Parte 2	44
Figura 24. Enrutado manual	44
Figura 25. Comparativa de enrutado	45
Figura 26. PCB de medición de consumos.....	45
Figura 27. Técnicas de Monitorización	49
Figura 28. Diagrama del Sistema de Captación de Sonido	50
Figura 29. USB-6212 Multifunción	50
Figura 30. Micrófono para Escucha	51
Figura 31. Menú principal de EAUDIO	52
Figura 32. Menú de captura de audio	53

Figura 33. Configuraciones desarrolladas	54
Figura 34. Driver de conexión.....	54
Figura 35. Configuración de conexión MySQL	55
Figura 36. Diagrama de bloques de conexión a MySQL	56
Figura 37. Panel frontal	56
Figura 38. Esquema y gráfica del chatter	57
Figura 39. Diagrama de bloques del Sistema de Aceleraciones.....	59
Figura 40. Placa con Acelerómetro y esquema del mismo.....	60
Figura 41. WEMOS D1 Mini Pro.....	60
Figura 42. Placa de carga.....	61
Figura 43. Dispositivo de medición de vibraciones	61
Figura 44. Portaherramientas con dispositivo de medición de vibraciones	62
Figura 45. Diagrama de la estructura Balanceador de Carga	64
Figura 46. Diagrama de funcionamiento del sistema LAMP	65
Figura 47. Clúster	66
Figura 48. Raspberry Pi 3 modelo B	66
Figura 49. Relaciones existentes en la Base de Datos.....	67
Figura 50. Base de Datos desarrollada	68
Figura 51. Tabla de consumos.....	69
Figura 52. Tabla de aceleraciones/vibraciones.....	69
Figura 53. Tabla de sonido	70
Figura 54. Ruta para establecer la conexión.....	71
Figura 55. Establecimiento de conexión	71
Figura 56. Creación de un esquema nuevo.....	72
Figura 57. Ventana de características del esquema	72
Figura 58. Añadir Cubo nuevo	72
Figura 59. Ventana de características del cubo	73
Figura 60. Tabla de Hechos.....	73
Figura 61. Esquema general de las dimensiones creadas	74
Figura 62. Configuración del elemento "Table" de la dimensión "Maquinaria"	75
Figura 63. Configuración del elemento "Nivel" de la dimensión "Maquinaria"	75
Figura 64. Configuración del elemento "Hierarchy" de la dimensión "Maquinaria"	76
Figura 65. Configuración de la dimensión "Maquinaria".....	77
Figura 66. Configuración del elemento "Table" de la dimensión "Fechas"	77

Figura 67. Configuración del elemento "Nivel" de la dimensión "Fechas"	78
Figura 68. Configuración del elemento "Hierarchy" de la dimensión "Fechas"	79
Figura 69. Configuración de la dimensión "Fechas"	79
Figura 70. Configuración del elemento "Table" de la dimensión "Potencia"	80
Figura 71. Configuración del elemento "Nivel" de la dimensión "Potencia"	80
Figura 72. Configuración del elemento "Hierarchy" de la dimensión "Potencia"	81
Figura 73. Configuración de la dimensión "Potencia"	82
Figura 74. Configuración del indicador.....	82
Figura 75. Opción para publicar	83
Figura 76. Ventana para realizar la publicación	83
Figura 77. Corriente eléctrica medida en el canal 1	84
Figura 78. Corriente eléctrica medida en el canal 2	85
Figura 79. Corriente eléctrica medida en el canal 3	85
Figura 80. Potencia total consumida.....	86
Figura 81. Medición de aceleraciones en μg	87
Figura 82. Medición de señales acústicas 1	88
Figura 83. Medición de señales acústicas 2.....	89
Figura 84. Propuesta de Interfaz.....	90
Figura 85. Diagrama de Gantt principal	93
Figura 86. Diagrama de Gantt secundario	94
Figura 87. Esquema Raspberry Pi 3 modelo B.....	107
Figura 88. Configuraciones del Clúster	121

Índice de tablas

Tabla 1. Identificación del trabajo	10
Tabla 2. Componentes para medición de consumos.....	42
Tabla 3. Componentes para medición de sonido	51
Tabla 4. Componentes para medición de vibraciones	62
Tabla 5. Tabla de tareas principales	92
Tabla 6. Tabla de tareas secundarias	94
Tabla 7. Cuadro de precios de los recursos hardware	96
Tabla 8. Cuadro de precios de los recursos software	96
Tabla 9. Cuadro de precios de los recursos humanos.....	96
Tabla 10. Presupuesto parcial de los recursos hardware	98

Tabla 11. Presupuesto parcial de los recursos software	98
Tabla 12. Presupuesto parcial de los recursos humanos.....	99
Tabla 13. Presupuesto final	99
Tabla 14. Especificaciones técnicas Deckel Maho.....	104
Tabla 15. Tabla de características NI USB-6212	105
Tabla 16. Tabla de características ADXL362	105
Tabla 17. Tabla de características WEMOS D1 Mini Pro	106
Tabla 18. Tabala de características Raspberry Pi 3 modelo B	106

Índice general

1	Introducción	10
1.1	Identificación del proyecto	10
1.2	Visión general del documento	10
1.3	Visión general del proyecto	11
1.4	Contexto histórico.....	12
1.4.1	Siglo XIX.....	12
1.4.2	Siglo XX.....	14
1.5	Situación actual de los centros de mecanizado	17
1.6	Objetivo	19
1.7	Alcance	19
2	Descripción del sistema.....	20
3	Descripción de maquinaria.....	24
3.1	Problemática en la desactualización de maquinaria.....	24
4	Prodintec, el entorno de trabajo y la innovación.....	25
4.1	Entorno de trabajo.....	25
4.1.1	Innovación	25
5	Internet de las cosas (IoT) y tecnologías relevantes.....	26
5.1	Lenguajes de programación	28
5.2	Bases de datos	30
5.2.1	MySQL.....	31
5.3	Sistema de análisis de datos.....	31
5.3.1	Cubos OLAP	32
6	Big data... ..	33
7	Desarrollo del proyecto	35
7.1	Desarrollo de sistema de medición de consumos	35
7.1.1	Desarrollo hardware	36
7.1.2	Desarrollo software.....	46
7.2	Desarrollo de sistema de medición de audio	48

7.2.1	Desarrollo hardware	49
7.2.2	Desarrollo software.....	51
7.3	Desarrollo de sistema de medición de vibraciones/aceleraciones	57
7.3.1	Desarrollo hardware	58
7.3.2	Desarrollo software.....	62
7.4	Sistema web de transferencia y tratamiento de datos	64
7.5	Desarrollo de base de datos MySQL	66
7.6	Desarrollo de sistema predictivo. Cubos OLAP	70
8	Resultados	84
9	Ampliaciones futuras	89
10	Planificación y presupuesto	92
10.1	Planificación	92
10.2	Presupuesto	95
10.2.1	Cuadro de precios	95
10.2.2	Presupuestos parciales	97
10.2.3	Presupuesto final	99
11	Conclusiones	100
12	Referencias.....	101
ANEXO A.	Especificaciones técnicas Deckel Maho	104
ANEXO B.	Especificaciones técnicas ni usb-6212.....	104
ANEXO C.	Especificaciones técnicas del acelerómetro	105
ANEXO D.	Especificaciones técnicas placa WEMOS D1 Mini Pro	105
ANEXO E.	Especificaciones técnicas Raspberry PI.....	106
ANEXO F.	Programación con Arduino. Medición de consumos.....	107
ANEXO G.	Programación con Arduino. Medición de vibraciones	111
ANEXO H.	Programación de archivo PHP para almacenar los datos de consumo en la base de datos.....	113
ANEXO I.	Programación de archivo PHP para almacenar los datos de aceleraciones en la base de datos.....	114
ANEXO J.	Programación de archivo PHP para almacenar los datos de sonido en la base de datos.....	119
ANEXO K.	Programación del clúster	120

1 Introducción

1.1 Identificación del proyecto

El actual Trabajo de Fin de Máster tiene como objetivo la finalización del Máster en Automatización e Informática Industrial según el plan de estudios de 2012. En la tabla 1, se muestran los datos de identificación del trabajo.

Título	Sistema de Monitorización Inteligente de Maquinaria Industrial
Empresa proponente	Fundación Pro dintec
Tutor en empresa	Fernando Jesús García Díaz
Tutor académico	José María Enguita González
Alumno proyectante	Javier Díez Fernández
D.N.I	
Correo electrónico	UO260271@uniovi.es
Fecha	Mayo de 2018

Tabla 1. Identificación del trabajo

1.2 Visión general del documento

El conjunto de páginas que forman este documento constituyen la memoria del Trabajo de Fin de Máster realizado el cual, consta de los siguientes apartados:

- ✚ El apartado 1 supone la introducción del trabajo, incluyendo los datos de identificación del trabajo, una visión general del mismo, el contexto histórico y situación actual de los centros de mecanizado, el objetivo principal y el alcance del trabajo.
- ✚ El apartado 2 incluye la descripción del sistema desarrollado, indicando su funcionamiento y los elementos (sensores, actuadores, etc.) que lo forman.
- ✚ En el apartado 3 se realiza la descripción de la máquina sobre la que se ha trabajado, así como los problemas de desactualización a los que se enfrentaba.
- ✚ El apartado 4 recoge la información más relevante sobre el entorno y lugar en los que se ha estado trabajando.
- ✚ El apartado 5 recoge conceptos sobre el “Internet de las cosas” y las tecnologías más relevantes asociadas.

- ✚ En el apartado 6 se explica el concepto de *Big Data* y lo que supone para el paradigma de la “Industria 4.0”.
- ✚ En el apartado 7 se especifica todo el proceso de desarrollo de los diferentes sistemas individuales, tanto a nivel de *hardware* como de *software*. Además, se incluye el proceso de creación de la base de datos y del equipo para transferir la información a la misma.
- ✚ En el apartado 8 se presenta un análisis de los resultados obtenidos.
- ✚ El apartado 9 recoge una serie de ampliaciones que se llevarán a cabo en el futuro con el fin de optimizar aún más el sistema.
- ✚ En el apartado 10 se recogen la planificación de las tareas principales y secundarias, y el presupuesto del trabajo, incluyendo un cuadro de precios, el presupuesto parcial y el presupuesto final.
- ✚ En el apartado 11 se presentan las conclusiones del trabajo.
- ✚ En el apartado 12 aparecen todas las referencias que han sido consultadas.
- ✚ Finalmente se incorporan una serie de anexos que recogen especificaciones sobre la máquina de trabajo y alguno de los componentes empleados, así como la programación informática y configuraciones que han sido elaboradas.

1.3 Visión general del proyecto

Hoy en día, la situación de poder estar conectado constantemente a la red supone un gran avance para todo el sector industrial. Y no solo la conectividad es un factor revolucionario, durante los últimos años ha surgido el concepto de análisis de datos, más conocido como “*Big Data*”, lo que permite que, a través del tratamiento de los datos recabados se puedan estimar predicciones en los comportamientos de las máquinas. Así, interpretando los datos de forma adecuada puede llegarse a ahorrar grandes cantidades de dinero en la producción. Sin embargo, la gran mayoría de las empresas que comenzaron su funcionamiento varios años atrás se ven limitados en lo que a conectividad “Hombre-Máquina” se refiere. Por este motivo, surge la necesidad de adaptarse a los nuevos tiempos, pero el problema que se encuentran las empresas es que, en la mayoría de los casos, es necesaria la adquisición de nueva maquinaria o la adecuación/inserción de dispositivos a precios desorbitados. No se debe olvidar que las empresas tienen como objetivo principal ganar dinero, por lo que las soluciones que pretenden incorporar son aquellas que menos inversión necesitan y más beneficios reportan.

A continuación, se presenta el contexto histórico del sector del mecanizado industrial (Aldabaldetrecu, 2002), su estado actual (Oscar Gonzalo, 2010), el objetivo a cumplir y el alcance del proyecto.

1.4 Contexto histórico

1.4.1 Siglo XIX

El mecanizado comenzó su desarrollo con la llega de la revolución industrial en el siglo XIX, tardando varios años en establecerse como un arte más. El desarrollo pudo ser posible gracias a la evolución de las herramientas de corte ya que, en sus inicios, estas eran un poco más duras que los elementos con los que iban a trabajar: hierro forjado, bronce y fundición gris.

Fue en 1800 cuando **Mudslay** construyó el primer “torno” totalmente metálico con “husillo guía patrón” para roscar tornillos. Para lograr mediciones precisas desarrolló un “micrómetro de tornillo” (1805) el cual, conseguía medir la milésima parte de la pulgada³. Otra de sus creaciones a destacar, es la construcción en 1803 de la primera “amortajadora vertical” para sacar chaveteros a poleas y engranajes. Sus influencias marcaron este siglo en lo que al desarrollo de máquinas se refiere.

En 1817, **Richard Roberts** fabricó el primer “cepillo puente práctico” empleado a nivel industrial, que incorpora una guía en V y otra plana para el movimiento de la mesa porta piezas. En los años venideros, este elemento sufrió mejoras por parte de otras personas del sector como **Whitworth**. Posteriormente, en el año 1836, **James Nasmyth** diseñó y construyó la primera limadora.

En torno al 1817, **Dietrich Uhlhöm** diseñó una “prensa acodada” para la acuñación de monedas. En los próximos años, surgieron diseños más nuevos que incluían mejoras significativas, como la del francés **Thonelier**, que incluye el sistema de “virola partida”, o la “prensa mecánica de fricción” presentada en 1867 por otro francés, **Cheret**.

En 1818 se construyó la primera máquina de fresado, de la mano del americano **Ely Whitney**, y que estaba formada por un caparazón de madera sujetado por cuatro patas de hierro forjado. Era una máquina que constaba de una mesa porta piezas con movimiento longitudinal sobre guías en forma de “cola de milano” y un “eje sinfín” que podía embragarse/desembragarse sobre una corona dentada situada en el husillo del carro. Es una de las máquinas que más cambios ha experimentado a lo largo de los años, dotándola cada

³ Una pulgada equivale a 25,40 mm.

vez de mayores elementos que ofrecían mayores precisiones y calidades en el proceso de mecanizado. Entre las aportaciones más importantes destaca la fresadora universal construida por **J. R. Brown**, que está dotada con divisor, consola con desplazamiento vertical, curso transversal y avance automático de la mesa con el sistema de la “transmisión Cardan”. Otra de las innovaciones que pueden ser destacadas es la construcción en 1894 por parte del francés **P. Huré** de una fresadora con un cabezal que permitía variedad de movimientos: horizontal, vertical, etc.

Ante la necesidad de taladrar materiales, en 1838 **Nasmyth** construyó un taladro de sobremesa metálica que incorporaba un sistema de “giro de eje” porta brocas, que podía ser accionado manualmente o por transmisión. Posteriormente, se construyeron máquinas con mejoras implementadas, entre las que destaca la construida en 1850 **Whitworth**, caracterizada por tener un taladro de columna cuyo accionamiento se realizaba por “transmisión a correa” y “giro del eje” porta brocas, todo ello mediante un conjunto de “engranajes cónicos”. Posteriormente, en 1860 tuvo lugar un gran acontecimiento, la fabricación por parte de **Martignon** de la “broca helicoidal”. Dicha broca supuso una gran evolución en la producción y duración de la herramienta.

Whitworth fue uno de los grandes fabricantes de máquinas y herramientas del siglo, de forma que su perfeccionamiento del torno paralelo en 1850 sigue siendo el vigente hasta nuestros días, con la diferencia que en 1890 fue incorporada la “Caja Norton” por parte de los americanos. También destacó por desarrollar un “sistema de roscas” (el cual lleva su nombre), que fue adoptado en 1841 por el “*Institute of Civil Engineers*” de Inglaterra.

Hasta aproximadamente 1850, los ingleses fueron los líderes en la fabricación de maquinaria ligera. Sin embargo, a partir de entonces se dedicaron más profundamente en la fabricación de maquinaria pesada, para poder solucionar la fabricación de piezas para el ferrocarril. Fue entonces cuando los americanos tomaron el relevo y consiguieron alzarse como líderes a nivel mundial en la fabricación de maquinaria ligera. Cabe destacar que los americanos optaron por el “*Sistema Sellar*”, en el año 1868, como normalización.

Ante la aparición de la necesidad de realizar varias operaciones sobre una pieza, se comenzaron a incorporar, alrededor de 1854, “torretas revolver” para la fabricación de tornillería y piezas de revolución. Sin embargo, fue en 1860 cuando empresas como “*Brown & Sharpe*” y “*Pratt & Whitney*” comenzaron a fabricar de manera regular este tipo de máquinas. A partir de 1870, empezaron a añadirse tornos automáticos con el fin de conseguir

solucionar los problemas derivados de la producción en grandes series de piezas pequeñas de revolución. Finalmente, en 1898 la empresa “*Pratt & Whitney*” construyó el primer torno automático y, “*The National Acme*” el primer torno multihusillo.

Como puede observarse, a partir de 1865 las máquinas aumentaron sus prestaciones ya que, las herramientas con las que se operaba habían sido notablemente mejoradas. Con la llegada del “acero *Mushet*” (acero aleado con manganeso, tungsteno y carbono) se lograron herramientas de corte más resistentes al desgaste y que soportaban mayores temperaturas de corte, permitiendo trabajar a velocidades de corte del orden de 10 m/min.

En 1870, la empresa “*Brown & Sharpe*” construyó la primera rectificadora universal. Sin embargo, hasta 1880 no puede ser tratada como tal, ya que fue entonces cuando fue dotada de un dispositivo que permitía la rectificación interna. Aun así, la verdadera etapa de desarrollo del rectificado no tuvo lugar hasta finales de siglo, ya que fue entonces que, por exigencias del sector automovilístico y al descubrimiento del carburo de silicio, fue necesaria su implementación. En esta recta final de siglo, muchas empresas como “*Churchil*”, “*Cincinnati*”, “*Norton*”, etc. ya habían desarrollado máquinas rectificadoras.

El fin de siglo llegó marcado por el descubrimiento, por parte de **Taylor** y **White**, del acero rápido. Este material permitió fabricar nuevas herramientas que dotaban a los procesos de mecanizado con velocidades que triplicaban a las que se usaban con anterioridad. Además, se aumentó la capacidad de arranque de viruta en un orden de siete.

1.4.2 Siglo XX

El empleo del sistema de generación polifásica desarrollado por **Tesla** en 1887, permitió emplear la electricidad a nivel industrial dando lugar a mejoras de los motores de corriente continua y alterna. De esta forma, poco a poco fueron adaptándose a la maquinaria industrial sustituyendo a las máquinas de vapor. El hecho de poder disponer de electricidad marcó la gran evolución del sector. Fue un siglo en el que evolucionaron especialmente los sectores automovilístico y aeronáutico.

Ante la llegada de la evolución, cada vez eran mayores los requisitos en cuanto a precisiones se refiere ya que, se exigían mejores calidades en el proceso productivo. Por ello, se desarrollaron micrómetros, cuyo sector dominaba Estados Unidos, que conseguían precisiones del orden de milésimas de metro.

Con el paso de los años fue necesaria la fabricación en masa, siendo en 1908 cuando **Henry Ford** fabrica el primer automóvil producido en serie y, en 1911 instala el primer transportador en cadena.

La evolución en la producción no fue sólo por la incorporación de mejoras en la maquinaria, como cabezales de cojinetes o rodamientos de bolas, sino que gracias y especialmente al desarrollo de nuevos materiales para la fabricación de herramientas se logró un aumento aún más notable. A continuación, se presentan los principales materiales con los que se desarrollan las herramientas de fabricación (Alcañiz, 2013).

Aceros rápidos (HSS)

Desarrollado por **Taylor y White**, el acero rápido es una aleación que cuenta con cromo y wolframio. De esta forma, se incrementa su contenido en carburos duros los cuales son resistentes al calor generado en su utilización. Es un material que permite realizar trabajos a altas velocidades de corte debido a que no pierde su dureza hasta los 600 °C. Durante los años treinta fueron mejorados convirtiéndose en “aceros extra rápidos” (HSS-E), permitiendo operar a velocidades de 70 m/min.

Metal duro (WIDIA)

Descubierto en 1926 por la empresa “*Krupp*” y presentado en 1927 con el seudónimo de “*Widia*”, el metal duro es una aleación con el 90% de carburos duros (carburo de tungsteno en sus inicios) y cobalto como material aglutinante. En sus inicios, las herramientas fabricadas con este material presentaban una rotura rápida del filo. Sin embargo, con el paso de los años fueron incorporándose mejoras como son la adición de partículas duras (carburos de titanio, tantalio y niobio) y el incremento de la tenacidad.

Cerámica

Las herramientas cerámicas se caracterizan por estar fabricadas por compuestos en los que predominan la alúmina (óxido de aluminio) y el nitruro de silicio. Entre las características principales, pueden destacarse la posibilidad de realizar trabajos a altas velocidades de corte y su resistencia a temperaturas de hasta 1500 °C. Por el contrario, presenta el inconveniente de ser muy frágil.

Nitruro de boro cúbico (CBN)

Descubierto en 1957 por **R.H.Wentorf** (General Electric), el nitruro de boro cúbico destaca por ser un material que mantiene su dureza hasta los 2000 °C. Esto es debido a su estructura

crystalina, la cual es similar a la del diamante. Como aglutinante se emplea nitruro de titanio o cerámica.

Diamante policristalino (PCD)

En torno a principios de la década de los años 70 hizo su aparición este compuesto el cual, es considerado como el más duro por lo que su resistencia a la abrasión es muy elevada. Sin embargo, como punto débil destaca su extremada fragilidad (no puede superar los 600 °C) y la imposibilidad de mecanizar elementos férricos (debido a su afinidad química). Es un buen material para operaciones en las que se necesitan buenos acabados superficiales y precisión.

Tras este análisis de las herramientas de mecanizado, se retoma el estudio de la evolución del proceso de mecanizado.

A partir de 1927, se realizó la automatización de algunas de las operaciones que debían realizar las máquinas. Todo ello fue posible gracias a la incorporación de motores eléctricos y sistemas hidráulicos, neumáticos y eléctricos.

En 1943 surgió un nuevo concepto revolucionario, el mecanizado por electroerosión. Fue en este año cuando se desarrollaron los primeros dispositivos para tal fin, pero fue en 1950 cuando aparecieron las primeras máquinas las cuales, incorporaban un generador, un depósito para el dieléctrico y electrodo con el molde a mecanizar. Cinco años más tarde surgieron las primeras máquinas consideradas como tal para realizar mecanizados por penetración. Con el paso de los años y la introducción del control numérico, nació la electroerosión por hilo.

Con la llegada de la electrónica y la informática, comenzaron a introducirse en el mecanizado las primeras computadoras electrónicas digitales, siendo la primera la conocida “ENAC”, desarrollada por **John W. Manclhy** y **J. Presper Ecker** (Universidad de Pennsylvania). Aun así, fue a partir de los años sesenta cuando hizo su aparición el control numérico por computadora (CNC), gracias al desarrollo de la microelectrónica. Durante los años ochenta se implantó este sistema de manera generalizada, dando lugar a una nueva era del mecanizado y, en el que incluso hoy en día se sigue avanzando.

El “CNC” (Control Numérico Computarizado) se incorporó en varias de las máquinas existentes (fresadoras, tornos, taladros, etc.) pero, entonces surgió la idea de una automatización superior, naciendo así el centro de mecanizado. El centro de mecanizado es un mecanismo que incorpora los procesos de fresado, taladrado, mandrinado, etc. y que incluye un almacén con las herramientas necesarias para cada operación junto con un sistema

automático de intercambio de las mismas, siendo el control numérico computacional el encargado de ajustar posiciones, velocidades de corte, giro de herramientas, etc.

A partir de entonces, el control numérico computacional ha asumido el papel principal de los centros de mecanizado, dando lugar al concepto de máquina-herramienta avanzada.

En el apartado siguiente, se analizará más profundamente los aspectos relativos a los centros de mecanizado de hoy en día.

1.5 Situación actual de los centros de mecanizado

Los centros de mecanizado de hoy en día están totalmente automatizados, a excepción de las configuraciones que deben ser realizadas por la persona con los conocimientos adecuados. Estos sistemas incorporan la tecnología “CNC” (Control Numérico Computarizado), de forma que aumentan notablemente la producción industrial, la flexibilidad y la precisión de trabajo.

También cabe destacar su versatilidad; los buenos acabados superficiales de las piezas; la producción uniforme, lo que permite que sean empleados en fabricaciones en serie; la capacidad para realizar varias operaciones sobre una misma pieza en un mismo proceso.

Generalmente los centros de mecanizado están preparados para desempeñar trabajos con herramientas de cortes rotatorios, como brocas y cortadores. Para ello, la máquina cuenta con un almacén con las diferentes herramientas que son necesarias, según los procesos que vaya a realizar. Además, el cambio lo realiza de forma automática sin necesidad de intervención por parte del usuario.

A la hora de diseñar los centros de mecanizado modernos, los fabricantes tienen en cuenta los materiales con los que van a construirlos de forma que:

- sea un mecanismo rígido, con el fin de minimizar deformaciones y vibraciones,
- se eviten deformaciones debidas al calor producido en las distintas operaciones,
- sea de una densidad baja, para minimizar fuerzas inerciales,
- tenga alta conductividad térmica, con el fin de disipar el calor de las operaciones lo más rápido posible,
- el coste sea el menor posible.

Otro de los aspectos a tener en cuenta en la actualidad es que existen dos tipos de centros de mecanizado: con husillo vertical y con husillo horizontal.

Centros de mecanizado con husillo vertical

Son máquinas con mucha rigidez y precisión que son utilizadas especialmente en trabajos con superficies planas y cavidades hondas como, por ejemplo, la fabricación de moldes. Estas máquinas son menos costosas que los centros de mecanizado con husillo horizontal.

Centros de mecanizado con husillo horizontal

Se utilizan para trabajar con piezas altas y de gran tamaño que requieren mecanizados en varias partes de su estructura. Una característica de estas máquinas es que, en ocasiones, la pieza a mecanizar es inclinada para adquirir ángulos de posición diferentes. Un ejemplo de este tipo de máquina es el centro de torneado.

De la misma forma que existen diferentes tipos de centros de mecanizado, también se encuentran diferentes tecnologías de fabricación. Así, cabe diferenciar tres tipos de tecnologías:

- 1. Mecanizado con arranque de viruta.** Esta tecnología emplea una herramienta de precisión para arrancar material de la pieza. Una de las tecnologías más empleadas y comunes es el fresado.
- 2. Mecanizado sin arranque de viruta.** No es una tecnología que complemente bien los conceptos de precisión y velocidad de producción, por lo que no suele usarse si se buscan estos dos parámetros. Destacan, entre otros, el laminado, estampado, fundición, etc.
- 3. Mecanizado por abrasión o electroerosión.** Es una tecnología que emplea un disco rotativo (también conocido como muela), que mediante el giro rápido de este desprende partículas pequeñas de material. Es un proceso muy lento pero que consigue precisiones muy elevadas.

Los centros de mecanizado más desarrollados hasta la fecha contienen multitud de sensores para garantizar un mecanizado más preciso, estable, flexible y rápido con el fin de conseguir un sistema productivo de la mayor calidad posible. Además, muchos de ellos también implementan sistemas de análisis del mecanizado y control remoto del proceso.

Poco a poco puede verse como las nuevas máquinas implementan la conectividad con ellas.

1.6 Objetivo

Debido a la evolución del sector del mecanizado en la industria hacia ámbitos cada vez más modernos que incluyen la implementación de nuevas tecnologías y equipos especializados, surgen necesidades empresariales para poder lograr estar al día en la cadena productiva, logrando así ser competitivos y alcanzar unos objetivos cada vez mayores y eficientes. Hay que recordar que se vive en un tiempo de auténtico “boom” tecnológico que afecta a todos los aspectos de la vida (social, empresarial, comercial, etc.), cuyos dispositivos asociados permiten facilitar la misma. Sin embargo, toda evolución conlleva a una serie de aumentos considerables en los costes asociados a la producción siendo, en este caso, aquellos relativos a la maquinaria física encargada de realizar las operaciones de fabricación.

Ante esta necesidad de actualización de la maquinaria industrial de hoy día, siendo esta aún viable para operar, lo que se pretende llevar a cabo con este proyecto es la **adaptación de un centro de mecanizado al paradigma de la “Industria 4.0”, mediante la incorporación de diferentes componentes y dispositivos de comunicación y un sistema capaz de optimizar los trabajos de mecanizado, ofreciendo una solución “Low Cost” (bajo coste), que permitirá a propietarios de máquinas similares su instalación sin costes desorbitados.**

1.7 Alcance

Para lograr diseñar, desarrollar e implementar el sistema mencionado anteriormente, existe un protocolo de actuación que incluye todas las tareas necesarias que permitirán cumplir con el objetivo del proyecto. Para ello, lo primero que debe hacerse es recopilar toda la información necesaria y estudiarla, con el fin de comprender muchos de los procedimientos, teorías y elementos que se planean usar. De esta forma, se determina la importancia de dicho análisis, ya que gracias a él se establecen las bases del proyecto sobre las que se van a trabajar. Una vez comprendidas cada una de las partes, se procede al diseño, desarrollo e implementación de tres tipos de dispositivos electrónicos: uno para medir los consumos eléctricos de la máquina, otro para medir las vibraciones/aceleraciones que sufre la herramienta de la máquina y un último para captar señales acústicas durante el proceso de fabricación. Tras lograr la implementación de cada sistema, se busca guardar toda la información relevante en una base de datos para, posteriormente, emplear los datos en el desarrollo de un sistema predictivo basado en “Cubos OLAP” (*On-Line Analytical Processing*), que consiste en matrices multidimensionales para analizar bases de datos relacionales de gran tamaño. Con el fin de ayudar a los usuarios que manipulen la máquina,

se propone el desarrollo de una interfaz (HMI, “Interfaz Hombre-Máquina”) que informe sobre el estado de la máquina en todo momento y, en caso de que suceda algún tipo de incidente relacionado con los dispositivos implementados, esta muestre las señales luminosas pertinentes y la información relativa al suceso. Para finalizar, se propone el establecimiento de una conexión remota para afianzar el concepto de “Industria 4.0” y, de esta forma, poder realizar comprobaciones a distancia sin necesidad de desplazarse a la ubicación de la máquina.

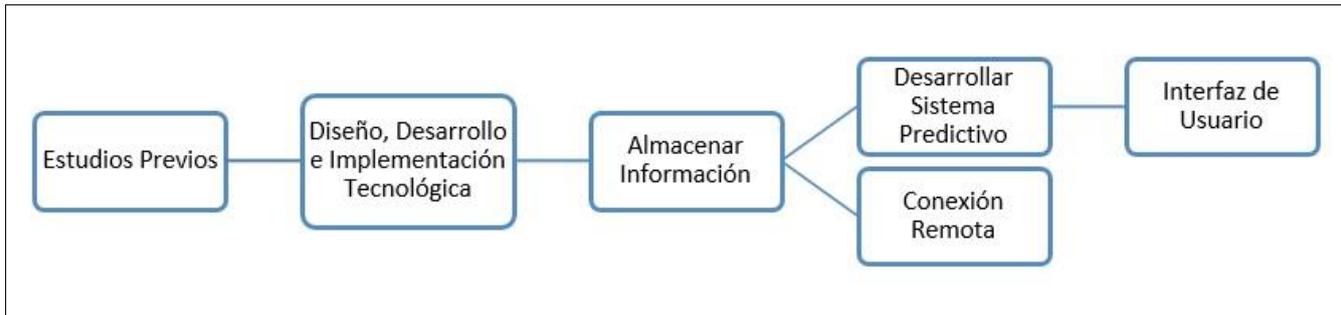


Figura 1. Diagrama del alcance del proyecto

2 Descripción del sistema

A lo largo de este capítulo, se especifica un análisis de todo el sistema en su conjunto para ofrecer una comprensión más profunda del objetivo del proyecto y poder entender el funcionamiento de todo el proceso de adquisición y tratamiento de los datos.

Dichos datos serían suministrados a una base de datos para ser almacenados y, en un futuro poder disponer de ellos para la realización de análisis inteligentes cuyo fin es la optimización del proceso de fabricación del centro de mecanizado. En la figura 2, se muestra un diagrama de bloques donde se especifica todos los elementos que componen el sistema.

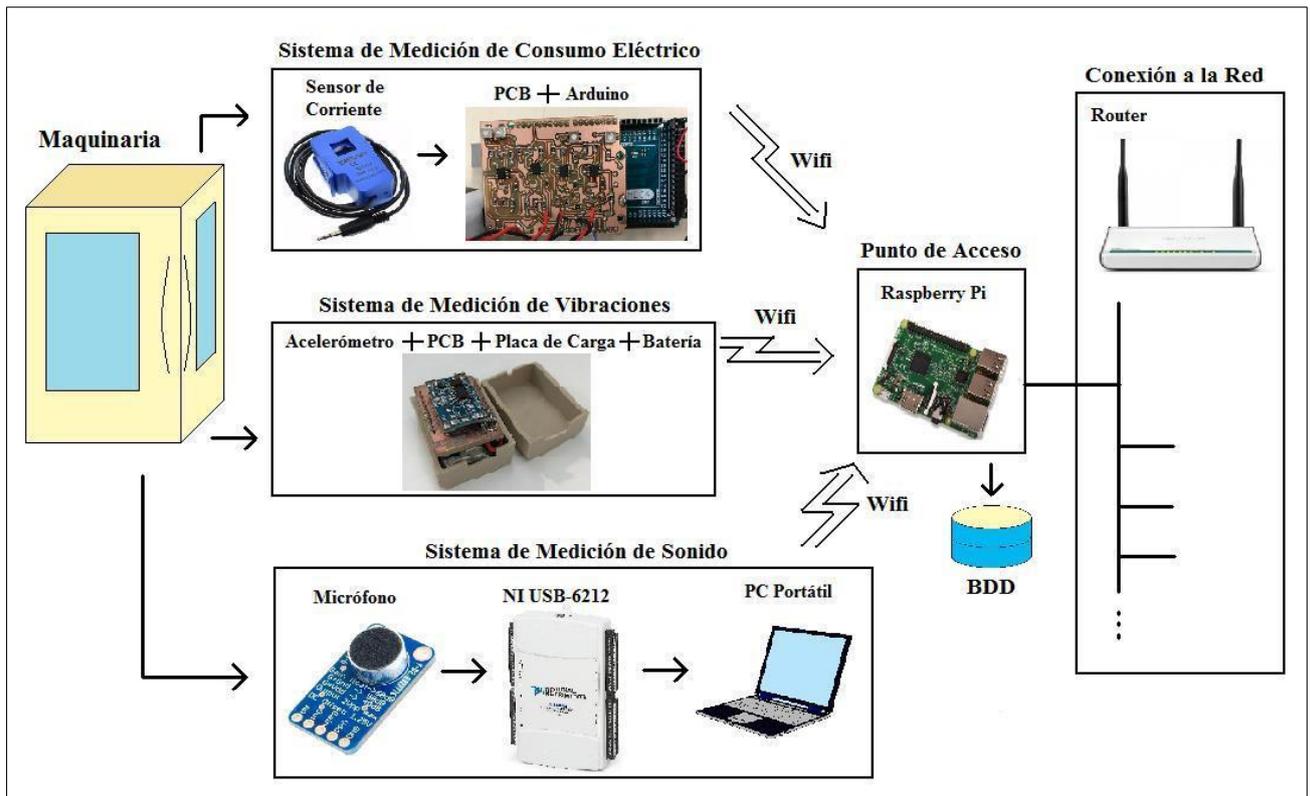


Figura 2. Diagrama de bloques del Sistema Global

En primer lugar, como puede verse en la figura 2, se propone el desarrollo de un sistema para la medición de consumo eléctrico del centro de mecanizado, compuesto por: un sensor de corriente no invasivo, una placa de “Arduino” junto con una placa para el tratamiento de la corriente y un módulo de comunicación wifi ESP-05 para, posteriormente almacenar la información en una base de datos. La decisión de emplear una placa de “Arduino” para la medición del consumo eléctrico reside en que es un elemento con la potencia suficiente para desempeñar su función, está disponible por un precio reducido, el software es de libre utilización, y hay disponible multitud de información y tutoriales de trabajo. Aunque hoy en día existe una placa de “Arduino” que trae incorporada la comunicación wifi, se ha decidido desarrollar el sistema con una placa de “Arduino” y un módulo de comunicación wifi de manera independiente ya que la otra placa todavía estaba en fase de desarrollo, en lo que a librerías para programación se refiere, y se encontraron multitud de impedimentos para lograr una configuración completa y funcional. Lo primero que se debe hacer es elaborar los esquemas eléctricos necesarios para, posteriormente, fabricar la PCB (del inglés: *Printed Circuit Board*) que se incorporará a “Arduino”. En dichos esquemas, se incluyen una serie de conectores para los sensores de corriente no invasivos y para el módulo de comunicación wifi, junto con varios conectores de unión para poder montar la PCB sobre la placa de “Arduino”. Una vez elaborados todos los esquemas y planos, se procede a la programación

de la placa de “Arduino” y del módulo wifi mediante el software libre que proporciona la casa a través de su página web: “<https://www.arduino.cc/en/Main/Software>”. Con el fin de proteger el sistema fabricado, se ha diseñado una caja para poder encerrarlo en su interior, evitando que factores como el polvo o personas ajenas al trabajo puedan entrar en contacto con él. Para constatar el funcionamiento del sistema, se ha añadido un diodo LED que parpadea si todo es correcto. Todo ello, aparece completamente especificado en el apartado 7.1.

En segundo lugar, se propone la fabricación de un sistema de captación de las vibraciones/aceleraciones generadas en el centro de mecanizado durante la producción. Dicho sistema estará compuesto por: un acelerómetro de tres ejes que detecta los desplazamientos derivados del contacto de la herramienta contra el material durante la fabricación, una batería que alimenta al dispositivo, una placa de carga para poder recargar la batería, una placa de comunicación wifi para poder enviar la información conseguida a la base de datos creada y, una placa que unifica cada uno de los elementos anteriores. Se ha decidido emplear el acelerómetro de tres ejes “ADXL362” y la placa de comunicación “WEMOS D1 Mini Pro” por el hecho de ser compatibles con la herramienta software de “Arduino”, lo que implica la facilidad de trabajar con una herramienta ya conocida e incluso la posibilidad de reutilización de código. Para poder incorporar el dispositivo en el interior del centro de mecanizado, se ha fabricado un portaherramientas con un hueco específico con las dimensiones del mismo. Además, para lograr el correcto funcionamiento de cada una de las partes que componen el dispositivo, se ha fabricado una pequeña caja a modo de recipiente para introducirlo y así encaje a presión en el hueco del portaherramientas. Todo ello aparece especificado en el apartado 7.3.

En tercer lugar, se propone el desarrollo de un sistema para la captación de señales acústicas con el fin de observar variaciones en relación a la frecuencia. En esta ocasión era necesaria la incorporación de elementos más potentes, ya que las señales acústicas requieren de ello. Quizás, habría sido posible la utilización de una “RaspBerry” ya que incorpora tarjeta de sonido, pero para asegurar totalmente un correcto funcionamiento sin sobrecargas se plantearon otros elementos. De esta forma, se dispone de un micrófono colocado en el interior del centro de mecanizado y un elemento multifunción de *National Instruments* para poder transmitir la señal acústica desde el micrófono al ordenador portátil. Desde el ordenador portátil, vía wifi y a través de una conexión establecida mediante *LabVIEW*® de *National Instruments* se realiza el envío de la información obtenida a la base de datos. Como

puede verse, en esta ocasión ha sido necesaria la programación mediante nuevas herramientas software, sin existir la posibilidad de reciclar alguna de las empleadas con anterioridad. Todas las especificaciones sobre el desarrollo de este sistema, aparece recogido en el apartado 7.2.

