



**Universidad  
de La Laguna**

**Escuela Superior  
de Ingeniería y Tecnología**

**Departamento de Ingeniería Industrial**

## **Trabajo de Fin de Grado**

**Protección a la Avifauna de la colisión contra  
estructuras peligrosas mediante señales ultrasónicas.**

**Titulación:** Grado en Ingeniería Electrónica Industrial y  
Automática

**Estudiante:**

Rubén Morales Rodríguez

**Tutor:** Fernando Luis Rosa González

**Co-Tutor:** Iván Rodríguez Méndez

11 de marzo de 2022

La publicación de este Trabajo Fin de Grado solo implica que el estudiante ha obtenido al menos la nota mínima exigida para superar la asignatura correspondiente, no presupone que su contenido sea correcto, aunque si aplicable. En este sentido, la Universidad de La Laguna no posee ningún tipo de responsabilidad hacia terceros por la aplicación total o parcial de los resultados obtenidos en este trabajo. También pone en conocimiento del lector, que según la ley de protección de propiedad intelectual, los resultados son propiedad intelectual del alumno, siempre y cuando se haya procedido a los registros de propiedad intelectual o solicitud de patentes correspondientes con fecha anterior a su publicación. Las denominaciones contenidas en este TFG referidas en género masculino, se entenderán realizadas y se utilizarán indistintamente en género masculino y femenino allí donde corresponda.

**IMPRESO DE AUTORIZACIÓN DEL  
TRABAJO DE FIN DE GRADO POR EL  
TUTOR**

**Curso 2021/2022**

---

El Dr. D. **Fernando Luis Rosa González**, con D.N.I. 43611314-W, como tutor del estudiante D. **Rubén Morales Rodríguez** en el Trabajo de Fin de Grado titulado

**Protección a la Avifauna de la colisión contra estructuras peligrosas mediante señales ultrasónicas.**

y cotutorizado por el ingeniero industrial D. **Iván Rodríguez Méndez** con D.N.I. 43380782-E,

**da su autorización,**

acreditada por la firma electrónica de este documento, para la presentación y defensa de dicho proyecto, a la vez que confirma que el estudiante ha cumplido con los objetivos generales y particulares que lleva consigo la realización del mismo.

La Laguna, 11 de marzo de 2022.





*“Hijo mío, recuerda lo que te he dicho ya muchas veces, la vida es como una fiesta y no debemos llorar por irnos, sino alegrarnos de haber podido disfrutar de ella. Cuando me marche, ten presente esto y no llores, disfruta de la fiesta que estás viviendo”.*

- Alfonso Morales Calvo.

## Agradecimientos

De manera eterna, a mi padre, un padre ejemplar. Por darme las oportunidades que nunca tuvo, por sus enseñanzas y por su cariño. Por ser un ejemplo de superación y valentía, por esa paz que transmitía mientras tan dura guerra vencía y por asegurarse de que el día en el que San Pedro llamase a su puerta, dejaba en este mundo un hombre capaz y dispuesto de llegar hasta aquí. Por como ya dije, ser un padre ejemplar.

A mi madre, una madre coraje. Por ese beso que todo lo cura, por nunca dejarme sólo, por tu valor inquebrantable. Por el cariño, que hasta el último suspiro, siempre mostraste a mi padre. Gracias por tanto y perdón por darte tan poco a cambio.

A Zaida Brito, por casi 25 años de bonita amistad. Por tu apoyo, por tu sonrisa, por ser siempre un ejemplo y referente para mí en el mundo de la ingeniería. Aunque ambos debamos mirar al cielo para celebrar nuestros exitos, siempre podremos mirar a un lado y encontrarnos el uno al otro.

A mis amigos, algunos me acompañan desde hace años y otros me los regaló la Universidad, pero todos han estado en los momentos en los que los he necesitado sin importar ni el dónde ni el cuándo. Gracias por apoyarme, aguantarme y darme ánimos durante todo este tiempo.

A mi tutor, el doctor Fernando Luis Rosa González, por confiar en mi desde un primer momento, por sus enseñanzas y sin duda alguna, por su paciencia. Para mí es todo un ejemplo de lo que es la profesionalidad y buena docencia, espero que el futuro nos de la oportunidad de volver a trabajar juntos.

A mi co-tutor, D. Iván Rodríguez Méndez, son numerosas las cosas que he de agradecerle, pero creo que todas se resumen en mostrarle mi gratitud por haberme hecho mejor ingeniero. Gracias de corazón por ayudarme incluso en los momentos más difíciles.

Al ingeniero industrial D. Augusto Samuel Hernández Martín, por la ayuda prestada cuándo hubo que “mancharse las manos” y por sus ánimos cuándo no salía nada. Conocerle ha sido un placer, pero trabajar contigo todo un orgullo.

A la Sociedad Española de Ornitología, y a su delegado en Canarias D. Yarci Acosta, por darle alas a este proyecto apoyandolo tanto etica cómo técnicamente.

## Resumen

En el presente Trabajo de Final de Grado, se ha diseñado un instrumento para la protección de la fauna aviar mediante señales ultrasónicas mediante el uso de *software* libre.

En el diseño de este instrumento se han usado técnicas de inteligencia artificial para la detección de aves y de teoría del *beamforming* para focalizar la señal ultrasónica de aviso que genera el instrumento. Este aviso potencialmente protege a las aves de la colisión contra estructuras peligrosas.

El instrumento se ha diseñado en cuatro módulos denominados: Detección, Procesamiento, Aviso y Alimentación. Cada módulo ha sido diseñado, fabricado y comprobado por separado. Para ello se ha utilizado simulaciones, ensayos y montajes en el laboratorio.

Su fabricación ha permitido probar el concepto de trabajo y el alcance de este tipo de instrumentos de aviso ultrasónico y abre la posibilidad de realizar pruebas en instalaciones reales.

## **Abstract**

In this Final Degree Project, a tool has been designed for the protection of avian fauna by means of ultrasonic signals using free software.

In the design of this instrument, artificial intelligence techniques have been used for the detection of birds and the theory of beamforming to focus the ultrasonic warning signal generated by the instrument. This warning potentially protects birds from collision with dangerous structures.

The instrument has been designed in four modules called: Detection, Processing, Warning and Power Supply. Each module has been designed, manufactured and tested separately. Simulations, tests and assemblies in the laboratory have been used for this purpose.

Its manufacture has made it possible to prove the working concept and scope of this type of ultrasonic warning instrument and opens up the possibility of carrying out tests in real installations.



**Universidad  
de La Laguna**

**Escuela Superior  
de Ingeniería y Tecnología**

**Departamento de Ingeniería Industrial**

## **Trabajo de Fin de Grado**

**Protección a la Avifauna de la colisión contra  
estructuras peligrosas mediante señales ultrasónicas.**

### **TOMO I**

#### **Memoria**

**Titulación:** Grado en Ingeniería Electrónica Industrial y  
Automática

#### **Estudiante:**

Rubén Morales Rodríguez

**Tutor:** Fernando Luis Rosa González

**Co-Tutor:** Iván Rodríguez Méndez

11 de marzo de 2022



# Índice general

Agradecimientos . . . . .	5
Resumen . . . . .	7
Abstract . . . . .	8
<b>I Memoria</b>	<b>9</b>
<b>1. Introducción</b>	<b>29</b>
1.1. Objetivos . . . . .	32
1.2. Metodología . . . . .	32
1.3. Estructura de esta memoria . . . . .	34
<b>2. Material y Método</b>	<b>37</b>
2.1. Programas informáticos . . . . .	37
2.1.1. Git . . . . .	38
2.1.2. L <sup>A</sup> T <sub>E</sub> X . . . . .	38
2.1.3. Kile . . . . .	38
2.1.4. Kate . . . . .	39
2.1.5. Python . . . . .	39
2.1.6. KiCAD 5.0.2 . . . . .	40
2.1.7. FlatCAM . . . . .	42
2.1.8. Inkscape . . . . .	42
2.1.9. FreeCAD . . . . .	42
2.1.10. PrusaSlicer . . . . .	43
2.2. Herramientas e instrumentos . . . . .	43
2.2.1. Generador de señales . . . . .	43
2.2.2. Osciloscopio . . . . .	44
2.2.3. Máquina CNC (control numérico computerizado) . . . . .	44
2.2.4. Horno de soldadura . . . . .	45
2.2.5. Impresora 3D . . . . .	45
2.2.6. Otras herramientas . . . . .	45