Como se muestra en la figura 2, la base de datos está contenida en una RaspBerry Pi y esta a su vez, se encuentra conectada a un *router* con acceso a la red. Mediante este punto de acceso, otros clientes con autorización pueden conectarse de forma remota y observar aquellas condiciones que sean de su interés.

Asimismo, se llevará a cabo la presentación de los resultados que se vayan consiguiendo a través de una visualización gráfica de los datos. De esta forma, se facilitará el entendimiento del comportamiento del centro de mecanizado en diferentes situaciones y con variedad de fuentes, puesto que se disponen de datos de tres dispositivos diferentes.

Finalmente, se establecerá el análisis inteligente de los datos mediante el sistema de “Cubos OLAP”, permitiendo identificar si el centro de mecanizado está operando bajo unas condiciones aceptables o, por el contrario, alguno de sus componentes, o el sistema en general, podría estar siendo sometido a unas condiciones de trabajo inadecuadas. A modo general, lo que se pretende es optimizar lo máximo el proceso de fabricación. En la figura 3, se presenta un diagrama de bloques con el análisis final tras la adquisición de suficientes datos.

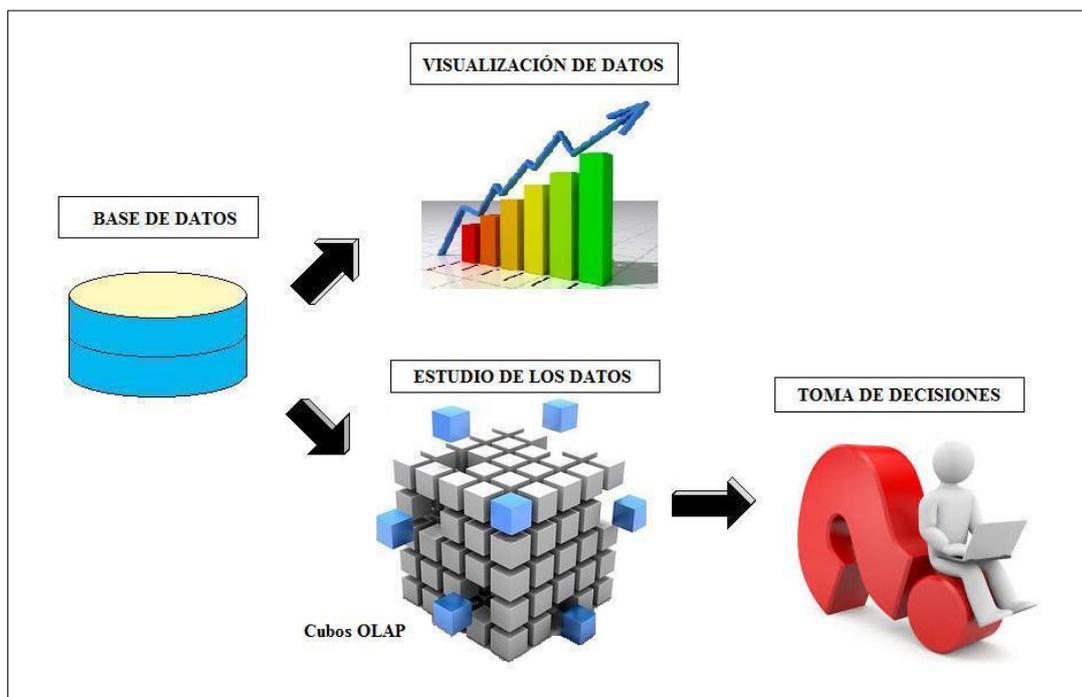


Figura 3. Diagrama de bloques del Análisis de Datos

3 Descripción de maquinaria

La **Fundación Pro dintec** cuenta con un centro de mecanizado para realizar la fabricación de piezas metálicas mediante arranque de viruta. Dicha labor, se desarrolla mediante una máquina “Deckel Maho”. Su función es la de emplear diferentes tipos de herramientas (brocas, fresas, etc.), situadas en un almacén en el interior de la máquina, para lograr la fabricación de las piezas diseñadas con las especificaciones establecidas. Las especificaciones de la máquina que se muestra en la figura 4, pueden consultarse en el anexo A.



Figura 4. Centro de mecanizado instalado

3.1 Problemática en la desactualización de maquinaria

La actualización de la maquinaria es un proceso importante a tener en cuenta ya que, aunque las máquinas pueden continuar desarrollando las labores con total normalidad, con el paso de los años los fabricantes desarrollan nuevas tecnologías que pueden dejar obsoletas a aquellas máquinas con ciertos años de antigüedad. Además, en el futuro todas las máquinas de este tipo deberán ser capaces de optimizar su funcionamiento y estar dotadas de sistemas de comunicación con el exterior debido a la imposición de la “Industria 4.0”, ofreciendo al usuario datos que informen sobre su estado o funcionamiento.

Por ello, ante el gran desembolso económico que supondría la compra de una de estas nuevas máquinas, la mejor solución para evitar la obsolescencia de la misma es implementar una serie de mejoras, basadas en elementos como sensores y sistemas de comunicación, para mantenerla en el mismo rango de competencia.

4 Prodintec, el entorno de trabajo y la innovación

Debido a que todos los entornos industriales han evolucionado hacia una organización y forma de trabajo casi totalmente tecnológica, existe la necesidad de actualizar la maquinaria adquirida hace algunos años para poder conseguir un mayor control sobre las mismas y los procesos productivos. Dicho control, permite reducir no solo los costes de fabricación sino, también aquellos asociados a consumos eléctricos y en evitar fallos en el proceso de fabricación. Como centro tecnológico fundado en 2004, la **Fundación Prodintec** está especializado en el diseño y producción industriales cuyo objetivo principal es potenciar la competitividad de las empresas industriales aplicando tecnologías y metodologías innovadoras a sus productos y procesos de fabricación y gestión.

4.1 Entorno de trabajo

El proyecto será llevado a cabo mediante el trabajo en equipo de los departamentos de electrónica y de tecnologías.

- **Departamento de ingeniería.** Es el encargado del estudio y desarrollo de toda la parte eléctrica/electrónica, lo que incluye: esquemas eléctricos, organización de componentes y dispositivos electrónicos necesarios, diseño de PCB's (placas de circuito impreso), etc. También se encargan del montaje y la puesta en marcha de los dispositivos desarrollados.
- **Departamento de tecnologías.** Es el encargado de la parte de control del proyecto, gestionando cada parte y que se vayan cumpliendo los objetivos.

Todos los trabajos serán llevados a cabo en las distintas instalaciones de la fundación, como son el laboratorio de electrónica, el laboratorio de pesados, donde está ubicada la máquina de mecanizado, y la oficina, donde se desarrollan las partes de programación y diseño.

4.1.1 Innovación

El presente proyecto marca una gran diferencia con las tecnologías existentes en el mercado ya que, no existe ninguna que permita realizar las mediciones que se buscan sin tener que invertir cantidades significativas de dinero. Por ello, muchas de las empresas se ven

obligadas a trabajar con maquinaria desactualizada o a realizar inversiones muy grandes de dinero para adquirir otras nuevas.

Gracias a este proyecto, se va a desarrollar un conjunto de dispositivos de **coste reducido** que permitirá actualizar centros de mecanizado sin la necesidad de realizar una gran inversión para tal fin.

5 Internet de las cosas (IoT) y tecnologías relevantes

El **IoT**, *Internet of Things* (según siglas en inglés), o más comúnmente conocido como el Internet de las Cosas (KUZU, 2017), consiste en la interconexión de dispositivos de última tecnología con el fin de facilitar la vida cotidiana. Ahora bien, en los últimos años se ha extrapolado el concepto a otros sectores como el de la industria.

La incorporación del **IoT** a la industria ha supuesto el incremento de la eficiencia y la productividad gracias al control y maximización de los procesos de fabricación. El hecho de dotar de **IoT** a un sistema productivo supone capacitar a la maquinaria de inteligencia, consiguiendo controlar los rendimientos en la producción, así como los tiempos en los que se desempeña cada trabajo, etc.

Otro de los aspectos a destacar es la posibilidad de almacenar grandes cantidades de datos. Además, tras la adquisición de los datos se realiza un análisis de los mismos, con el fin de detectar fallos o puntos débiles del sistema para que dejen de serlo.

Por consiguiente, si conocemos el funcionamiento de la maquinaria y analizamos los parámetros clave de la misma, se podrá realizar una gestión de los recursos óptima y con ello reducir los costes de producción y aumentar la productividad, así como aumentar la seguridad de los trabajadores.

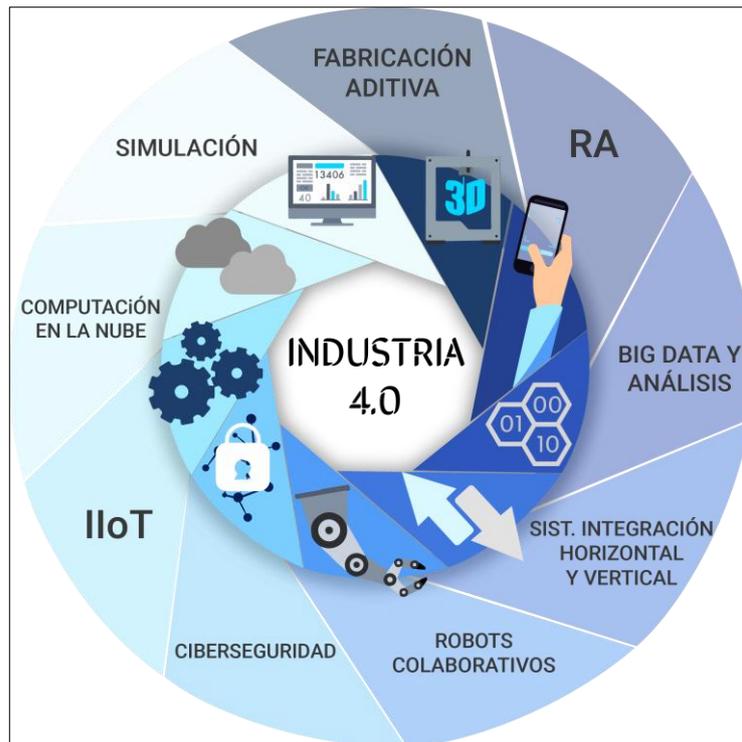


Figura 5. Industria 4.0 (Mítica Technology. (2017). Ilustración de Industria 4.0. [Figura]. Recuperado de: <https://www.miticotechnology.com/index.php/transformacion-digital/industria-4-0>)

En la figura 5 se muestra una composición general de los elementos y actividades que intervienen en el concepto de Industria 4.0. En ella, pueden observarse temas principalmente orientados a la conectividad y a la tecnología más actual, como son los sistemas de simulación, el análisis de grandes cantidades de datos, la ciberseguridad, sistemas de conectividad, etc.

Estamos inmersos en una nueva era, en la que se busca la máxima conectividad posible. Sin embargo, aún queda mucho camino por andar y el número de dispositivos conectados crecerá de forma masiva en los próximos años. En la figura 6 se muestra una gráfica con las instalaciones reflejadas en el año 2016 a nivel mundial (NCTA, 2015):

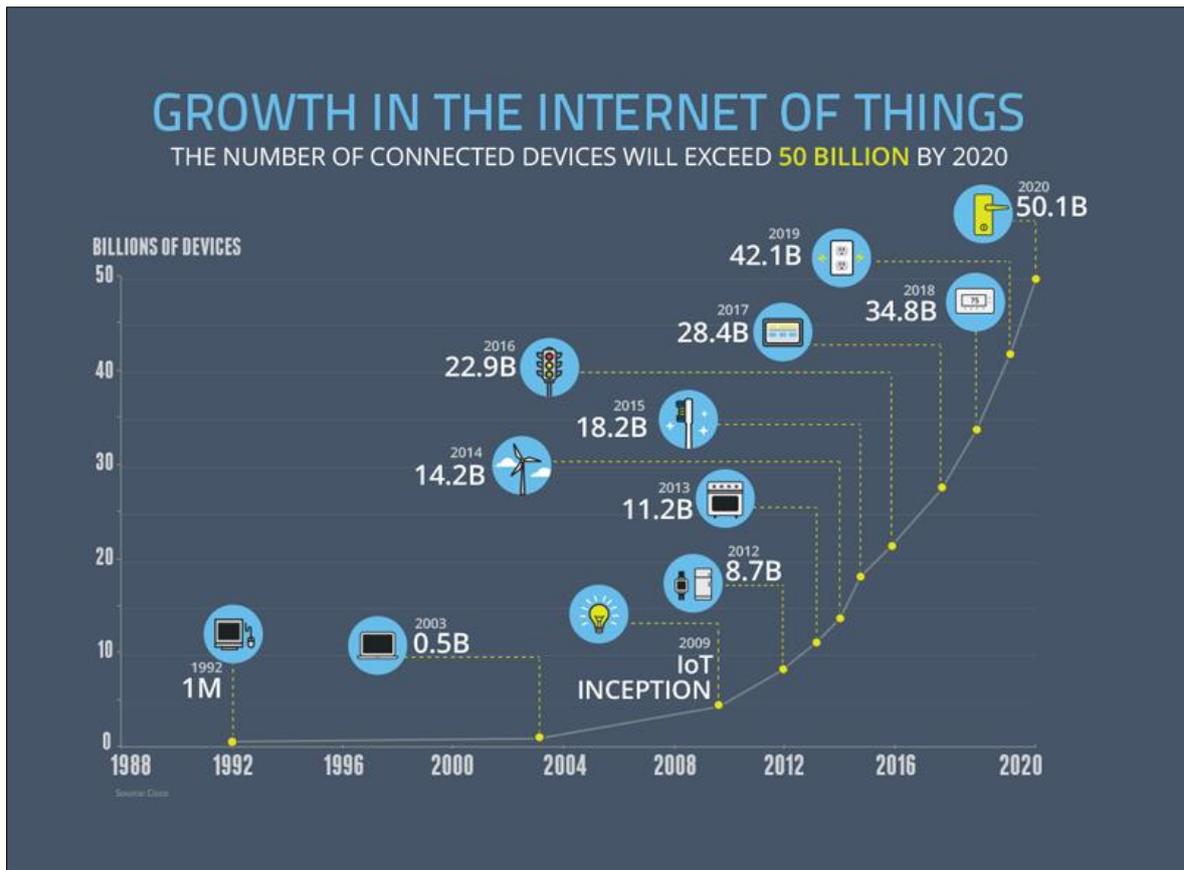


Figura 6. Representación de dispositivos conectados (NCTA. (2015). Ilustración de crecimiento de IoT. [Figura]. Recuperado de: <https://www.ncta.com/whats-new/behind-the-numbers-growth-in-the-internet-of-things-2>)

5.1 Lenguajes de programación

Como no podía ser de otra forma, el **IoT** va acompañado de una gran parte de programación informática, necesaria para la organización, estructuración y acciones a desempeñar por el sistema que se ha creado. Para ello, es necesario identificar los lenguajes de programación con los que se va a trabajar o son más comunes:

- C
- C++
- G
- PHP
- JAVASCRIPT

En esta ocasión, los más importantes para el proyecto son el lenguaje “C” y “PHP”, ya que son con los que se realizará la mayoría de las aplicaciones que permiten el funcionamiento de los sistemas de adquisición y la comunicación con la base de datos.

El lenguaje C es el lenguaje de programación básico en informática por excelencia, de nivel medio, que dispone de estructuras de lenguajes de alto nivel y, a su vez, de estructuras que permiten un control a bajo nivel. En esta ocasión, este lenguaje cobra un gran protagonismo debido a que será empleado para la programación de los dispositivos electrónicos de medición de consumos y de vibraciones/aceleraciones.

Aunque no sea un lenguaje de programación, también se empleará “HTML” (Herrera, 2015) para el diseño de páginas web y que procede de las siglas *HyperText Markup Language* (en inglés) o *Lenguaje de Marcas de Hipertexto* (en español). HTML no incluye el diseño gráfico de las páginas web sino, que indica el orden del contenido de esta. Para ello, emplea una serie de marcas de hipertexto denominadas *etiquetas* o *tags*, según el lenguaje sajón. La estructura de “HTML” suele seguir una misma pauta: cabecera, título y cuerpo. Para indicar el comienzo y final de cada una de estas partes, se emplean los símbolos “<”, “>” y “/”, usando este último únicamente para marcar el final. Por ejemplo, si se quisiera marcar el inicio de la cabecera se usaría la siguiente expresión: <head> y, para el final: </head>.

“PHP” (Codecademy, 2017) es un lenguaje de programación de propósito general de código del lado del servidor (colaboradores de Wikipedia, 2018) que permite realizar multitud de funciones como, evaluar datos enviados desde un navegador, comunicarse con una base de datos, construir contenido web, etc. Procede del acrónimo recursivo en inglés *HyperText Processor* (procesador de hipertexto) (PHP, 2017) y puede ser incrustado en el lenguaje “HTML”, apareciendo encerrado bajo los símbolos “<?””, para entrar y “?>”, para salir. Es un lenguaje caracterizado por no permitir la lectura del código al recibir un resultado, por lo que los usuarios/clientes no pueden tener conocimiento de lo que se esconde tras la respuesta recibida.

El lenguaje gráfico (colaboradores de Wikipedia, 2017), también conocido como G, será empleado para programaciones relativas al sistema electrónico de captación de señales acústicas mediante la herramienta software “LabView”. Dicho lenguaje se caracteriza por el hecho de que no se escribe, sino que se dibuja (de ahí lo de lenguaje gráfico). El software incluye una gran variedad de bloques pre-diseñados, lo que permite al usuario prestar una mayor atención a la interfaz gráfica.

Aunque en esta ocasión no se vaya a usar, JavaScript es un lenguaje muy presente en este tipo de trabajos. Es un lenguaje de programación interpretado, orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico (colaboradores de Wikipedia,

2018). Suele emplearse en su forma del lado del cliente, implementado como parte de un navegador web lo que permite mejoras en interfaces de usuario y páginas web dinámicas.

5.2 Bases de datos

Una base de datos (CCM, 2017) es una entidad que permite el almacenamiento de datos de forma estructurada. A modo simplista, es como una biblioteca con multitud de datos.

Las bases de datos deben permitir a varios programas el acceso a su información a través de la red, ya sea de forma remota o local. Por ello, las bases de datos están totalmente ligadas al concepto de red ya que lo que se busca es la compartición de información. En la figura 7, puede verse como sería la estructura de un sistema conectado a una base de datos.

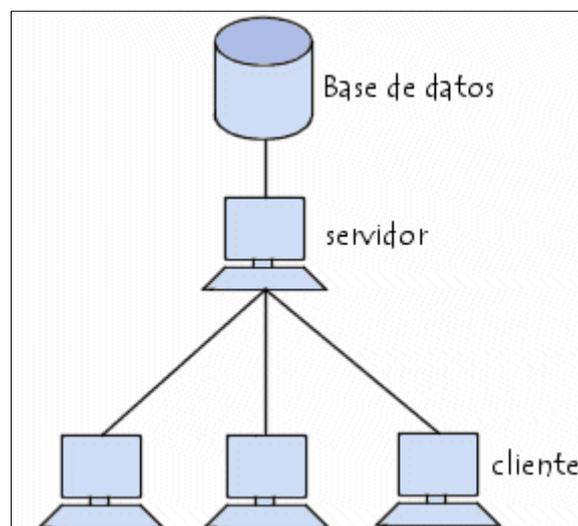


Figura 7. Estructura de un sistema conectado a una Base de Datos (CCM. (2017). Ilustración de sistema de información. [Figura]. Recuperado de: <https://es.ccm.net/contents/66-introduccion-a-las-bases-de-datos>)

Las bases de datos implican una gran variedad de ventajas (Cibertareas, 2013):

- Almacenamiento de multitud de información,
- compartición de la información entre una gran variedad de clientes,
- acceso simple, sencillo y rápido a la información,
- permite mejorar la productividad,
- reducción de dispositivos de almacenamiento.

Sin embargo, no se debe olvidar de que nos encontramos en un entorno de Internet y, por lo tanto, debe tenerse muy en cuenta el tema de la seguridad. Este concepto es vital para evitar robos o ediciones indebidas de los datos.

Los tipos de bases de datos existes hoy en día son múltiples, pero, en este caso, se ha decido emplear un base de datos del tipo “MySQL” (idesweb, 2012) ya que es una base de datos “Open Source”, algo muy bueno ya que no supone ningún coste y existe mucha información sobre ella como se verá a continuación.

5.2.1 MySQL

“MySQL” es uno de los sistemas de gestión de bases de datos de código abierto más usado a nivel mundial hoy en día. Está disponible para multitud de sistemas operativos: *Linux*, *Windows*, *Mac OS X*, etc. Es un sistema que se caracteriza por su alto rendimiento, la confianza que aporta y la facilidad de su uso, especialmente para desarrollos basados en la Web.

Es un sistema que se caracteriza por su seguridad ya que incorpora privilegios y contraseñas mediante verificación basada en el *host*. Las contraseñas permanecerán totalmente a salvo durante todo el tráfico de datos ya que éste está encriptado durante la conexión a un servidor.

Otra de las características es que trabaja con archivos programados en lenguaje PHP, por lo que la información quedará totalmente oculta.

Permite realizar conexiones mediante sockets TCP/IP, conectores ODBC (*Open DataBase Connectivity*) para clientes que usen “MS Access” y JDBC (*Java DataBase Connectivity*) para clientes con plataforma “Java”.

Además, permite ejecutar consultas en lenguaje “SQL” para chequear, optimizar y reparar tablas de datos.

5.3 Sistema de análisis de datos

Un sistema de análisis de datos es una herramienta que permite analizar y visualizar la información contenida en una base de datos con el fin de aplicar métodos de toma de decisiones, permitiendo la ejecución de acciones preventivas y correctivas en el ámbito necesario, siendo este el productivo, económico, gestión, etc. En la figura 8, puede verse un esquema que recoge estos sectores.



Figura 8. Concepto gráfico de sistema de análisis de datos (Shutterstock. (2017). Ilustración de análisis de bases de datos. [Figura]. Recuperado de: <https://www.shutterstock.com/es/image-illustration/database-analytics-banner-flat-set-data-310367603>)

Existen muchas plataformas software para realizar todas las tareas anteriormente mencionadas, pero en este proyecto se va a trabajar con el sistema de “Cubos OLAP”.

5.3.1 Cubos OLAP

Los “Cubos OLAP” (Logicalis, 2017) son estructuras multidimensionales que permiten realizar análisis de bases de datos relacionales muy voluminosas y con gran variedad de datos rápida y ágilmente, dando lugar a una reducción en los tiempos y recursos del análisis. El acrónimo “OLAP” procede de las palabras en inglés “*On-Line Analytical Processing*”, reservadas para los sistemas de análisis de datos basados en estructuras multidimensionales.

Hay que destacar que no todos los sistemas “OLAP” (Logicalis, 2015) son iguales, sino que, dentro de esta categoría existen varios tipos de sistemas. A continuación, se explica cada uno de ellos con sus principales características:

1. **Sistemas ROLAP.** Proviene de las palabras “*Procesamiento Analítico OnLine Relacional*”. Estos sistemas se desarrollan sobre bases de datos relacionales con una estructura en copo de nieve o estrella. Se usan cuando quieren analizarse grandes cantidades de datos.
2. **Sistemas MOLAP.** Proviene de las palabras “*Procesamiento Analítico OnLine Multidimensional*”. Son sistemas desarrollados sobre bases de datos con una estructura “OLAP” de tipo multidimensional. Si se compara con el modelo anterior, este sistema es más rápido en lo que al almacenamiento de datos se refiere, más eficiente en el uso de la memoria y durante el proceso de extracción de los datos. Sin

embargo, existe el riesgo de duplicaciones durante el análisis, especialmente si se va a trabajar con volúmenes de datos muy grandes.

- 3. Sistemas HOLAP.** Proviene de las palabras “*Procesamiento Analítico OnLine Híbrido*”. Este sistema combina las características de los dos anteriores, es decir, trabaja sobre bases de datos relacionales y con una estructura multidimensional. De esta forma, la recomendación a la hora de trabajar con este sistema es, almacenar los datos más recientes en la plataforma “MOLAP” y, lo más antiguos o que se usen con menos frecuencia en la plataforma “ROLAP”.

Una vez que se ha explicado el concepto de “Cubos OLAP” y las especificaciones del mismo, hay que destacar que para que el sistema pueda usarse de la forma más eficiente y correcta posible, la base de datos con la que se va a trabajar debe estar correctamente estructurada y organizada.

6 Big data

Ante toda esta cantidad de conceptos relacionados con la adquisición y el tratamiento de datos, surge un nuevo concepto conocido como “Big Data” el cual, se basa en el análisis masivo de datos. El mundo ha evolucionado de forma que anteriormente era necesario aplicar procesos costosos y complejos para adquirir unos pocos datos y, ahora se habla de la medida de exabytes en cuanto a cantidad de almacenamiento (Logicalis, 2017). La cantidad es tal que actualmente hay generados unos 2.5 trillones de datos y el número crece día a día. Según varios expertos, esta nueva era de revolución tecnológica constituye la cuarta revolución industrial, en la que se destacan las siguientes claves:

- Optimizar las propuestas de valor, gracias a las interacciones en tiempo real, pudiendo aplicar soluciones inmediatas ante posibles mejoras.
- Los datos en sí mismos carecen de valor, la clave es emplearlos de forma adecuada.
- El empleo de las redes sociales permite al productor conocer las tendencias y aprovechar en beneficio mutuo este factor.
- Es importante mezclar los datos de varias fuentes (internas y externas).
- La experimentación y la digitalización son muy importantes a la hora de optimizar los resultados tras la explotación de los datos.

La evolución del “Big Data” será imparable y seguirá transformando el mundo productivo debido a la aplicación de *redes neuronales artificiales*, el *aumento de la conectividad* y el

uso de *datos no estructurados de carácter no textual* (Logicalis, 2015), siendo este último uno de los conceptos que más está siendo investigado por parte de las empresas. Para aclarar este último concepto, los datos no estructurados son aquellos que no están contenidos en una base de datos o están registrados en una estructura de datos diferente. Además, al ser *no textuales* significa que están contenidos en imágenes, videos o archivos de audio. En definitiva, este tipo de datos incrementaría la habilidad de una empresa ahorrando costes, tomando las mejores decisiones y evolucionar de la mano con las nuevas tecnologías.

Otro de los aspectos que más se deben tener en cuenta es el de la seguridad. Nunca se debe olvidar que una vez que se establece conexión a través de la red, se es vulnerable a cualquier tipo de ataque cibernético, causando problemas como el robo de información o el borrado de archivos (lo que ocasiona graves pérdidas a las compañías). Por todo ello, siempre se debe tener en mente la capacidad de localizar y prevenir anomalías que puedan perjudicar a la información. Existen tres aplicaciones fundamentales para prevenir este tipo de intrusiones:

1. **Aumentar la vigilancia inteligente.** Analizar movimientos de datos para encontrar patrones de actuación
2. **Aplicar el análisis en tiempo real para minimizar los ataques.** Analizar el tráfico de la red para encontrar posibles amenazas y prevenir ataques.
3. **Predecir y prevenir un ataque.** Analizar datos de la red de telecomunicaciones y de las redes sociales con el fin de detectar amenazas y adelantarse a los ataques.



Figura 9. Concepto de Big Data (TICbeat. (2017). Ilustración de big data. [Figura]. Recuperado de: <http://www.ticbeat.com/educacion/experto-en-big-data-profesion-de-referencia-de-2017/>)

Como puede verse en la figura 9, el “Big Data” conlleva a una nueva era, la era de los datos, con la que se llegará a tener un control prácticamente total de un proceso productivo, pudiendo optimizar los mismos y resolviendo problemas de una forma correcta y segura.

7 Desarrollo del proyecto

A lo largo de este capítulo, se presentan los distintos dispositivos y sistemas desarrollados para lograr introducir el centro de mecanizado en el sector de la industria 4.0. Para cada uno de ellos, se especifican los fundamentos teóricos del sistema (para comprender su utilidad), su funcionamiento, la distribución hardware y la programación software.

7.1 Desarrollo de sistema de medición de consumos

A la hora de realizar este tipo de dispositivos, se debe tener en cuenta el tipo de señal eléctrica que va a ser tratada. En esta ocasión, se va a trabajar con una señal eléctrica de corriente alterna a 400 V de tensión y 50 Hz de frecuencia (Por localización geográfica es la que corresponde). En la figura 10, se representa una señal eléctrica alterna junto con su radio vector giratorio.

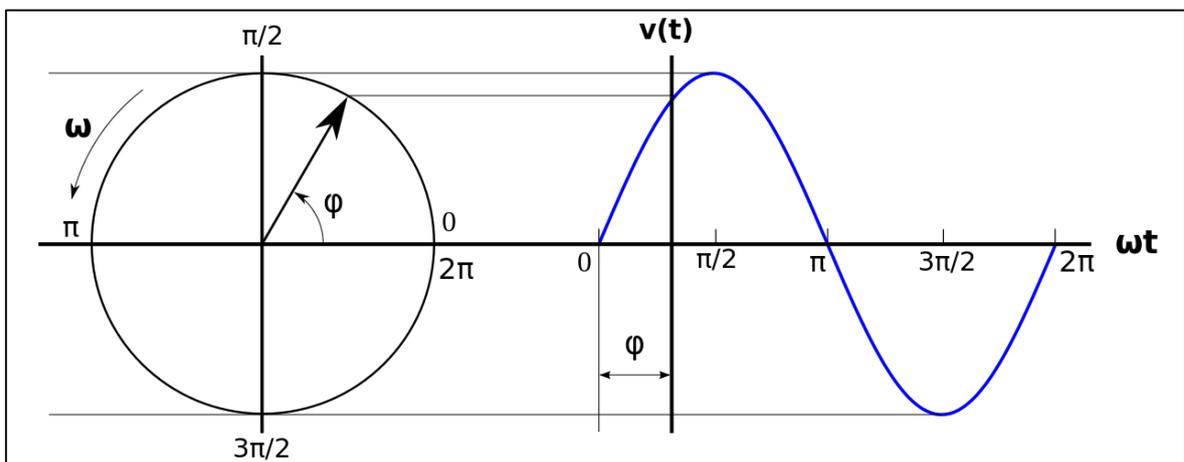


Figura 10. Diagrama de una Señal Eléctrica Alterna (Wikimedia Commons. (2009). Ilustración onda senoidal. [Figura]. Recuperado de: <https://commons.wikimedia.org/wiki/File:OndaSenoidal2.svg>)

Al tratarse de una señal de corriente alterna, se debe tener en cuenta que esta circula en ambos sentidos, es decir, se van a tratar tanto valores positivos como negativos lo que presenta problemas por parte de los dispositivos ya que, la mayoría de ellos solo pueden tratar valores positivos, como, por ejemplo, los dispositivos de “Arduino” (Arduino Mega 2560 en este caso). Sin embargo, la solución a este problema es muy sencilla: basta con la utilización de un rectificador e inversor de señal el cual, incorpora un “AOP” (configurado como inversor) y un diodo para evitar la circulación de las corrientes inversas. En la figura 11, se muestra la configuración anterior.

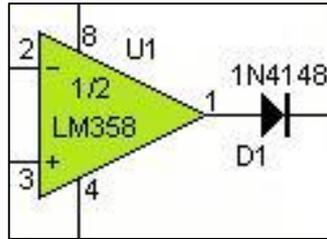


Figura 11. Elementos para Rectificación e Inversión de la señal

Tras la aplicación de esta configuración, la señal pasará a tener únicamente valores positivos. En la figura 12, se muestra el aspecto de la señal tras aplicar la configuración mencionada anteriormente.

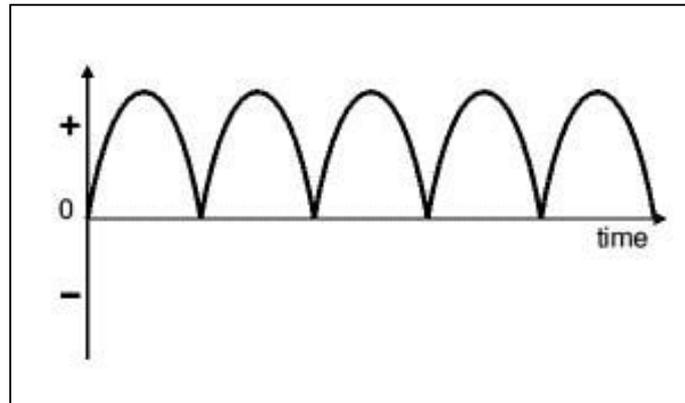


Figura 12. Señal Eléctrica Rectificada e Invertida (Curiosoando. (2014). Ilustración onda de salida completa DC. [Figura]. Recuperado de: <https://curiosoando.com/que-es-un-puerto-rectificador-de-diodos>)

En el siguiente apartado, se visualizan los diferentes dispositivos electrónicos donde se implementan las especificaciones anteriores.

7.1.1 Desarrollo hardware

Para lograr medir los consumos eléctricos de la máquina, se ha desarrollado un dispositivo que integra un circuito para la correcta lectura y tratamiento de la corriente eléctrica. Una vez registrados y tratados los datos, estos son enviados mediante comunicación wifi a una base de datos. A continuación, en la figura 13 se muestra un diagrama de bloques para comprender el funcionamiento del dispositivo electrónico en su conjunto.

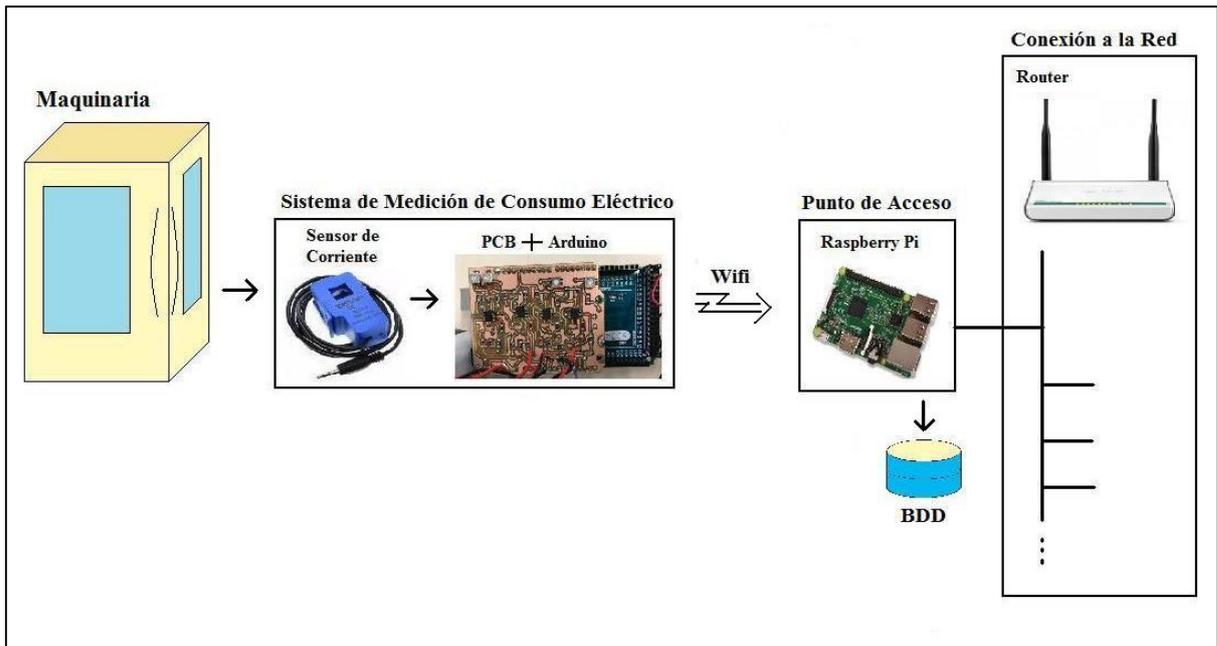


Figura 13. Diagrama de Bloques del Sistema de Medición de Consumo Eléctrico

A la hora de desarrollar el esquema, hay que tener claro una serie de aspectos para poder adquirir el valor correcto durante la medición. Puesto que se va a trabajar con una placa de “Arduino” se debe saber que esta no es capaz de leer mediciones negativas por lo que, será necesario rectificar la señal de entrada. Posteriormente, se realiza una amplificación de la señal muestreada con el fin de que su lectura sea mejor. En este último caso, se ha diseñado la circuitería para que el segundo integrado tenga una ganancia del valor la unidad, evitando así su saturación. Para poder regular la medición y ajustarla correctamente, se implementa un potenciómetro de una vuelta. También se incorporan unos condensadores a modo de filtro. A continuación, en la figura 14 se muestra el esquema general del sistema (no olvidar que son cuatro iguales).

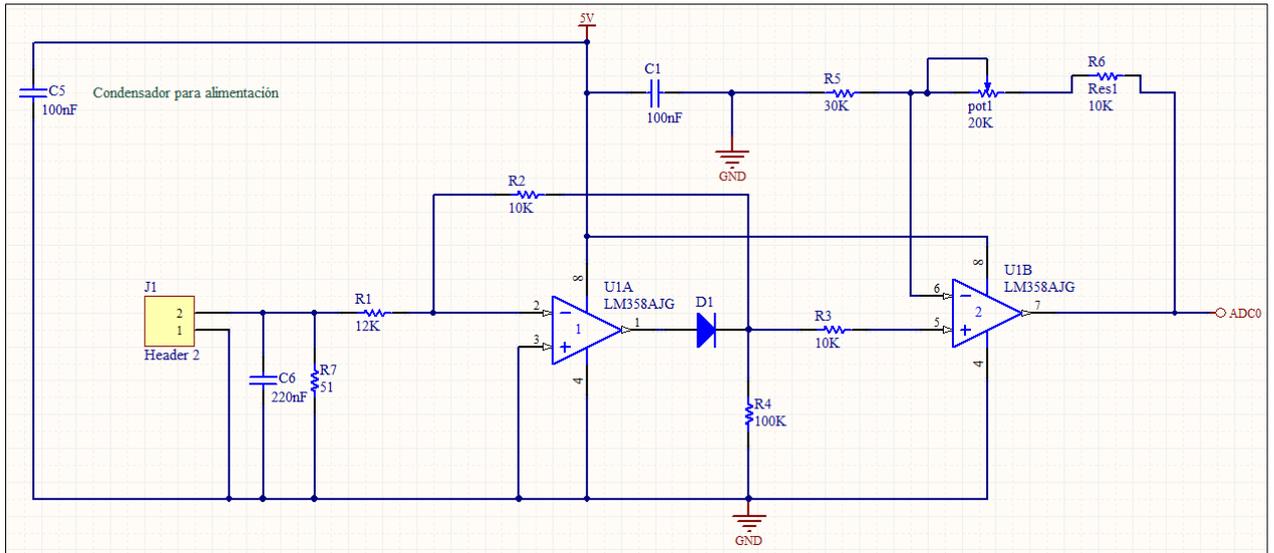


Figura 14. Esquema eléctrico general

Además, puesto que el módulo wifi con el que se va a trabajar solo puede ser alimentado a 3.3V, ha sido necesaria la inclusión de un regulador de tensión, pasando de una alimentación de 5V a la anteriormente mencionada de 3.3V. Este tipo de dispositivo es necesario para evitar que el módulo sufra daños en alguno de sus componentes. En la figura 15, se muestra el esquema del regulador de tensión.