<b>3. Módulos de detección y aviso</b>	<b>49</b>
3.1. Módulo de detección	50
3.1.1. Red neuronal artificial	50
3.1.2. Modelos de redes neuronales artificiales estudiados	51
3.1.3. Birdwatcher	52
3.2. Módulo de aviso	56
3.2.1. Generación, filtrado y amplificación de señales ultrasónicas	56
3.2.2. Programación ICSP	57
3.2.3. Introducción a la teoría del <i>Beamforming</i>	58
3.2.4. Aplicación de la teoría del <i>Beamforming</i>	59
3.2.5. Propagación y suma de señales	65
<b>4. Módulos de procesamiento y alimentación</b>	<b>69</b>
4.1. Módulo de procesamiento	69
4.1.1. Unidad de procesamiento	70
4.1.2. Comunicación - Bus I2C	71
4.2. Modulo de Alimentación	72
4.2.1. Batería	73
4.2.2. Módulo fotovoltaico	73
4.2.3. Regulador de Carga	74
4.2.4. UPS del módulo de procesamiento	75
<b>5. Ensayos y montajes de laboratorio</b>	<b>77</b>
5.1. Ensayos y montajes del emisor	77
5.1.1. Frecuencias inferiores a 20KHz	77
5.1.2. Frecuencias ultrasónicas	77
5.2. Generación de señales ultrasónicas	80
5.3. Filtrado y amplificación de la señal ultrasónica	81
<b>6. Fabricación del prototipo funcional</b>	<b>87</b>
6.1. Fabricación de la "PCB" del módulo de aviso	87
6.1.1. Corte y realizado de pasantes	87
6.1.2. Transferencia de fotolitos	87
6.1.3. Ataque químico	88
6.1.4. Fijación y conexionado de los componentes	89
6.2. Conexionado y encapsulado de los Emisores implementados	90
6.2.1. Conexionado de los emisores	90
6.2.2. Encapsulado de los emisores	93
6.2.3. Fijación de los encapsulados de los emisores	93
6.3. Diseño y fabricación del soporte de la cámara	94
6.4. Integración mecánica del sistema	95



<i>ÍNDICE GENERAL</i>	13
<b>7. Discusión y mejoras futuras</b>	<b>97</b>
7.1. Módulo de detección	97
7.2. Módulo de procesamiento	97
7.3. Módulo de aviso	98
7.3.1. Módulo de alimentación	98
7.4. Uso del instrumento en otros casos	98
<b>8. Discussion and future improvement</b>	<b>99</b>
8.1. Detection module	99
8.2. Processing module	99
8.3. Warning module	100
8.4. Power supply module	100
8.5. Use of the instrument in other cases	100
<b>II Medición y presupuesto</b>	<b>105</b>
<b>9. Mediciones</b>	<b>107</b>
9.1. Medición de componentes	107
9.2. Medición de horas de diseño y programación	107
9.3. Medición de horas de fabricación y pruebas	108
9.4. Medición de horas de redacción de documentación	108
9.5. Total de horas medidas	109
<b>10. Presupuesto</b>	<b>111</b>
10.1. Presupuesto de los componentes del módulo de alimentación	111
10.2. Presupuesto de los componentes del módulo de procesamiento	112
10.3. Presupuesto de los componentes del módulo de detección	112
10.4. Presupuesto de los componentes del módulo de aviso	113
10.5. Presupuesto del total de los componentes	113
10.6. Presupuesto total del proyecto	114
<b>III Anexos</b>	<b>115</b>
<b>A. Anexo de cálculos</b>	<b>117</b>
A.1. Cálculos del módulo de aviso	117
A.1.1. Filtro “RLC”	117
A.1.2. Amplificador de señal	119
A.2. Cálculos del módulo de alimentación	121
A.2.1. Cálculos justificativos de la batería usada	121

A.2.2. Cálculos justificativos del módulo fotovoltaico elegido . . .	122
<b>B. Caracterización del Piezoeléctrico usado</b>	<b>125</b>
B.1. Análisis de un material piezoeléctrico . . . . .	125
B.1.1. El efecto de los terminales . . . . .	127
B.2. Experimento Realizado . . . . .	127
B.3. Tratamiento de los datos . . . . .	128
B.3.1. Operaciones realizadas con los datos extrídos del ensayo .	128
B.3.2. Resultados gráficos del tratamiento de los datos . . . . .	129
<b>C. Simulaciones realizadas</b>	<b>133</b>
C.1. Simulaciones gráficas de imágenes reales de aves . . . . .	133
C.2. Simulaciones gráficas de imágenes de diferentes animales . . . . .	154
C.3. Simulaciones gráficas de imágenes tratadas . . . . .	164
C.4. Simulaciones gráficas de imágenes complejas . . . . .	174
C.5. Comparativa de resultados obtenidos en las simulaciones realiza- das con imágenes reales . . . . .	184
C.6. Comparativa de resultados obtenidos en las simulaciones realiza- das con imágenes de diferentes animales y paisajes . . . . .	185
C.7. Comparativa de resultados obtenidos en las simulaciones realiza- das con imágenes de aves tratadas . . . . .	186
C.8. Comparativa de resultados obtenidos en las simulaciones realiza- das con imágenes complejas . . . . .	187
<b>D. Datasheets de los componentes utilizados</b>	<b>189</b>
D.1. Módulo de alimentación . . . . .	189
D.1.1. Datasheet de la batería seleccionada . . . . .	190
D.1.2. Datasheet del módulo fotovoltaico seleccionado . . . . .	192
D.1.3. Datasheet del controlador de carga seleccionado . . . . .	194
D.2. Módulo de procesamiento . . . . .	202
D.2.1. Datasheet Raspberry Pi 3 Model B . . . . .	203
D.2.2. Diagrama Raspberry Pi 3 Model B . . . . .	205
D.2.3. Datasheet UPS Raspberry Pi . . . . .	206
D.3. Módulo de detección . . . . .	212
D.3.1. Datasheet Módulo cámara Raspberry Pi V2 . . . . .	213
D.4. Módulo de aviso . . . . .	214
D.4.1. Amplificador operacional TCA0372 . . . . .	215
D.4.2. Regulador de tensión 7805 . . . . .	225
D.4.3. Microcontrolador ATM328 . . . . .	227
D.4.4. Cable de audio . . . . .	231

<b>E. Esquematicos utilizados para el desarrollo del dispositivo</b>	<b>233</b>
E.0.1. Etapa de generación, filtrado y amplificación de señal . . .	234
E.0.2. Esquematico para la comunicación I2C . . . . .	235
<b>F. Información geográfica</b>	<b>237</b>
F.1. Zonas de especial protección avifauna . . . . .	237
F.1.1. Zonas de especial protección avifauna de El Hierro . . . .	238
F.1.2. Zonas de especial protección avifauna de La Gomera . . . .	241
F.1.3. Zonas de especial protección avifauna de La Palma . . . .	246
F.1.4. Zonas de especial protección avifauna de Tenerife . . . .	251
F.1.5. Zonas de especial protección avifauna de Gran Canaria . . .	260
F.1.6. Zonas de especial protección avifauna de Lanzarote . . . .	269
F.1.7. Zonas de especial protección avifauna de Fuerteventura . .	276
F.2. Datos de insolación . . . . .	285
F.2.1. Resultados de insolación a 45 <sup>0</sup> en La laguna . . . . .	286
F.2.2. Resultados de insolación a 45 <sup>0</sup> en Las Palmas de Gran Canaria . . . . .	287
F.2.3. Resultados de insolación optimizados en La laguna . . . .	288
F.2.4. Resultados de insolación optimizados en Las Palmas de Gran Canaria . . . . .	289
<b>G. Scripts utilizados</b>	<b>291</b>
G.1. Generación de señal . . . . .	291
G.2. Adquisición de los datos del Osciloscopio . . . . .	294
G.3. Tratamiento de datos . . . . .	298
G.4. Generación de gráficas . . . . .	301
G.5. Simulaciones acústicas . . . . .	304
G.5.1. Simulaciones de propagación de ondas . . . . .	304
G.5.2. Simulación de propagación del frente de onda . . . . .	305
G.5.3. Simulación del cambio de presión . . . . .	307
G.5.4. Detección de aves . . . . .	310
G.6. Creación del servicio del sistema . . . . .	312
G.6.1. Texto plano del servicio del sistema . . . . .	312
G.6.2. Instrucciones para el servicio del sistema . . . . .	312



# Índice de figuras

1.1. Anagrama del grupo de <i>BFmSD</i> . . . . .	31
1.2. Esquema de la metodología seguida en el desarrollo de este proyecto. . . . .	33
2.1. Imagen del generador de señal HM8030-3 de HAMEG utilizado en los diferentes ensayos y montajes en el laboratorio de BFmSD .	43
2.2. Imagen del Osciloscopio TDS210 de Tektronix utilizado en los diferentes ensayos y montajes en el laboratorio de BFmSD . . . . .	44
2.3. Imagen de la máquina de corte por control numérico disponible en el laboratorio de electrónica de la ESIT de la ULL usada para la fabricación de la “PCB” . . . . .	45
2.4. Imagen del horno T962 disponible en el laboratorio de electrónica de la ESIT de la ULL usada para la fabricación de la “PCB” . . . . .	46
2.5. Imagen de la impresora 3D Ender-3 disponible en el laboratorio de BFmSD . . . . .	47
3.1. Diagrama de bloques del diseño general del instrumento . . . . .	49
3.2. Diagrama de bloques que esquematiza el funcionamiento de este módulo . . . . .	50
3.3. Imagen que esquematiza el funcionamiento de una neurona artificial.	51
3.4. Imagen de un ejemplar Lagópodo Alpino . . . . .	53
3.5. Ave detectada en la imagen . . . . .	53
3.6. Imagen de una pareja de Gaviota patiamarilla . . . . .	54
3.7. Detectada la pareja de aves . . . . .	54
3.8. Imagen de un pájaro sin especificar . . . . .	55
3.9. Ave detectada en la imagen . . . . .	55
3.10. Imagen de un pájaro sin especificar . . . . .	55
3.11. Detectada la pareja de aves . . . . .	55
3.12. Diagrama de bloques de la etapa de generación, filtrado y amplificación de la señal otorgada a los emisores . . . . .	56

3.13. Imagen de la etapa “Step-up” seleccionado en el dimensionado de la etapa de amplificación de señal de este módulo . . . . .	57
3.14. Imagen del anverso del diseño de la PCB . . . . .	58
3.15. Imagen del reverso del diseño de la PCB . . . . .	58
3.16. Situación de emisores y receptor . . . . .	60
3.17. Propagación del sonido a través del aire para el caso de dos emisores	61
3.18. Desplazamiento temporal del pulso . . . . .	61
3.19. Situación de dos emisores y un receptor con una posición dada . .	63
3.20. Comportamiento de las ondas . . . . .	63
3.21. Simulaciones de propagación de onda . . . . .	66
3.22. Simulaciones de propagación del frente de onda . . . . .	67
3.23. Simulaciones de propagación de la presión acústica . . . . .	68
4.1. Diagrama de bloques que esquematiza el funcionamiento del módulo de procesamiento . . . . .	69
4.2. Diagrama del modelo de Raspberry PI seleccionado . . . . .	70
4.3. Diagrama de bloques del módulo de alimentación . . . . .	72
4.4. Imagen de la batería seleccionada . . . . .	73
4.5. Imagen del modulo fotovoltaico seleccionado . . . . .	74
4.6. Imagen del regulador de carga seleccionado . . . . .	74
4.7. Imagen de la UPS seleccionada . . . . .	75
5.1. Diagrama de bloques del ensayo realizado a frecuencias inferiores a los 20KHz . . . . .	78
5.2. Imagen del montaje del ensayo en el espectro audible . . . . .	78
5.3. Diagrama de bloques del ensayo realizado a frecuencias superiores a los 20KHz con dos emisores . . . . .	78
5.4. Montaje del ensayo de emisión a frecuencias superiores a 20KHz .	79
5.5. Conexión y ensamblaje de los “piezos” emisor/receptor . . . . .	79
5.6. Señal obtenida por el “piezo” receptor a 35KHz . . . . .	79
5.7. Imagen de un piezoeléctrico comercial de 12 mm de diámetro. . .	80
5.8. Diagrama de bloques del ensayo realizado a frecuencias superiores a los 20KHz con con un piezoeléctrico comercial como receptor.	80
5.9. Montaje del ensayo de emisión a frecuencias superiores a 20KHz .	81
5.10. Conexión emisor/receptor comercial . . . . .	81
5.11. Señal obtenida por el piezoeléctrico comercial a 35KHz . . . . .	81
5.12. Diagrama de bloques del ensayo realizado para la comprobación de la generación de señales superiores a 20KHz . . . . .	82
5.13. Montaje del ensayo de generación de señal con el microcontrolador ATM328 . . . . .	82
5.14. Señal cuadrada obtenida a la salida del microcontrolador ATM328	82

5.15. Diagrama de bloques del ensayo realizado para comprobar el funcionamiento de la etapa de filtrado y amplificación . . . . .	83
5.16. Montaje en <i>protoboard</i> del sistema de filtrado y amplificación de señal . . . . .	84
5.17. Detalle del montaje en <i>protoboard</i> del sistema de filtrado y amplificación de la señal. . . . .	84
5.18. Detalle de la etapa <i>Step-Up</i> previa al montaje en <i>protoboard</i> del sistema de filtrado y amplificación de señal . . . . .	84
5.19. Comparativa de señales de entrada y de salida en la etapa de filtrado y amplificación . . . . .	85
6.1. Imagen de la <i>PCB</i> tras realizarle los orificios y cortes necesarios .	88
6.2. Imagen del anverso de la <i>PCB</i> tras realizarle la transferencia de fotolitos . . . . .	88
6.3. Imagen del reverso de la <i>PCB</i> tras realizarle la transferencia de fotolitos . . . . .	88
6.4. Imagen del anverso de la <i>PCB</i> tras ser atacada químicamente . . .	89
6.5. Imagen del reverso de la <i>PCB</i> tras ser atacada químicamente . . .	89
6.6. Imagen del anverso de la <i>PCB</i> tras ser soldados sus componentes .	89
6.7. Imagen del reverso de la <i>PCB</i> tras ser soldados sus componentes .	89
6.8. Imagen de la parte delantera de uno de los emisores. . . . .	90
6.9. Imagen de la parte posterior de uno de los emisores. . . . .	90
6.10. Imagen del cable de audio seleccionado . . . . .	91
6.11. Imagen del trenzado de cable fabricado . . . . .	92
6.12. Imagen de la parte delantera de uno de los emisores tras ser conectado. . . . .	92
6.13. Imagen de la parte posterior de uno de los emisores tras ser conectado. . . . .	92
6.14. Imagen detallada del frontal de la cápsula fabricada para los emisores. . . . .	93
6.15. Imagen detallada de la sujeción de la cápsula . . . . .	93
6.16. Imagen detallada del lateral de la fijación de la cápsula. . . . .	94
6.17. Imagen detallada del reverso de la fijación de la cápsula. . . . .	94
6.18. Imagen detallada de la fijación de las cápsulas de los emisores al tubo portante . . . . .	94
6.19. Imagen detallada de la fijación de las cápsulas cableadas al tubo portante . . . . .	94
6.20. Imagen detallada del frontal del soporte de la cámara. . . . .	95
6.21. Imagen detallada del lateral del soporte de la cámara. . . . .	95
6.22. Imagen de la caja antes de ser retirada de la impresora 3D. . . . .	96
6.23. Imagen de la caja antes de ser retirada de la impresora 3D. . . . .	96

6.24. Imagen detallada de la fijación del soporte de la cámara la caja. . . . .	96
6.25. Imagen detallada de la <i>interface</i> de alimentación de la caja. . . . .	96
A.1. Esquemático del filtro “RLC” . . . . .	118
A.2. Esquema de un amplificador operacional en configuración no inversora . . . . .	119
A.3. Esquemático del circuito de amplificación . . . . .	120
B.1. Circuito equivalente de un piezoeléctrico . . . . .	126
B.2. Circuito equivalente de un piezoeléctrico atendiendo a los efectos de sus terminales . . . . .	127
B.3. Esquema del montaje del experimento realizado . . . . .	128
B.4. Gráfica de la impedancia del piezoeléctrico frente a la frecuencia de la señal de entrada . . . . .	130
B.5. Gráfica de la ganancia del piezoeléctrico frente a la frecuencia de la señal de entrada . . . . .	130
B.6. Gráfica de la fase frente a la frecuencia de la señal de entrada . . . . .	131
C.1. Ejemplar de Agateador Euroasiático . . . . .	133
C.2. Detección como “Pájaro” en una imagen de un Agateador Euroasiático . . . . .	134
C.3. Detección como “Pájaro” en una imagen de un Agateador Euroasiático . . . . .	134
C.4. Ejemplar de Aguila Real volando . . . . .	134
C.5. Detección como “Pájaro” en una imagen de un Aguila real . . . . .	135
C.6. Detección como “Pájaro” en una imagen de un Aguila real . . . . .	135
C.7. Ejemplar de un Alcaudón chico . . . . .	135
C.8. Detección como “Pájaro” en una imagen de un Alcaudón chico . . . . .	136
C.9. Detección como “Pájaro” en una imagen de un Alcaudón chico . . . . .	136
C.10. Ejemplar de un Avefría Europea volando . . . . .	136
C.11. Detección como “Pájaro” en una imagen de un Avefría Europea volando . . . . .	137
C.12. Detección como “Pájaro” en una imagen de un Avefría Europea volando . . . . .	137
C.13. Ejemplar de un Alcaudón dosirrojo . . . . .	137
C.14. Detección como “Pájaro” en una imagen de un Alcaudón dosirrojo . . . . .	138
C.15. Detección como “Pájaro” en una imagen de un Alcaudón dosirrojo . . . . .	138
C.16. Ejemplar de un Cernícalo vulgar volando . . . . .	138
C.17. Detección como “Pájaro” en una imagen de un Cernícalo vulgar volando . . . . .	139