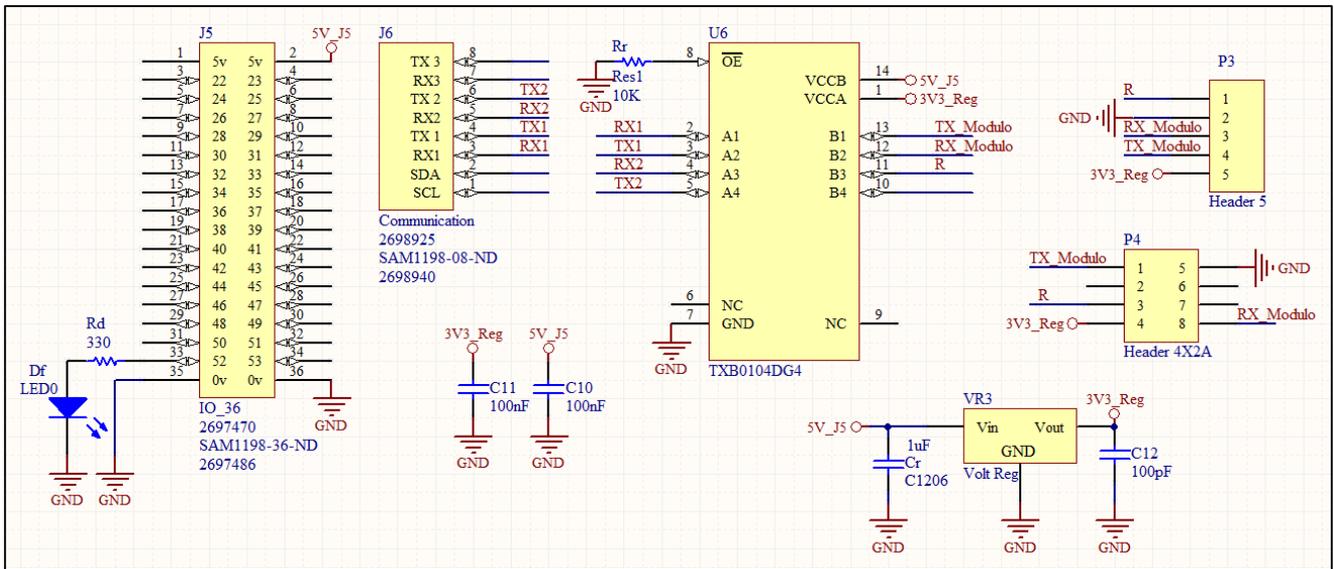


Figura 15. Esquema Eléctrico del Regulador de Tensión

Para el sensor de corriente no invasivo seleccionado, se debe destacar que es necesario añadirle entre las conexiones una resistencia de 51 Ω . El motivo es que este sensor no la trae incorporada de serie por lo que, las medidas de entrada son en intensidad y lo que se buscan son medidas en tensión ya que se está trabajando con “Arduino”. El cálculo de la resistencia

de carga puede realizarse de forma teórica o experimental. En este caso, se ha realizado de forma experimental para garantizar el correcto y más fiable funcionamiento del sistema.

Una vez diseñados los esquemas eléctricos y la PCB, se procede a su fabricación mediante una máquina similar a un pequeño centro de mecanizado, la cual puede verse en la figura 16.

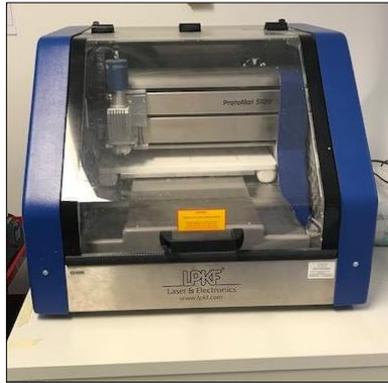


Figura 16. Máquina de fabricación de PCB's

Finalmente, se sueldan los componentes a la placa y se conecta con la placa “Arduino”. Hay que destacar que, para evitar cableados auxiliares excesivos, se ha diseñado la placa con las dimensiones del “Arduino” con el que se trabaja, coincidiendo las conexiones de los distintos pines de ambas placas. En las figuras 17 y 18, se muestra la PCB con los componentes soldados y montados sobre la placa de “Arduino”.

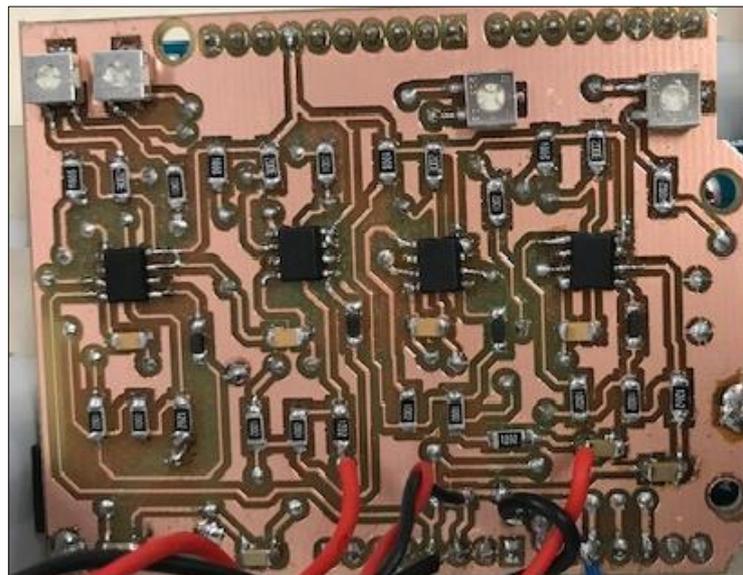


Figura 17. PCB fabricada

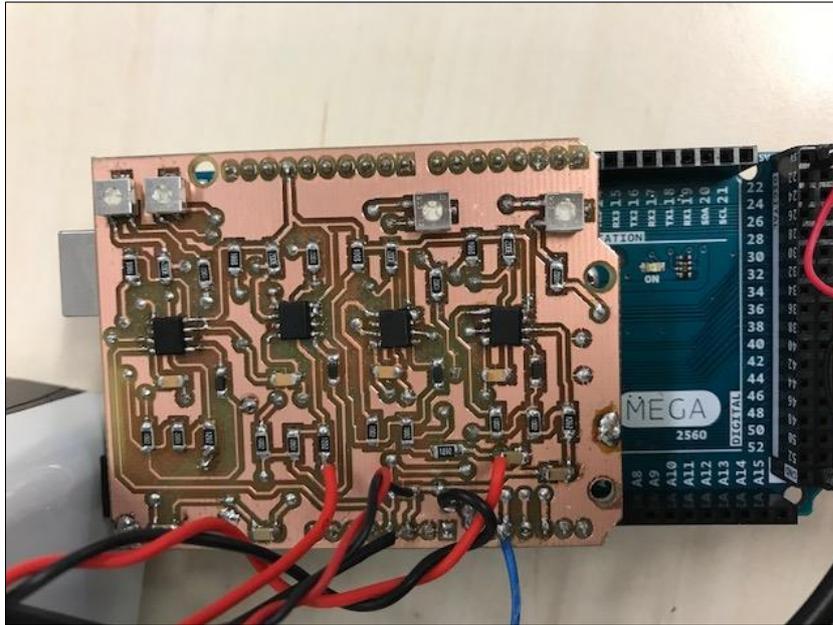


Figura 18. PCB conectada a Arduino

A la hora de poder enviar la información a la base de datos, se ha dotado al dispositivo de comunicación wifi. Para ello, se ha añadido un módulo ESP-05 especialmente diseñado para que trabaje con “Arduino”. En la figura 19, se muestra el módulo wifi indicándose cada uno de sus pines de conexión.

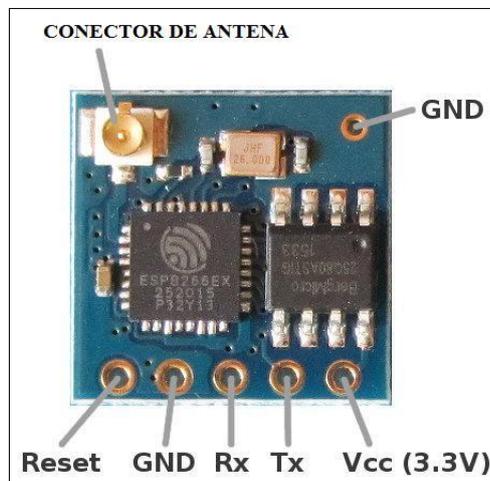


Figura 19. Módulo wifi ESP-05

Para instalar el dispositivo fabricado, se ha diseñado una caja para introducirlo y así, permanezca aislado del entorno de trabajo, evitando que pueda ser manipulado de forma accidental o se dañe por acumulación de suciedad o algún golpe. Además, se ha añadido un “LED” para poder detectar cuando está funcionando. En la figura 20, aparece el dispositivo final montado en el centro de mecanizado.



Figura 20. Dispositivo instalado en máquina

Para finalizar, se ha elaborado una tabla que contiene todos los elementos necesarios para desarrollar el dispositivo.

ELEMENTO	UNIDADES	VALOR	DESCRIPCIÓN
Resistencia	4	12 K Ω	-
Resistencia	12	10 K Ω	-
Resistencia	4	100 K Ω	-
Resistencia	4	30 K Ω	-
Resistencia	3	51 Ω	Resistencia de carga
Potenciómetro	4	20 K Ω	Permite ajustar la medida
Condensador	4	100 nF	Permiten reducir el ruido
Condensador	4	220 nF	
Circuito Integrado	4	-	Modelo LM358

Diodo	4	-	Solo permite el paso de la parte positiva de la señal
Conector	4	-	Modelo Jack 3.5 mm para conectar sensor
Sensor de corriente	4	-	Modelo SCT-013-000
Arduíno Mega 2560	1	-	Placa de desarrollo hardware
Módulo wifi	1	-	Modelo ESP-05

Tabla 2. Componentes para medición de consumos

Hay que destacar que las unidades registradas se corresponden con el total de los elementos empleados ya que, aunque se desarrolla un único dispositivo, los circuitos eléctricos a implementar son cuatro, uno para cada canal de medición.

Para el diseño de los esquemas eléctricos y la PCB, se ha empleado una herramienta software conocida como “Altium”. Dicha herramienta permite incorporar, gracias a las librerías implementadas, todos los componentes que son necesarios (resistencias, condensadores, etc.). El entorno de trabajo es fácil e intuitivo, que contiene una barra de herramientas principal con diferentes opciones. En este caso, las más usadas han sido la biblioteca de componentes (recuadrada en azul) y la conexión de los mismos (recuadrado en rojo). En la figura 21, se muestran las especificaciones anteriores.

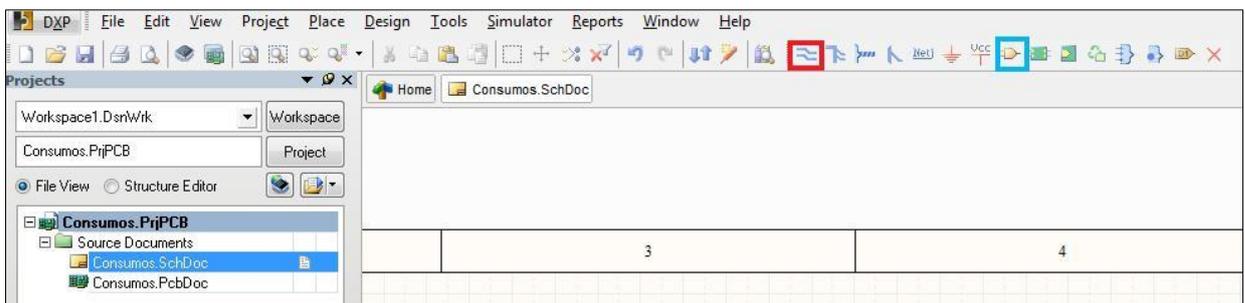


Figura 21. Barra de herramientas Altium

El diseño se realiza mediante lenguaje gráfico, de forma que se aumenta la sencillez del sistema. Además, la gran ventaja que aporta esta herramienta es que permite realizar la PCB en el mismo proyecto creado, es decir, que tras diseñar el esquema eléctrico se puede generar un archivo que contiene la PCB. Ahora bien, es tarea del usuario la ordenación de cada uno

de los componentes de forma que esta permita la fabricación más sencilla posible. La conexión entre los elementos puede realizarse automáticamente, mediante el autoenrutado, o manualmente. En este caso, la opción a seguir ha sido la segunda, ya que es la forma de conseguir una mejor estructura de la placa. Así mismo, será necesario respetar una serie de configuraciones a la hora de crear la PCB, con el fin de evitar daños en los componentes o en la propia placa. Tales configuraciones son:

- **Pistas.** Las pistas son los caminos de unión entre los componentes y deben tener una anchura mínima de entre 0.4mm y 0.5 mm, pero lo más favorable sería una anchura de 0.7 mm. A mayor anchura, menor calentamiento de las pistas.
- **Vías.** Son las canales de comunicación entre las caras superior e inferior. Es necesario cuando algunas de las conexiones se encuentran en caras diferentes. En este caso, los tamaños ideales son 1.7 mm de diámetro y 0.5 mm de taladro.
- **Distancia entre elementos.** Es necesario que los componentes, pistas y vías respeten una distancia mínima entre sí para evitar problemas como cortocircuitos, pudiendo dañar el sistema. Por ello, en este caso se ha establecido una separación mínima de 0.3 mm entre cada uno de ellos.
- **Plano de masa/tierra.** Es necesario implementar un plano de masa tanto en la capa superior como en la inferior. Dicho plano blindo el sistema del resto de líneas y lo protege.

Para poder modificar todos estos parámetros, se debe seguir la siguiente ruta a través de la barra de herramientas: *Design* → *Rules*. Posteriormente, se desplegará una ventana llamada *PCB Rules and Constraints Editor*, donde podrán realizarse las modificaciones especificadas anteriormente. En las figuras 22 y 23, se representan las configuraciones correspondientes.

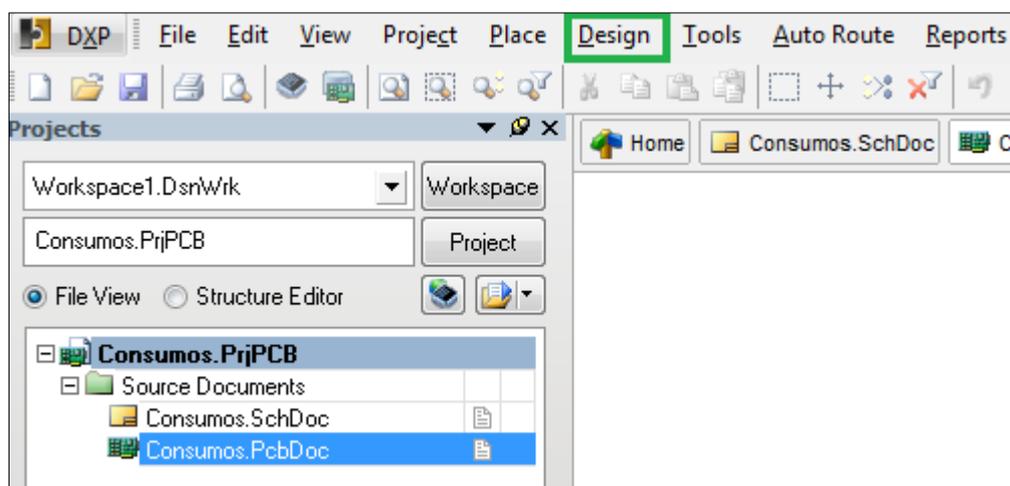


Figura 22. Configuración de parámetros. Parte 1

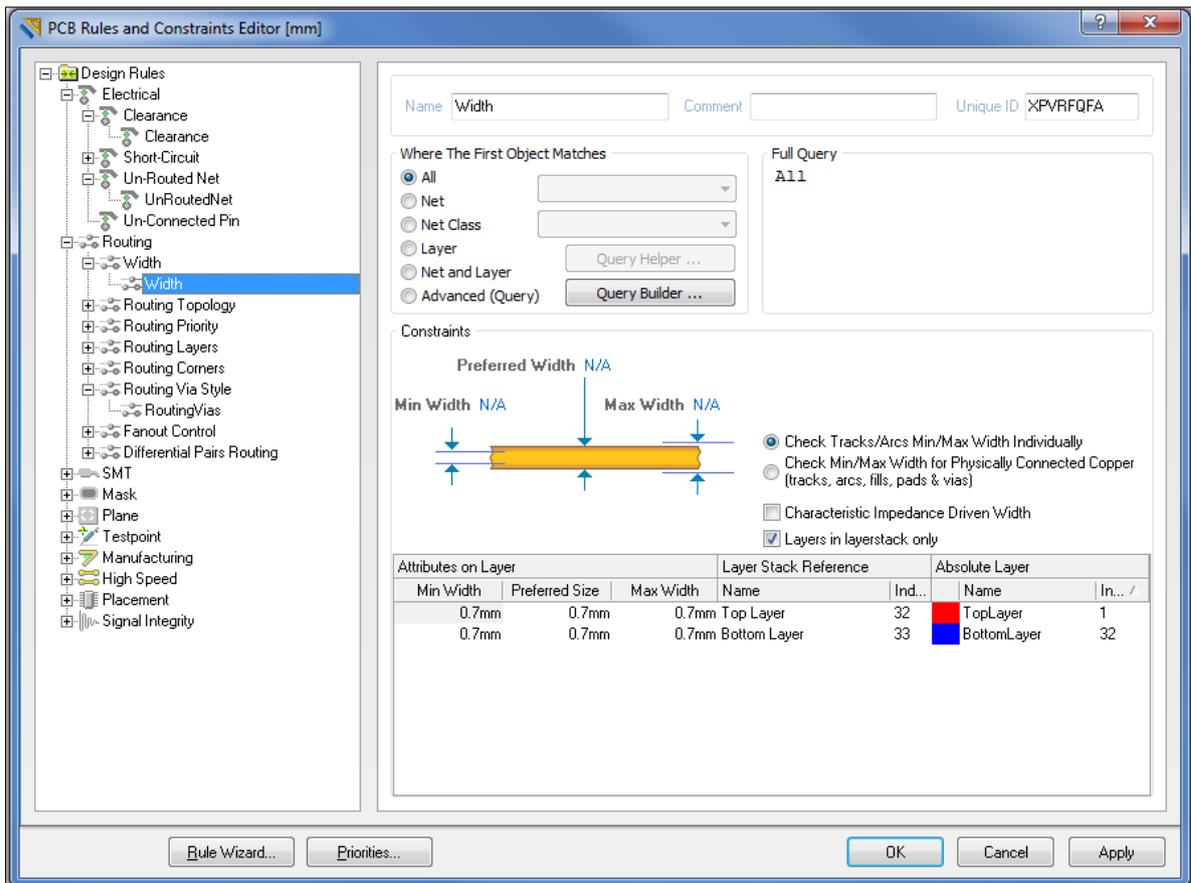


Figura 23. Configuración de parámetros. Parte 2

A la hora de llevar a cabo el enrutado manual, se debe seleccionar la opción *Interactive Routing* (recuadrado en azul) y, a continuación, conectar los extremos de los componentes que se desea unir. En la figura 24, se representa la opción que deber ser elegida.

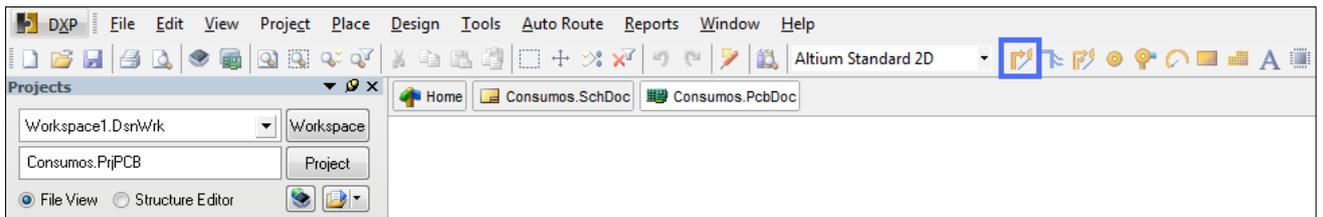


Figura 24. Enrutado manual

A la hora de enrutar se deben establecer las conexiones de la forma más eficiente posible, es decir, establecer el camino más favorable sin obstaculizar al resto de conexiones. Además, se aconseja evitar formar ángulos de 90° en las pistas para favorecer el flujo de corriente y evitar sobrecalentamientos. En la figura 25, se muestra un ejemplo de enrutado correcto e incorrecto:

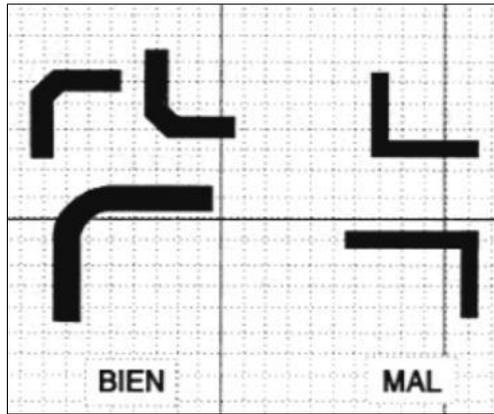


Figura 25. Comparativa de enrutado

Tras realizar todas las configuraciones y conocer las nociones básicas sobre la normativa, se ha realizado el diseño de la PCB. Como puede observarse en la figura que se muestra a continuación, todos los componentes se encuentran ensamblados en la cara “TOP” (superior), empleando únicamente la cara “BOTTOM” (inferior) para conexiones entre componentes. En la figura 26, aparece representada la PCB diseñada.

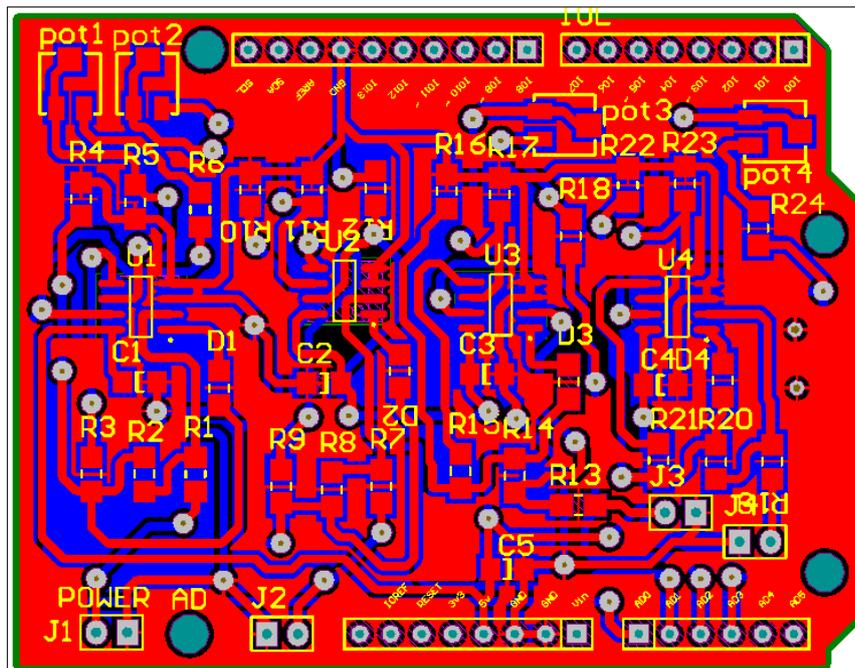


Figura 26. PCB de medición de consumos

7.1.2 Desarrollo software

7.1.2.1 Programación de arduino

Para programar la placa de “Arduino” se emplea el software “Open Source” propio de la marca, el cual puede descargarse desde la página oficial de “Arduino”: “<https://www.arduino.cc/en/main/software>”.

A lo largo de este apartado, se procederá a explicar las configuraciones y funcionalidades programadas a la placa “Arduino Mega 2560” con la que se va a trabajar. La programación “in situ”, podrá consultarse en el anexo B.

Todo programa de “Arduino” tiene dos funciones fundamentales:

- **Setup.** Función en la que se realiza la inicialización de variables y puertos serie. Se ejecuta una sola vez, tras la alimentación de la placa.
- **Loop.** Función que se ejecuta cíclicamente una y otra vez (bucle), ejecutando las funciones programadas en él.

Independientemente de las funciones anteriores, pueden desarrollarse tantas funciones nuevas como sea necesario, lo único que debe tenerse en cuenta es realizar las llamadas a las mismas desde una de las dos funciones fundamentales, recordando que si la llamada se establece en “setup” sólo se ejecutará una vez y, si se hace en “loop” lo hará de manera cíclica, respetando los retrasos que sean implementados (si los hay).

Para este caso, aparte de de las funciones preestablecidas se han creado cuatro nuevas:

1. **IniciaWifi.** Esta función es la encargada de establecer la configuración del módulo wifi ESP-05, así como realizar la conexión a la red wifi correspondiente. Todo ello, se realiza recorriendo un “String” que contiene los comandos “AT” que sirven para configurar el módulo anteriormente mencionado, empleando para ello un bucle “while”. Tras comenzar dicho bucle, se realiza la llamada a la función *GetLineWifi* donde, en función de lo que devuelva se pasará al siguiente comando o se repetirá el mismo si existe algún fallo.
2. **GetLineWifi.** Esta función se encarga de devolver la respuesta del módulo wifi.

Las posibilidades de respuesta son:

- ❖ READY
- ❖ OK
- ❖ NO CHANGE

❖ FAIL

Para ello, es necesario realizar la lectura del puerto serial al que está conectado el módulo wifi.

3. **Lectura_A.** Esta función se encarga de leer la entrada analógica del canal correspondiente. Es necesario que se le indique a la función el canal con el que se quiere trabajar, por lo que es imprescindible pasarle dicho valor. A continuación, se realiza un doble bucle cuyo fin consiste en obtener diez valores máximos para, posteriormente realizar la media de dichos valores. Una vez obtenida la media, se realiza la transformación del valor calculado a un valor en el rango de tensión. Finalmente, se devuelve el valor medido y se almacena en una variable. Entre medias se ha realizado el cálculo de la potencia y se ha configurado la impresión por pantalla de los valores de corriente y potencia calculados. Sin embargo, estos valores solo serán visibles si el dispositivo se encuentra conectado a un ordenador.
4. **Enviar.** Esta función se encarga de enviar la información registrada en las variables correspondientes a la base de datos. Para ello, se deben realizar los siguientes pasos, por orden de escritura:
 - ❖ Iniciar conexión con base de datos.
 - ❖ Iniciar el envío de los datos.
 - ❖ Indicar la información a enviar con la estructura correcta. Es necesario indicar el archivo sobre el que vamos a escribir, así como los parámetros adecuados.
 - ❖ Cerrar la comunicación con la base de datos y así poder realizar un nuevo envío.

A continuación, se procederá a explicar la programación de las funciones “setup” y “loop”:

1. **Setup.** En ella se realiza la inicialización de todas las variables globales y de los puertos serial del monitor (para la visualización por pantalla) y del módulo wifi. También se realiza la llamada a la función *IniciaWifi*.
2. **Loop.** Mediante la función cíclica, se realiza la llamada a la función *Lectura_A* para los tres canales de medida. El valor devuelto en cada llamada es almacenado en una variable distinta. Posteriormente, se calcula la potencia total del sistema consumida por los tres canales. Finalmente, se establece un retraso de 2 segundos antes de que se realice un nuevo ciclo. Entre cada una de las lecturas se ha

establecido un retraso de 0.5 segundos para evitar que el procesador se sature y tenga tiempo para almacenar correctamente cada una de las lecturas.

7.2 Desarrollo de sistema de medición de audio

Durante los últimos años, el uso del sonido como mecanismo de monitorización de procesos de fabricación ha aumentado, especialmente para los procesos de acabado como molienda o el torneado. Gracias al estudio de este tipo de señales, pueden realizarse evaluaciones durante los procesos de fabricación demostrando así, la correlación entre la emisión acústica y la integridad de la superficie que está siendo sometida al proceso, dando lugar a una estimación de la calidad del producto. Todo ello, tiene como objetivo lograr procesos de fabricación más optimizados que permitan evitar rechazos o “reacabados” (varias operaciones de acabado) así como, asegurar costes económicos y ecológicos menores. Hay que destacar que muchos de los sistemas de la actualidad se basan en análisis de postproducción y que, por lo tanto, no pueden determinar directamente la integridad del material fabricado.

Los sistemas de monitorización mediante señales acústicas permiten medir las condiciones de las herramientas de trabajo o del propio proceso en sí mismo, reportando los problemas que puedan surgir mediante algún mensaje o estados del sistema (H.K. Tönshoff, 1999). Generalmente estos sistemas tienen como tareas principales:

- Funciones de control y localización de fallos en los componentes de la maquinaria.
- Procesos de control de reconocimiento de procesos de fallo.
- Reconocimiento de inexactitud de la maquinaria, lo que conlleva a la ausencia de calidad.
- Apoyo y operación de mantenimiento.

Los sistemas de monitorización de este calibre, pueden clasificarse dependiendo de la complejidad del proceso a analizar. De esta forma, se tendrá un sistema basado en el análisis de una señal, una comparativa entre señales o una clasificación. En la figura 27, aparecen las tres técnicas mencionadas anteriormente.

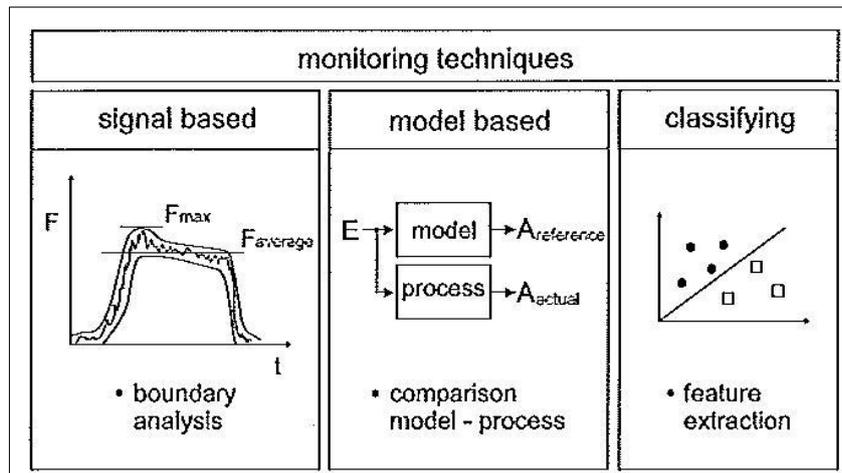


Figura 27. Técnicas de Monitorización (H.K. Tönshoff, 1999)

Los dos primeros modelos se determinan o bien de forma empírica o mediante relaciones físicas, siendo de gran importancia emplear las variables de entrada más adecuadas para lograr un análisis correcto. El último modelo tiene como objetivo la unión de un vector de características con una clase en concreto de características de calidades.

7.2.1 Desarrollo hardware

En esta ocasión, el sistema de monitorización a emplear está basado en la captura del mayor número posible de señales acústicas durante el proceso de fabricación de diferentes productos, siendo estos de diferentes materiales. Así mismo, los procesos de fabricación también serán variados, con el fin de lograr la mayor información posible y abarcar un mayor rango de operaciones y posibles estados de la maquinaria.

La estructura del sistema consta de un micrófono, un sistema de adquisición de señales analógicas, un ordenador portátil y la base de datos. En la figura 28, se muestra el diagrama del sistema.

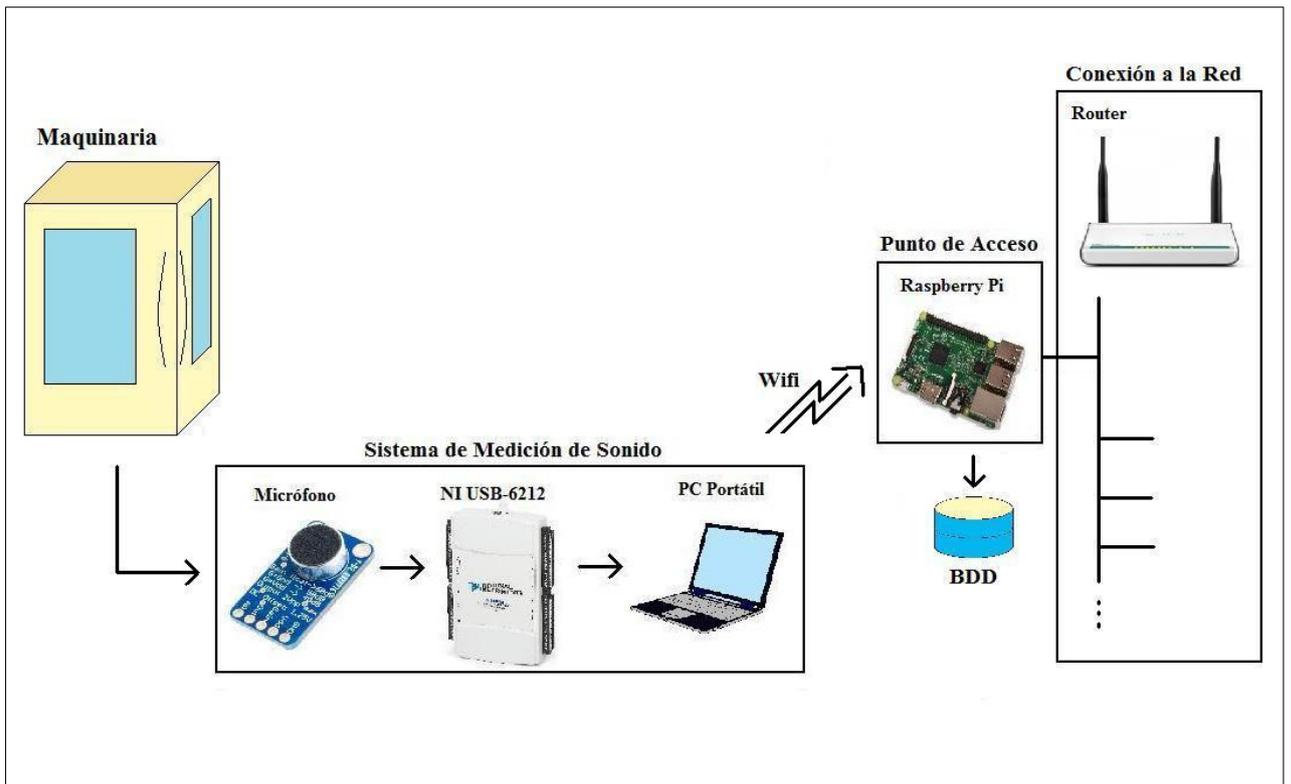


Figura 28. Diagrama del Sistema de Captación de Sonido

El funcionamiento del sistema consiste en la escucha, por parte del micrófono, del sonido generado durante el proceso de fabricación. Posteriormente, el sistema de adquisición de señales analógicas, mostrado en la figura 29, recoge la señal captada por el micrófono (introducido en la máquina), mostrado en la figura 30. A continuación, en el ordenador portátil se recoge el archivo que contiene la señal acústica y, donde se realiza el tratamiento de la misa. Al mismo tiempo, aquí se realiza una comparativa de las diferentes señales capturadas. Por último, se extrae la información más relevante de las señales y se envía a la base de datos creada.



Figura 29. USB-6212 Multifunción (National Instruments, 2018)



Figura 30. Micrófono para Escucha

Finalmente, en la tabla que se muestra a continuación se reflejan los diferentes dispositivos necesarios para poder realizar establecer el sistema de monitorización mediante señales acústicas.

ELEMENTO	DESCRIPCIÓN
Micrófono	Para captar la señal acústica
NI USB-6212 E/S Multifunción	Dispositivo para almacenar la señal acústica y transmitirla al ordenador
Ordenador portátil	Necesario para poder visualizar y almacenar en un archivo la señal acústica

Tabla 3. Componentes para medición de sonido

7.2.2 Desarrollo software

Para el desarrollo software, se va a contar con dos herramientas fundamentales:

1. **EAudio.**
2. **LabVIEW® de National Instruments.**

La primera de ellas está orientada a la adquisición y visualización de las señales acústicas captadas, mientras que la segunda se emplea para realizar el establecimiento de la conexión a la base de datos.

7.2.2.1 EAudio

Gracias a *EAudio*, se lleva a cabo la adquisición de la señal acústica y su posterior almacenamiento en un archivo, el cual se crea tras pulsar la opción “Captura de audio y obtención de espectros” (ver figura 31).



Figura 31. Menú principal de EAudio

Tras crear el archivo donde se almacenará la información, aparecerá una ventana en la que podrán configurarse algunos parámetros y comenzar a realizar una captura de sonido (ver figura 32).

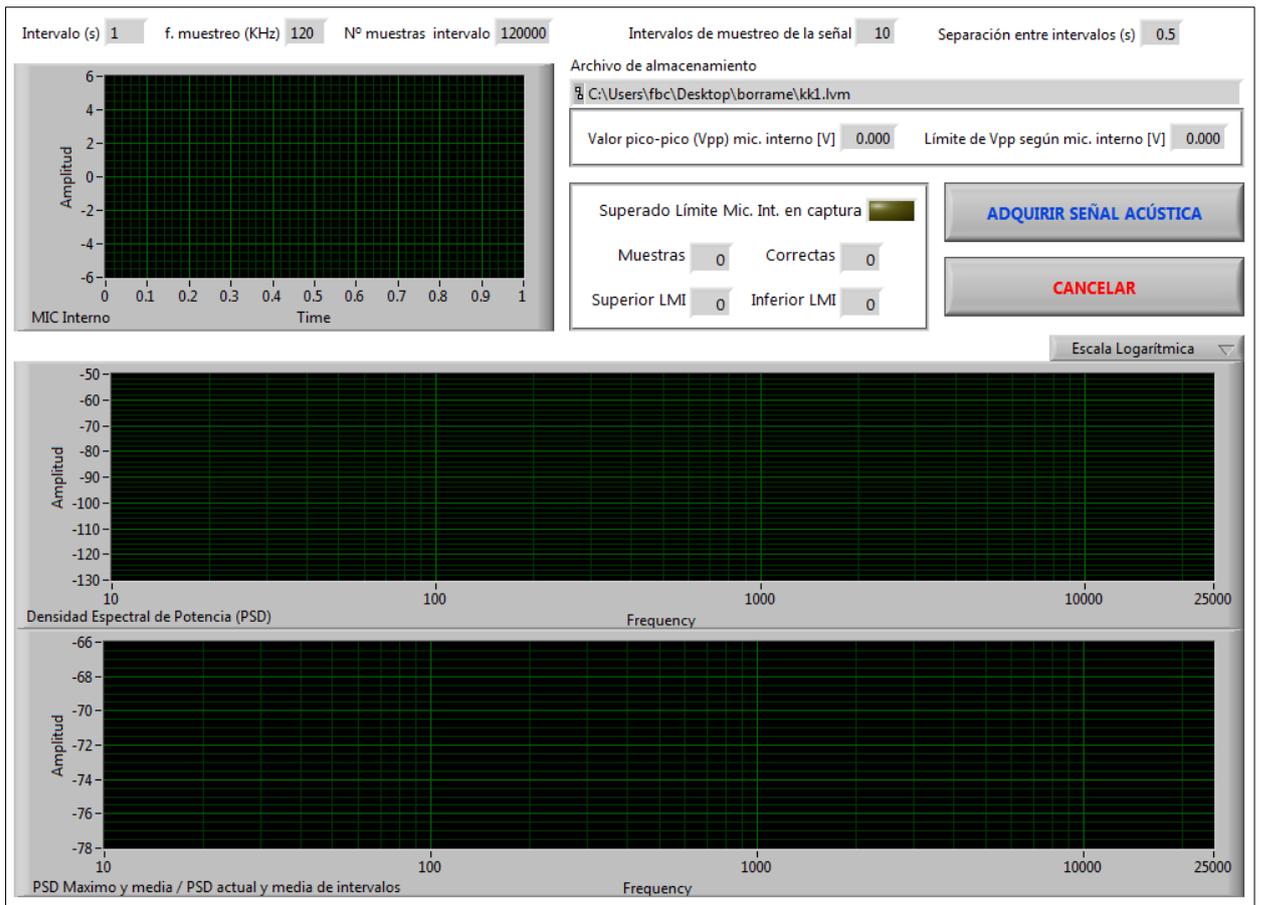


Figura 32. Menú de captura de audio

7.2.2.2 Labview

“LabVIEW® de National Instruments” es una herramienta que trabaja con un lenguaje gráfico por lo que, es un estilo de programación intuitivo basado en la relación entre bloques. A través de esta herramienta software, se va a realizar el análisis y envío de toda la información recabada anteriormente. De esta forma, se procederá a analizar las señales acústicas capturadas para poder detectar picos en las señales. Posteriormente, esta información será enviada a la base de datos creada.