C.18. Detección como “Pájaro” en una imagen de un Cernícalo vulgar volando . . . . .	139
C.19. Ejemplar de un Cernícalo vulgar . . . . .	139
C.20. Detección como “Pájaro” en una imagen de un Cernícalo vulgar . . . . .	140
C.21. Detección como “Pájaro” en una imagen de un Cernícalo vulgar . . . . .	140
C.22. Ejemplar de un Charrán común volando . . . . .	140
C.23. Detección como “Pájaro” en una imagen de un Charrán común volando . . . . .	141
C.24. Detección como “Pájaro” en una imagen de un Charrán común volando . . . . .	141
C.25. Ejemplar de un Camachuelo trompetero . . . . .	141
C.26. Detección como “Pájaro” en una imagen de un Camachuelo trompetero . . . . .	142
C.27. Detección como “Pájaro” en una imagen de un Camachuelo trompetero . . . . .	142
C.28. Ejemplar de un Chorlito gris volando . . . . .	142
C.29. Detección como “Pájaro” en una imagen de un Chorlito gris volando . . . . .	143
C.30. Detección como “Pájaro” en una imagen de un Chorlito gris volando . . . . .	143
C.31. Ejemplar de un Correlimos gordo . . . . .	143
C.32. Detección como “Pájaro” en una imagen de un Correlimos gordo . . . . .	144
C.33. Detección como “Pájaro” en una imagen de un Correlimos gordo . . . . .	144
C.34. Ejemplar de un Correlimos de temminck volando . . . . .	144
C.35. Detección como “Pájaro” en una imagen de un Correlimos de temminck volando . . . . .	145
C.36. Detección como “Pájaro” en una imagen de un Correlimos de temminck volando . . . . .	145
C.37. Ejemplar de Escribano hortelano . . . . .	145
C.38. Detección como “Pájaro” en una imagen de un Escribano hortelano . . . . .	146
C.39. Detección como “Pájaro” en una imagen de un Escribano hortelano . . . . .	146
C.40. Ejemplar de Culebrera Europea volando . . . . .	146
C.41. Detección como “Pájaro” en una imagen de una Culebrera Europea volando . . . . .	147
C.42. Detección como “Pájaro” en una imagen de una Culebrera Europea volando . . . . .	147
C.43. Ejemplar de Falaropo picofino . . . . .	147
C.44. Detección como “Pájaro” en una imagen de un Falaropo picofino . . . . .	148
C.45. Detección como “Pájaro” en una imagen de un Falaropo picofino . . . . .	148
C.46. Ejemplar de Ostrero Euroasiático volando . . . . .	148
C.47. Detección como “Pájaro” en una imagen de un Ostrero Euroasiático volando . . . . .	149

C.48. Detección como “Pájaro” en una imagen de un Ostrero Euroasiático volando . . . . .	149
C.49. Ejemplar de Torcecuellos Euroasiático . . . . .	149
C.50. Detección como “Pájaro” en una imagen de un Torcecuellos Euroasiático . . . . .	150
C.51. Detección como “Pájaro” en una imagen de un Torcecuellos Euroasiático . . . . .	150
C.52. Ejemplar de Pagalo pomarino volando . . . . .	150
C.53. Detección como “Pájaro” en una imagen de un Pagalo pomarino volando . . . . .	151
C.54. Detección como “Pájaro” en una imagen de un Pagalo pomarino volando . . . . .	151
C.55. Ejemplar de Vuelvepedras común . . . . .	151
C.56. Detección como “Pájaro” en una imagen de un Vuelvepedras común	152
C.57. Detección como “Pájaro” en una imagen de un Vuelvepedras común	152
C.58. Ejemplar de Quebrantahuesos volando . . . . .	152
C.59. Detección como “Pájaro” en una imagen de un Quebrantahuesos volando . . . . .	153
C.60. Detección como “Pájaro” en una imagen de un Quebrantahuesos volando . . . . .	153
C.61. Imagen de un gato sin especificar . . . . .	154
C.62. Detección como “gato” en una imagen de un gato sin especificar . . . . .	154
C.63. Detección como “gato” en una imagen de un gato sin especificar . . . . .	154
C.64. Imagen de un gato sin especificar . . . . .	155
C.65. Detección como “gato” en una imagen de un gato sin especificar . . . . .	155
C.66. Detección como “gato” en una imagen de un gato sin especificar . . . . .	155
C.67. Imagen de un gato sin especificar . . . . .	155
C.68. Detección como “gato” en una imagen de un gato sin especificar . . . . .	156
C.69. Detección como “gato” en una imagen de un gato sin especificar . . . . .	156
C.70. Imagen de un perro sin especificar . . . . .	156
C.71. Detección como “perro” en una imagen de un perro sin especificar . . . . .	157
C.72. Detección como “perro” en una imagen de un perro sin especificar . . . . .	157
C.73. Imagen de un perro sin especificar . . . . .	157
C.74. Detección como “perro” en una imagen de un perro sin especificar . . . . .	158
C.75. Detección como “perro” en una imagen de un perro sin especificar . . . . .	158
C.76. Imagen de un perro sin especificar . . . . .	158
C.77. Detección como “perro” en una imagen de un perro sin especificar . . . . .	159
C.78. Detección como “perro” en una imagen de un perro sin especificar . . . . .	159
C.79. Imagen de tortugas sin especificar . . . . .	159
C.80. Detección como “tortuga” en una imagen de tortugas sin especificar . . . . .	160
C.81. Detección como “tortuga” en una imagen de tortugas sin especificar . . . . .	160

C.82. Imagen de tortugas sin especificar . . . . .	160
C.83. Detección como “tortuga” en una imagen de tortugas sin especificar	161
C.84. Detección como “tortuga” en una imagen de tortugas sin especificar	161
C.85. Imagen de un paisaje sin especificar . . . . .	161
C.86. Detección en un paisaje sin especificar . . . . .	162
C.87. Detección en un paisaje sin especificar . . . . .	162
C.88. Imagen de un paisaje sin especificar . . . . .	162
C.89. Detección en un paisaje sin especificar . . . . .	163
C.90. Detección en un paisaje sin especificar . . . . .	163
C.91. Imagen de un pajaros sin especificar . . . . .	164
C.92. Detección como “Pájaro” en una imagen de un pajaros sin especificar	164
C.93. Detección como “Pájaro” en una imagen de un pajaros sin especificar	164
C.94. Imagen de un pajaros sin especificar . . . . .	165
C.95. Detección como “Pájaro” en una imagen de un pajaros sin especificar	165
C.96. Detección como “Pájaro” en una imagen de un pajaros sin especificar	165
C.97. Imagen de un pajaros sin especificar . . . . .	166
C.98. Detección como “Pájaro” en una imagen de un pajaros sin especificar	166
C.99. Detección como “Pájaro” en una imagen de un pajaros sin especificar	166
C.100 Imagen de un pajaros sin especificar . . . . .	167
C.101 Detección como “Pájaro” en una imagen de un pajaros sin especificar	167
C.102 Detección como “Pájaro” en una imagen de un pajaros sin especificar	167
C.103 Imagen de un pajaros sin especificar . . . . .	168
C.104 Detección como “Pájaro” en una imagen de un pajaros sin especificar	168
C.105 Detección como “Pájaro” en una imagen de un pajaros sin especificar	168
C.106 Imagen de un pajaros sin especificar . . . . .	169
C.107 Detección como “Pájaro” en una imagen de un pajaros sin especificar	169
C.108 Detección como “Pájaro” en una imagen de un pajaros sin especificar	169
C.109 Imagen de un pajaros sin especificar . . . . .	170
C.110 Detección como “Pájaro” en una imagen de un pajaros sin especificar	170
C.111 Detección como “Pájaro” en una imagen de un pajaros sin especificar	170
C.112 Imagen de un pajaros sin especificar . . . . .	171
C.113 Detección como “Pájaro” en una imagen de un pajaros sin especificar	171
C.114 Detección como “Pájaro” en una imagen de un pajaros sin especificar	171
C.115 Imagen de un pajaros sin especificar . . . . .	172
C.116 Detección como “Pájaro” en una imagen de un pajaros sin especificar	172
C.117 Detección como “Pájaro” en una imagen de un pajaros sin especificar	172
C.118 Imagen de un pajaros sin especificar . . . . .	173
C.119 Detección como “Pájaro” en una imagen de varios pajaros sin especificar . . . . .	173
C.120 Detección como “Pájaro” en una imagen de varios pajaros sin especificar . . . . .	173

C.121	Imagen de una bandada de Aguja colinegra . . . . .	174
C.122	Detección como “Pájaro” en una imagen de una bandada de Aguja colinegra . . . . .	174
C.123	Detección como “Pájaro” en una imagen de una bandada de Aguja colinegra . . . . .	174
C.124	Imagen de una bandada de Avefría Europea . . . . .	175
C.125	Detección como “Pájaro” en una imagen de una bandada de Avefría Europea . . . . .	175
C.126	Detección como “Pájaro” en una imagen de una bandada de Avefría Europea . . . . .	175
C.127	Imagen de una bandada de Combatiente . . . . .	176
C.128	Detección como “Pájaro” en una imagen de una bandada de Combatiente . . . . .	176
C.129	Detección como “Pájaro” en una imagen de una bandada de Combatiente . . . . .	176
C.130	Imagen de un grupo de Aguja colinegra . . . . .	177
C.131	Detección como “Pájaro” en una imagen de un grupo de Aguja colinegra . . . . .	177
C.132	Detección como “Pájaro” en una imagen de un grupo de Aguja colinegra . . . . .	177
C.133	Imagen de un grupo de Charrán patinegro . . . . .	178
C.134	Detección como “Pájaro” en una imagen de un grupo de Charrán patinegro . . . . .	178
C.135	Detección como “Pájaro” en una imagen de un grupo de Charrán patinegro . . . . .	178
C.136	Imagen de un grupo de Gaviota de Auodin . . . . .	179
C.137	Detección como “Pájaro” en una imagen de un grupo de Gaviota de Auodin . . . . .	179
C.138	Detección como “Pájaro” en una imagen de un grupo de Gaviota de Auodin . . . . .	179
C.139	Imagen de un grupo de Gaviota de Auodin . . . . .	180
C.140	Detección como “Pájaro” en una imagen de de un grupo de Gaviota de Auodin . . . . .	180
C.141	Detección como “Pájaro” en una imagen de un grupo de Gaviota de Auodin . . . . .	180
C.142	Imagen de una pareja de Charrán bengalí . . . . .	181
C.143	Detección como “Pájaro” en una imagen de una pareja de Charrán bengalí . . . . .	181
C.144	Detección como “Pájaro” en una imagen de una pareja de Charrán bengalí . . . . .	181
C.145	Imagen de una pareja de Gaviota del Atlántico . . . . .	182

C.146	Detección como “Pájaro” en una imagen de una pareja de Gaviota del Atlántico . . . . .	182
C.147	Detección como “Pájaro” en una imagen de una pareja de Gaviota del Atlántico . . . . .	182
C.148	Imagen de una pareja de Gaviota patiamarilla . . . . .	183
C.149	Detección como “Pájaro” en una imagen de una pareja de Gaviota patiamarilla . . . . .	183
C.150	Detección como “Pájaro” en una imagen de una pareja de Gaviota patiamarilla . . . . .	183



# Índice de tablas

4.1. Comandos para la comunicación I2C. . . . .	72
6.1. Relación del par de trenzado del cableado de los emisores. . . . .	91
9.1. Medición presupuestaria de las horas de diseño y programación . .	108
9.2. Medición presupuestaria de las horas de fabricación y pruebas . .	108
9.3. Medición presupuestaria de las horas de redacción de documenta- ción . . . . .	109
9.4. Medición presupuestaria de horas totales de dedicación . . . . .	109
10.1. Presupuesto de los componentes del módulo de alimentación . .	111
10.2. Presupuesto de los componentes del módulo de procesamiento . .	112
10.3. Presupuesto de los componentes del módulo de detección . . . . .	112
10.4. Presupuesto de los componentes del módulo de aviso . . . . .	113
10.5. Presupuesto del total de los componentes . . . . .	114
10.6. Presupuesto total del proyecto . . . . .	114
10.7. Presupuesto total del instrumento . . . . .	114
C.1. Estado de las aves y niveles de accuracy obtenidos al hacer uso de los modelos preentrenados de inteligencia artificial estudiados en imagenes reales. . . . .	184
C.2. Niveles de accuracy obtenidos al hacer uso de los modelos pre- entrenados de inteligencia artificial estudiados en imágenes con distintos animales. . . . .	185
C.3. Niveles de accuracy obtenidos al hacer uso de los modelos preen- trenados de inteligencia artificial estudiados con imagenes trata- das mediante filtros. . . . .	186
C.4. Niveles de accuracy obtenidos al hacer uso de de los modelos pre- entrenados de inteligencia artificial estudiados con imagenes don- de se muestran varias aves. . . . .	187





# Capítulo 1

## Introducción

Los avances de la tecnología han permitido un mayor disfrute de los servicios y una mejora de la comodidad en todo tipo de actividades, aumentando también el uso de la energía eléctrica. Esto ha hecho que los tendidos eléctricos se hayan convertido en un elemento habitual en el entorno en el que vivimos. Las líneas eléctricas atraviesan montañas y llanuras mientras que las torretas eléctricas aparecen en cualquier rincón o pueblo, alterando el paisaje.

El consumo y la generación de electricidad en España dejará de crecer especialmente por la llegada de la movilidad eléctrica y la expansión de las nuevas instalaciones de energías renovables como eólicas y fotovoltaicas, ya que los compromisos internacionales fijados el 25 de septiembre de 2015, donde los líderes mundiales adoptaron un conjunto de objetivos globales para erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda de desarrollo sostenible, establecen un aumento del uso de las energías renovables. Mientras esto ocurre, las aves sufren los tendidos eléctricos por partida doble. Por un lado, especies de gran envergadura sufren daños por **colisión** al chocar contra los cables por los que discurre la electricidad. Mientras que los postes o apoyos eléctricos suponen otra amenaza para aquellas especies de aves que utilizan estos lugares como oteaderos <sup>1</sup> o lugares de descanso. Debido a la cercanía de cables a su zona de posada, sufren descargas eléctricas, conocidas como **electrocución** [6].

---

<sup>1</sup>Se entiende por oteadero cualquier lugar situado a una altura relativamente privilegiada y con amplio campo de visión que les sirve a las aves de presa y a las propias presas para vigilar su entorno.

Aunque en España existe legislación vigente que nace de la necesidad de evitar la muerte de aves por las causas que se describen anteriormente, como la ley **26/2007** de Responsabilidad Medioambiental, que estipula que el generador de una actividad económica está obligado a evitar que esta provoque un daño al medio ambiente mediante medidas correctoras. Y que de no cumplirse esto, se aplicarán sanciones mediante el principio conocido como «*quien contamina, paga*». O el **RD 1432/2008** aprobado el 29 de agosto de ese año, por el que se establecen medidas para la protección de la avifauna contra la colisión y la electrocución en líneas eléctricas de alta tensión. La situación de mortalidad de las aves se ha visto afectada negativamente con el aumento de los parques eólicos, debido a la influencia que ejercen los aerogeneradores sobre estas. Tal y cómo expone *SEO/BirdLife*, la Sociedad Española de Ornitología, fundada en 1954, en la tercera versión de las directrices para la evaluación del impacto de los parques eólicos en aves y murciélagos [25], “teniendo en cuenta las estimas más conservadoras sobre el porcentaje de superficie muestreada, se podrían establecer cifras de entre 6 y 18 millones de aves y murciélagos muertos en los 17.780 aerogeneradores instalados en España”.

Por todo esto, se podría establecer que los compromisos de la agenda 2030 estarían entrando en conflicto entre sí. Ya que cada uno de los 17 objetivos propuestos tiene metas específicas que deben alcanzarse en los próximos años. Dentro de esos objetivos adoptados, el objetivo número 7 en su segundo apartado propone “De aquí a 2030, aumentar considerablemente la proporción de energía renovable en el conjunto de fuentes energéticas” situación que haría aumentar el número de parques eólicos. Mientras que por otro lado el objetivo número 15 concretamente en su quinto apartado propone “Adoptar medidas urgentes y significativas para reducir la degradación de los hábitats naturales, detener la pérdida de la diversidad biológica y, para 2020, proteger las especies amenazadas y evitar su extinción”.

Atendiendo a esto, se ha desarrollado este proyecto de la mano del grupo de investigación *BFmSD* (**B**ioacústica **F**ísica y **multi-Sensores Distribuidos**) del *Departamento de Ingeniería Industrial* de la *Universidad de La Laguna*, cuyo anagrama se puede observar en la figura 1.1. El desarrollo de este proyecto consiste en el diseño y fabricación de un instrumento capaz de mitigar y/o eliminar el impacto existente entre las aves y ciertas estructuras peligrosas para ellas, como podrían ser los aerogeneradores o las líneas de transporte de alta tensión. Con la creación y uso de este instrumento se podría reducir el número de aves muertas a consecuencia de las circunstancias anteriormente descritas, contribuyendo con ello al cumplimiento de los objetivos de la agenda 2030 y por ende con el desarrollo sostenible.