Antes de realizar el envío de la información a la base de datos es necesario instalar un servidor “ODBC”, el cual permite el tráfico de datos entre el programa y la base de datos. Tras la instalación, se debe configurar un nuevo origen de datos “ODBC” por lo que, será necesario dirigirse a la siguiente ruta: *Panel de control → Todos los elementos de Panel de control → Herramientas administrativas* y hacer doble clic sobre *orígenes de datos ODBC*. Tras este hecho, aparecerá una ventana nueva con las configuraciones actuales y una serie de opciones (ver figura 33).

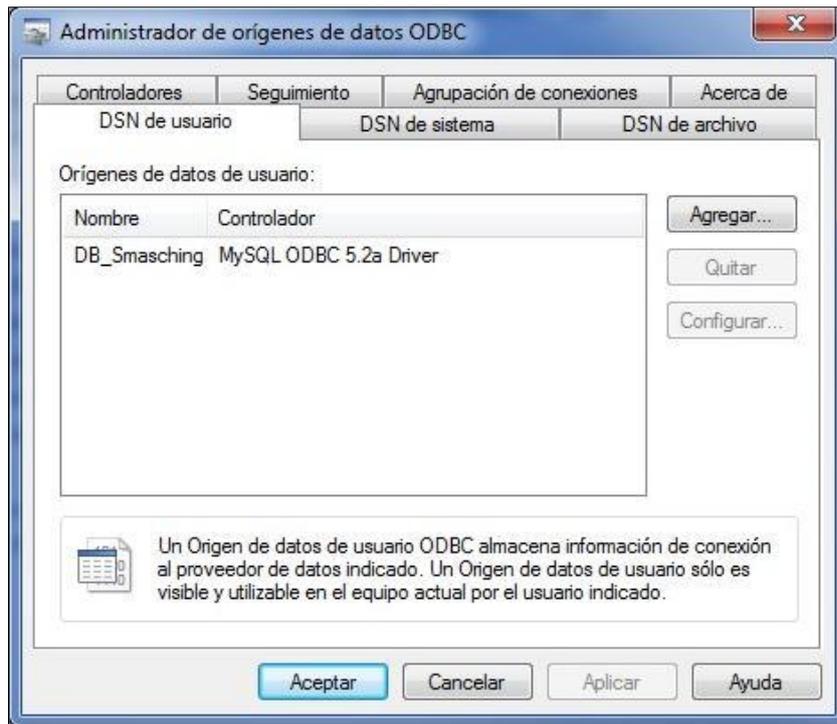


Figura 33. Configuraciones desarrolladas

En esta ocasión, se debe pulsar *Agregar* y, en la ventana que aparecerá después elegir la conexión “MySQL ODBC 5.2 Driver” (ver figura 34).

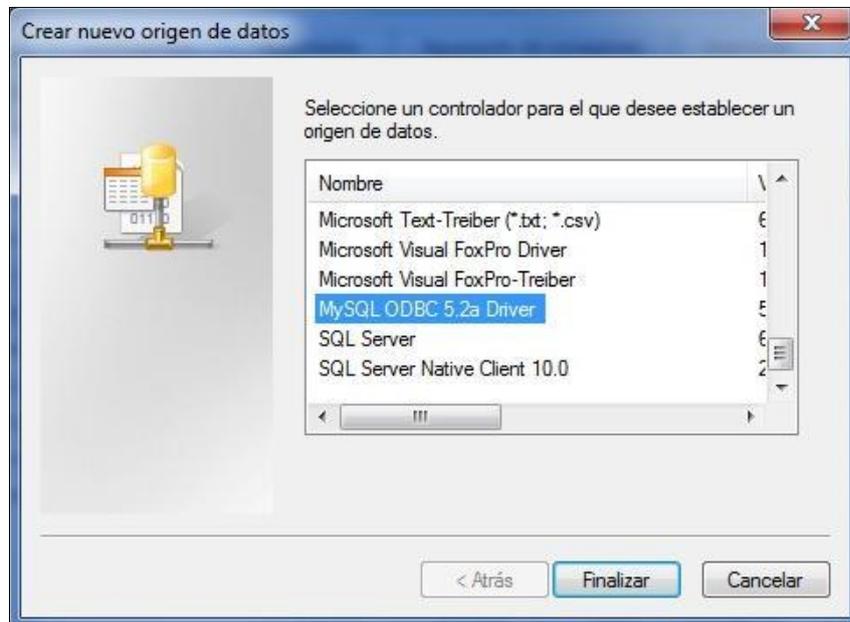


Figura 34. Driver de conexión

Tras pulsar el botón de *Finalizar*, surgirá una nueva ventana en la que se realiza la configuración de la conexión a “MySQL” (ver figura 35).

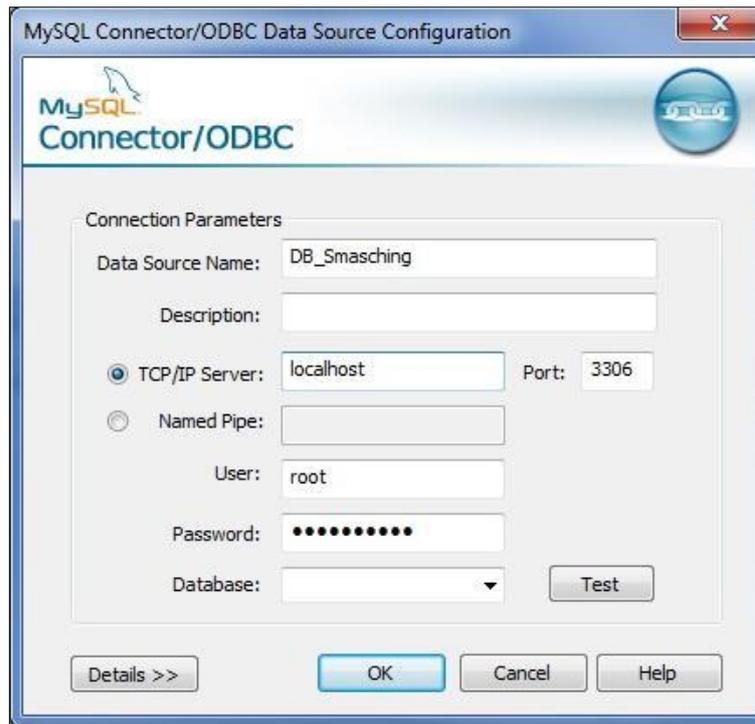


Figura 35. Configuración de conexión MySQL

Una vez realizada la configuración de la conexión, se deben situar los bloques necesarios para la misma en la carpeta de “Labview” denominada “lib”. Ahora ya puede comenzarse a desarrollar el programa para la conexión con la base de datos.

Al crear un nuevo proyecto en “Labview”, aparecerán dos ventanas llamadas *Panel frontal* y *Diagrama de bloques*. En el diagrama de bloques se lleva a cabo la programación gráfica, insertando los bloques necesarios. Para este caso, es necesario insertar los siguientes bloques:

- **ADO Create.** Crea una conexión nueva.
- **ADO Open.** Abre una conexión con una base de datos indicada.
- **ADO Connection.** Conecta con la base de datos indicada, permitiendo escribir en ella.
- **ADO Close.** Cierra la conexión con la base de datos.

Para poder insertarlos, hay que dirigirse a la paleta de funciones y pulsar *Select a VI* (figura 36, recuadro rojo) y, en la carpeta en la que situamos las librerías, elegir cada uno de ellos de forma individual. Una vez insertados se debe realizar la conexión entre los bloques (ver figura 36), además se debe establecer dos elementos de control en los que se especifica a qué base de datos se debe conectar el programa y los datos que debe introducir en ella.

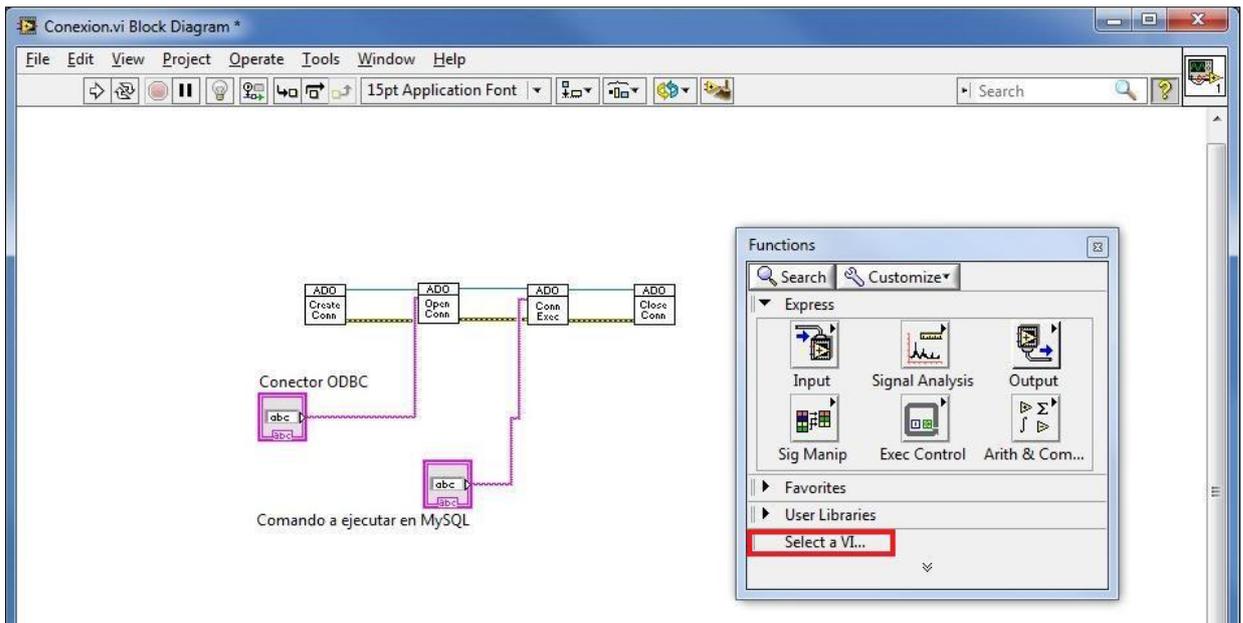


Figura 36. Diagrama de bloques de conexión a MySQL

En la ventana *Panel frontal*, se realiza la programación de los elementos de control. Como puede verse en la figura 36, se han añadido dos de tipo “String”. Ahora bien, en la nueva ventana debe establecerse el código necesario que ejecutará el bloque al que se encuentran conectados (ver figura 37).

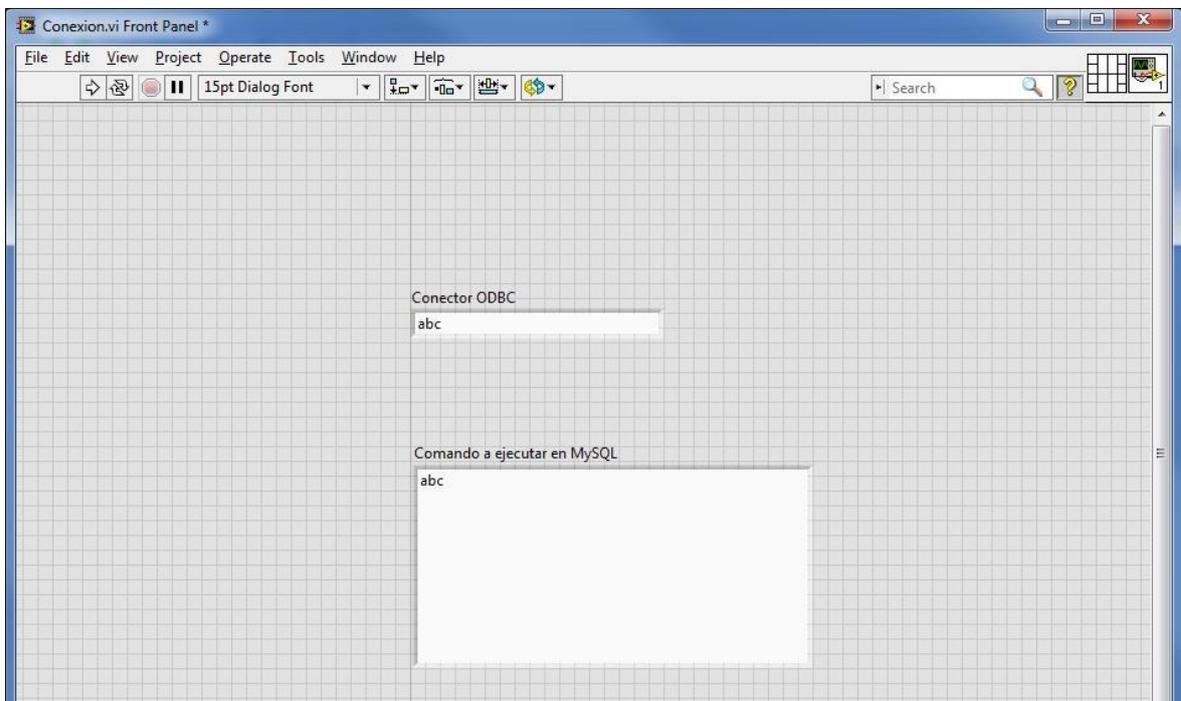


Figura 37. Panel frontal

En el elemento nombrado como “Conector ODBC”, se indica el nombre de la conexión que fue configurada con anterioridad y a la que se denominó “DB_Smasching”. La forma de escribir dicho código será: DSN=DB_Smasching.

Finalmente, en el elemento al se ha denominado “Comando a ejecutar en MySQL”, se desarrolla la programación para especificar en qué tabla de la base de datos se va a escribir y los datos que se van a enviar. Para ello, se debe especificar el siguiente código: INSERT INTO nombre_tabla (parámetros) VALUES (valores).

7.3 Desarrollo de sistema de medición de vibraciones/aceleraciones

Dentro del sector del mecanizado, uno de los principales problemas que suelen surgir son las vibraciones auto-excitadas conocidas como “chatter” (Federico Fernández G., 1994). Dicho suceso tiene lugar al paso de la herramienta de corte a través del material que se está mecanizando, derivando en un aumento de la intensidad de tal forma que, puede aumentarse el desgaste de la herramienta o incluso la rotura de la misma. En la figura 38, se muestra una imagen del paso de la herramienta por el material y una gráfica con los puntos estables e inestables.

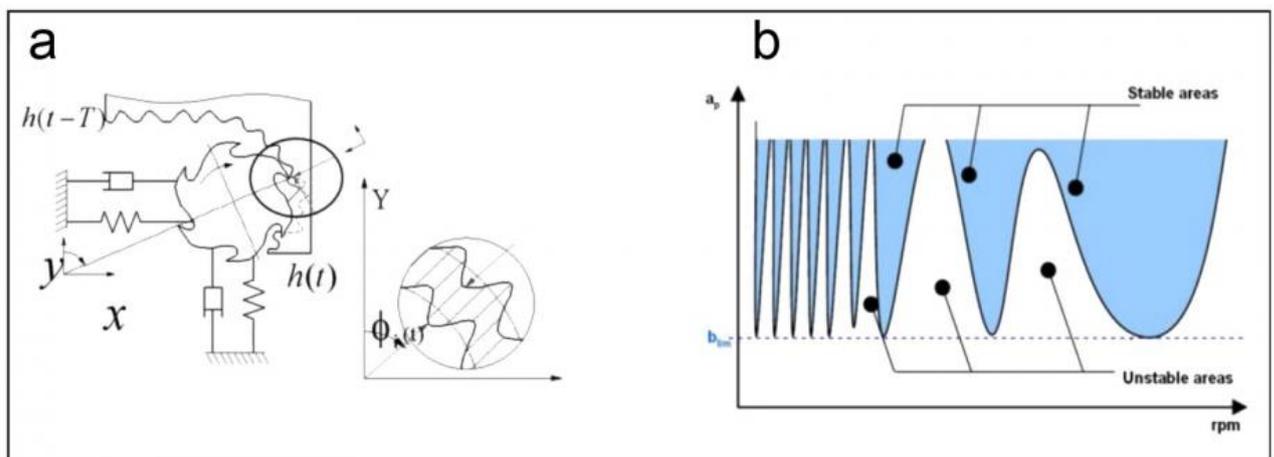


Figura 38. Esquema y gráfica del chatter

La aparición de este fenómeno puede deberse a dos causas:

1. **Regeneración de ondulaciones.** Al pasar la herramienta de corte por el material, cada diente de esta encuentra la superficie producida por el diente precedente. En función de la fase entre las ondulaciones producidas por cada diente, la amplitud de las vibraciones será mayor o menor.
2. **Vibración forzosa resonante.** Tiene lugar cuando la frecuencia de paso de cada diente de la herramienta es próxima a la frecuencia natural producida por el sistema

máquina-herramienta. Puede evitarse configurando una correcta velocidad de fabricación.

Ante toda esta situación, surge la necesidad de desarrollar un dispositivo electrónico que sea capaz de detectar las vibraciones excesivas durante el proceso de mecanizado, evitando así el desgaste acelerado de la herramienta de corte o la rotura de la misma. Las posibilidades de realizar un dispositivo de este calibre son varias:

- ❖ Micrófonos
- ❖ Acelerómetros
- ❖ Velocímetros

Pero, en este caso, se ha decidido la instalación de un acelerómetro ya que su precio es muy asequible y su incorporación al sistema es sencillo, gracias a que su tamaño es muy pequeño. Añadido a esto, se encuentra el hecho de que toda la programación del sistema puede realizarse a través del software de “Arduino”, una gran ventaja teniendo en cuenta que se está familiarizado con el entorno de esta herramienta al haber sido usada para el desarrollo del dispositivo de medición de consumos. A parte de este elemento, serán necesarios otros para poder completar un dispositivo capaz de adquirir la información y enviarla a la base de datos. A lo largo de este apartado, se especifica el desarrollo de este sistema tanto a nivel de hardware como de software.

7.3.1 Desarrollo hardware

Lo que se busca con el desarrollo de este dispositivo es poder determinar si durante el proceso de fabricación tienen lugar sobreesfuerzos por parte del centro de mecanizado. Para ello, el dispositivo creado será insertado en el interior de un portaherramientas de forma que, puedan medirse las aceleraciones producidas durante todo el proceso. Además, todos los datos son enviados a la base de datos para registrar toda la operación. En la figura 39, se representa un diagrama de bloques que refleja el proceso de captación y guardado de la información.

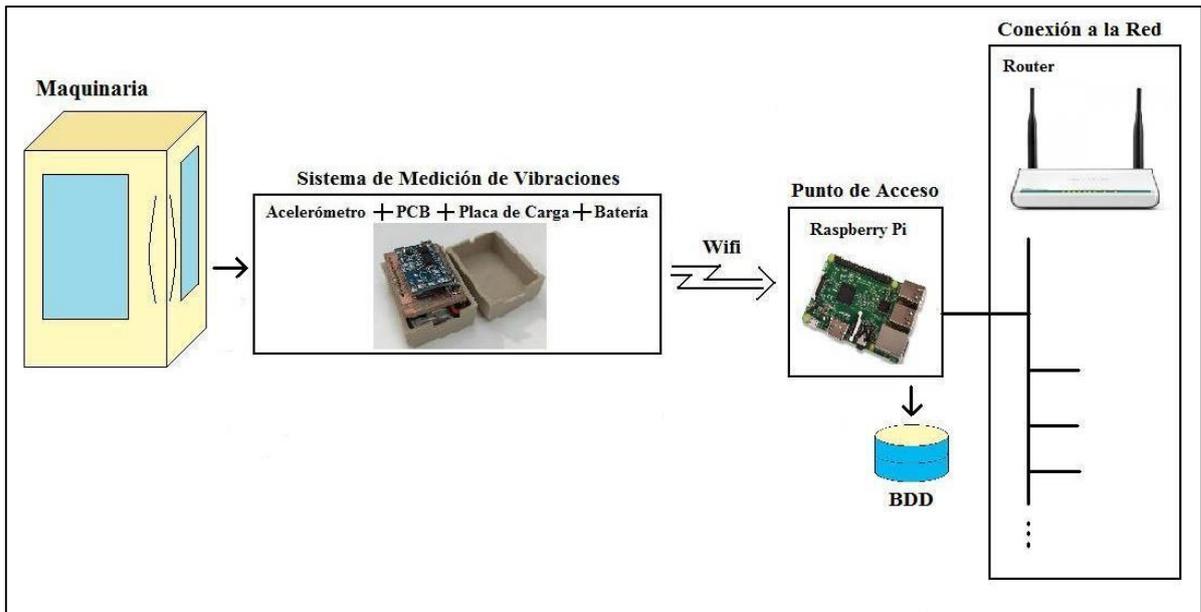


Figura 39. Diagrama de bloques del Sistema de Aceleraciones

El acelerómetro ADXL362 (Analog Devices, 2017), es un componente que permite la medida de aceleraciones en tres ejes con un consumo muy bajo de potencia. Las medidas se corresponden con aceleraciones debidas a movimientos, impactos/vibraciones e inclinaciones. Los datos se reportan digitalmente y la comunicación está basada en el protocolo SPI. Las medidas son captadas gracias a un sensor que consiste en una estructura de polisilicio con superficie micromecánica construida sobre una oblea de silicio. Los muelles de polisilicio, colocados sobre la oblea, muestran una resistencia en contra de las fuerzas de aceleración. Las desviaciones en la estructura son medidas gracias a un condensador, el cual está basado en unas placas fijas y otras adjuntas, para medir el movimiento. La medida que muestra el sensor es proporcional a la aceleración. Finalmente, la sensibilidad de la fase indica la magnitud y polaridad de la aceleración. En la figura 40, puede verse una imagen de la placa empleada con el acelerómetro incorporado.

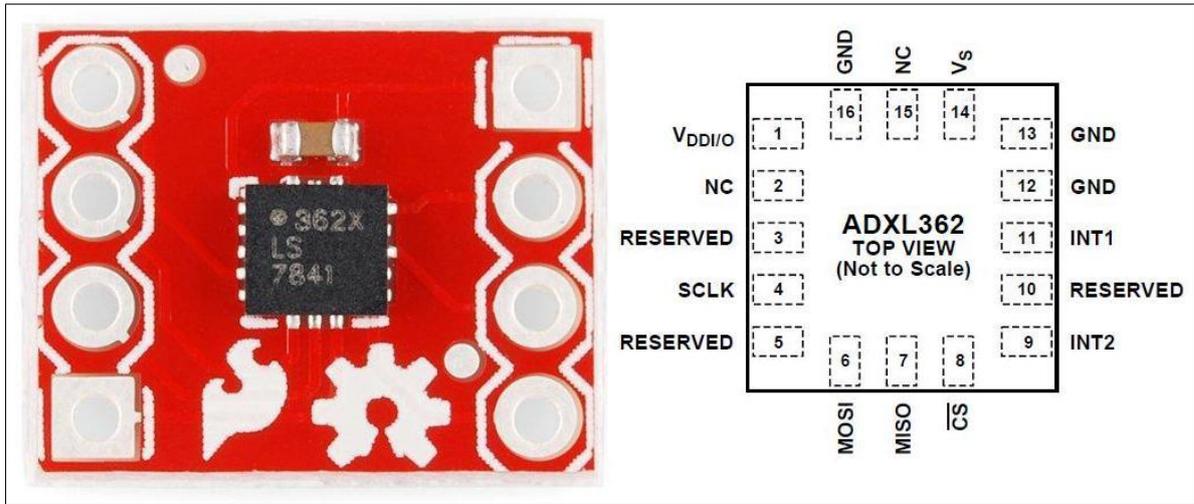


Figura 40. Placa con Acelerómetro y esquema del mismo

Para la comunicación con la base de datos se emplea una placa de desarrollo WEMOS D1 Mini Pro, la cual está basada en el módulo ESP8266 (módulo wifi para “Arduino”), que incorpora un chip wifi con antena cerámica. En la figura 41, puede verse este elemento.

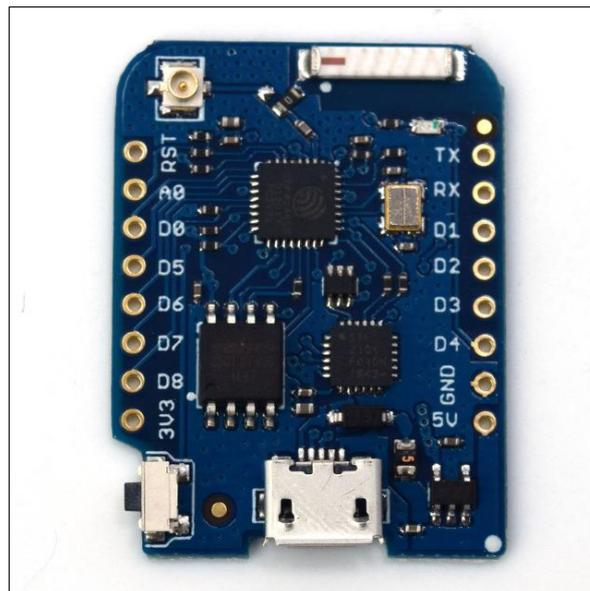


Figura 41. WEMOS D1 Mini Pro

Por último, para el diseño de la batería se ha tenido en cuenta el consumo de ambos elementos en conjunto, ya que el dispositivo final está compuesto por ambos funcionando a la vez. De forma individual, el acelerómetro consume unos 3 uA a 400 Hz de frecuencia y, la placa “WEMOS” unos 73.2 mAh, por lo que el valor del acelerómetro puede despreciarse. Si se emplea una batería de Litio-Polímero, existen en el mercado diferentes modelos asequibles, pero, en este caso, se necesita una con un tamaño similar a la placa por lo que la mejor opción es una batería de 380 mAh, que aportará una duración de unas 5 horas (suponiendo un funcionamiento a pleno rendimiento de forma continuada).

Para poder cargar la batería, se ha incorporado una placa de carga basada en la placa TP4056 (ver figura 42), la cual permite cargar la batería y a la vez poder cargar una configuración a la placa.

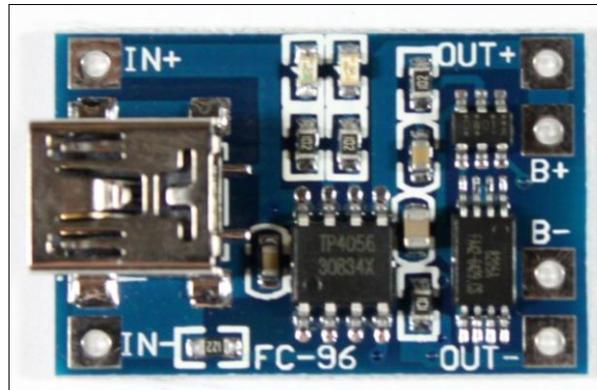


Figura 42. Placa de carga

Una vez desarrollado de forma individual cada uno de los elementos, solo queda unirlos a todos juntos formando un único dispositivo capaz de realizar medidas de vibraciones, transmitirlos a la base de datos y de forma autónoma (gracias a la batería). En la figura 43, puede verse el dispositivo final insertado en una pequeña caja para almacenarlo.

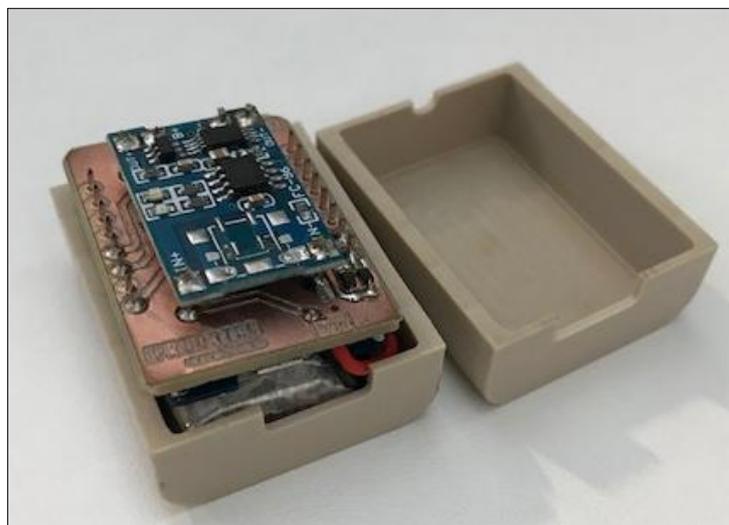


Figura 43. Dispositivo de medición de vibraciones

Adicionalmente, ha sido necesaria la fabricación de un portaherramientas con cono HSK63A, ya que es la sujeción que incorpora el centro de mecanizado “Deckel”. En la figura 44, puede verse dicho portaherramientas junto con el dispositivo almacenado en su interior.



Figura 44. Portaherramientas con dispositivo de medición de vibraciones

Para finalizar, se presenta una tabla que resume los diferentes elementos hardware, necesarios para poder realizar este dispositivo electrónico.

ELEMENTO	DESCRIPCIÓN
Acelerómetro	Modelo ADXL362
Placa de desarrollo con conexión wifi	Modelo WEMOS D1 mini Pro
Batería compacta	Para alimentación autónoma
Placa de carga	Placa que permite cargar la batería. Modelo FC-96

Tabla 4. Componentes para medición de vibraciones

7.3.2 Desarrollo software

Para desarrollar el programa que permita la recolección de los datos y el envío de los mismos a la base de datos, se ha empleado el software de “Arduino”. Gracias a que los componentes usados están basados en la estructura interna de otros compatibles con la casa “Arduino”, es posible operar con esta herramienta.

Antes de explicar las operaciones que tienen lugar en cada una de ellas, se va a proceder a explicar algunos pasos previos necesarios para implementar librerías y variables globales. De esta forma, es necesario incluir las librerías correspondientes al acelerómetro, a la comunicación “SPI” y a la placa “WEMOS” que, en este caso, puede emplearse la misma que para el módulo “ESP8266”. Además, es necesario incluir una librería para la gestión del dispositivo en el modo “Low Power”, un modo de operación de consumo mínimo de potencia. Posteriormente, se han declarado parámetros globales para el acelerómetro como los tres ejes de medición, la temperatura del sensor; para la comunicación wifi como la red a la que se desea conectar (identificador y contraseña) y; para la comunicación “SPI” como el canal de comunicación, la clave de comunicación, el servidor y el periodo de muestreo. Además, se han declarado una instancia para la comunicación wifi y otra para el acelerómetro.

Una vez que se han establecido las variables necesarias, se procede a explicar el contenido de las funciones “setup”, “loop” e “interruptFunction”.

- **Setup.** En ella se realiza la inicialización del puerto serie, la conexión a la red mediante el paso del nombre y la contraseña de la misma y, del acelerómetro para que comience a realizar mediciones. Además, se ha establecido un bucle tipo *while* de forma que, tras la conexión mediante wifi, si esta no está correctamente conectada se mantenga en él con un *delay* de 500 milisegundos. También se establece el pin 7 como una salida de conexión aun diodo LED que servirá como indicador visual del modo del dispositivo (dormido/desperto). Además, se han incorporado las condiciones que establecen los límites de activación/desactivación del acelerómetro, lo que permitirá al dispositivo cambiar de manera automática del modo despierto al modo dormido. Finalmente, se establecen los mapeados de los distintos modos de funcionamiento, se realiza un chequeo de las configuraciones realizadas y se da comienzo al proceso de medición del acelerómetro.
- **Loop.** Dependiendo del modo de operación en el que se encuentre el dispositivo, las funciones a realizar por parte de este variarán. Si el dispositivo se encuentra en modo de inactividad o dormido, el dispositivo entrará en modo de bajo consumo (apagando el diodo LED a modo de indicador visual de inactividad) y desactivando el conversor ADC. Por el contrario, si el dispositivo está operando en modo despierto o actividad, se registran las lecturas de los tres ejes de trabajo (X, Y, Z) y se imprimen por pantalla junto a un mensaje preestablecido. A continuación, si el cliente se ha conectado correctamente al servidor y puerto correctos se mostrará un mensaje indicando el lugar al que se ha conectado, se realizará la lectura de la entrada analógica correspondiente, se construye el cuerpo de solicitud API y, finalmente, se envía la información a la base de datos. En caso de que la conexión falle al inicio, se mostrará un mensaje de error de conexión y el cliente detendrá las peticiones. En último lugar, se establece un *delay* de 20 milisegundos antes de comenzar un nuevo registro.

7.4 Sistema web de transferencia y tratamiento de datos

Para lograr que toda la información sea almacenada y tratada, se ha llevado a cabo la instalación y configuración de un sistema que permita trabajar con todos los datos. Dicho sistema consiste en un clúster, lo que implica la unión de un conjunto de ordenadores a través de una red comportándose como si fuera un único ordenador. Algunas de sus utilidades más comunes son, el uso de software para aplicaciones críticas, servidores web, bases de datos de alto rendimiento, etc.

Puesto que uno de los objetivos principales del proyecto es la de desarrollar un sistema de bajo coste, se ha implementado un clúster compuesto por cuatro RPi (Raspberry Pi) y un router con una arquitectura de conexión para el servidor web denominada *balanceador de carga*. En la figura 45, se representa el diagrama con dicha estructura.

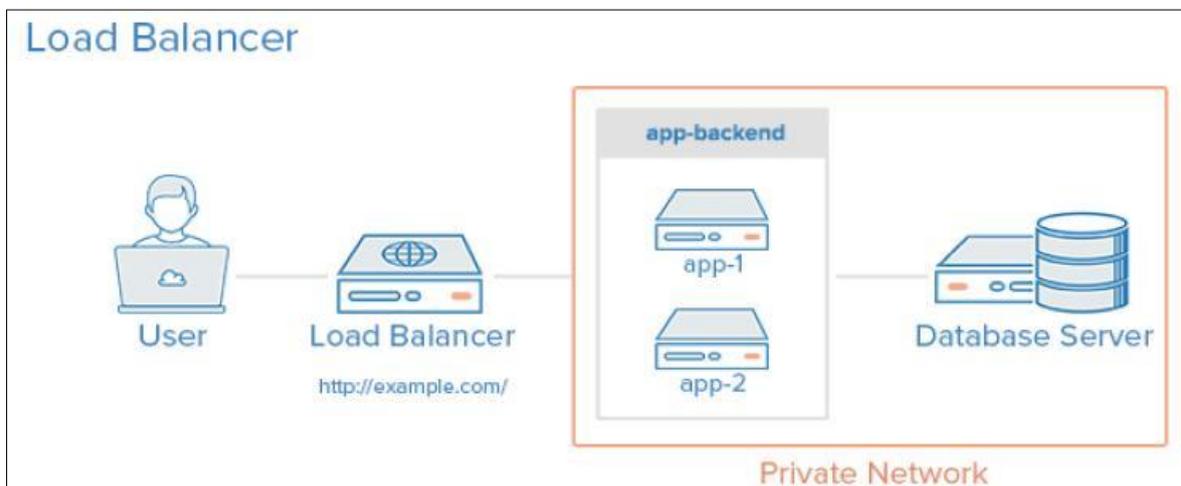


Figura 45. Diagrama de la estructura Balanceador de Carga

Esta estructura consiste en un equipo principal que actúa como balanceador de cargas (peticiones al servidor), dos equipos espejo que contienen el servidor web y un cuarto equipo que se encarga exclusivamente de gestionar la base de datos. En todos ellos, se ha procedido a instalar el sistema operativo "Ubuntu 16.04" y configurándolo como servidor "LAMP" (Linux Apache MySQL PHP), mostrado en la figura 46. El motivo de realizar este tipo de configuración es debido a que todo este software es "Open Source" gratuito por lo que, de esta forma, se ahorran los costes derivados del software asumiendo únicamente los asociados al hardware.

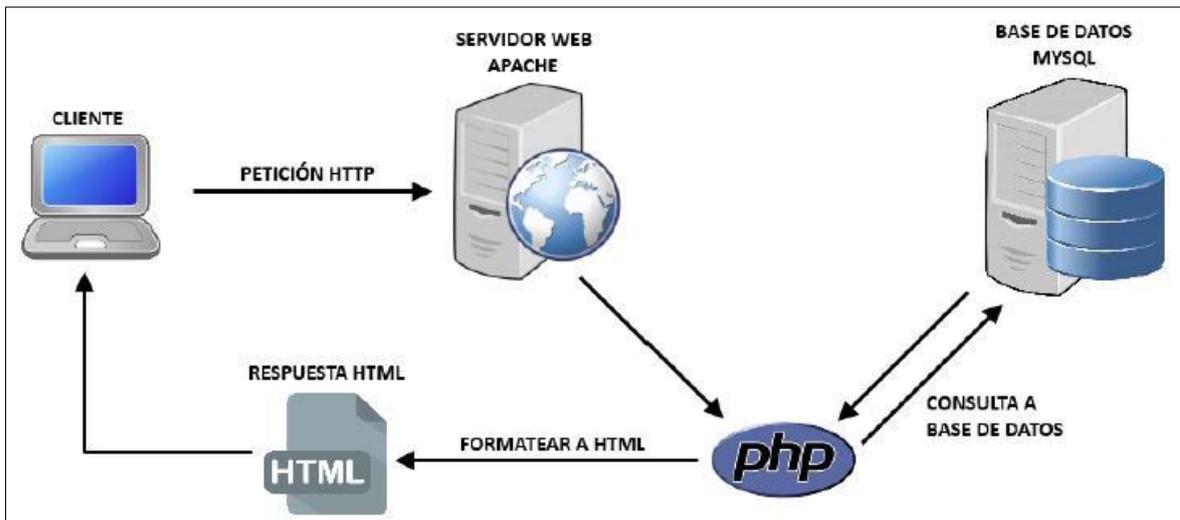


Figura 46. Diagrama de funcionamiento del sistema LAMP

Con este tipo de configuración, un cliente realiza una consulta al servidor, este se comunica a través de un archivo en lenguaje “PHP” con la base de datos realizando una consulta y, finalmente, la respuesta que recibe el cliente es traducida a “HTML”.

Como puede observarse, la web del servidor ha sido desarrollada en “HTML” y “PHP”. Ambos lenguajes unidos, permiten desarrollar páginas web dinámicas que interactúan con el usuario, solicitando información y configurada según las necesidades de este. El funcionamiento consiste en la generación del archivo “PHP” en el servidor, el cual genera la página web con la configuración del usuario, y envía el código “HTML” al equipo que hizo la consulta. De esta forma, el cliente únicamente recibe el archivo “HTML” sin visualizar el código de programación de la web. El motivo de este proceso es por seguridad, evitando así modificaciones indeseadas del código.

Finalmente, se ha procedido a la instalación de todo el hardware sobre una estructura única y rígida para poder desplazarla en cualquier momento. En las figuras 47 y 48, que se muestran a continuación, puede observarse en primer lugar el clúster físico y, en segundo lugar, la **Raspberry Pi 3 modelo B** empleada para todo el sistema desarrollado.

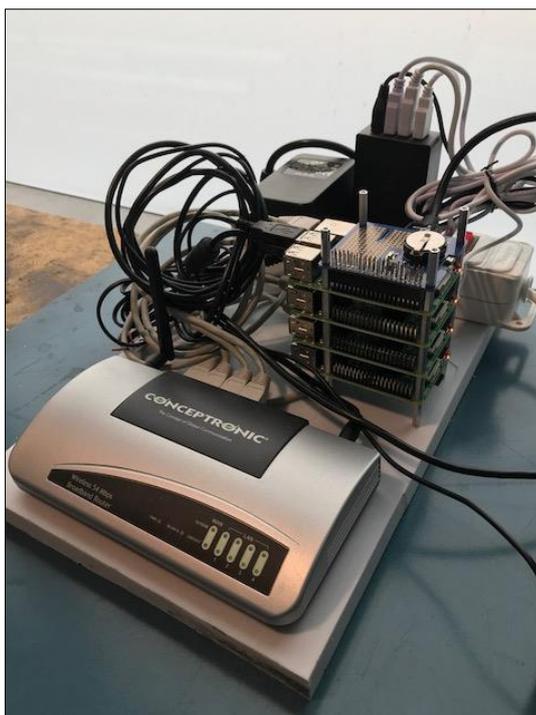


Figura 47. Clúster



Figura 48. Raspberry Pi 3 modelo B

7.5 Desarrollo de base de datos MySQL

Como ya se indicó en el apartado 3.2, la base de datos con la que se va a trabajar es del tipo “MySQL” (consultar características en el apartado 3.2.1).

Debido a la estructura que se ha establecido en el clúster, únicamente una de las RPi’s es la encargada de gestionar la base de datos (la RPi3 en esta ocasión). Para ello, se emplea el software “PHPMyadmin”, una herramienta “Open Source” gratuita, que controla el acceso a la base de datos y las modificaciones en la misma, como creación de tablas o el borrado de registros.

Para comenzar, se ha realizado un esquema para mostrar las relaciones que existen entre las diferentes tablas creadas (ver figura 49). Para ello, se ha empleado la herramienta “Microsoft Access” donde se incorpora cada una de las tablas con sus campos correspondientes.

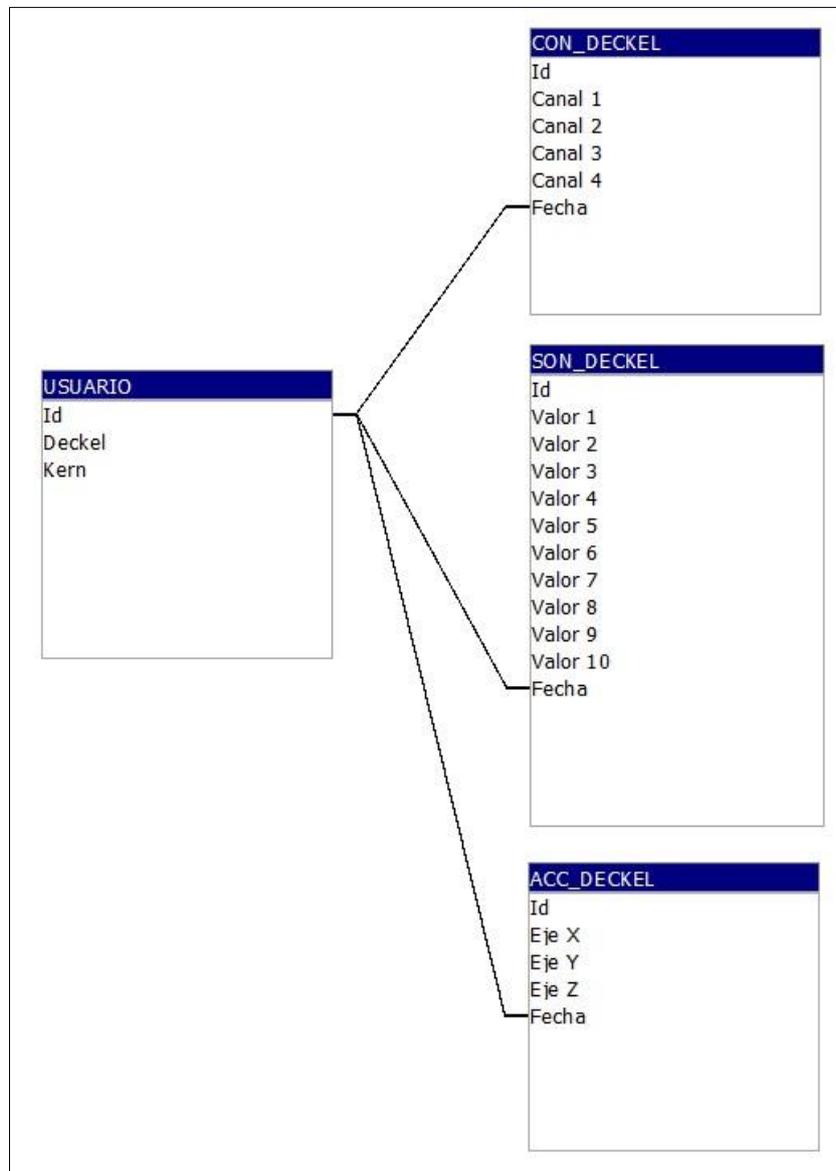


Figura 49. Relaciones existentes en la Base de Datos

Como se muestra en la figura 49, en ella pueden verse las relaciones existentes entre el identificador de usuario correspondiente a la máquina que se encuentra en funcionamiento y las fechas en las que han tenido lugar los registros de los datos para cada una de las tablas.

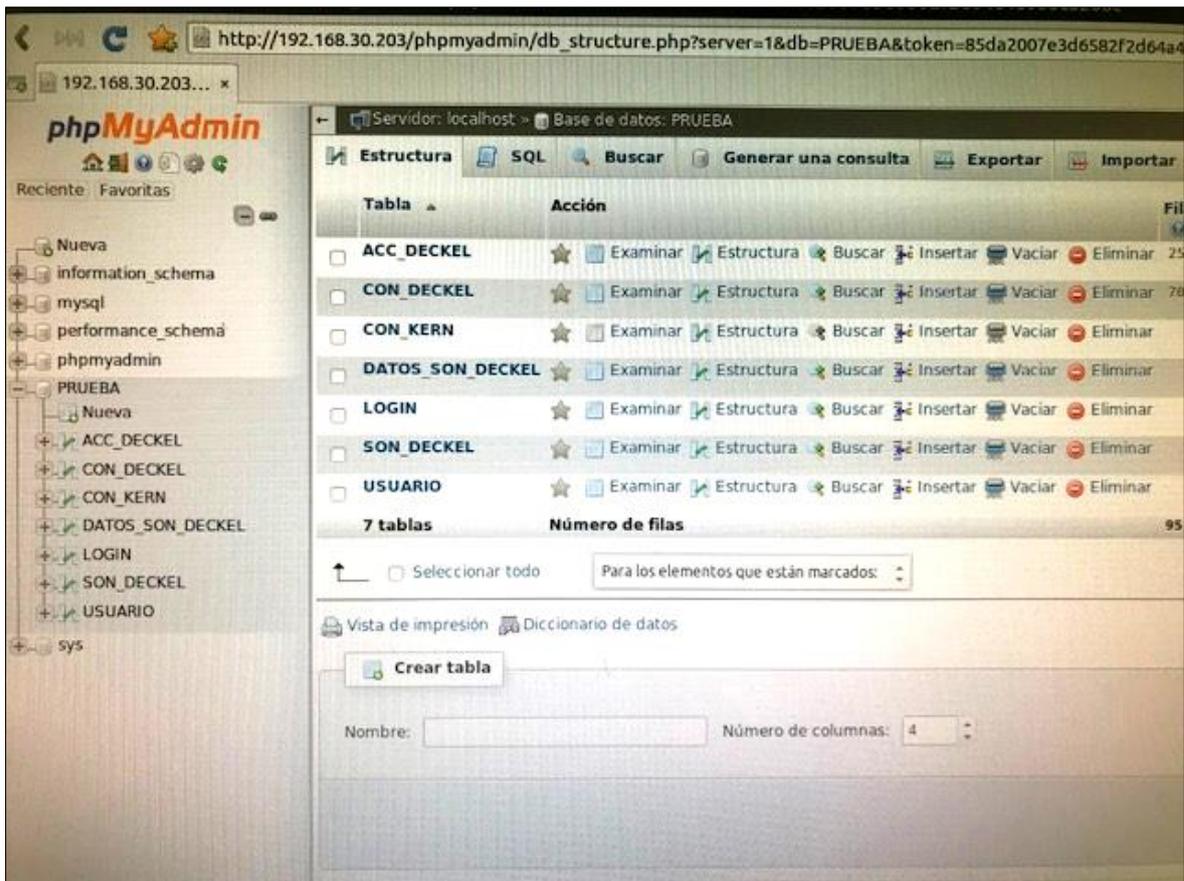


Figura 50. Base de Datos desarrollada

En la figura 50, se muestra la base de datos desarrollada, en ella puede observarse que existen creadas varias tablas de registros para poder almacenar la información relativa a los consumos eléctricos, las aceleraciones/vibraciones y el sonido del centro de mecanizado, así como otra máquina de iguales características, pero de la casa “Kern”.

Hay que especificar que la base de datos puede modificarse mediante la creación de tantas tablas como sean necesarias o modificando alguna de las ya existentes. Para crear una tabla nueva, se debe pulsar en “Nueva” (opción ubicada en el árbol de la izquierda), a continuación, indicarle un nombre a la tabla (en el campo “Nombre”) y establecer las columnas que queramos (en el campo “Número de columnas”).

En todas las tablas creadas, se ha establecido una columna que indica el “ID” numérico de cada registro, las columnas para las medidas y, una última para indicar la fecha (formato año/mes/día hora) en la que se ha realizado el registro. Para concretar más profundamente, en las figuras que se observan a continuación, muestran los campos de cada una de las tablas de interés.

		Número de filas: 25		Filtrar filas: Buscar en esta tabla			
Ordenar según la clave: Ninguna							
+ Opciones							
← T →							
		id	canal1	canal2	canal3	canal4	fecha
<input type="checkbox"/>	Editar Copiar Borrar	70897	273	117	283	131	2017-12-27 11:21:29
<input type="checkbox"/>	Editar Copiar Borrar	70896	264	122	151	130	2017-12-27 11:21:24
<input type="checkbox"/>	Editar Copiar Borrar	70895	254	137	229	131	2017-12-27 11:21:19
<input type="checkbox"/>	Editar Copiar Borrar	70894	264	127	283	131	2017-12-27 11:21:15
<input type="checkbox"/>	Editar Copiar Borrar	70893	254	156	278	138	2017-12-27 11:21:10
<input type="checkbox"/>	Editar Copiar Borrar	70892	254	117	215	125	2017-12-27 11:21:05
<input type="checkbox"/>	Editar Copiar Borrar	70891	269	112	288	128	2017-12-27 11:21:00
<input type="checkbox"/>	Editar Copiar Borrar	70890	259	127	352	130	2017-12-27 11:20:55

Figura 51. Tabla de consumos

En la figura 51, se representa la tabla con los registros de consumo de la máquina “Deckel Maho”. En ella, se ha establecido (como se mencionó anteriormente) una columna para identificar numéricamente cada registro, tres columnas (canal1, canal2 y canal3) para registrar las mediciones de cada una de las fases de la máquina, otra columna (canal4) donde se establece el consumo total de potencia de las tres fases y, una última columna (fecha) para registrar la fecha y la hora de cada registro. Cada uno de ellos corresponde a una fila de la tabla.

		Número de filas: 25		Filtrar filas: Buscar en esta tabla			
Ordenar según la clave: PRIMARY (Descendente)							
+ Opciones							
← T →							
		id	X_acc	Y_acc	Z_acc	fecha	
<input type="checkbox"/>	Editar Copiar Borrar	25236	-40	-44	1072	2017-12-19 13:18:20	
<input type="checkbox"/>	Editar Copiar Borrar	25235	102	24	1192	2017-12-19 13:18:20	
<input type="checkbox"/>	Editar Copiar Borrar	25234	-45	-52	1086	2017-12-19 13:18:15	
<input type="checkbox"/>	Editar Copiar Borrar	25233	-41	-51	1072	2017-12-19 13:18:14	
<input type="checkbox"/>	Editar Copiar Borrar	25232	-42	-47	1076	2017-12-19 13:18:14	
<input type="checkbox"/>	Editar Copiar Borrar	25231	-36	-48	1084	2017-12-19 13:18:13	
<input type="checkbox"/>	Editar Copiar Borrar	25230	-39	-47	1075	2017-12-19 13:18:13	

Figura 52. Tabla de aceleraciones/vibraciones

En la figura 52, se representa la tabla de registros de las aceleraciones/vibraciones en la que, igual al caso anterior, se ha establecido una columna con identificación numérica de cada registro, tres columnas que corresponden a las mediciones de los ejes X, Y, Z y, una columna con la fecha y hora de cada registro.

id_son	fecha	VALOR1	VALOR2	VALOR3	VALOR4	VALOR5	VALOR6	VALOR7	VALOR8	VALOR9	VALOR10
107	2017-11-21 12:10:34	1	2	3	4	5	6	7	8	9	11
106	2017-11-21 09:22:46	1	2	3	4	5	6	7	8	9	15
105	2017-11-21 09:22:38	1	2	3	4	5	6	7	8	9	15
104	2017-11-21 09:22:31	1	2	3	4	5	6	7	8	9	15
103	2017-11-21 09:22:22	1	2	3	4	5	6	7	8	9	15
102	2017-11-21 09:22:14	1	2	3	4	5	6	7	8	9	15
101	2017-11-21 09:22:01	1	2	3	4	5	6	7	8	9	15
100	2017-11-21 09:21:50	1	2	3	4	5	6	7	8	9	15

Figura 53. Tabla de sonido

En la figura 53, se muestra la tabla que recoge los registros de sonido más significativos de una señal captada. De esta forma, aparte de las columnas de identificación numérica y de la fecha y hora de cada registro, se han establecido diez columnas para albergar los valores más significativos de la señal analizada.

Las configuraciones de la base de datos también pueden ser realizadas a través del terminal que incorpora “Linux” pero, ante la facilidad que ofrece una interfaz visual como “PHPMysqladmin”, es preferible realizar todo el proceso a través de dicha herramienta con el fin de lograr un entendimiento sencillo para el usuario. Además, de esta forma se evita el tener que estar memorizando o consultando comandos de ejecución, por lo que se reduce el tiempo en todo el proceso de configuración de la base de datos.

7.6 Desarrollo de sistema predictivo. Cubos OLAP

Tras todo el proceso de adquisición y almacenamiento de información en la base de datos por parte de los dispositivos electrónicos fabricados, es hora de emplear todos ellos para poder realizar análisis inteligentes y detectar situaciones críticas o complicadas para, poder tomar decisiones y solucionar o evitar problemas. Para lograr tal fin, se va a emplear un

sistema conocido como “Cubos OLAP”. Gracias a este sistema, se podrán estudiar todos los datos a un mismo tiempo ya que, se basa en análisis multidimensionales de información⁴.

Lo primero que se debe de establecer, es la idea de la información que se quiere analizar a partir de la base de datos con la que se va a trabajar. De esta forma, y en este caso, lo que se plantea analizar son los datos relativos a los consumos eléctricos, las aceleraciones y el sonido registrados para una maquinaria específica y en unas fechas determinadas. Para evitar hacer un desarrollo largo y repetitivo, se va a presentar la configuración y el diseño de un cubo para analizar los consumos eléctricos registrados por el centro de mecanizado “Deckel Maho”. Para poder analizar el resto de datos, bastará con repetir el proceso de configuración y diseño, pero, teniendo en cuenta las tablas correspondientes en cada caso.

Tras tener claros los conceptos a analizar, el siguiente paso será comenzar a realizar las primeras configuraciones con la herramienta de diseño del cubo, llamada “Mondrian Schema Workbench”. Lo primero que se debe hacer con esta herramienta, es establecer la conexión con la base de datos correcta, usando el driver específico para ello. Para acceder a este apartado, basta con seguir la siguiente ruta: *Tools* → *Connection*, y rellenar los campos que aparecen en la ventana (ver figuras 54 y 55).



Figura 54. Ruta para establecer la conexión

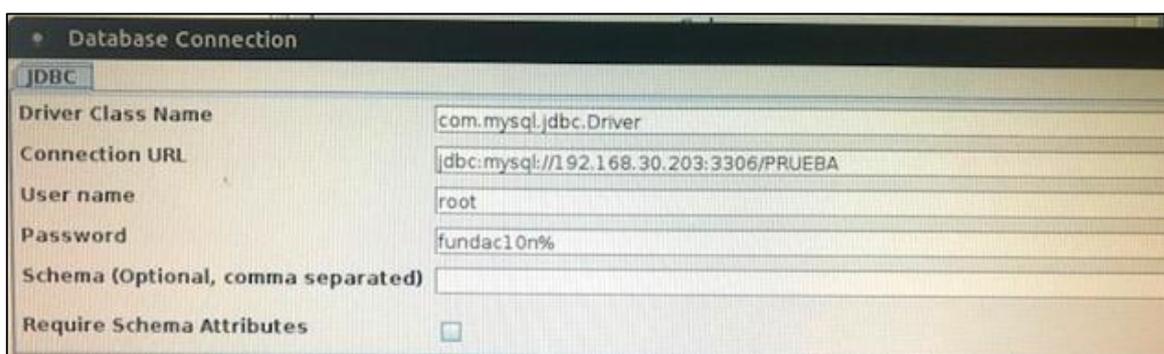


Figura 55. Establecimiento de conexión

Con la conexión establecida, ya se puede comenzar con el desarrollo del cubo. Para ello, se crea un esquema nuevo siguiendo los siguientes pasos: *File* → *New* → *Schema*. Seleccionando el nuevo esquema creado, aparecerá una ventana en la parte derecha donde se especifica la

⁴ Consultar apartado 3.3.1 para comprender claramente el concepto multidimensional de los Cubos OLAP.

información del mismo (ver figuras 56 y 57). Es aquí donde se establecerá el nombre que se desee al esquema.

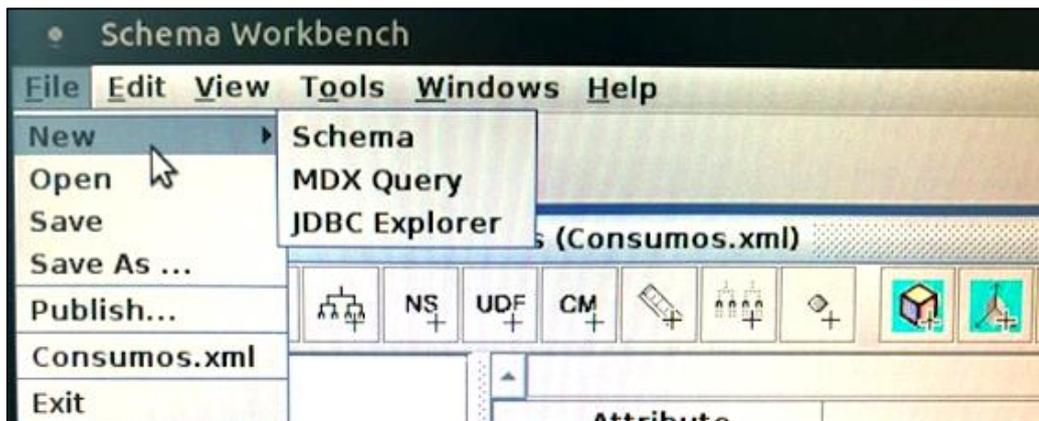


Figura 56. Creación de un esquema nuevo

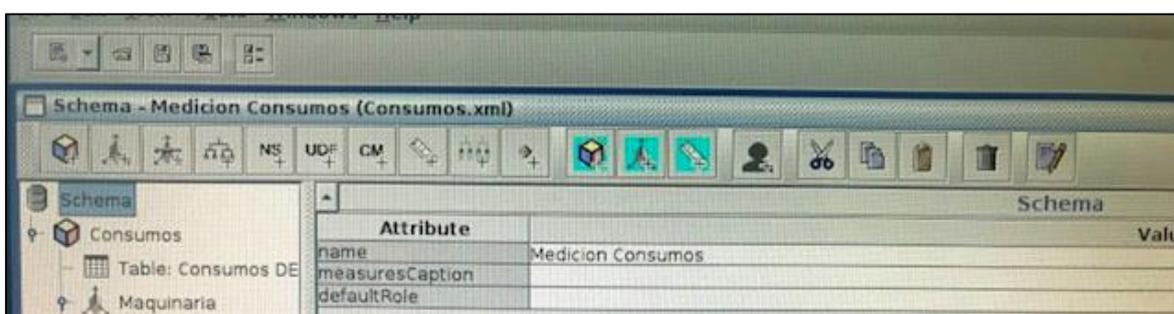


Figura 57. Ventana de características del esquema

Una vez creado el esquema, el siguiente paso consiste en crear un cubo nuevo siguiendo los siguientes pasos: *Marcar esquema* → *Pulsar botón Add Cube* (ver figura 58). A continuación, se marca el cubo nuevo creado y, en la ventana que aparece en el lateral derecho, indicar el nombre que se desee al cubo (ver figura 59). A la hora de rellenar los campos, se debe prestar atención a la parte inferior de la ventana ya que, aquí se verán reflejados los errores que puedan existir.

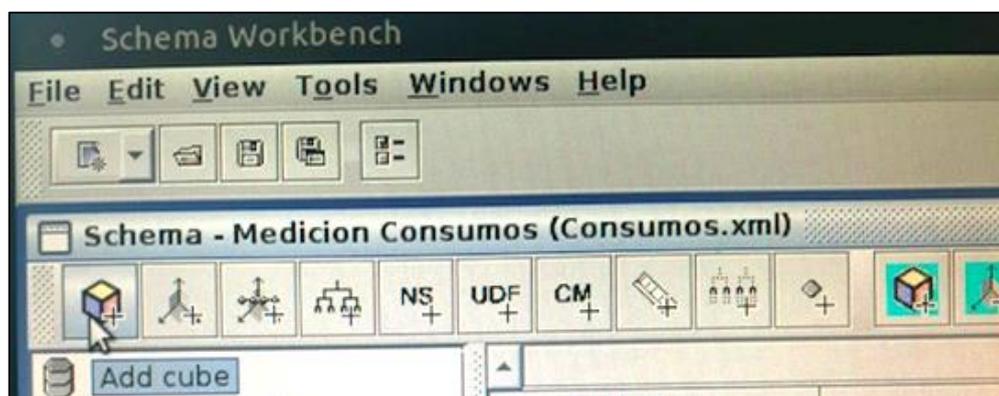


Figura 58. Añadir Cubo nuevo

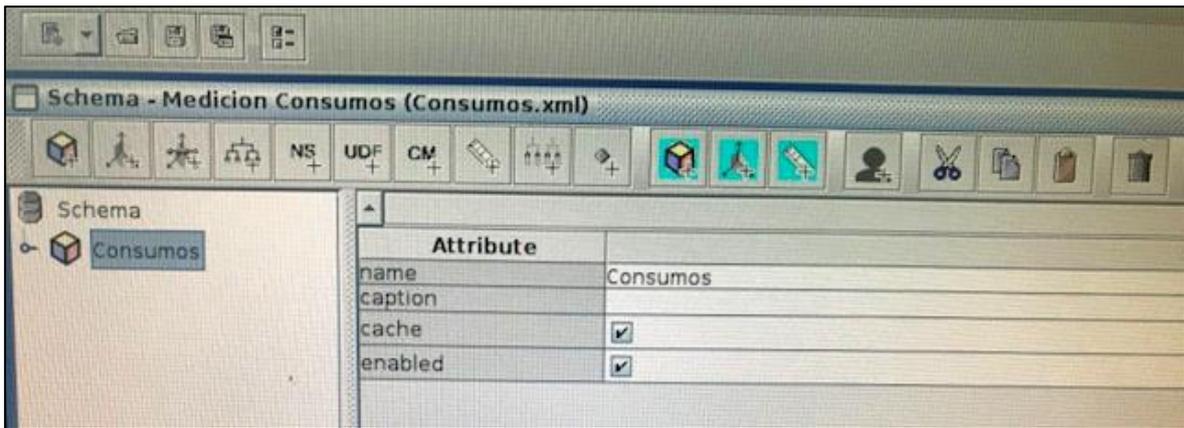


Figura 59. Ventana de características del cubo

El siguiente paso será la definición de la “Tabla de Hechos”, la cual puede visualizarse si se hace clic sobre el pequeño símbolo a la izquierda del cubo creado, en el árbol general. La “Tabla de Hechos” es aquella que contiene la información que se quiere analizar siendo, en este caso, la tabla que contiene las dos máquinas disponibles con todos los datos asociados a ellas y que fue nombrada como “USUARIO”. Para determinar la tabla con la que se va a trabajar, será necesario indicarlo en el campo “name” del panel que aparece a la derecha (ver figura 60), tras seleccionar el elemento del árbol (situado en el panel de la izquierda).

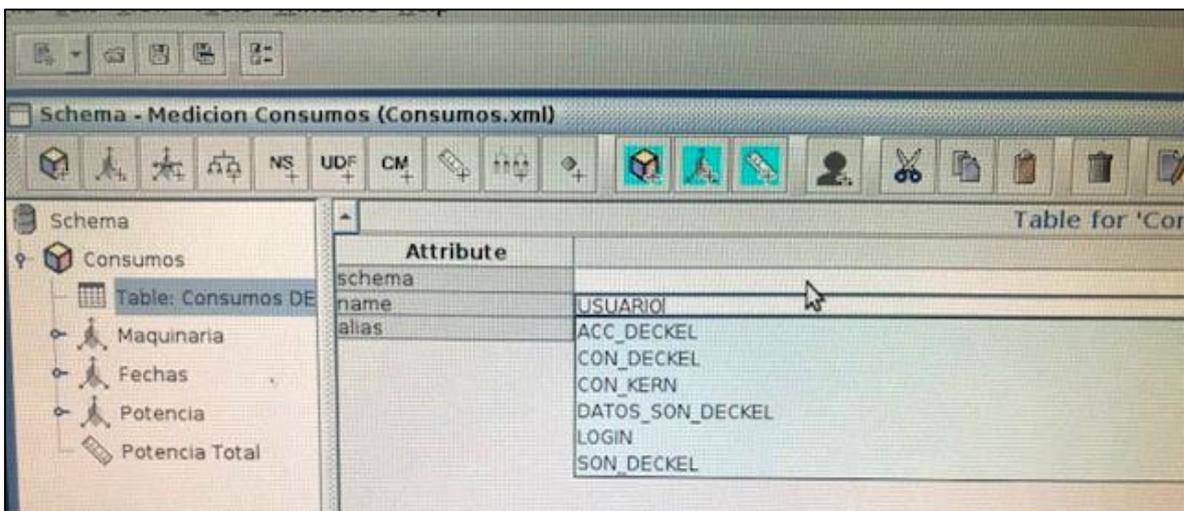


Figura 60. Tabla de Hechos

Tras la definición de la “Tabla de Hechos”, se debe seleccionar el cubo de nuevo y hacer clic derecho sobre él para, posteriormente, elegir la opción *Add Dimension*. A continuación, puede observarse como se ha creado un nuevo elemento en el árbol de organización del panel izquierdo. Es aquí donde se hace referencia a las tablas de dimensiones que se van a utilizar. Al mismo tiempo, este nuevo elemento contiene otros dos denominados “Hierarchy” y “Table”. Se debe destacar que, a la dimensión creada hay que establecerla un nombre y, el

elemento “Table” que contiene adquiere el mismo nombre que la tabla que se le asocie. Para mantener un orden correcto y lo menos dificultoso posible, la configuración ha sido llevada a cabo desde el nivel inferior (elemento “Table”) hasta el principal (la dimensión creada). De esta forma, se selecciona el elemento “Table” y en el campo “name” de la ventana que aparece a la derecha, se le asocia la tabla que se quiera estudiar. Además, también será necesario añadir un elemento adicional llamado “Nivel” en el que se especifica un campo en concreto de la tabla asociada. Para ello, se debe seleccionar el elemento “Hierarchy” y hacer: *clic derecho* → *Add Level*. En esta ocasión, el proceso se ha repetido otras dos veces dando lugar a un total de tres dimensiones denominadas “Maquinaria”, “Fechas” y “Potencia” (ver figura 61). A continuación, se procede a la configuración de cada una de las dimensiones creadas.

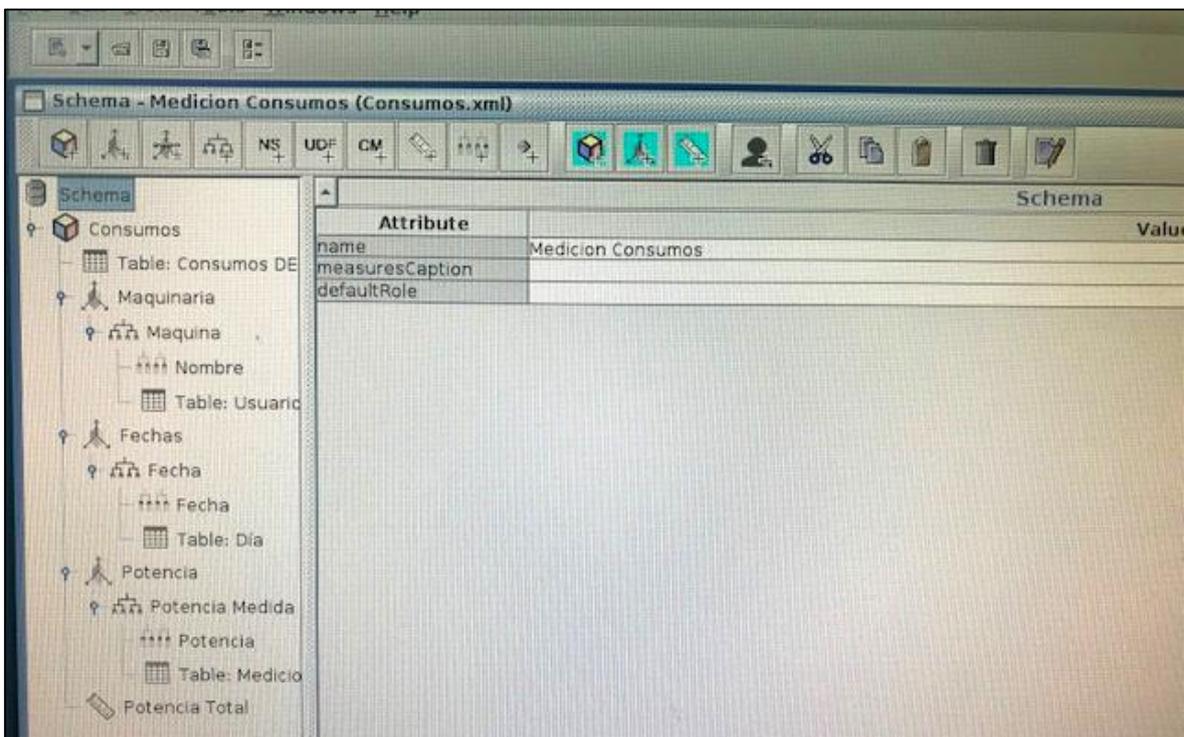


Figura 61. Esquema general de las dimensiones creadas

DIMENSIÓN MÁQUINA

Se comienza configurando el elemento “Table” (ver figura 62), asignándole en el campo “name” la tabla con la que se quiere trabajar. En este caso, se establece la tabla “USUARIO” para poder determinar claramente el centro de mecanizado del que se quiere realizar un análisis. En el campo “alias”, puede establecerse una breve descripción del elemento.

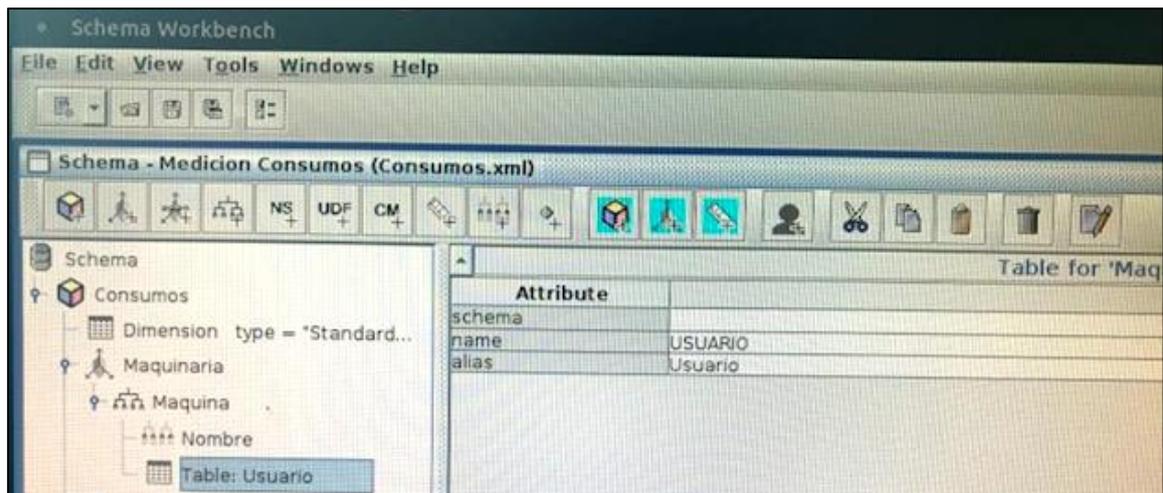


Figura 62. Configuración del elemento "Table" de la dimensión "Maquinaria"

El siguiente paso, es la configuración del elemento “Nivel” (ver figura 63). Tras seleccionarlo, pueden verse varios campos para rellenar, pero, no es necesario completarlos todos e incluso algunos vienen configurados por defecto. En esta ocasión, lo más importante es el establecimiento de un nombre en el campo “name” y de establecer la columna concreta de la tabla asociada con la que se quiere trabajar a través del campo “column”, escogiendo para este caso “Nombre” como identificador y la columna “maquina”.

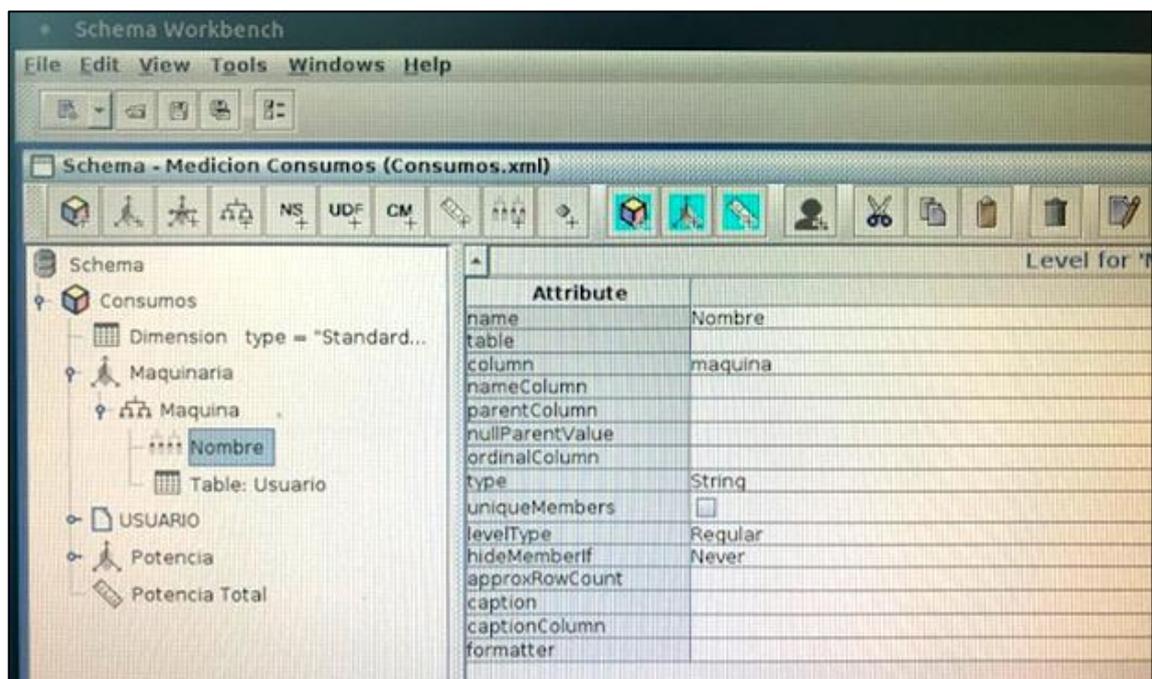


Figura 63. Configuración del elemento "Nivel" de la dimensión "Maquinaria"

A continuación, hay que subir un escalón más para realizar la configuración del elemento “Hierarchy” que, en esta ocasión, ha sido nombrado como “Maquina” a través del campo “name” (ver figura 64). Al igual que en el caso anterior, aquí también se encontrarán varios

campos, pero solo es necesario rellenar tres de ellos: “name”, “allMemberName” y “primaryKey”. Como el primero de ellos ya ha sido rellenado, se pasa al siguiente de forma que, en este campo debe establecerse el mismo nombre que se introdujo para la tabla asociada, es decir, “Usuario”. Por último, en el campo “primaryKey” se debe situar el nombre de la columna principal que se quiere tener como referencia. Como puede observarse, el elemento “Hierarchy” permite establecer la relación entre los dos elementos inferiores que ya han sido configurados independientemente.

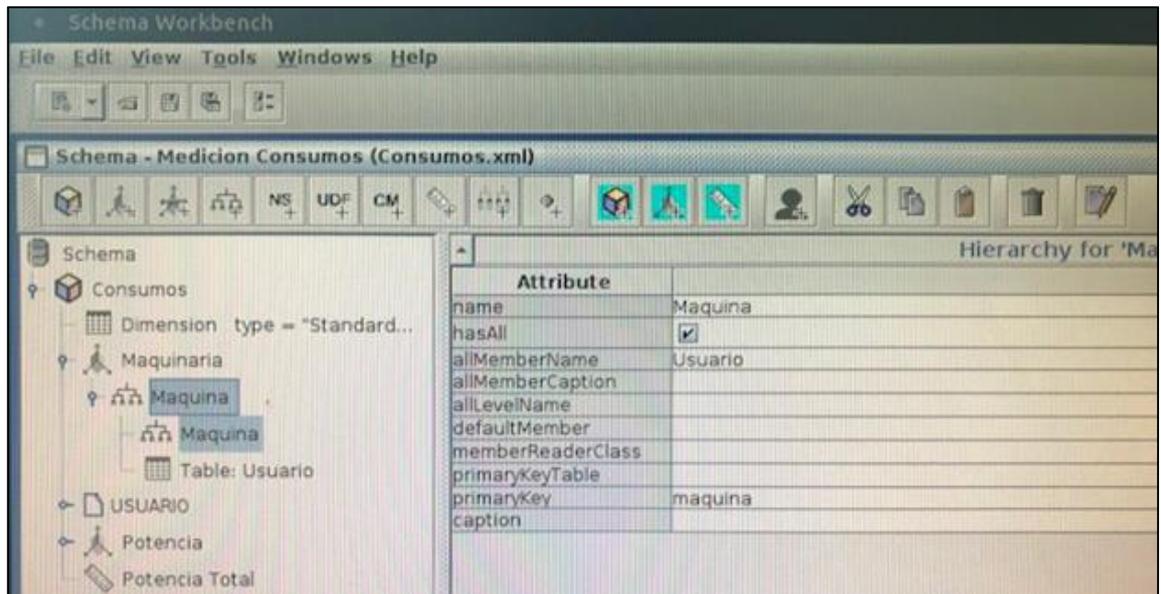


Figura 64. Configuración del elemento "Hierarchy" de la dimensión "Maquinaria"

Finalmente, se pasa a la configuración del elemento principal, conocido como “Dimensión” (ver figura 65). Al igual que con todos y cada uno de los elementos que componen el cubo, aquí también se encontrará un panel en el lateral derecho con diferentes campos a rellenar. En el campo “name” se establece un nombre para la dimensión, en este caso, “Maquina”. En el campo “foreignKey”, se debe establecer la columna de la tabla asociada que ya fue configurada en niveles inferiores y que corresponde a la columna “maquina”. El resto de campos vienen configurados por defecto.