Figura 1.1: Anagrama del grupo de *BFmSD*

De manera paralela y aunque el objetivo principal de este proyecto es el preservar la vida de la fauna aviar, se han encontrado estudios que desde el año 1999 plantean que la colisión o intrusión en los distintos sistemas de generación y transporte eléctrico, constituyen de por sí un gasto económico para la ciudadanía en cuanto a tiempo de reparaciones y/o desabastecimiento total o parcial de energía eléctrica [19], situación que el objeto de este proyecto podría solventar sin ser su principal línea de estudio. En la misma literatura se hace referencia a que alguna de las medidas que a día de hoy se presentan en el propio **RD 1432/2008** como medidas a implementar con el fin de mitigar las muertes de la fauna aviar, no son del todo efectivas o presentan ciertos aspectos que las convierten en poco eficientes. Situación que sumada a lo anteriormente descrito, hace necesaria la creación e implementación de nuevas medidas que consigan reducir de manera notoria las cifras de aves muertas por estas causas. Hoy en día, existen equipos, como *dtbird*, que utilizan medidas sonoras como dispositivo de alarma para las aves ante las diferentes estructuras que presentan riesgo vital para ellas [3]. Las medidas que se han estudiado utilizan rangos de sonido audibles para el ser humano, es decir inferiores a los 20 KHz. Algunos estudios han determinado que las aves presentan un rango de audición sensible de entre los 25 y 55 KHz, al menos a distancias cortas [27]. Además existen informes técnicos que establecen que las aves pueden terminar acostumbrándose a este tipo de sonidos de frecuencias audibles para el ser humano, como podrían ser sonidos de alarmas que se emplean de manera habitual en distintos sistemas. Por otro lado, al consultar este tipo de información con ornitólogos de la Sociedad Española de Ornitología, afirmaron no tener constancia de medidas sonoras que fuesen efectivas más allá de los denominados “Cañones espantapájaros”. Es por ello por lo que se plantea, partir de la base de estos “cañones espantapájaros” pero haciendo uso de señales de ultrasonido para conseguir salvaguardar la vida de las aves en estas situaciones.

Para llevar a cabo la detección de la presencia de aves se han utilizado redes neuronales, método que ya se ha usado en proyectos de conservación animal me-

diante el uso de fototrampeo <sup>2</sup> en lugares como por ejemplo el parque de Doñana [2]. Por ello, este proyecto se basa en esta técnica de estudio e intenta, utilizando la inteligencia artificial, ir un paso más allá en la conservación animal.

Además de lo descrito anteriormente, es resaltable la importancia del uso de medidas de esta índole en determinados lugares, como por ejemplo, la comunidad autónoma de Canarias, la cual posee una gran extensión de territorio protegido mediante las siglas ZEPA <sup>3</sup> tal y como se puede observar en el anexo F. Hacer uso de medidas como la desarrollada en este proyecto podría facilitar el aprovechamiento de los territorios bajo esta demarcación para distintos fines sin que estos interfieran en la vida de la fauna aviar.

## 1.1. Objetivos

Los objetivos del presente proyecto son los siguientes:

- Diseñar un instrumento capaz de mitigar el impacto de las aves contra estructuras peligrosas.
- Estudiar la aplicabilidad la teoría del *beamforming* en el sistema de emisión.
- Estudiar el uso de redes neuronales para detectar la presencia cercana de aves.
- Fabricar un prototipo funcional que permita realizar una prueba de concepto.

Para la consecución de estos objetivos se ha planteado hacer un uso mayoritario de *software libre*.

## 1.2. Metodología

Para llevar a cabo la realización de este proyecto de una manera satisfactoria, se ha utilizado una metodología que ha servido de guía para abordar ordenadamente las diferentes tareas. Dicha metodología se detalla a continuación y se puede ver de manera gráfica en la figura 1.2.

---

<sup>2</sup>El fototrampeo es la actividad que consiste en fotografiar a los animales en su medio natural con la ayuda de cámaras fotográficas diseñadas para tal fin.

<sup>3</sup>ZEPA: Zona de especial protección avifauna

1. Revisión bibliográfica.
  - a) Estudio del uso de redes neuronales para la detección de aves.
  - b) Estudio de la emisión de ultrasonidos.
  - c) Estudio de la teoría del *beamforming*.
2. Diseño del instrumento.
3. Fabricación de las distintas partes del prototipo funcional.
4. Redacción del proyecto

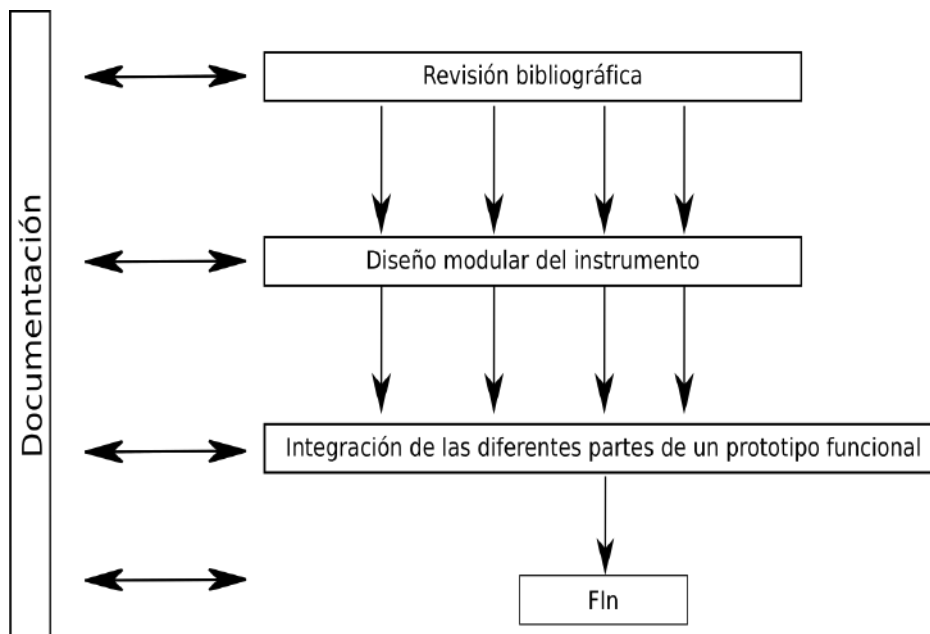


Figura 1.2: Esquema de la metodología seguida en el desarrollo de este proyecto.

El proyecto comenzó con la revisión bibliográfica de los diferentes sistemas que, aplicando procedimientos similares al propuesto, conseguían obtener resultados como los esperados en éste. Además con esta revisión bibliográfica se pretenden adquirir conocimientos lo suficientemente profundos e importantes como para sentar unas bases sólidas que lleven al correcto desarrollo del proyecto.

Posteriormente, se procedió con el diseño del instrumento que se dividió en diferentes módulos, atendiendo a lo que se había estudiado previamente en la revisión bibliográfica. Dicho diseño se vio sometido a cambios durante el propio desarrollo atendiendo a las diferentes cuestiones que se iban abordando, y que,

mientras se descartaban algunos sistemas propuestos en un principio, se aupaban otros que otorgaban rigor y fiabilidad al desarrollo del proyecto.

Una vez planteado el diseño, se procedió con las diferentes simulaciones que mostrasen de manera simulada la aplicación de los diferentes conceptos estudiados, además de la realización de las diferentes pruebas necesarias para proceder con la fabricación del prototipo funcional que permitiese llevar a cabo una prueba de conceptos.

Por último se realizó la recopilación de toda la información que se había ido documentando durante el desarrollo del proyecto, para con esta, elaborar de manera justificada los diferentes apartados de la presente memoria.

### 1.3. Estructura de esta memoria

El presente documento consta de tres partes, un primer tomo denominado como *Memoria*, un segundo tomo que ha sido definido como *Medición y presupuesto* y por último los *Anexos*.

En el primer capítulo de la Memoria, capítulo 1, además de introducir al lector en la temática que redacta el presente documento, se establecen los objetivos del proyecto y la metodología llevada a cabo durante el desarrollo del mismo. En el capítulo 2 se describen los materiales, métodos y las herramientas empleadas para llevar a cabo este proyecto. Por otro lado en el capítulo 3 se establece la propuesta de diseño de los módulos de detección y aviso que conforman el instrumento, mientras que en el capítulo 4 se establece el diseño de los módulos de procesamiento y alimentación. En el capítulo 5 se describen todos los ensayos y montajes de laboratorio realizados previamente a la fabricación. Esta última parte, la parte de fabricación, se explica en el capítulo 6 donde se establece todo lo referente a la fabricación del prototipo de la prueba de conceptos. Por último en el capítulo 7 se discute el diseño y sus mejoras futuras. Además este último capítulo se redacta en la lengua extranjera inglés, en el capítulo 8.

El tomo de Medición y Presupuesto, ha sido dividido en dos capítulos, un primer capítulo, capítulo 9, donde se exponen las mediciones llevadas a cabo en cuanto a tareas realizadas y componentes se refiere. Y un segundo capítulo, el capítulo 10 donde se expone tanto el coste material de los componentes como el amortizable de la maquinaria utilizada para la fabricación del instrumento y el coste total del proyecto ateniendo a las diferentes tareas justificadas en el capítulo 9.