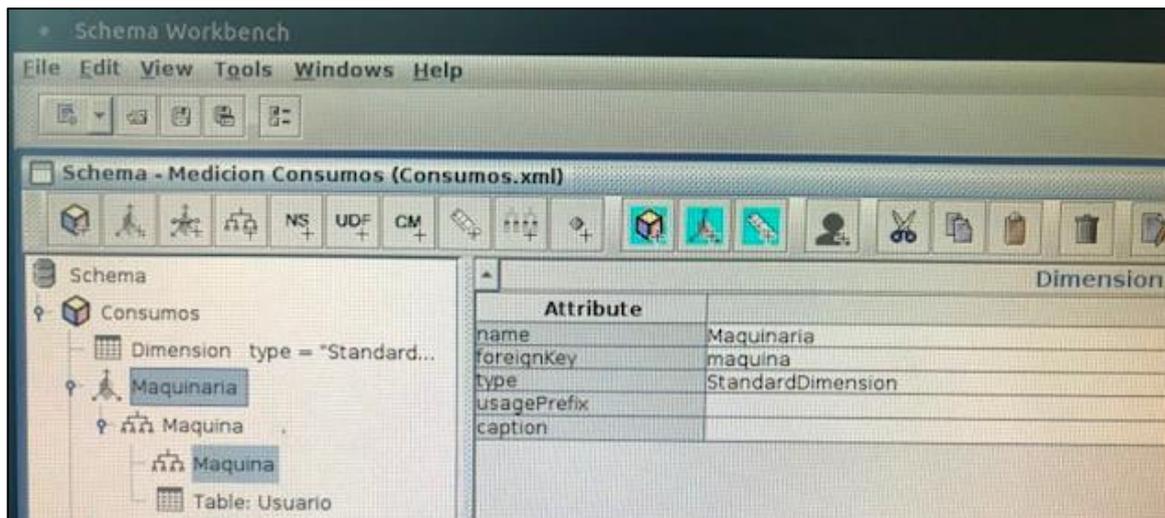


Figura 65. Configuración de la dimensión "Maquinaria"

DIMENSIÓN FECHAS

Se comienza configurando el elemento "Table", asignándole en el campo "name" la tabla con la que se quiere trabajar (ver figura 66). En este caso, se establece la tabla "CON_DECKEL" para poder determinar los consumos de potencia registrados por el centro de mecanizado "Deckel Maho". En el campo "alias", puede establecerse una breve descripción del elemento.

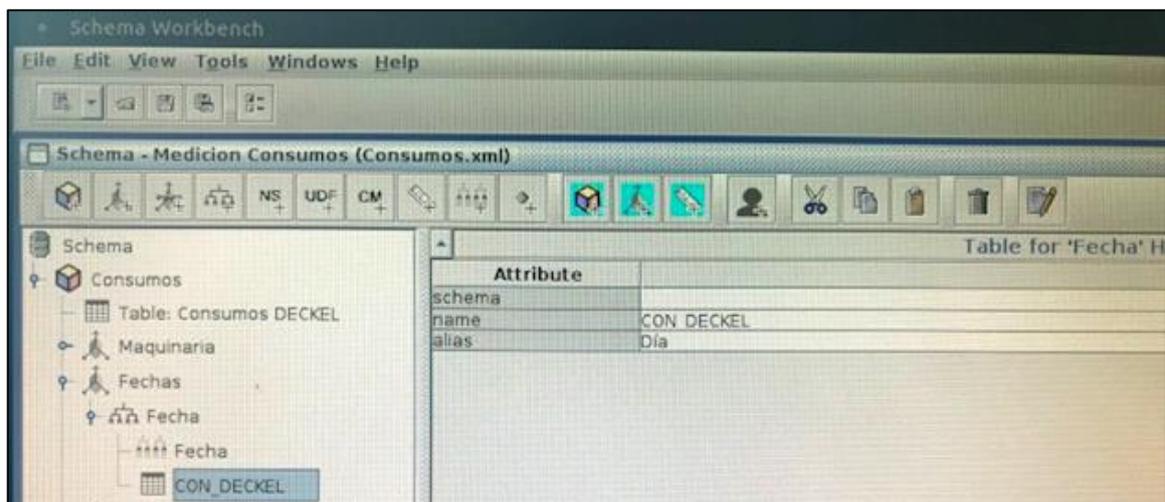


Figura 66. Configuración del elemento "Table" de la dimensión "Fechas"

El siguiente paso, es la configuración del elemento "Nivel" (ver figura 67). Tras seleccionarlo, pueden verse varios campos para rellenar, pero, no es necesario completarlos todos e incluso algunos viene configurados por defecto. En esta ocasión, lo más importante es el establecimiento de un nombre en el campo "name" y de establecer la columna concreta de la tabla asociada con la que se quiere trabajar a través del campo "column", escogiendo para este caso la columna "fecha".

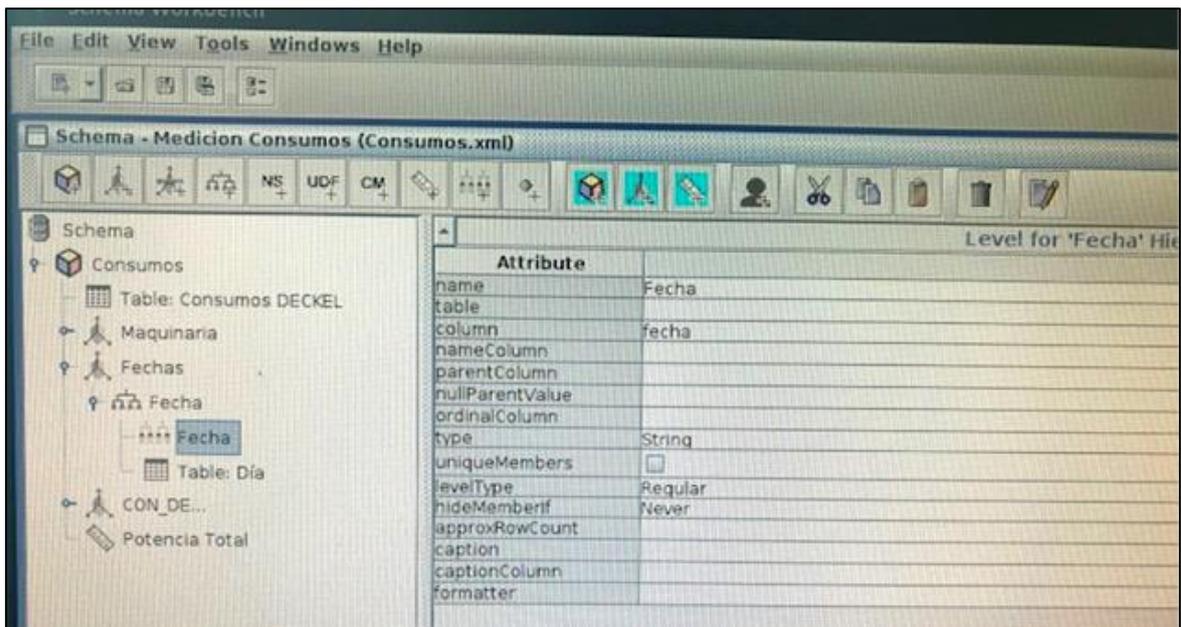


Figura 67. Configuración del elemento "Nivel" de la dimensión "Fechas"

A continuación, hay que subir un escalón más para realizar la configuración del elemento “Hierarchy” (ver figura 68) que, en esta ocasión, ha sido nombrado como “Fecha” a través del campo “name”. Al igual que en el caso anterior, aquí también se encontrarán varios campos, pero solo es necesario rellenar tres de ellos: “name”, “allMemberName” y “primaryKey”. Como el primero de ellos ya ha sido rellenado, se pasa al siguiente de forma que, en este campo debe establecerse el mismo nombre que se introdujo para la tabla asociada, es decir, “Día”. Por último, en el campo “primaryKey” se debe situar el nombre de la columna principal que se quiere tener como referencia. Como puede observarse, el elemento “Hierarchy” permite establecer la relación entre los dos elementos inferiores que ya han sido configurados independientemente.

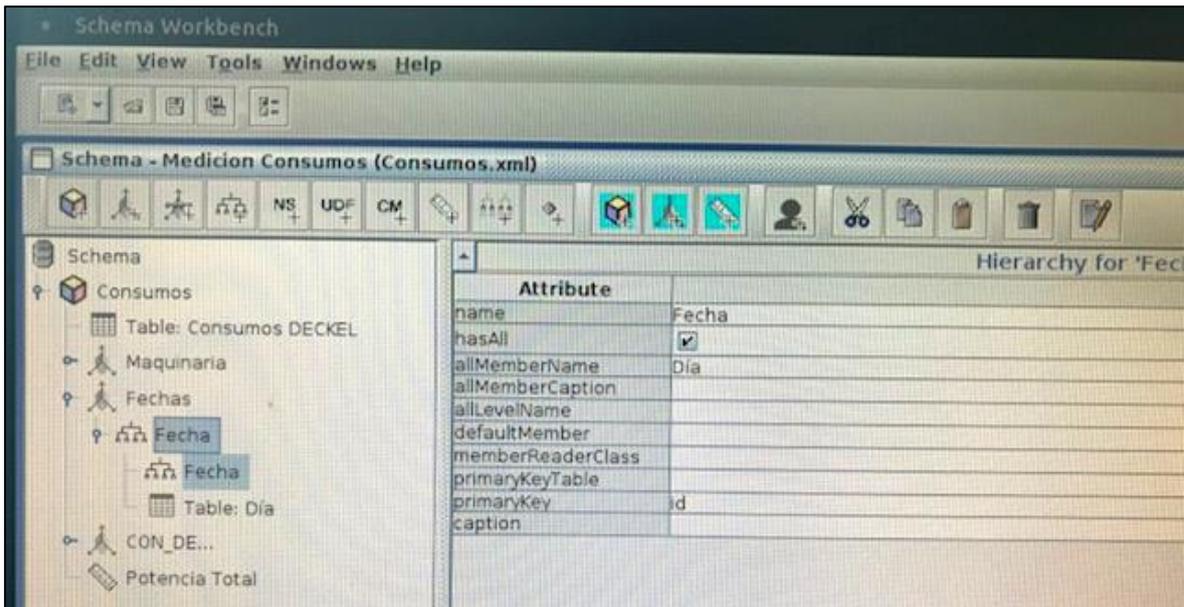


Figura 68. Configuración del elemento "Hierarchy" de la dimensión "Fechas"

Finalmente, se pasa a la configuración del elemento principal, conocido como “Dimensión” (ver figura 69). Al igual que con todos y cada uno de los elementos que componen el cubo, aquí también se encontrará un panel en el lateral derecho con diferentes campos a rellenar. En el campo “name” se establece un nombre para la dimensión, en este caso, “Fechas”. En el campo “foreignKey”, se debe establecer la columna de la tabla asociada que ya fue configurada en niveles inferiores y que corresponde a la columna “fecha”. El resto de campos vienen configurados por defecto.

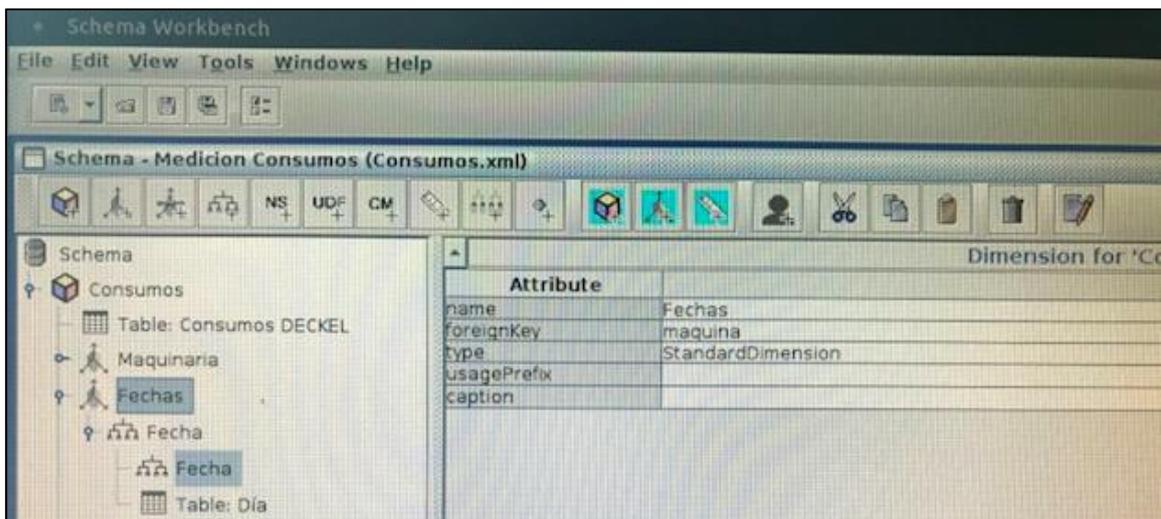


Figura 69. Configuración de la dimensión "Fechas"

DIMENSIÓN POTENCIA

Se comienza configurando el elemento “Table”, asignándole en el campo “name” la tabla con la que se quiere trabajar (ver figura 70). En este caso, también se establece la tabla

“CON_DECKEL” para poder determinar los consumos de potencia registrados por el centro de mecanizado “Deckel Maho”. En el campo “alias”, puede establecerse una breve descripción del elemento.

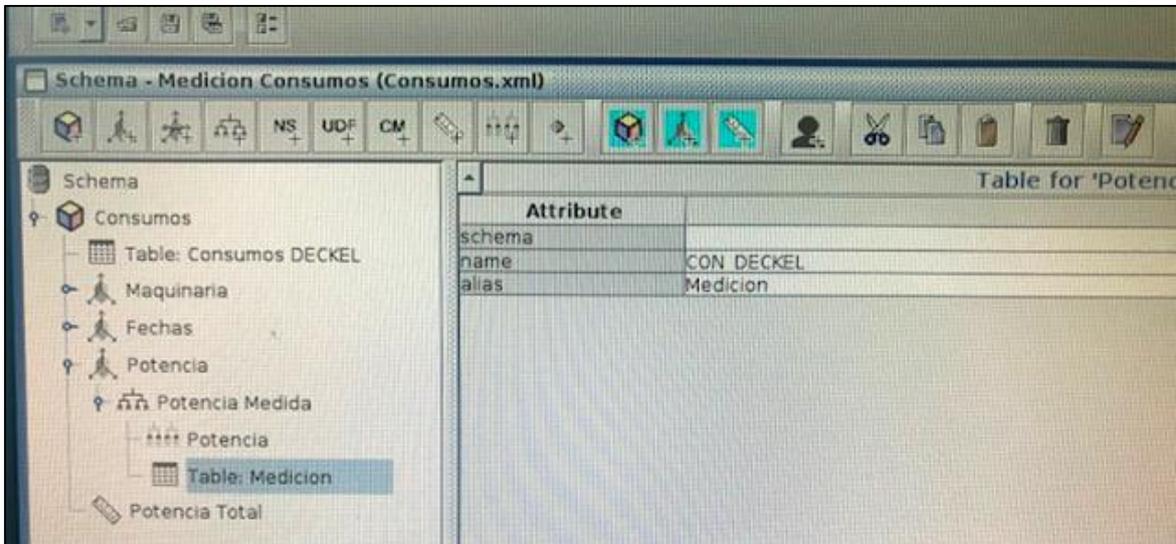


Figura 70. Configuración del elemento "Table" de la dimensión "Potencia"

El siguiente paso, es la configuración del elemento “Nivel” (ver figura 71). Tras seleccionarlo, pueden verse varios campos para rellenar, pero no es necesario completarlos todos e incluso algunos vienen configurados por defecto. En esta ocasión, lo más importante es el establecimiento de un nombre en el campo “name”, “Potencia” para esta configuración, y de establecer la columna concreta de la tabla asociada con la que se quiere trabajar a través del campo “column”, escogiendo para este caso la columna “canal4”.

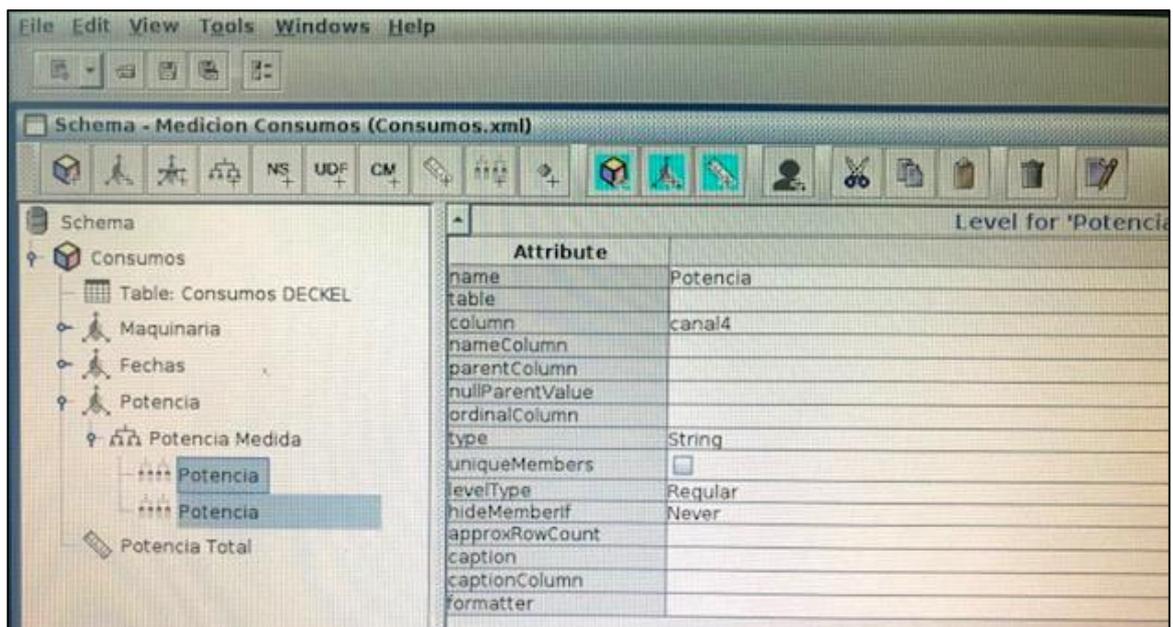


Figura 71. Configuración del elemento "Nivel" de la dimensión "Potencia"

A continuación, hay que subir un escalón más para realizar la configuración del elemento “Hierarchy” (ver figura 72) que, en esta ocasión, ha sido nombrado como “Potencia Medida” a través del campo “name”. Al igual que en el caso anterior, aquí también se encontrarán varios campos, pero solo es necesario rellenar tres de ellos: “name”, “allMemberName” y “primaryKey”. Como el primero de ellos ya ha sido rellenado, se pasa al siguiente de forma que, en este campo debe establecerse el mismo nombre que se introdujo para la tabla asociada, es decir, “Medicion”. Por último, en el campo “primaryKey” se debe situar el nombre de la columna principal que se quiere tener como referencia. Como puede observarse, el elemento “Hierarchy” permite establecer la relación entre los dos elementos inferiores que ya han sido configurados independientemente.

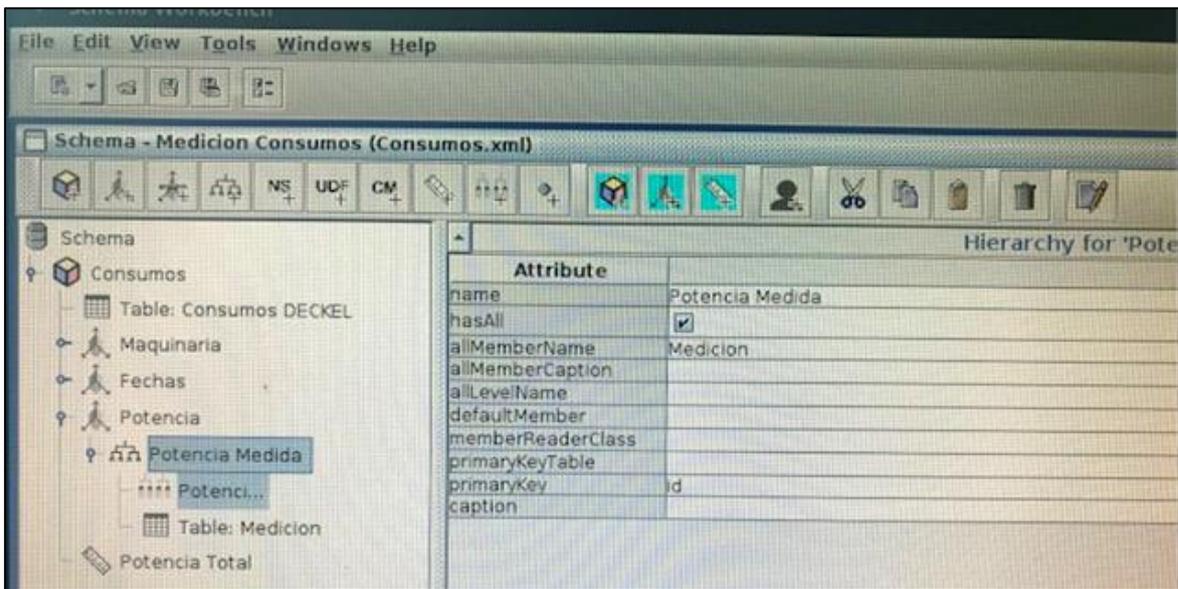


Figura 72. Configuración del elemento "Hierarchy" de la dimensión "Potencia"

Finalmente, se pasa a la configuración del elemento principal, conocido como “Dimensión” (ver figura 73). Al igual que con todos y cada uno de los elementos que componen el cubo, aquí también se encontrará un panel en el lateral derecho con diferentes campos a rellenar. En el campo “name” se establece un nombre para la dimensión, en este caso, “Potencia”. En el campo “foreignKey”, se debe establecer la columna de la tabla asociada que ya fue configurada en niveles inferiores y que corresponde a la columna “maquina”. El resto de campos vienen configurados por defecto.

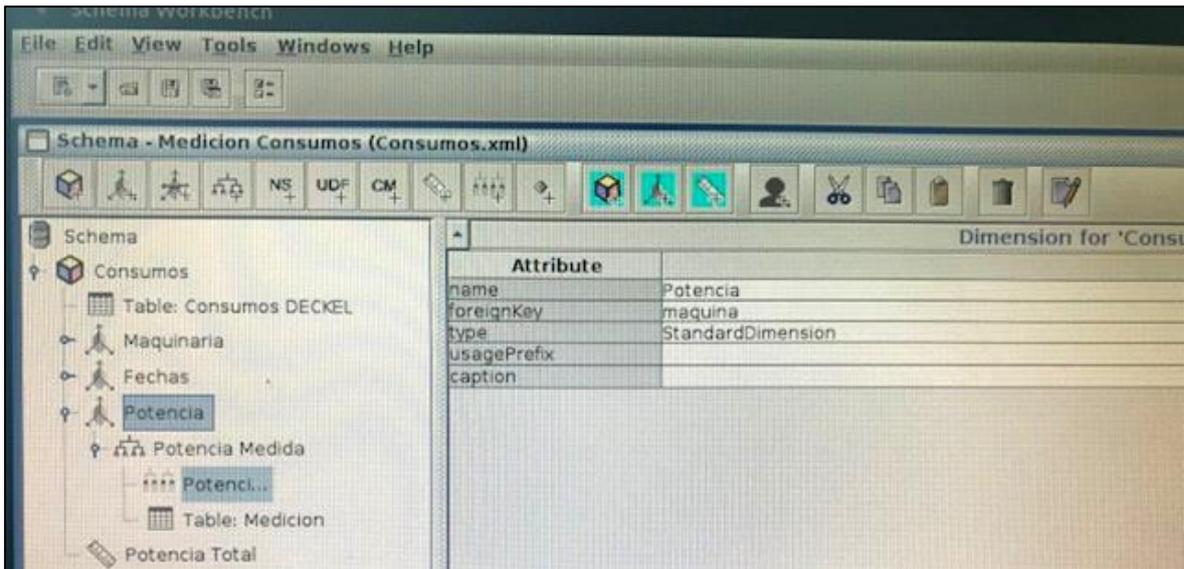


Figura 73. Configuración de la dimensión "Potencia"

Tras el desarrollo de las diferentes dimensiones, el siguiente paso consiste en la incorporación de un indicador que permita realizar el cálculo de las potencias totales consumidas (ver figura 74). Para añadir un indicador hay que seleccionar el cubo creado y seguir los siguientes pasos: *clic derecho* → *Add Measure*.

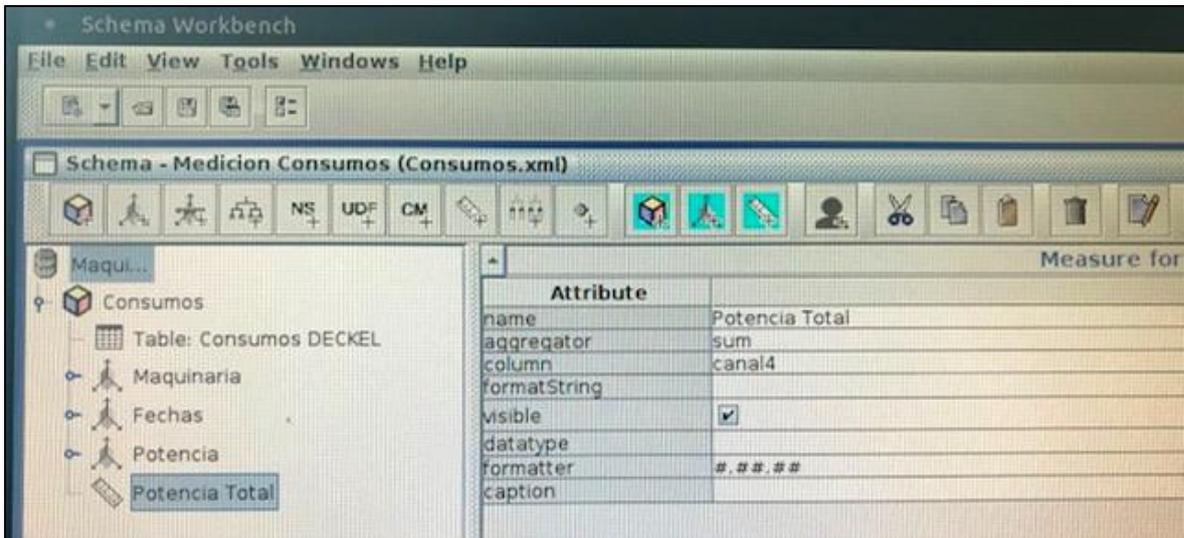


Figura 74. Configuración del indicador

Como puede verse en la figura 74, se disponen de una serie de campos en la ventana que se despliega en el lateral derecho de la pantalla de los cuales, serán necesarios rellenar los tres primeros y el penúltimo. En el primer campo, denominado “name”, se especifica el nombre del indicador; en el segundo, denominado “aggregator”, el tipo de operación que realizará el indicador, siendo la suma para esta ocasión; en el tercero, denominado “column”, se indica la columna de la tabla “CON_DECKEL” sobre la que se quiere trabajar y, finalmente; en el

penúltimo campo, denominado “formatter”, se indica el formato de la operación. El resto de campos están configurados por defecto.

Una vez diseñado y configurado el cubo, lo siguiente que debe hacerse es guardarlo. Para ello, basta ir a *File* → *Save as* e indicar el nombre del archivo y el lugar de guardado.

Finalmente, para poder visualizar el cubo a través de la herramienta “Pentaho” es necesario publicarlo en el servidor de la herramienta mencionada (ver figura 75). Para realizar la publicación ir a *File* → *Publish* y, aparecerá una ventana donde será necesario introducir la URL del servidor, la contraseña de publicación y las credenciales de acceso de “Pentaho” (ver figura 76).

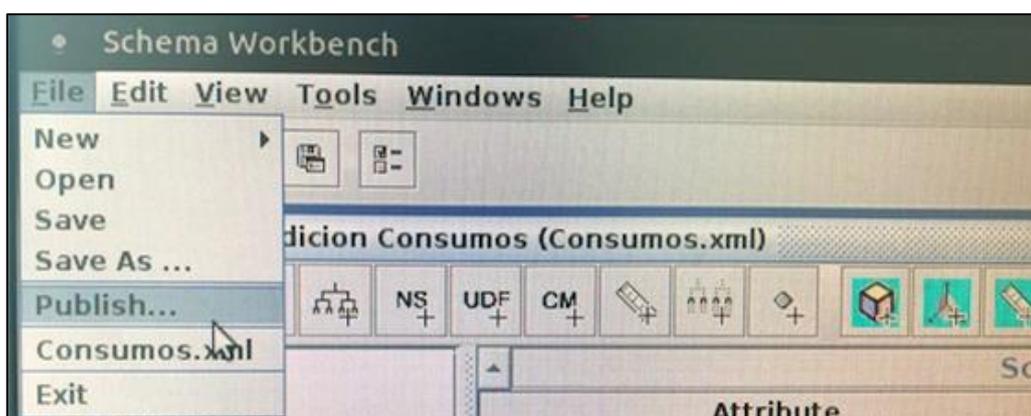


Figura 75. Opción para publicar



Figura 76. Ventana para realizar la publicación

8 Resultados

Tras todo el trabajo realizado, en el que se han desarrollado tres dispositivos electrónicos diferentes para la medición de consumo eléctrico, la medición de aceleraciones y la captación del sonido durante el proceso de fabricación de un centro de mecanizado, se dispone a la presentación mediante gráficas de los resultados obtenidos.

En primer lugar, se procede a mostrar la corriente eléctrica de cada uno de los canales de medición de manera independiente (ver figuras 77, 78 y 79), así como la potencia consumida en total por los tres canales (ver figura 80). Debido a que los datos de los que se dispone son demasiados, se van a presentar los resultados relativos a la producción de un día completo correspondiendo, en este caso, al día 25 de enero de 2018.

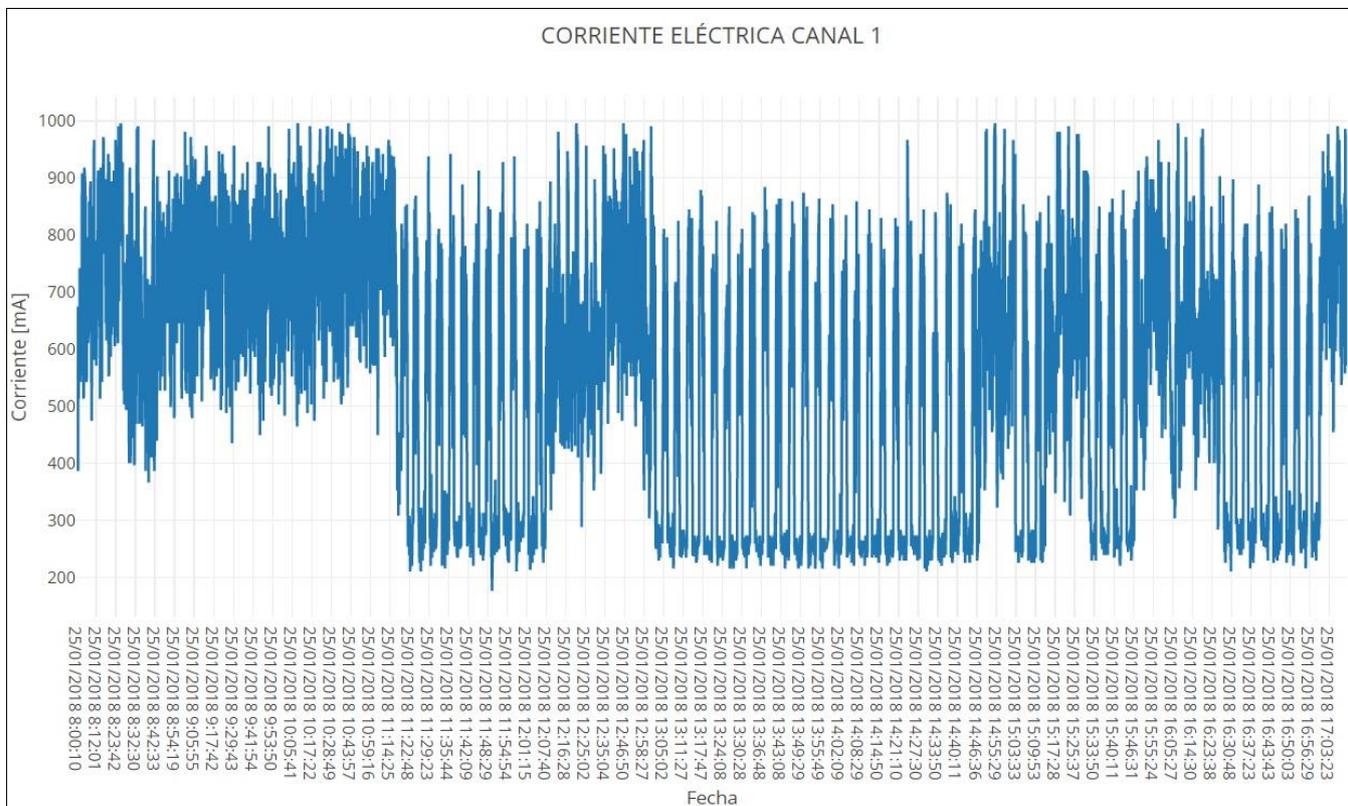


Figura 77. Corriente eléctrica medida en el canal 1

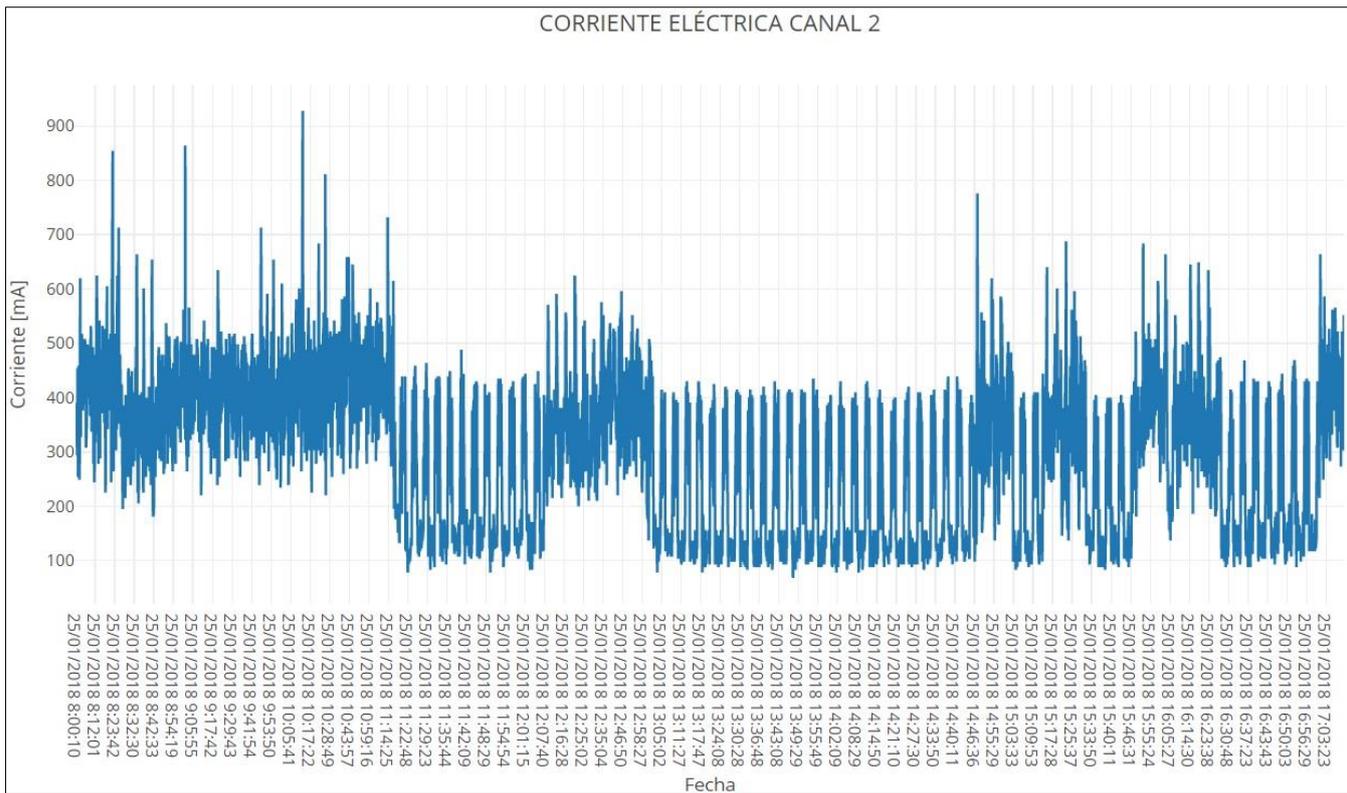


Figura 78. Corriente eléctrica medida en el canal 2

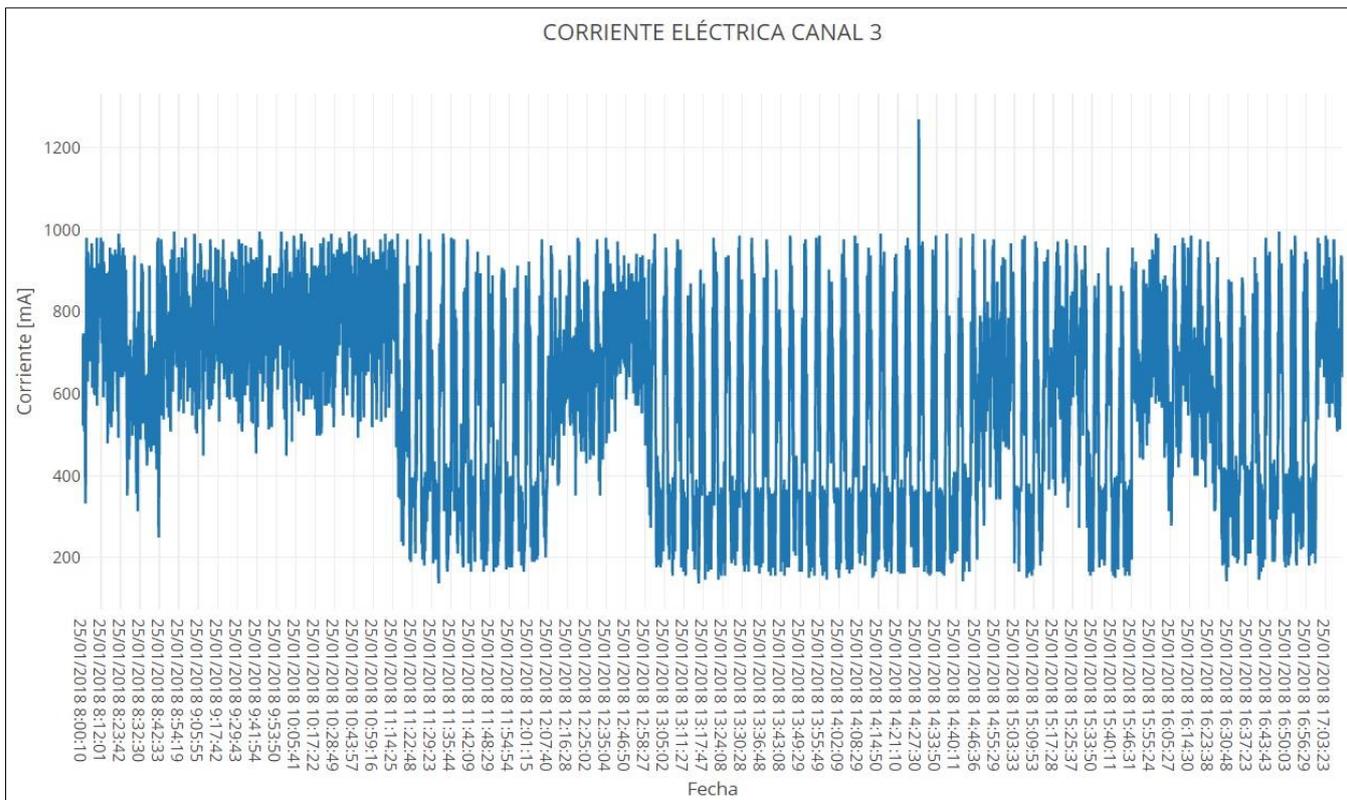


Figura 79. Corriente eléctrica medida en el canal 3

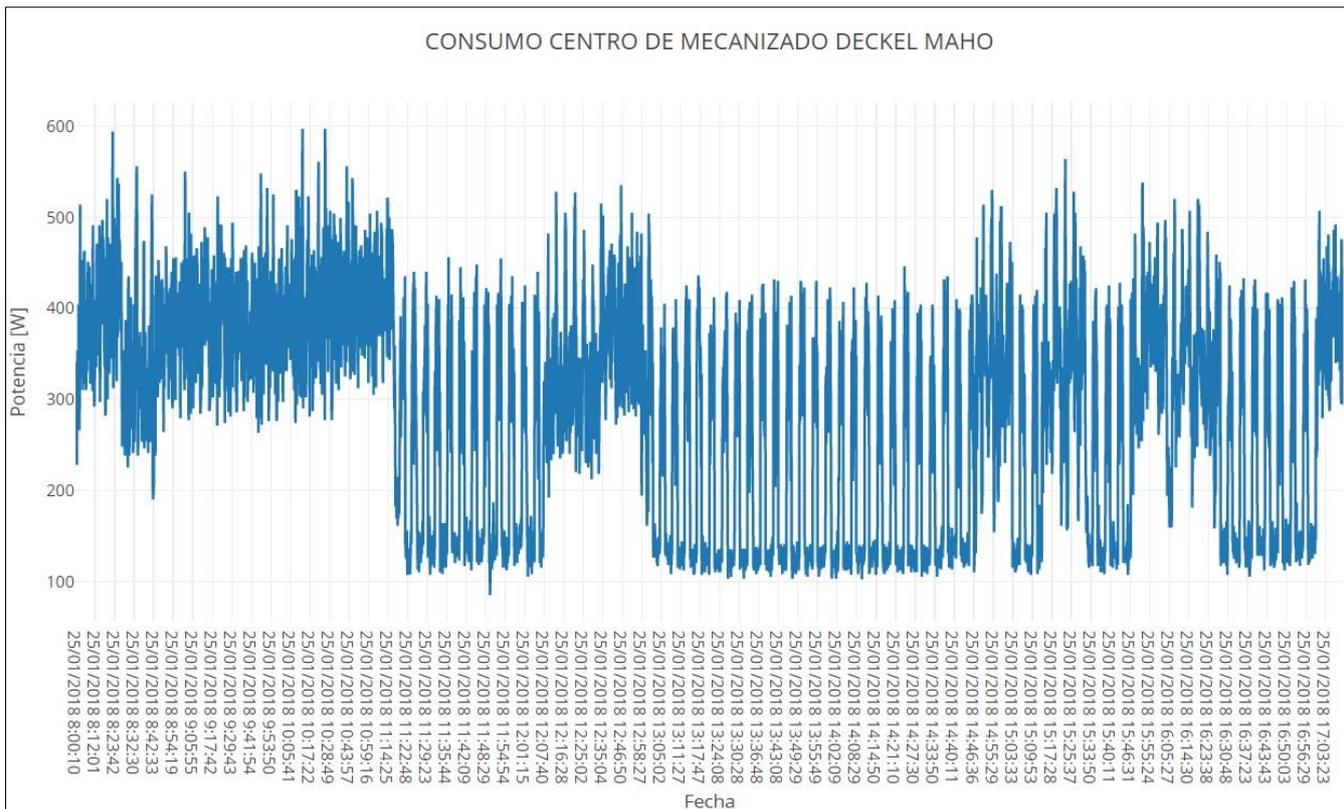


Figura 80. Potencia total consumida

Como puede verse en las figuras de medición de corriente de cada uno de los canales, la medición es aproximada a los 1000 mA (1 A), siendo considerada esta medida como la estándar o normal y todas aquellas superiores pueden suponer indicios de algún tipo de anomalía en el proceso de fabricación. Sin embargo, no siempre este tipo de situaciones serán indicativos de anomalías, sino que pueden deberse a situaciones ajenas a problemas como un pico de corriente en la red eléctrica o temas similares. De esta forma, el parámetro que más información fiable aporta al sistema es la potencia total consumida ya que, es aquí donde se recoge la información relativa a los tres canales de medición estableciéndose como valor máximo tolerable 600 W de potencia consumida. Así, los registros superiores a esta potencia serán considerados como problemas que afectan a la integridad de la herramienta de trabajo del centro de mecanizado y deberá notificarse al usuario dicha situación mediante la interfaz que se propone diseñar.