La última parte, denominada Anexos, se encuentra dividida en varias secciones, un primer anexo, Anexo A, denominado anexo de cálculos donde se justifican los componentes que conforman el dispositivo. Un segundo anexo, Anexo B, donde se recoge toda la información del comportamiento eléctrico del emisor seleccio-

nado, otro anexo donde se muestran las simulaciones realizadas con los diferentes modelos preentrenados de inteligencia artificial que se han estudiado, anexo **C**. Un cuarto anexo, Anexo **D**, que contiene los listados de los componentes que debido a sus características técnicas y a su extensión se han extraído del texto de la memoria para que además de justificar lo realizado, sirva de referencia al lector interesado. Otro anexo donde se han adjuntado los esquemáticos referentes al proyecto, Anexo **E**, y otro donde se encuentran las hojas de información sobre las zonas ZEPA en Canarias así como las diferentes hojas de rendimiento fotovoltaico obtenidas, Anexo **F**. Por último se adjunta otro anexo, Anexo **G**, en el que se incluyen los *scripts* elaborados en este proyecto.





# Capítulo 2

## Material y Método

En este capítulo se describen los diferentes materiales y métodos empleados para llevar a cabo este proyecto. Las distintas herramientas informáticas necesarias para el análisis y el diseño del instrumento, se han seleccionado entre las disponibles que usan *software* libre <sup>1</sup>.

En cuanto a las herramientas *hardware* e instrumentos utilizados, estos han sido seleccionados atendiendo a los requisitos de las distintas pruebas y/o ensayos llevados a cabo, así como a los requerimientos de fabricación, seguridad y fiabilidad necesarios.

### 2.1. Programas informáticos

Para el desarrollo de este instrumento, se han usado distintas herramientas de *software* libre. El *software* libre es aquel cuya licencia ofrece a los usuarios las siguientes cuatro libertades [1]:

- **Libertad 0:** la libertad de usar el programa, con cualquier propósito.
- **Libertad 1:** la libertad de estudiar cómo funciona el programa y adaptarlo a tus necesidades.
- **Libertad 2:** la libertad de distribuir copias, con lo que puedes ayudar a tu vecino.
- **Libertad 3:** la libertad de mejorar el programa y hacer públicas las mejoras a los demás, de modo que toda la comunidad se beneficie.

---

<sup>1</sup>El *software* libre es un *software* cuyo código fuente puede ser estudiado, modificado, y utilizado libremente con cualquier finalidad y redistribuido con cambios o mejoras sobre ellas.

### 2.1.1. Git

Para efectuar el control de versiones del proyecto y el seguimiento del mismo por parte del tutor, se ha utilizado un repositorio virtual, gestionado mediante Git. Git es un software de control de versiones que se utiliza para llevar a cabo el registro de los cambios efectuados en los archivos informáticos compartidos de un proyecto dado y coordinar el trabajo que las diferentes personas o equipos realizan sobre estos, almacenándolos en un repositorio remoto y permitiendo a las distintas partes recuperar versiones antiguas del proyecto o actualizar éste con los archivos generados por las otras partes [10]. Git se distribuye bajo licencia GPLv2, aunque algunas de sus partes están cubiertas por licencias distintas pero todas compatibles con GPLv2 [9].

### 2.1.2. L<sup>A</sup>T<sub>E</sub>X

Utilizado para la realización de esta memoria, L<sup>A</sup>T<sub>E</sub>X es un sistema de composición de textos, orientado a la creación de documentos escritos con alta calidad tipográfica. Debido a su versatilidad a la hora de representar expresiones matemáticas, L<sup>A</sup>T<sub>E</sub>X es especialmente usado para la creación de publicaciones científicas, otorgando a estas una calidad tipográfica que se considera adecuada a las necesidades de una editorial científica de primera línea. Dicha versatilidad se debe a que LaTeX está formado por un gran conjunto de macros de TEX, lo que facilita el uso de este lenguaje de composición tipográfica [16]. L<sup>A</sup>T<sub>E</sub>X se distribuye bajo licencia LPPL[17].

### 2.1.3. Kile

Utilizado para redactar y generar el archivo “.pdf” de esta memoria. Kile [15] es un editor de TeX L<sup>A</sup>T<sub>E</sub>X que funciona, vía KDE <sup>2</sup>, en varios OSs y que se distribuye bajo licencia GPLv2. Sus características son las siguientes:

- Autocompletado de comandos LaTeX.
- Coloreado de sintaxis. Kile marca los comandos LaTeX automáticamente y resalta los paréntesis.
- Puede trabajar con múltiples ficheros simultáneamente.
- Ofrece un catálogo de plantillas y patrones para facilitar la creación de documentos.

---

<sup>2</sup>KDE es una comunidad internacional de desarrollo de software libre que produce un entorno de escritorio (actualmente Plasma 5, sucesor de KDE 4), aplicaciones e infraestructura de desarrollo para diversos OSs.

- Plegado de código.

#### 2.1.4. Kate

Utilizado en este proyecto como editor de textos base para la creación de diferentes *scripts*, Kate [13] es un editor de textos KDE cuyas siglas significan KDE *Advanced Text Editor*. Es decir, Editor de Textos Avanzado para KDE. Forma parte del paquete KDEBase desde la versión 2.2 de KDE, del 15 de agosto del 2002. Gracias a la tecnología KParts, que es parte de KDE, es posible utilizar Kate como herramienta de edición en cualquier otra aplicación de KDE.

Kate se distribuye bajo licencia LGPLv2. Entre otras, sus características son:

- Resaltado de sintaxis, extensible mediante archivos XML.
- Búsqueda y reemplazo de texto usando expresiones regulares.
- Seguimiento de código para Python, C++, C, PHP y otros.
- Permite trabajar con múltiples documentos simultáneamente.
- Tutoriales guiados.
- Gestión de archivos.
- Emulador de terminal basado en Konsole.
- Numeración de líneas.

#### 2.1.5. Python

Utilizado en este proyecto para la creación de diferentes *scripts* para el tratamiento de datos y para la realización de las diferentes simulaciones entre otras cosas. Python es un lenguaje de programación interpretado cuya filosofía hace hincapié en una sintaxis que favorezca un código legible. Se trata de un lenguaje de programación dinámico y multiparadigma que soporta tanto programación orientada a objetos como programación imperativa y, en menor medida, programación funcional.

Python es administrado por la *Python Software Foundation* bajo una licencia denominada *Python Software Foundation License* [11]. Compatible con la Licencia Pública General (GPL) de GNU a partir de la versión 2.1.1, e incompatible con algunas versiones anteriores.

### 2.1.6. KiCAD 5.0.2

Siguiendo con la misma filosofía de uso de software libre, se ha utilizado el software **EDA** (del inglés, *Electronic Design Automation*) KiCAD en su versión 5.0.2. KiCAD es un paquete de *software* libre para la automatización del diseño electrónico que permite el diseño esquemático de circuitos electrónicos y la posterior conversión de estos a placa de circuito impreso o *PCB* (del inglés, *Printed Circuit Board*). A su vez, cuenta con una serie de herramientas que permiten, entre otras cosas, crear una lista de materiales, ilustraciones, archivos GERBER<sup>3</sup> y vistas 3D de la PCB y sus componentes [14].

#### Eeschema

Eeschema es la herramienta que permite el diseño de circuitos mediante la interconexión de símbolos electrónicos a través de líneas de conducción. El esquema generado de este modo se almacena en un archivo cuya extensión es “.sch”. Por otro lado, la interconexión de los distintos componentes es almacenada en el archivo *netlist*, de extensión “.net”, que será leído por la herramienta de diseño de *PCB*, *PCBNew*. Eeschema permite, además, editar símbolos mediante su herramienta *LibEdit*, con la que el usuario puede tanto crear símbolos como modificar símbolos existentes, que serán guardados en un archivo denominado, de forma genérica, librería y cuya extensión es “.lib”, que durante el desarrollo del proyecto fue utilizado para crear símbolo de componentes que no se encontraban en la librerías del programa. Con esta extensión, también se almacenan archivos que contienen modelos SPICE<sup>4</sup> de los distintos componentes, especialmente semiconductores y circuitos integrados (en adelante, CIs). Eeschema también cuenta con una interfaz para el simulador ngSPICE, que permite modificar el valor de los componentes sin tener que acudir al esquemático del circuito. Con esta interfaz, el usuario puede generar las gráficas correspondientes a los distintos tipos de análisis que ngSPICE puede ejecutar. El tipo de análisis a realizar viene dado por el comando de simulación y/o líneas de control escritas en el propio esquemático. Una vez definido su tipo y ejecutado el análisis, esta interfaz permitirá observar la evolución de la señal en los nodos del esquemático seleccionados.