En segundo lugar, se presentan los resultados del sistema de medición de aceleraciones (ver figura 81), pero en este caso, en vez de representar todo un día de trabajo se hará de una hora de trabajo. El motivo de presentar únicamente este período de tiempo es que, al ser un sistema alimentado por baterías no puede estar funcionando continuamente, sino que necesita ser extraído de la máquina y cargar dichas baterías. Además, lo que interesa

realmente es ver el comportamiento del dispositivo en puntos clave del proceso de fabricación. En la figura 81, se muestra la gráfica del funcionamiento del sistema durante una hora de trabajo del día 19 de diciembre de 2017.

Antes de comenzar el análisis de los resultados de este sistema, hay que especificar que el acelerómetro tiene dos configuraciones principales: una de medición continua y otra de bajo consumo (registra datos cuando detecta desplazamientos superiores a cierta medida). En esta ocasión se ha empleado la primera de las dos configuraciones para evitar perderse algún dato importante que pudiera implicar la no detección de una situación de riesgo. Mediante esta configuración, los datos registrados nunca van a ser cero porque el acelerómetro, al registrar todos los desplazamientos a los que está sometido, en esta ocasión se encuentra posicionado verticalmente al ser introducido en el portaherramientas y, por consiguiente, sufre una serie de desplazamientos que, como puede verse en la figura 81 son constantes y no es hasta cuando se producen desplazamientos más grandes cuando se pueden observar dichas variaciones, lo que indica el proceso de fabricación.

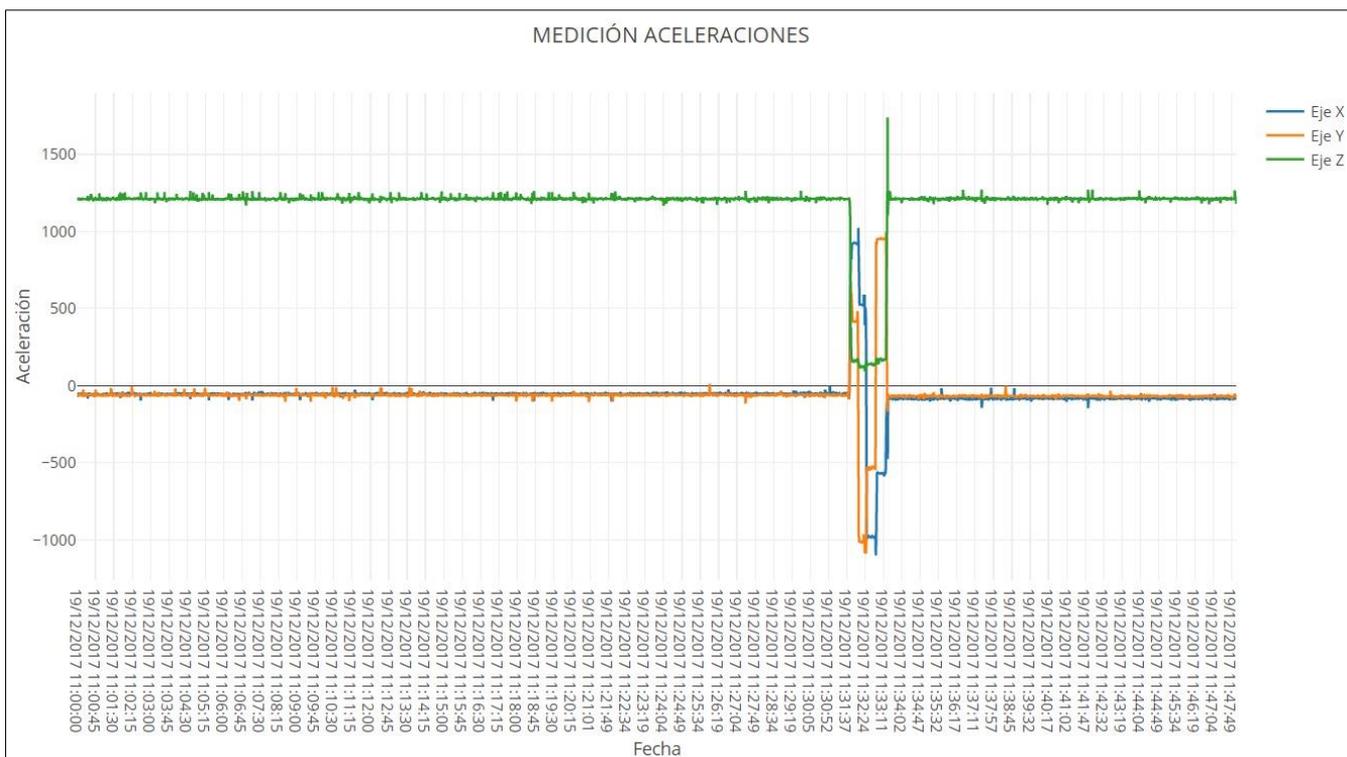


Figura 81. Medición de aceleraciones en μg

Como puede verse en la figura 81, se muestran en colores diferentes las medidas registradas para cada uno de los ejes (eje X en azul, eje Y en naranja y eje Z en verde). El hecho de que se vean valores negativos se debe a que el acelerómetro registra datos en función de la dirección en la que surgen los desplazamientos, funcionando exactamente igual que los ejes

cartesianos X, Y y Z. En esta ocasión, hay que fijarse que los valores de fuerza registrados no sean excesivamente altos como, por ejemplo, obtener valores de 2000 μg o más ya que, sería en esos momentos en los que se están produciendo vibraciones excesivas que podrían dañar la herramienta de trabajo. Al igual que con el sistema de medición de consumo eléctrico, también se propone el aviso al usuario encargado de manipular la máquina a través de la interfaz propuesta.

Finalmente, se procede a mostrar los resultados obtenidos con el sistema de medición de señales acústicas. Para ello, se interpretan varias de las señales conseguidas en diferentes procesos de fabricación los cuales, se configuran bajo parámetros como la velocidad de corte, la profundidad del corte y la herramienta para desempeñar el proceso.

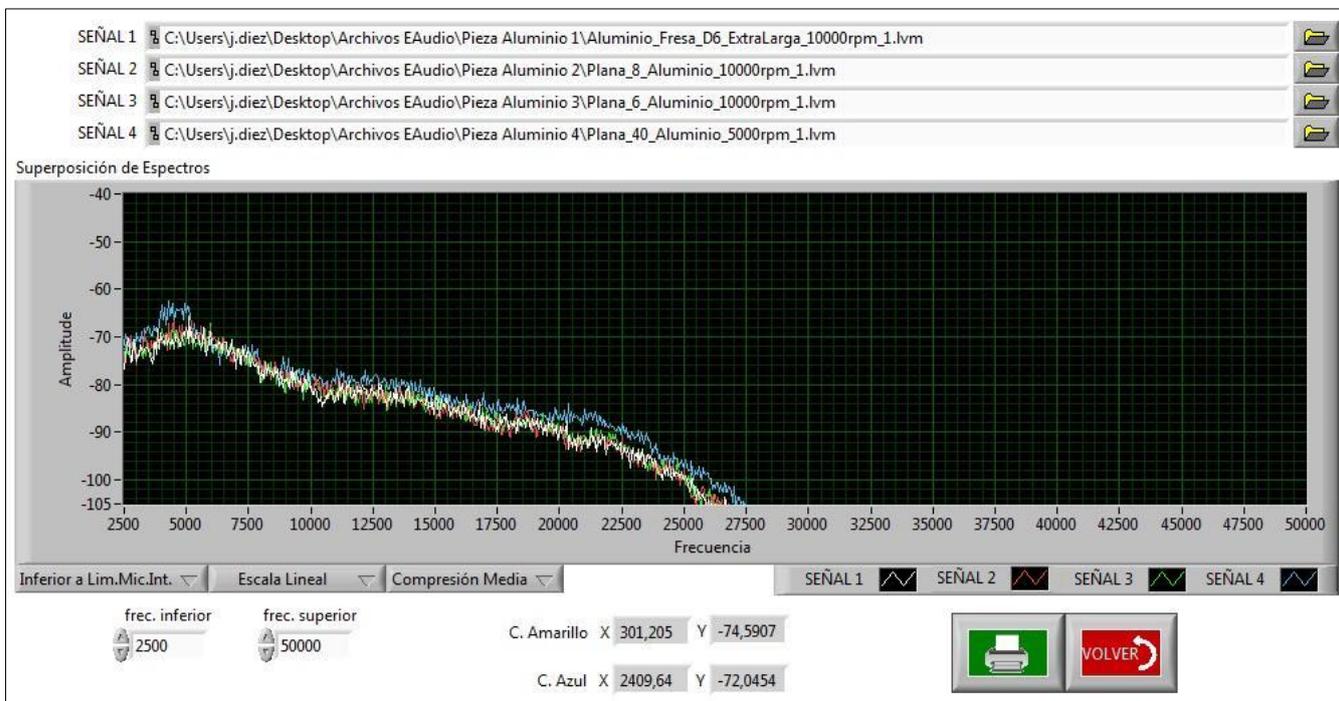


Figura 82. Medición de señales acústicas 1

En la figura 82, se muestra la comparación entre cuatro piezas de aluminio a 1000 y 5000 revoluciones por minuto con una fresa de diámetro 6 extra larga para la primera, una plana de diámetro 8 para la segunda, una plana de diámetro 6 para la tercera y una plana de diámetro 40 para la cuarta. Como puede apreciarse cuanto mayor es la velocidad de corte, mayor es la amplitud registrada. A la hora de poder establecer una cota de amplitud máxima para considerar tal como riesgo para la herramienta de fabricación, deben conocerse de forma estimada los límites entre los que debe operar la máquina. La parte de mayor estudio y en la que más se debe centrar el sistema es en torno a los 5 kHz.



Figura 83. Medición de señales acústicas 2

En la figura 83, se muestra la comparativa entre una pieza de aluminio y otra de resina empleando una fresa de diámetro 6 para la primera y una plana de diámetro 2 para la segunda y, a una velocidad de corte en ambos casos de 10000 revoluciones por minuto. Como puede verse, a 5 kHz no se presenta ningún tipo de pico que sobresalga por encima de los demás estados.

9 Ampliaciones futuras

Todo sistema puede ser mejorado con el tiempo mediante la implementación de nuevos equipos o simplemente optimizando los ya existentes. En esta ocasión, lo que se plantea que podrían ser buenas mejoras para el sistema desarrollado en todo su conjunto es, la adición de nuevos sistemas de medición de parámetros como la presión a la que está sometida una herramienta durante el proceso de fabricación y la temperatura generada durante dicho proceso. Además, se propone el desarrollo de una interfaz gráfica “Hombre-Máquina” (ver figura 84) en la que se pudieran visualizar aquellas variables supuestas como críticas para el sistema y, en caso de que se superen ciertos rangos, alertar al usuario mediante señales de emergencia tales como carteles textuales de que se está produciendo alguna anomalía o algún tipo de iluminación en rojo. También sería de gran ayuda la generación de informes tras una situación de riesgo y posibilidad de sugerir posibles actuaciones para solventar el problema.

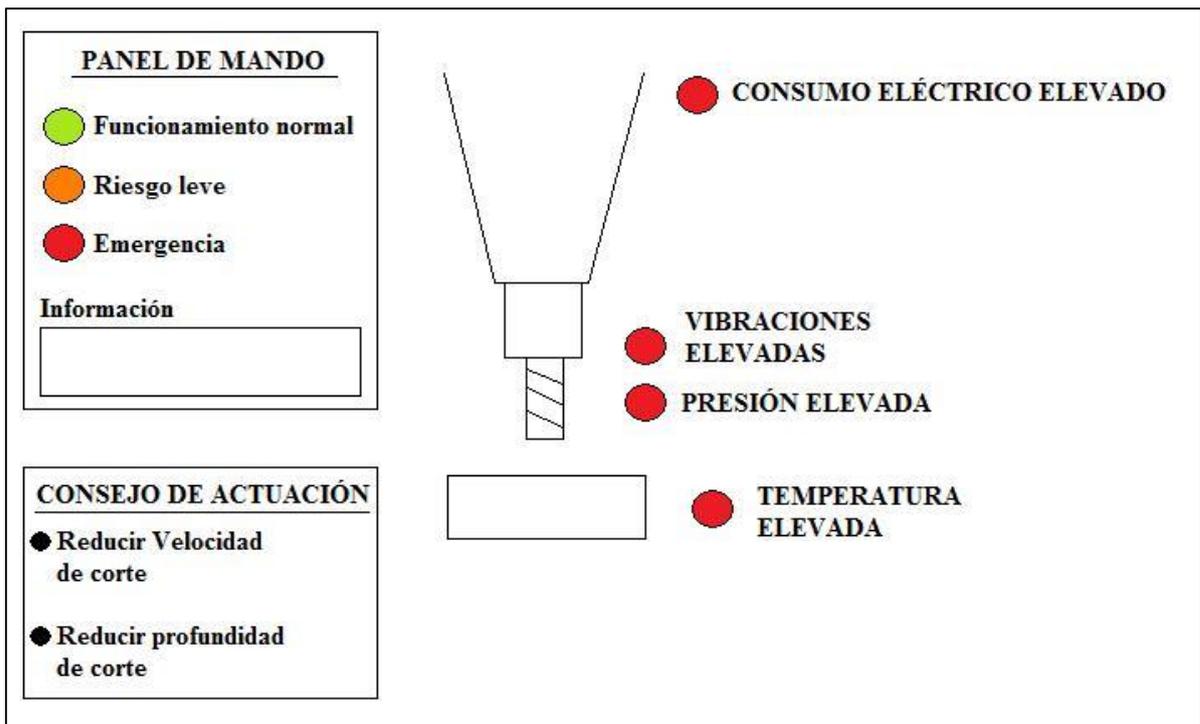


Figura 84. Propuesta de Interfaz

Otra necesidad que se debe implementar es el establecimiento de una conexión remota para poder introducir al sistema desarrollado de forma completa en el sector de la “Industria 4.0”. Existen varios protocolos y mecanismos (Velasco, 2017) para implementar una conexión remota como son:

- SSH
- VNC
- RDP

El protocolo **SSH** (colaboradores de Wikipedia, 2018) permite la conexión remota a un terminal del sistema operativo Linux, pero únicamente en modo texto por lo que se tendrán que tener los conocimientos necesarios para manejarlo. Este protocolo trabaja de forma similar a *telnet*, con la diferencia de que SSH emplea técnicas de cifrado, haciendo que la información viaje de forma no legible y evitar que terceras personas puedan tener acceso a ella. A pesar de todo, este tipo de protocolo puede ser atacado mediante ataques de *replay*. La conexión puede realizarse desde otra terminal Linux o a través de herramientas como “Putty” si se está trabajando con Windows.

La herramienta **VNC** (REALVNC, 2018) permite la conexión remota al escritorio del sistema operativo Linux que se desee. A diferencia del anterior, esta herramienta emplea una interfaz gráfica por lo que su uso es más sencillo y permite navegar a través del escritorio

como en cualquier otro sistema operativo. Como desventaja, puede surgir el problema de una lenta tasa de refresco lo que provoca problemas a la hora de ver vídeos, pero en este caso al no ser un elemento necesario, no supone ningún inconveniente grave.

El protocolo propietario de Microsoft **RDP** permite la conexión remota a cualquier otro ordenador con sistema Windows sin depender de ninguna otra aplicación. Si se desea establecer una conexión remota a un sistema operativo Linux, se deberá instalar la versión no oficial de este protocolo.

Tras conocer cada una de las opciones, en esta ocasión la opción más viable y sencilla es emplear la herramienta software **VNC** ya que es más sencilla que las demás y no requiere ningún conocimiento exhaustivo sobre un sistema operativo. Además, permitirá el establecimiento de una conexión remota gráfica con un asistente de configuración muy fácil de seguir, evitándose así la configuración por línea de comandos. A continuación, se explica el proceso a seguir para configurar una conexión remota a través de una red local y fuera de la misma, es decir, desde cualquier punto con acceso a Internet.

Lo primero que se debe hacer es instalar un servidor en el ordenador que se desea controlar y habilitar un cliente **VNC** en el ordenador que controlará remotamente al otro. En esta ocasión, el servidor se instalaría en la Raspberry Pi principal y el cliente VNC en el ordenador desde el que se desea realizar el control remoto. Se aconseja emplear la herramienta “*TightVNC*” como servidor ya que está optimizado para dispositivos y redes lentas, además de ser de distribución libre. Tras descargar e instalar el programa, se cargará el servidor **VNC**. A continuación, aparecerá una ventana donde se establece la configuración del servidor en la que, se debe establecer una contraseña para la conexión y donde puede observarse que por defecto la conexión será establecida a través del puerto principal 5900 o el puerto HTTP 5800 (a través de la web). Se aconseja mantener la configuración por defecto a excepción de la contraseña. Una vez finalizada, se debe pulsar la opción “Apply” y, posteriormente la opción “OK”. A partir de este momento, ya puede establecerse cualquier conexión remota desde el ordenador que se desee. En los siguientes pasos se expone el procedimiento para tal fin.

Para establecer una conexión remota se debe abrir el cliente **VNC** descargado y, en la pantalla que aparecerá acto seguido, introducir la dirección IP del dispositivo a controlar (Raspberry Pi en este caso) y que tiene el servidor instalado y en funcionamiento. Tras introducir la dirección IP, se debe pulsar la opción “OK” y aparecerá una nueva pantalla

solicitando la contraseña configurada en el servidor. Una vez introducida la contraseña válida, lo siguiente que se verá es la pantalla del ordenador que está siendo controlado de forma remota (la Raspberry Pi principal).

También es posible realizar una conexión remota a través de la web. Para ello, abrir un navegador cualquiera y escribir en la barra del buscador lo siguiente: **http://IP.Servidor:5800**. A continuación, aparecerá en la pantalla la solicitud del ingreso de la contraseña configurada para el servidor.

Ahora bien, si lo que se desea es establecer una conexión remota fuera de la red local, lo que se debe hacer es en primer lugar, habilitar en el router los puertos 5500, 5800 y 5900 en **TCP**. A continuación, abrir el cliente **VNC** o un navegador web e introducir la IP pública que tiene en la red local del servidor. Finalmente, se debe introducir la contraseña que fue configurada inicialmente para el servidor y ya se habrá establecido una conexión remota externa a la red local.

10 Planificación y presupuesto

10.1 Planificación

DIAGRAMA DE GANTT PRINCIPAL			
TAREA	INICIO	DURACIÓN (DÍAS)	FIN
Estudios Previos	09/10/2017	5	13/10/2017
Desarrollo Disp. Elec. Medición de Consumo	16/10/2017	31	17/11/2017
Desarrollo Disp. Elec. Medición de Sonido	20/11/2017	15	04/12/2017
Desarrollo Disp. Elec. Medición de Aceleraciones	05/12/2017	31	05/01/2018
Visualización de Resultados	08/01/2018	5	12/01/2018
Desarrollo Sistema Preventivo	15/01/2018	39	23/02/2018

Tabla 5. Tabla de tareas principales

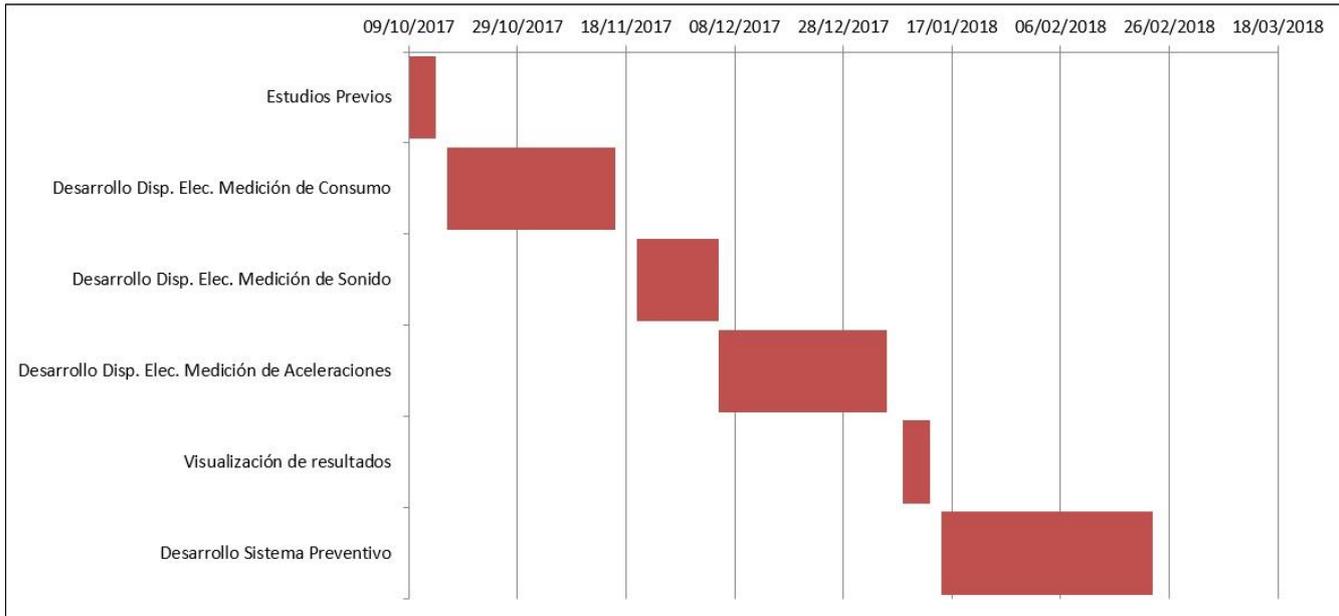


Figura 85. Diagrama de Gantt principal

DIAGRAMA DE GANTT SECUNDARIO				
TAREA PRINCIPAL	SUBTAREA	INICIO	DURACIÓN (DÍAS)	FIN
Estudios Previos	Comprobar Documentación	09/10/2017	3	11/10/2017
	Comprobar Estado del Arte	12/10/2017	2	13/10/2017
Desarrollo Disp. Elec. Medición de Consumo	Diseño Esquema Eléctrico	16/10/2017	3	18/10/2017
	Diseño PCB	19/10/2017	6	24/10/2017
	Fabricación PCB	25/10/2017	0,042	25/10/2017
	Soldar Componentes	25/10/2017	7	01/11/2017
	Programación Software	02/11/2017	11	13/11/2017
	Puesta en Marcha	14/11/2017	4	17/11/2017
Desarrollo Disp. Elec.	Diseño	20/11/2017	4	23/11/2017
	Programación Software	24/11/2017	7	30/11/2017

Medición de Sonido	Puesta en Marcha	01/12/2017	4	04/12/2017
Desarrollo	Diseño	05/12/2017	7	12/12/2017
Disp. Elec.	Programación Software	13/12/2017	16	29/12/2017
Medición de Aceleraciones	Puesta en Marcha	02/01/2018	4	05/01/2018
Visualización de Resultados	Programación Software	08/01/2018	5	12/01/2018
Desarrollo Sistema Preventivo	Desarrollo Cubos OLAP	15/01/2018	39	23/02/2018

Tabla 6. Tabla de tareas secundarias

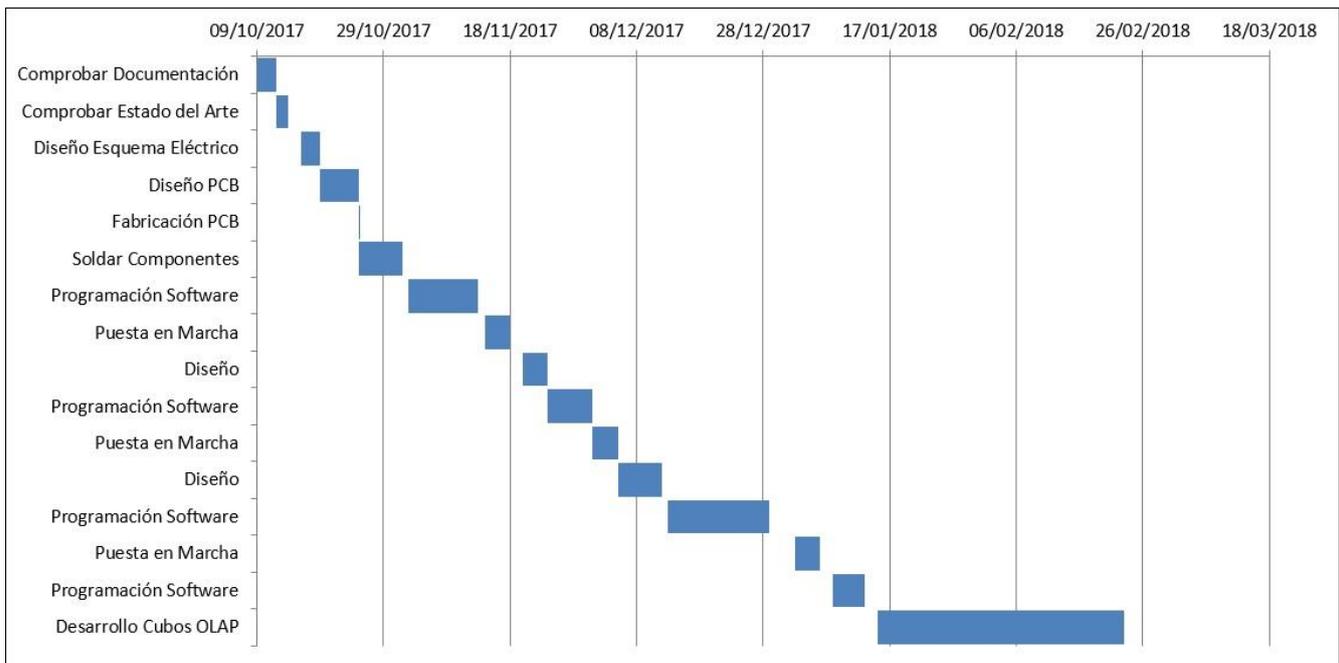


Figura 86. Diagrama de Gantt secundario

10.2 Presupuesto

10.2.1 Cuadro de precios

RECURSOS HARDWARE

ID UNIDAD	DESCRIPCIÓN	UNIDADES	PRECIO UNITARIO
HW01	Resistencia tipo SMD de 51 Ω	3	0,04 €
HW02	Resistencia tipo SMD de 10 K Ω	12	0,003 €
HW03	Resistencia tipo SMD de 12 K Ω	4	0,01 €
HW04	Resistencia tipo SMD de 30 K Ω	4	0,01 €
HW05	Resistencia tipo SMD de 100 K Ω	4	0,01 €
HW06	Condensador tipo SMD de 100 nF	4	0,03 €
HW07	Condensador tipo SMD de 220 nF	4	0,06 €
HW08	Conector Jack de 3,5 mm	3	2,36 €
HW09	Circuito Integrado LM358	4	0,16 €
HW10	Sensor de corriente no invasivo SCT-013-000	3	10,59 €
HW11	Diodo 1N4148 tipo SMD	4	0,14 €
HW12	Diodo LED tipo SMD	1	0,56 €
HW13	Arduino Mega 2560	1	27,65 €
HW14	Módulo wifi ESP-05	1	1,94 €
HW15	Micrófono MAX9814	1	7,82 €
HW16	NI USB-6212 E/S Multifunción	1	1098,89 €
HW17	Acelerómetro ADXL362	1	12,42 €

HW18	Placa Wemos D1 Mini Pro	1	7,41 €
HW19	Placa de carga FC-96	1	1,39 €
HW20	Batería Lipo recargable de 350 mah	1	7,74 €
HW21	RaspBerry Pi 3B	4	23,98 €
HW22	Ordenador portátil	1	544,59 €

Tabla 7. Cuadro de precios de los recursos hardware

RECURSOS SOFTWARE

ID UNIDAD	DESCRIPCIÓN	UNIDADES	PRECIO UNITARIO
SW01	Windows Vista	1	Incluido en PC
SW02	Arduino	1	Software libre gratuito
SW03	<i>Altium</i> ®	1	164,61 €
SW04	<i>LabVIEW</i> ® de <i>National Instruments</i>	1	157,61 €

Tabla 8. Cuadro de precios de los recursos software

RECURSOS HUMANOS

ID UNIDAD	DESCRIPCIÓN	PERÍODO TEMPORAL	PRECIO UNITARIO
RH01	Investigaciones Previas	117,5 Horas	30 €
RH02	Diseño e Implementación Hardware	237,5 Horas	20 €
RH03	Diseño e Implementación Software	258 Horas	25 €
RH04	Dirección	137 Horas	45 €

Tabla 9. Cuadro de precios de los recursos humanos

10.2.2 Presupuestos parciales

RECURSOS HARDWARE

ID UNIDAD	DESCRIPCIÓN	IMPORTE
HW01	Resistencia tipo SMD de 51 Ω	0,12 €
HW02	Resistencia tipo SMD de 10 $K\Omega$	0,03 €
HW03	Resistencia tipo SMD de 12 $K\Omega$	0,05 €
HW04	Resistencia tipo SMD de 30 $K\Omega$	0,06 €
HW05	Resistencia tipo SMD de 100 $K\Omega$	0,02 €
HW06	Condensador tipo SMD de 100 nF	0,11 €
HW07	Condensador tipo SMD de 220 nF	0,25 €
HW08	Conector Jack de 3,5 mm	7,08 €
HW09	Circuito Integrado LM358	0,66 €
HW10	Sensor de corriente no invasivo SCT-013-000	95,31 €
HW11	Diodo 1N4148 tipo SMD	0,56 €
HW12	Diodo LED tipo SMD	0,56 €
HW13	Arduino Mega 2560	27,65 €
HW14	Módulo wifi ESP-05	1,94 €
HW15	Micrófono MAX9814	7,82 €
HW16	NI USB-6212 E/S Multifunción	1098,89 €
HW17	Acelerómetro ADXL362	12,42 €
HW18	Placa Wemos D1 Mini Pro	7,41 €
HW19	Placa de carga FC-96	1,39 €
HW20	Batería Lipo recargable de 350 mah	7,74 €

HW21	RaspBerry Pi 3B	95,92 €
HW22	Ordenador portátil	232,30 €
TOTAL		1598,29 €

Tabla 10. Presupuesto parcial de los recursos hardware

El ordenador portátil se imputa como coste por medio de amortización, cuyo tiempo corresponde a 2 años. A continuación, se muestra el cálculo mediante “amortización en línea recta”.

$$\begin{aligned}
 &\text{Precio} = 544,59 \text{ €} \\
 &\underline{\text{Estimación de venta} = 80 \text{ €}} \\
 &\text{Resta} = 464,59 \text{ €} \\
 &\underline{\hspace{1.5cm}} \\
 &\text{Amortización anual} = 232,30 \text{ €}
 \end{aligned}$$

RECURSOS SOFTWARE

ID UNIDAD	DESCRIPCIÓN	IMPORTE
SW01	Windows Vista	Incluido en PC
SW02	Arduino	Software libre gratuito
SW03	<i>Altium</i> ®	164,61 €
SW04	<i>LabVIEW</i> ® de <i>National Instruments</i>	157,61 €
TOTAL		322,22 €

Tabla 11. Presupuesto parcial de los recursos software

Las herramientas software “*Altium* ®” y “*LabVIEW* ® de *National Instruments*” se imputan como coste por medio de amortización, cuyo tiempo corresponde a 2 años. A continuación, se muestra el cálculo para cada uno de ellos mediante “amortización en línea recta”.

$$\begin{aligned}
 &\text{Precio } \textit{Altium} \text{ ®} = 329,21 \text{ €} \\
 &\underline{\text{Estimación de venta} = 0 \text{ €}} \\
 &\text{Resta} = 329,21 \text{ €} \\
 &\underline{\hspace{1.5cm}} \\
 &\text{Amortización anual} = 164,61 \text{ €}
 \end{aligned}$$

$$\begin{aligned}
 &\text{Precio } \textit{LabVIEW} \text{ ® de } \textit{National Instruments} = 315,21 \text{ €} \\
 &\underline{\text{Estimación de venta} = 0 \text{ €}} \\
 &\text{Resta} = 315,21 \text{ €} \\
 &\underline{\hspace{1.5cm}} \\
 &\text{Amortización anual} = 157,61 \text{ €}
 \end{aligned}$$

RECURSOS HUMANOS

ID UNIDAD	DESCRIPCIÓN	IMPORTE
RH01	Investigaciones Previas	3525 €
RH02	Diseño e Implementación Hardware	4750 €
RH03	Diseño e Implementación Software	6450 €
RH04	Dirección	6165 €
TOTAL		20890 €

Tabla 12. Presupuesto parcial de los recursos humanos

10.2.3 Presupuesto final

CAPÍTULO	IMPORTE TOTAL
Recursos Hardware	1598,29 €
Recursos Software	322,22 €
Recursos Humanos	20890 €
TOTAL	22810,51 €

Tabla 13. Presupuesto final

Presupuesto de Ejecución Material = 22810,51 €

Beneficio Industrial (10%) = 2281,10 €

Costes Generales (15%) = 3421,60 €

Suma de Costes y Beneficios = 28513,21 €

I.V.A (21%) = 5987,77 €

Presupuesto de Ejecución por Contrata = 34500,98 €

El presupuesto de ejecución por contrata asciende a la cantidad de **treinta y cuatro mil quinientos euros con noventa y ocho céntimos**.

Fdo.: Javier Díez Fernández

11 Conclusiones

A lo largo del presente proyecto se han desarrollado numerosas actividades diferentes para lograr introducir un centro de mecanizado en la nueva era de la “Industria 4.0”. Gracias a todas ellas, se han adquirido capacidades nuevas y mejorado y ampliado otras, entre las que destacan:

- Manejo de nuevas herramientas software de diseño de placas.
- Conocimientos del ámbito de la electrónica.
- Conocer y comprender el funcionamiento de un centro de mecanizado.
- Conocimientos de fabricación de piezas.
- Conocimientos sobre señales acústicas y equipos para su captación.
- Conocimientos sobre vibraciones/aceleraciones producidas durante un proceso de fabricación y los equipos empleados para su adquisición.
- Conocimientos informáticos: lenguajes de programación, comandos para trabajar con el sistema operativo Linux, etc.
- Conocimientos sobre bases de datos.
- Conocimientos del ámbito de las comunicaciones y envío de información.
- Conocimientos sobre conexiones remotas.

Tras diseñar y desarrollar cada uno de los dispositivos que se plantearon, estos fueron implementados físicamente en el centro de mecanizado para que cumplieran sus funciones de adquisición de información y envío de la misma a una base de datos para, posteriormente poder analizarla y mostrar las gráficas que aportan los datos más significativos para el usuario. Todo ello pudo comprobarse rigurosamente y asegurando el completo y correcto funcionamiento de todos y cada uno de los dispositivos.

Sin embargo, cabe destacar que este tipo de sistemas son más eficientes y eficaces cuanto más tiempo están en funcionamiento y la cantidad de datos es mayor ya que, es aquí cuando el sistema es capaz de interpretar y localizar patrones donde pueden aparecer anomalías en la maquinaria.

Finalmente se debe destacar que, aunque hay tareas que no han podido llevarse a cabo, estas han sido planteadas indicando cada uno de los pasos que deben seguirse para poder implementarlo en un futuro por el equipo pertinente.

12 Referencias

- Alcañiz, D. (10 de Junio de 2013). *BlogSeas*. Recuperado el 30 de Octubre de 2017, de https://www.seas.es/blog/disenio_mecanico/la-historia-de-la-herramienta-de-corte/
- Aldabaldetrecu, P. (1 de Febrero de 2002). *Interempresas*. Recuperado el 25 de Octubre de 2017, de <http://www.interempresas.net/MetalMecanica/Articulos/1435-Evolucion-tecnica-de-la-maquina-herramienta-Resena-historica.html>
- Analog Devices. (2017). *Analog*. Retrieved Diciembre 15, 2017, from <http://www.analog.com/media/en/technical-documentation/data-sheets/ADXL362.pdf>
- CCM. (Diciembre de 2017). *CCM*. Recuperado el 7 de Diciembre de 2017, de <http://es.ccm.net/contents/66-introduccion-a-las-bases-de-datos>
- Cibertareas. (18 de Septiembre de 2013). *Cibertareas*. Recuperado el 7 de Diciembre de 2017, de <https://cibertareas.info/ventajas-y-desventajas-de-las-bases-de-datos-herramientas-informaticas-1.html>
- Codecademy. (2017). *Codecademy*. Recuperado el 29 de Noviembre de 2017, de https://www.codecademy.com/es/courses/web-beginner-en-aqtuv/0/1?curriculum_id=53a32baefed2a8232e000001
- colaboradores de Wikipedia. (13 de Diciembre de 2017). *LabVIEW*. (l. e. Wikipedia, Editor) Recuperado el 31 de Enero de 2018, de LabVIEW: <https://es.wikipedia.org/wiki/LabVIEW>
- colaboradores de Wikipedia. (24 de Abril de 2018). *JavaScript*. Recuperado el 31 de Enero de 2018, de JavaScript: <https://es.wikipedia.org/wiki/JavaScript>
- colaboradores de Wikipedia. (13 de Febrero de 2018). *Wikipedia*. (l. e. Wikipedia, Editor) Recuperado el 20 de Febero de 2018, de Wikipedia: https://es.wikipedia.org/wiki/Secure_Shell
- colaboradores de Wikipedia. (23 de Abril de 2018). *Wikipedia*. (l. e. Wikipedia, Editor) Recuperado el 5 de Diciembre de 2017, de Wikipedia: <https://es.wikipedia.org/wiki/PHP>
- ExaPro. (2017). *exapro*. Recuperado el 13 de Diciembre de 2017, de <https://www.exapro.es/centro-de-mecanizado-5-ejes-deckel-dmc-75-v-linear-p71031104/>

- Federico Fernández G., R. H. (1994). *sea-acustica*. Recuperado el 15 de Diciembre de 2017, de http://www.sea-acustica.es/fileadmin/publicaciones/publicaciones_4340yq002.pdf
- H.K. Tönshoff, M. J. (1999). *Using acoustic emission signal for monitoring of production processes*. Hannover, Germany: Elsevier.
- Herrera, J. F. (2015). *Códigofacilito*. Recuperado el 29 de Noviembre de 2017, de <https://codigofacilito.com/articulos/que-es-html>
- idesweb. (2012). *idesweb*. Recuperado el 8 de Diciembre de 2017, de <http://idesweb.es/proyecto/proyecto-prac10-php-mysql-acceso-base-de-datos>
- Instruments, National. (2018). *National Instruments*. Recuperado el 8 de Diciembre de 2017, de <http://www.ni.com/es-es/support/model.usb-6212.html>
- KUZU. (16 de Noviembre de 2017). La Internet de las cosas (IoT) y el futuro de la industria.
- Logicalis. (24 de Julio de 2015). *Logicalis*. Recuperado el 4 de Diciembre de 2017, de <https://blog.es.logicalis.com/analytics/cubos-olap-y-estructuras-multidimensionales-todo-lo-que-hay-que-saber>
- Logicalis. (13 de Abril de 2015). *Logicalis*. Recuperado el 11 de Diciembre de 2017, de https://blog.es.logicalis.com/analytics/big-data-analytics-adelantate-al-manana?__hstc=61804339.20469dad3c1f40c7977911c9af516295.1509352762950.1509352762950.1509352762950.1&__hssc=61804339.1.1509352762950&__hsfp=2686770518
- Logicalis. (10 de Julio de 2017). *Logicalis*. Recuperado el 11 de Diciembre de 2017, de <https://blog.es.logicalis.com/analytics/las-claves-del-big-data-de-analytics-al-negocio-segun-once-expertos>
- Logicalis. (4 de Abril de 2017). *Logicalis*. Recuperado el 11 de Diciembre de 2017, de <https://blog.es.logicalis.com/analytics/base-de-datos-multidimensional-respuesta-a-las-cuestiones-mas-complejas-del-e-commerce>
- National Instruments. (2018). *National Instruments*. Obtenido de National Instruments: <http://www.ni.com/es-es/support/model.usb-6212.html>
- National Instruments. (2018). *National Instruments*. Retrieved Diciembre 12, 2017, from <http://www.ni.com/pdf/manuals/375196c.pdf>
- NCTA. (2015, Agosto 13). *NCTA*. Retrieved Noviembre 27, 2017, from <https://www.ncta.com/whats-new/behind-the-numbers-growth-in-the-internet-of-things-2>

Oscar Gonzalo, I. Q. (15 de Septiembre de 2010). *Interempresas*. Recuperado el 25 de Octubre de 2017, de <http://www.interempresas.net/MetalMecanica/Articulos/43234-Nuevas-tendencias-para-el-mecanizado.html>

PHP. (2017). *PHP*. Recuperado el 29 de Noviembre de 2017, de <http://php.net/manual/es/intro-what-is.php>

Prodintec. (2018). *Prodintec*. Recuperado el 22 de Noviembre de 2017, de <http://www.prodintec.es/es/nosotros/quienes-somos>

Raspberry Pi. (2017). *europa-electrocomponents*. Retrieved Diciembre 26, 2017, from <http://docs-europe.electrocomponents.com/webdocs/14ba/0900766b814ba5fd.pdf>

REALVNC. (14 de Marzo de 2018). *realvnc*. Obtenido de realvnc: <https://www.realvnc.com/es/products/vnc/>

Velasco, R. (12 de Mayo de 2017). *Redes Zone*. Obtenido de Redes Zone: <https://www.redeszone.net/2017/05/12/controlar-remota-linux-raspberry-pi/>

WEMOS Electronic. (2018). *wiki.wemos*. Recuperado el 21 de Diciembre de 2017, de https://wiki.wemos.cc/products:d1:d1_mini_pro

ANEXO A. Especificaciones técnicas Deckel Maho

En este anexo, se presenta una tabla con las principales características técnicas de la máquina “Deckel Maho” (ExaPro, 2017).

CARACTERÍSTICA	VALOR
Recorrido longitudinal (eje X)	885 mm
Recorrido transversal (eje Y)	600 mm
Recorrido vertical (eje Z)	600 mm
Tipo de CNC	iTNC 530 HEIDENHAIN SMART NC
Tipo de ejes	Cabezal universal
Longitud de la mesa	750 mm
Ancho de la mesa	750 mm
Peso máximo autorizado	800 Kg
Número de herramientas	30
Cono del husillo	HSK A 63
Velocidad máxima del husillo	28000 rpm
Potencia del motor del husillo	55 kW
Tabla rotativa	Sí
Longitud x Anchura x Altura	4600 mm x 2500 mm x 3000 mm
Peso	10200 Kg
Con normativa CE	Sí
Condición	Bajo tensión

Tabla 14. Especificaciones técnicas Deckel Maho

ANEXO B. Especificaciones técnicas ni usb-6212

El dispositivo NI USB-6212, es un dispositivo DAQ multifunción encargado de tomar muestras de señales como, por ejemplo, el sonido. En la tabla que se muestra a continuación, pueden consultarse las principales características del mismo. Si se desea consultar otras características en mayor profundidad, consultar la datasheet del producto (National Instruments, 2018).

CARACTERÍSTICA	DESCRIPCIÓN
Entradas	Analógicas y Digitales
Salidas	Analógicas y Digitales
ADC	16 bits
Nº de canales	8 diferenciales o 16 simples
Tiempo de resolución	50 ns
Precisión de temporización	50 ppm de tasa de muestras
Rango de entrada	± 0.2 V, ± 1 V, ± 5 V, ± 10 V
Máxima tensión de trabajo	± 10.4 V of AI GND
Conexión USB	Sí

Tabla 15. Tabla de características NI USB-6212

ANEXO C. Especificaciones técnicas del acelerómetro

En este apartado, se presenta una tabla con las principales especificaciones técnicas del acelerómetro ADXL362. En caso de que sea necesario consultar otras propiedades, consultar datasheet del producto (Analog Devices, 2017).

CARACTERÍSTICA	DESCRIPCIÓN
Encapsulado	3 mm \times 3.25 mm \times 1.06 mm
Modos de trabajo	3
Ejes de medida	3
Consumo	< 2uA a 100 Hz
Resolución de salida	12 bits
Rangos de medidas	± 2 g, ± 4 g y ± 8 g
Multimodo de salida FIFO	Sí
Sensor de temperatura	Sí
Rango de trabajo	1.6V a 3.5V

Tabla 16. Tabla de características ADXL362

ANEXO D. Especificaciones técnicas placa WEMOS D1 Mini Pro

A lo largo de este apartado, se presentan las principales características de la placa WEMOS D1 Mini Pro. En caso de necesidad o interés de consulta de otra serie de características, consultar *datasheet* del producto (WEMOS Electronic, 2018).

CARACTERÍSTICA	DESCRIPCIÓN
Entradas digitales	11
Salidas digitales	11
Entradas analógicas	1 (Máximo a 3.2V)
Memoria Flash	16M Bytes
Comunicación	Interrupt/pwm/I2C/one-wire
Antena	Cerámica
Puerto USB	CP2104 USB-TO-UART IC
Encapsulado	34,2 mm x 25,6 mm x 5 mm

Tabla 17. Tabla de características WEMOS D1 Mini Pro

ANEXO E. Especificaciones técnicas Raspberry Pi

En este apartado, se muestra una tabla con las características principales de la Raspberry Pi empleada para el desarrollo del sistema web, junto con un esquema de la placa.

CARACTERÍSTICA	DESCRIPCIÓN
Procesador	Chip Broadcom BCM2387. 1.2GHz Quad-Core ARM Cortex-A53 802.11 b/g/n Wireless LAN y Bluetooth 4.1 (Bluetooth Classic y LE)
GPU	Dual Core VideoCore IV® Multimedia Co-Procesador.
Memoria	1GB LPDDR2
Sistema Operativo	Sistema desde tarjeta Micro SD, versiones de Linux o Windows 10 IoT
Dimensiones	85mm x 56 mm x 17mm
Potencia	Conector Micro USB 5V1, 2.5A

Tabla 18. Tabala de características Raspberry Pi 3 modelo B

En caso de ser necesario consultar con mayor detalle alguna otra característica, consultar datasheet del producto (Raspberry Pi, 2017).

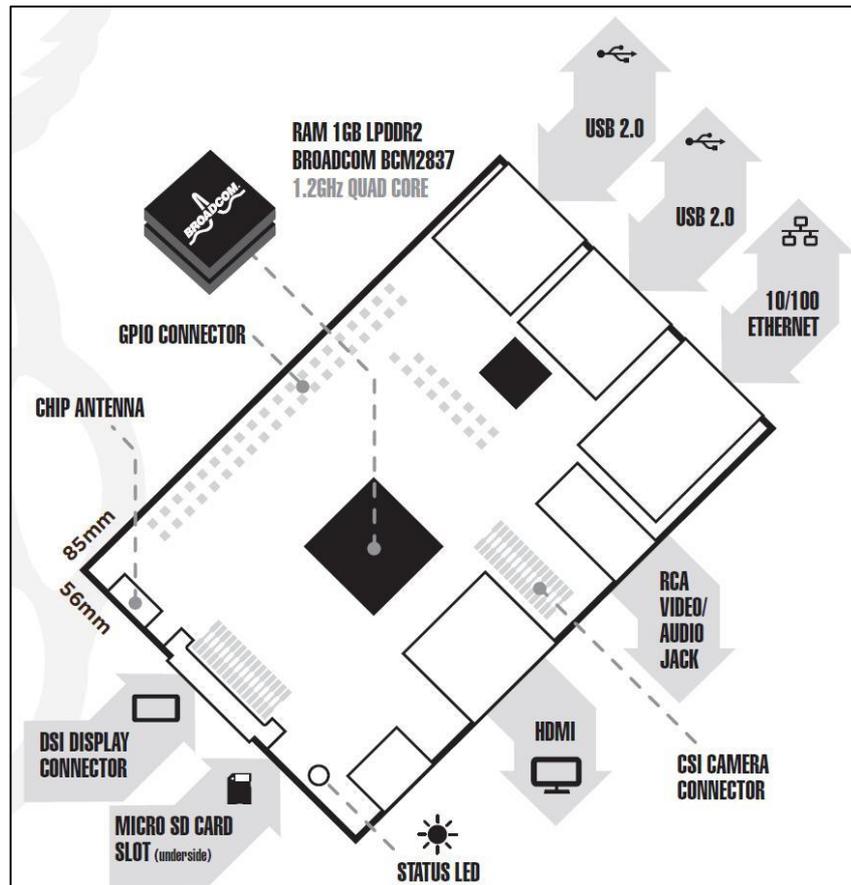


Figura 87. Esquema Raspberry Pi 3 modelo B

ANEXO F. Programación con Arduino. Medición de consumos

```

1. //Se declaran algunos parámetros globales:
2.
3.
4. int cont=0; //Contador para recorrer el array.
5.
6.
7. // -- variables and pins definition -----
8. const int analogInPin=A0;           // analog input pin
9. const int analogInPin1=A1;
10. const int analogInPin2=A2;
11. const int analogInPin3=A3;
12.
13. int retardo=2 ; // (tiempo (s.) entre visionados)
14.
15. float lectura, ff, pKW, iA, vV, vS, S_Ratio;
16. float lectura1, pW1, iA1, vS1;
17. float lectura2, pKW2, iA2, vS2;
18. float lectura3, pKW3, iA3, vS3;
19. int lectura_prueba;
20. float ANREAD[100];
21. float Maxval=0;
22. float Maxval_avg;
23. int i, z, j, k;
24.

```

```

25.     float P_0,P_1,P_2,P_3;
26.
27.
28.     void setup() {
29.
30.         Serial.begin(115200);
31.         Serial1.begin(115200);
32.
33.         pinMode(52, OUTPUT);
34.         digitalWrite(52, LOW);
35.         analogReference(DEFAULT);
36.
37.         Serial.println("MEDICIÓN DE CONSUMOS Y ENVÍO DE RESULTADOS
A BASE DE DATOS");
38.
39.         S_Ratio=20;           // Sensor/ratio (mV/mA ) : 36.5
40.         vV=400;              // valor de tension a computar
41.         ff=5; // freq. factor / (50Hz -> 5 / 60Hz -> 4.15)
42.         //Se inicializan el resto de variables para evitar
problemas en los cálculos posteriores
43.         lectura=0.0;
44.         vS=0.0;
45.         iA=0.0;
46.         pKW=0.0;
47.         lectural=0.0;
48.         vS1=0.0;
49.         iA1=0.0;
50.         pW1=0.0;
51.         lectura2=0.0;
52.         vS2=0.0;
53.         iA2=0.0;
54.         pKW2=0.0;
55.         lectura3=0.0;
56.         vS3=0.0;
57.         iA3=0.0;
58.         pKW3=0.0;
59.         P_0=0.0;
60.         P_1=0.0;
61.         P_2=0.0;
62.         P_3=0.0;
63.
64.         delay(500);
65.
66.         IniciaWifi(); //Esta función se encarga de ejecutar los
distintos comandos AT del array.
67.
68.     }
69.
70.     void loop() {
71.         Serial.println("Visualización de registro");
72.         digitalWrite(52, HIGH);
73.
74.
75.         P_0=Lectura_A(0);
76.         delay(500);
77.         P_1=Lectura_A(1);
78.         delay(500);
79.         P_2=Lectura_A(2);
80.         delay(200);
81.         P_3=((P_0*400*0.84)+(P_1*400*0.84)+(P_2*400*0.84))/1000;
82.

```

```

83.
84.
85.
86.     Serial.println("Iniciando envío...");
87.     delay(200);
88.     Enviar();
89.     digitalWrite(52, LOW);
90.     delay(retardo*1000);
91. }
92.
93.
94. void IniciaWifi()
95. {
96.     String Comandos[]=
97.     {
98.         "AT+RESTORE",
99.         "AT+CWMODE=3", //Configurado como cliente y AP (punto de
    acceso).
100.        "AT+CWJAP_CUR=\"CLUSTER\", \"fundaci0n%\"",
101.        "AT+CIPMUX=1",
102.        "AT+CIPSERVER=1,80",
103.        "END" //Marca el fin de los comandos AT.
104.    };
105.
106.    while (Comandos[cont]!="END")
107.    {
108.        Serial1.println(Comandos[cont++]);
109.        Serial.print("vuelta: ");
110.        Serial.println(cont);
111.        while(true)
112.        {
113.            String s=GetLineWifi();
114.            delay(1000);
115.
116.            if(s!="")
117.            {
118.                Serial.println(s);
119.            }
120.            if(s.startsWith("no change"))
121.            {
122.                break;
123.            }
124.            if(s.startsWith("OK"))
125.            {
126.                break;
127.            }
128.            if(s.startsWith("ready"))
129.            {
130.                break;
131.            }
132.            if(s.startsWith("FAIL"))
133.            {
134.                Serial1.println(Comandos[2]);
135.                delay(1000);
136.            }
137.
138.        }
139.        Serial.println("*****");
140.        delay(1000);
141.    }
142. }

```

```

143.
144.     String GetLineWifi ()
145.     {
146.         String S = "" ;
147.         if (Serial1.available())
148.         {
149.             char c = Serial1.read(); ;
150.             while ( c != '\n' )           //Hasta que el caracter
sea intro
151.             {
152.                 S = S + c ;
153.                 delay(25) ;
154.                 c = Serial1.read();
155.             }
156.             return( S ) ;
157.         }
158.     }
159.
160.
161.
162.     //----- FUNCIÓN LECTURA ENTRADA ANALÓGICA -----
-----
163.     float Lectura_A(int canal)
164.     {
165.         int CANAL=canal;
166.         Maxval=0;
167.         Maxval_avg=0;
168.         delay(2);
169.         //Obtenemos un array de 100 lecturas del adc
170.         for (z=0; z<100; z++)
171.         {
172.             for (j=0; j<100; j++)
173.             {
174.                 if (CANAL==0)
175.                 {
176.                     ANREAD[j]=analogRead (A0) ;
177.                     break;
178.                 }
179.                 else if (CANAL==1)
180.                 {
181.                     ANREAD[j]=analogRead (A1) ;
182.                     break;
183.                 }
184.                 else if (CANAL==2)
185.                 {
186.                     ANREAD[j]=analogRead (A2) ;
187.                     break;
188.                 }
189.                 delayMicroseconds (1) ;
190.             }
191.
192.             //Buscamos el maximo valor dentro del array
193.             for (k=0; k<100; k++)
194.             {
195.                 if (Maxval<ANREAD[k])
196.                     Maxval=ANREAD[k];
197.             }
198.             Maxval_avg=(Maxval+Maxval_avg);
199.
200.         }
201.

```

```

202.     Maxval_avg=(Maxval_avg/10.0);
203.
204.     lectural=((Maxval*5.0)/1024.0); //lectura en voltios
205.
206.     iA1=lectural;
207.     pW1=(400*iA1*0.9);           // Potencia (kW)
208.
209.     Serial.print("Canal=");
210.     Serial.print(canal);
211.     Serial.print(", Corriente [mA]=" );
212.     Serial.print(lectural,2);
213.
214.
215.     Serial.print(", Power [W]=" );
216.     Serial.print(pW1);
217.     Serial.print("\n" );
218.     return (iA1*1000); //Valor devuelto
219. }
220.
221. //----- Función para enviar los datos vía Wifi -----
-----
222. void Enviar()
223. {
224.     Serial.println("Enviando...");
225.     Serial1.println("AT+CIPSTART=0,\"TCP\", \"192.168.30.203\",
80");
226.     delay(500);
227.     Serial1.println("AT+CIPSEND=0,115");
228.     delay(500);
229.     Serial1.print("GET /consumos_deckel.php?&canal1=");
230.     Serial1.print(P_0);
231.     Serial1.print("&canal2=");
232.     Serial1.print(P_1);
233.     Serial1.print("&canal3=");
234.     Serial1.print(P_2);
235.     Serial1.print("&canal4=");
236.     Serial1.print(P_3);
237.     Serial1.println("
HTTP/1.0\r\nHost:192.168.30.203\r\n\r\n");
238.     delay(500);
239.     Serial.println("Datos recibidos.");
240.     Serial1.println("AT+CIPCLOSE=0");
241. }

```

ANEXO G. Programación con Arduino. Medición de vibraciones

```

1. #include <ESP8266WiFi.h>
2.
3. ////////////////
4. //ACCELEROMETRO
5. #include <SPI.h>
6. #include <ADXL362.h>
7.

```

```

8. ADXL362 xl;
9.
10.     int16_t temp;
11.     int16_t XValue, YValue, ZValue, Temperature;
12.     //////////////////////////////////
13.
14.     // Wi-Fi Settings
15.     const char* ssid = "CLUSTER";//Nombre de la red (SSID)
16.     const char* password = "fundac10n%";//Contraseña de red
17.
18.     WiFiClient client;
19.
20.     // ThingSpeak Settings
21.     const int channelID = 233555;
22.     String writeAPIKey = "XJJV3AXY1LGRQUC1"; // Clave para canal
23.     const char* server = "192.168.30.200";
24.     const int postingInterval = 500; // Intervalo de registro
25.
26.     void setup() {
27.         Serial.begin(115200);
28.         /*Serial.println("Reseteando configuración ESP...");
29.         ESP.eraseConfig();
30.         ESP.reset();
31.         Serial.print("Espere");
32.         delay(1000);
33.         Serial.print(".");
34.         delay(1000);
35.         Serial.print(".");
36.         delay(1000);
37.         Serial.println(".");
38.         delay(1000);*/
39.         Serial.println("Configurando Wifi...");
40.         WiFi.begin(ssid, password);
41.
42.         while (WiFi.status() != WL_CONNECTED) {
43.             delay(500);
44.         }
45.         Serial.println("Establecimiento hecho!!");
46.
47.         xl.begin(0); // Establecimiento protocolo
SPI
48.         xl.beginMeasure(); // Activar modo medición del
ADXL362
49.     }
50.
51.     void loop() {
52.         xl.readXYZTData(XValue, YValue, ZValue, Temperature);
53.         String message = "Conectando a ";
54.         message += String(server);
55.         Serial.println(XValue);
56.         Serial.println(YValue);
57.         Serial.println(ZValue);
58.         Serial.println(message);
59.         if (client.connect(server, 80)) {
60.
61.             Serial.println("Conectado a albertozapicoania.com");
62.
63.             // Medición de entrada analógica (A0)
64.             int valueA0 = analogRead(A0);
65.
66.             // Construit cuerpo requerimiento API

```

```

67.     String body = "field1=";
68.     body += String(XValue);
69.     body += "\n";
70.     body += "field2=";
71.     body += String(300);
72.     body += "\n";
73.     body += "field3=";
74.     body += String(300);
75.
76.     client.print( "GET /add_secure.php?");
77.     client.print("&X_acc=");
78.     client.print( String(XValue) );
79.     client.print("&Y_acc=");
80.     client.print( String(YValue) );
81.     client.print("&Z_acc=");
82.     client.print( String(ZValue) );
83.     client.println( " HTTP/1.1");
84.     client.println( "Host: albertozapicoania.com" );
85.     client.println( "Content-Type: application/x-www-form-
urlencoded" );
86.     client.println( "Connection: close" );
87.     client.println();
88.     client.println();
89.
90.     }
91.     else
92.         Serial.println("Error conectando al servidor
albertozapicoania.com");
93.         client.stop();
94.
95.         // Esperar y volver a enviar
96.         delay(postingInterval);
97.     }

```

ANEXO H. Programación de archivo PHP para almacenar los datos de consumo en la base de datos

```

1. <?php
2.     function Connection(){
3.         $servername = "192.168.30.203";
4.         $username = "root";
5.         $password = "fundac10n%";
6.         $dbname = "PRUEBA";
7.         $query_sucess = false;
8.
9.         $conn = new mysqli($servername,$username,$password,
$dbname);
10.
11.         if (!$conn) {
12.             die('MySQL ERROR: ' . mysqli_error($conn));
13.         }
14.
15.         mysqli_select_db($conn,$dbname) or die( 'MySQL
L ERROR: ' . mysqli_error($conn) );

```

```

16.
17.             return $conn;
18.         }
19.
20.         $link=Connection();
21.         $xvalue1=($_GET['canal1']);
22.         $xvalue2=($_GET['canal2']);
23.         $xvalue3=($_GET['canal3']);
24.         $xvalue4=($_GET['canal4']);
25.
26.
27.
28.
29.         $query1 = "INSERT INTO `CON_DECKEL` (`canal1`,
30. `canal2`, `canal3`, `canal4`)
31.         VALUES
32. ('".$xvalue1."', '".$xvalue2."', '".$xvalue3."', '".$xvalue4."')";
33.
34.             if ($link-
35. >query(htmlspecialchars($query1)))
36.             {
37.                 $msg = "Fichero actualizado";
38.                 $query_success = true;
39.             }
40.             else
41.             {
42.                 $msg = "Error al
43. actualizar" . $link->error;
44.             }
45.
46.
47.
48.         echo $msg;
49.         mysqli_close($link);
50.
51.     ?>

```

ANEXO I. Programación de archivo PHP para almacenar los datos de aceleraciones en la base de datos

```

1. <!DOCTYPE html>
2. <?php
3. $servername = "192.168.30.203";
4. $username = "root";
5. $password = "fundac10n%";
6. $dbname = "PRUEBA";
7. $query_success = false;
8. // Create connection
9. $conn = new mysqli($servername, $username, $password, $dbname);
10.
11.     $fecha_tratada = str_replace("T", " ", $_GET['fecha']);
12.     // Check connection
13.     if (!$conn->connect_error)

```

```

14.     {
15.         if($_GET['maquina'] === "deckel")
16.         {
17.             $sql = "SELECT * FROM `ACC_DECKEL` WHERE
fecha < (SELECT STR_TO_DATE('{$_GET['fecha']}','%Y-%m-%dT%H:%i')
ORDER BY `fecha` DESC LIMIT 10";
18.         }
19.         else if ($_GET['maquina'] === "kern")
20.         {
21.             $sql = "SELECT * FROM `ACC_KERN` WHERE fecha
< (SELECT STR_TO_DATE('{$_GET['fecha']}','%Y-%m-%dT%H:%i') ORDER
BY `fecha` DESC LIMIT 10";
22.         }
23.         //echo $sql;
24.         if ($resultado = $conn->query($sql)) {
25.             $query_success = true;
26.             $valores = array();
27.             //var_dump($valores);
28.             while ($row = $resultado->fetch_object())
29.             {
30.                 array_push($valores, array($row-
>X_acc, $row->Y_acc, $row->Z_acc, $row->fecha));
31.             }
32.             $resultado->close();
33.         }
34.         //var_dump($valores);
35.         $conn->close();
36.     } else {
37.         $msg = "Error de conexión: " . $conn->connect_error;
38.     }
39.     ?>
40.     <html>
41.         <head>
42.             <title>SMASCHING4</title>
43.             <meta name="viewport" content="width=device-
width, initial-scale=1.0">
44.             <link rel="stylesheet" type="text/css"
href="../main-style.css">
45.             <style>
46.                 .contenedor_graficas {
47.                     display: flex;
48.                     flex-direction: row;
49.                     flex-wrap: wrap;
50.                 }
51.                 .grafica {
52.                     width: 20em;
53.                     height: 20em;
54.                     padding: .5em;
55.                     background-color: white;
56.                 }
57.             </style>
58.             <script type="text/javascript"
src="https://www.gstatic.com/charts/loader.js"></script>
59.             <script type="text/javascript">
60.                 google.charts.load('current',
{packages: ['corechart', 'line']});
61.                 google.charts.setOnLoadCallback(drawB
ackgroundColor);
62.
63.                 function drawBackgroundColor()
64.                 {

```

```

65.         var data = new
google.visualization.DataTable();
66.         data.addColumn('string', 'X');
67.         data.addColumn('number', 'acc');
68.
69.         data.addRows([
70.             <?php
71.             $count_valores = count($valores);
72.             for ($i = $count_valores - 1; $i >= 0; $i--) {
73.                 echo "[".$valores[$i][3]."', '.$valores[$i][0]."]";
74.                 if($i > 0)
75.                     echo ",";
76.             }
77.             ?>
78.         ]);
79.
80.         var options =
81.         {
82.
83.             Axis: {
84.                 title: 'Time'
85.             },
86.             vAxis: {
87.                 title:
88.                 'Aceleracion'
89.             },
90.             backgroundColor:
91.             '#f1f8e9'
92.         };
93.
94.         var chart = new
google.visualization.LineChart(document.getElementById('top_x_div')
);
95.         chart.draw(data, options);
96.
97.         var data2 = new
google.visualization.DataTable();
98.         data2.addColumn('string', 'X');
99.         data2.addColumn('number', 'acc');
100.
101.         data2.addRows([
102.             <?php
103.             $count_valores = count($valores);
104.             for ($i = $count_valores - 1; $i >= 0; $i--) {
105.                 echo "[".$valores[$i][3]."', '.$valores[$i][1]."]";
106.                 if($i > 0)
107.                     echo ",";
108.             }
109.             ?>
110.         ]);
111.
112.         var options2 =
113.         {
114.
115.             Axis: {
116.                 title: 'Time'

```

```

117.                                     backgroundColor:
    '#f1f8e9'
118.                                     };
119.
120.                                     var chart2 = new
    google.visualization.LineChart(document.getElementById('right_x_div
    '));
121.                                     chart2.draw(data2, options2);
122.
123.                                     var data3 = new
    google.visualization.DataTable();
124.                                     data3.addColumn('string', 'X');
125.                                     data3.addColumn('number', 'acc');
126.
127.                                     data3.addRows([
128.                                     <?php
129.                                     $count_valores = count($valores);
130.                                     for ($i = $count_valores - 1; $i >= 0; $i--) {
131.                                     echo "[".$valores[$i][3]."', {$valores[$i][2]}]";
132.                                     if($i > 0)
133.                                     echo ",";
134.                                     }
135.                                     ?>
136.                                     ]);
137.
138.                                     var options3 =
139.                                     {
140.
141.                                     Axis: {
142.                                     title: 'Time'
143.                                     },
144.                                     vAxis: {
145.                                     title:
    'Aceleracion'
146.                                     },
147.                                     backgroundColor:
    '#f1f8e9'
148.                                     };
149.                                     var chart3 = new
    google.visualization.LineChart(document.getElementById('left_x_div'
    ));
150.                                     chart3.draw(data3,
    options3);
151.                                     };
152.
153.
154.                                     </script>
155.                                     </head>
156.                                     <body>
157.                                     <header>
158.                                     <p>SMASCHING4</p>
159.                                     </header>
160.                                     <main>
161.                                     <article>
162.                                     <h1>Mostrar información</h1>
163.                                     <form
    action="aceleraciones.php" method="get">
164.                                     <p>Maquina:
165.                                     <select
    name="maquina" <?php if ($estadoBD != 0) echo 'disabled ';?>>

```

```

166.                                     <opti
on
value="deckel" <?php if($maquinaBD == "deckel") echo 'selected="se
lected"'; ?>/>Deckel</option>
167.                                     <opti
on
value="kern" <?php if ($maquinaBD == "kern") echo 'selected="select
ed"'; ?>/>Kern</option>
168.                                     </select>
169.                                     </p>
170.                                     <p>Fecha inicio:
<input type="datetime-local" name="fecha" /></p>
171.
172.                                     <p><input
type="submit" /></p>
173.
174.                                     <div
class="contenedor_graficas">
175.                                     <div
id="top_x_div" class="grafica
card"></div>
176.                                     <div
id="right_x_div" class="grafica card"></div>
177.                                     <div
id="left_x_div" class="grafica card"></div>
178.                                     </div>
179.                                     </form>
180.                                     <?php
181.                                     if($msg)
182.                                     {
183.                                     ?>
184.                                     <p><?php echo $msg; ?></p>
185.                                     <?php } ?>
186.                                     <?php
187.                                     if($msg_estado)
188.                                     {
189.                                     ?>
190.                                     <p><?php echo $msg_estado; ?>
</p>
191.                                     <?php } ?>
192.                                     </article>
193.                                     </main>
194.                                     <footer>
195.                                     <p>PROGRAMA ASTURIAS</p>
196.                                     </footer>
197.                                     </body>
198.                                     </html>

```

ANEXO J. Programación de archivo PHP para almacenar los datos de sonido en la base de datos

```
1. <?php
2.     function Connection(){
3.         $servername = "192.168.30.203";
4.         $username = "root";
5.         $password = "fundac10n%";
6.         $dbname = "PRUEBA";
7.         $query_sucess = false;
8.
9.         $conn = new mysqli($servername,$username,$password,
10. $dbname);
11.
12.         if (!$conn) {
13.             die('MySQL ERROR: ' . mysqli_error($conn));
14.         }
15.         mysqli_select_db($conn,$dbname) or die('MySQ
16. L ERROR: ' . mysqli_error($conn) );
17.         return $conn;
18.     }
19.
20.     $link=Connection();
21.     $xvalue1=($_GET['VALOR1']);
22.     $xvalue2=($_GET['VALOR2']);
23.     $xvalue3=($_GET['VALOR3']);
24.     $xvalue4=($_GET['VALOR4']);
25.     $xvalue5=($_GET['VALOR5']);
26.     $xvalue6=($_GET['VALOR6']);
27.     $xvalue7=($_GET['VALOR7']);
28.     $xvalue8=($_GET['VALOR8']);
29.     $xvalue9=($_GET['VALOR9']);
30.     $xvalue10=($_GET['VALOR10']);
31.
32.
33.     $query1 = "INSERT INTO `SON_DECKEL` (`VALOR1`,
34. `VALOR2`, `VALOR3`, `VALOR4`, `VALOR5`, `VALOR6`, `VALOR7`, `VALOR8`,
35. `VALOR9`, `VALOR10`)
36.     VALUES
37. ('".$xvalue1."', '".$xvalue2."', '".$xvalue3."', '".$xvalue4."', '".$xv
38. alue5."', '".$xvalue6."', '".$xvalue7."', '".$xvalue8."', '".$xvalue9."
39. ', '".$xvalue10."')";
40.
41.     if ($link->query(htmlspecialchars($query1)))
42.     {
43.         $msg = "Fichero actualizado";
44.         $query_sucess = true;
45.     }
46.     else
47.     {
48.         $msg = "Error al
49. actualizar" . $link->error;
50.     }
```

```
48.  
49.  
50.         echo $msg;  
51.         mysqli_close($link);  
52.  
53.         ?>
```

ANEXO K. Programación del clúster

Para evitar realizar una memoria muy cargada de programación, en el presente anexo se describen las programaciones relativas al sistema web de transferencia y tratamiento de datos (clúster).

En primer lugar, se ha llevado a cabo la instalación de los sistemas operativos de cada una de las Raspberry Pi, siendo este el “Linux Ubuntu MATE 16.04.2 Xenial Xerus” para procesadores **ARMv7**. A continuación, se asigna una dirección IP fija de red local a la RPi0, la cual actúa como *balanceador de carga*, y se instala el servidor web **Apache2**. Para que dicho servidor funcione como *balanceador de carga*, se deben activar una serie de módulos que tiene incorporados:

```
$ sudo a2enmod lbmethod_byrequests slotmem_shm  
$ sudo a2enmod proxy_http  
$ sudo a2enmod proxy_balancer
```

Tras la activación de los módulos, se procede a establecer las direcciones IP de los servidores web en espejo a través del archivo que se muestra a continuación:

```
ServerAdmin webmaster@localhost  
DocumentRoot /var/www/html  
  
ErrorLog ${APACHE_LOG_DIR}/error.log  
CustomLog ${APACHE_LOG_DIR}/access.log combined  
ProxyRequests Off  
<Proxy balancer://cluster>  
BalancerMember http://192.168.XXX.YYY:80  
BalancerMember http://192.168.XXX.YYY:80  
AllowOverride None  
Order allow,deny  
allow from all  
ProxySet lbmethod=byrequests  
</Proxy>  
<Location /balancer-manager>  
SetHandler balancer-manager  
Order allow,deny  
allow from 192.168.2
```

```

</Location>
ProxyPass /balancer-manager !
ProxyPass / balancer://cluster/
ErrorLog ${APACHE_LOG_DIR}/error.log
# Possible values include: debug, info, notice, warn, error, crit,
# alert, emerg.
LogLevel warn
CustomLog ${APACHE_LOG_DIR}/access.log combined
# vim: syntax=apache ts=4 sw=4 sts=4 sr noet

```

La instalación de las dos RPi's (RPi1 y RPi2) con el servidor web en espejo, es igual en ambos casos. Se debe instalar el servidor web **Apache2** en ambas y asignarlas una dirección IP fija a cada una. Hay que tener en cuenta que se deben copiar los archivos de la página web en las dos RPi's para un correcto funcionamiento.

Retornando a la RPi0, abrir un navegador y escribir la URL: <http://192.168.30.200/balancer-manager> y, si todos los pasos anteriores se hicieron correctamente, debería aparecer la ventana que se observa en la figura.

Load Balancer Manager for 192.168.30.200

Server Version: Apache/2.4.18 (Ubuntu)
 Server Built: 2016-07-14T12:32:26
 Balancer changes will NOT be persisted on restart.
 Balancers are inherited from main server.
 ProxyPass settings are inherited from main server.

LoadBalancer Status for [balancer://rpicluster](#) [p861c0747_rpicluster]

MaxMembers	StickySession	DisableFailover	Timeout	FailoverAttempts	Method	Path	Active
2 [2 Used]	(None)	Off	0	1	byrequests	/	Yes

Worker URL	Route	RouteRedir	Factor	Set	Status	Elected	Busy	Load	To	From
http://192.168.30.201			1	0	Init Ok	2656	0	-1	680K	93K
http://192.168.30.202			1	0	Init Ok	2645	0	1	676K	90K

Apache/2.4.18 (Ubuntu) Server at 192.168.30.200 Port 80

Figura 88. Configuraciones del Clúster

Como puede observarse en la figura 88, la RPi0 (balanceador de carga) tiene asignada la dirección IP 192.168.30.200, la RPi1 la dirección IP 192.168.30.201 y la RPi2 la dirección IP 192.168.30.202. Pueden añadirse un total de 250 unidades de servidores web en espejo.

Para finalizar, hay que asignar una dirección IP fija a la última RPi (denominada RPi3, en este caso) para que pueda acceder de forma remota a la base de datos que contiene y poder configurar la misma con permisos de acceso desde la red local. Para tal fin, se debe modificar el archivo llamado *my.conf*, añadiendo las líneas:

```
Skip-external-locking  
Bind-address = 192.168.30.200
```

La dirección IP indicada es la que tendrá acceso remoto, siendo en este caso la RPi0 y que ha sido configurada como *balanceador de carga*.

Gracias a esta configuración y a la conexión de las cuatro RPi's a un router wifi, se podrán registrar todos los datos recabados por los dispositivos elaborados e instalados en el centro de mecanizado "Deckel Maho".