---

<sup>3</sup>GERBER es un formato abierto de vectores ASCII para el diseño de PCBs. Es el estándar de facto del software de la industria de *PCBs*, utilizado para el trazado de las imágenes de las placas de circuito impreso: capas de cobre, máscara de soldadura, leyendas, datos de taladrado, etc.

<sup>4</sup>SPICE (del inglés, *Simulation Program with Integrated Emphasis*) es un estándar internacional para la simulación de circuitos electrónicos analógicos desarrollado por la Universidad de California, Berkeley, en 1973 por Donald O. Pederson y Laurence W. Nagel.

### CvPCB

Esta herramienta es necesaria para llevar a cabo la asignación de huellas a los distintos componentes que integran los circuitos electrónicos. Estas huellas se almacenan en un directorio de extensión “.pretty” que contiene los ficheros con extensión “Kicad\_mod” con cada una de las huellas de los diferentes elementos. A cada componente se le asigna una huella o *footprint* cuya imagen, junto con la información de la interconexión de los pines de ésta, será almacenada en el correspondiente *netlist* para ser leído por el editor de PCBs. Por lo general cuando se crea un simbolo como se explicó en el párrafo anterior, es necesario crear su huella, lo cual se realizó en este proyecto con algún componente para poder añadirlo a la *PCB*.

### PCBNew

PCBNew es la herramienta de diseño de circuitos impresos de KiCAD. Una vez asignadas las huellas y generado el netlist, *PCBNew* muestra un entorno donde el usuario puede modificar la posición de los componentes dentro de la placa, definir las dimensiones de ésta y trazar las pistas que unirán eléctricamente todos los componentes del circuito. El diseño realizado será almacenado en un archivo de extensión “.kicad-pcb”. Además, PCBNew permite la creación y la modificación de las *footprints* o huellas, que serán guardadas en archivos de extensión “.mod”, mediante la herramienta *FootprintEditor*. Por otro lado, mediante su visor 3D, *PCBNew* nos ofrece una vista tridimensional de la *PCB* diseñada, lo cual permite tener una idea previa del aspecto físico final de la placa. *PCBNew* permite gestionar hasta 32 capas de cobre, 14 capas técnicas (cortes, serigrafía, máscara de soldadura, etc.) y 4 capas auxiliares (comentarios, dibujos y *ratsnest*). *PCBNew* puede generar archivos con la información correspondiente a estas capas para la fabricación automatizada de las placas. A estos archivos se los conoce como GERBER y poseen distintas extensiones en función de la información contenida en ellos. Estas extensiones se describen a continuación:

- **.gbl**: información de trazado de la cara inferior de cobre.
- **.gtl**: información de trazado de la cara superior de cobre.
- **.gtp**: información de la máscara de pasta de soldar.
- **.gto**: información de la serigrafía.
- **.gts**: información de la máscara de soldadura.
- **.gtr**: información de los bordes de la placa.

### 2.1.7. FlatCAM

Para la creación de ficheros de trayectoria para máquinas de control numérico se ha utilizado FlatCAM. FlatCAM [28] es un software *open source* que permite llevar diseños a un enrutador CNC, o lo que es lo mismo, generar un archivo que una máquina de control sea capaz de interpretar. Este *software* puede abrir ficheros tipo Gerber, Excellon y generar un archivo G-Code que es la extensión de ficheros para este tipo de máquinas. Este software está escrito en Python y funciona sin problemas en la mayoría de las plataformas de modo independiente del sistema operativo.

### 2.1.8. Inkscape

Inkscape [5] es un editor de gráficos vectoriales gratuito y de código libre u *open source*. Inkscape puede crear y editar diagramas, líneas, gráficos, logotipos, e ilustraciones complejas. El formato principal que utiliza el programa es el *Scalable Vector Graphics* (SVG). Este programa tiene como objetivo proporcionar a los usuarios una herramienta de elaboración de gráficos en formato vectorial escalable (SVG) que cumpla completamente con los estándares XML, SVG y CSS2. Este programa se encuentra desarrollado principalmente para los sistemas operativos GNU/Linux, pero es una herramienta multiplataforma y funciona en Windows, Mac OS X, y otros sistemas derivados de Unix.

### 2.1.9. FreeCAD

En el presente proyecto, FreeCAD ha sido utilizada para el diseño de las distintas partes mecánicas del instrumento, esta es una aplicación *open-source* de diseño 3D asistido por ordenador (CAD) e ingeniería asistida por computadora (CAE), para la asistencia en ingeniería mecánica y el diseño de piezas mecánicas. Está basado en *Open CASCADE TECHNOLOGY*<sup>5</sup> (OCCT) y programado en los lenguajes C++ y Python. FreeCAD se distribuye bajo dos licencias distintas, *Lesser General Public Licence*, versión 2 o superior (LGPL2+) para el código fuente y *Creative Commons Attribution 3.0 License* (CC-BY-3.0) para su documentación.

FreeCAD cuenta con varios entornos de trabajo, cada uno de ellos dedicado a tareas específicas dentro del diseño de piezas 3D. Estas tareas van desde la realización de bocetos 2D, mediante los cuales se podrán generar las piezas 3D (extrusión, revolución, etc.), hasta la generación de planos técnicos que cumplen con distintas normas y estándares. Además, FreeCAD lee y almacena varios formatos

---

<sup>5</sup>*Open CASCADE TECHNOLOGY* es una plataforma de desarrollo de *software open-source* para CAD, CAM, CAE, etc. En 3D desarrollada y mantenida por la compañía *Open CASCADE*, subsidiaria de Capgemini.

de open file, tales como STEP, IGES, STL, SVG, DXF, OBJ, IFC, DAE y muchos otros [8]. Por defecto, los modelos 3D creados con FreeCAD se almacenan en archivos cuya extensión es “.fcstd“ pero, a efectos de fabricación, la información de las piezas elaboradas con este *software* se exporta a archivos con extensión “.stl“, formato compatible con el *software* PrusaSlicer, descrito a continuación y utilizado para la fabricación de dichas piezas mediante impresión 3D.

### 2.1.10. PrusaSlicer

PrusaSlicer [22] (antes conocido como Slic3r Prusa Edition o Slic3r PE) es un *software* de rebanado desarrollado internamente basado en el proyecto de código abierto Slic3r. PrusaSlicer es una herramienta de código abierto, rica en características y con actualización frecuente, que contiene todo lo que se necesita para exportar los archivos de impresión para una impresora 3D Original Prusa.

## 2.2. Herramientas e instrumentos

A continuación se exponen y especifican las diferentes herramientas e instrumentos utilizados en el desarrollo y fabricación del objeto de este proyecto.

### 2.2.1. Generador de señales

Se trata de un instrumento capaz de proporcionar señales eléctricas. En concreto, se utiliza para obtener señales periódicas (la tensión varía periódicamente en el tiempo) controlando su periodo (tiempo en que se realiza una oscilación completa) y su amplitud (máximo valor que toma la tensión de la señal). En la figura 2.1 se puede observar el modelo HM8030-3 de HAMEG utilizado en los ensayos de laboratorio.



Figura 2.1: Imagen del generador de señal HM8030-3 de HAMEG utilizado en los diferentes ensayos y montajes en el laboratorio de BFmSD

### 2.2.2. Osciloscopio

El Osciloscopio TDS210 de Tektronix permite visualizar las lecturas de la señal enviada y recibida en una misma línea temporal. Además permite realizar capturas y el volcado de datos a una unidad de memoria mediante su puerto “RS232”. En la figura 2.2 se puede observar el instrumento utilizado para llevar a cabo las medidas de los diferentes ensayos realizados.



Figura 2.2: Imagen del Osciloscopio TDS210 de Tektronix utilizado en los diferentes ensayos y montajes en el laboratorio de BFmSD

### 2.2.3. Máquina CNC (control numérico computerizado)

Para la fabricación de la placa de circuito impreso, además de las tecnologías de fabricación clásicas con químicos, se ha utilizado maquinaria de control numérico. El CNC o control numérico por computadora es un sistema que permite el control de la posición de un elemento montado en el interior de una máquina-herramienta mediante un software especialmente diseñado para ello. Su funcionamiento está basado en el posicionamiento sobre los ejes X, Y, Z. Gracias a esto una misma pieza se puede taladrar, cortar, roscar, fresar, o desbastar en todos sus planos de forma totalmente automática. Para realizar esto se han utilizado los ficheros “.gcode” obtenidos del programa “KiCad” al diseñar la *PCB*. La maquinaria para ejecutar estos ficheros puede observarse en la figura 2.3, en dicha figura puede verse la CNC4030Z, de Shenzhen Scotle Technology Ltd., disponible en el laboratorio de electrónica de la escuela superior de ingeniería y tecnología de la Universidad de la Laguna. Esta CNC tiene una resolución de aproximadamente  $21\mu\text{m}$  en cada uno de sus ejes. Su *spindle* está refrigerado por una bomba de agua, lo cuál proporciona una buena refrigeración a la maquina durante largos periodos de trabajo.





Figura 2.3: Imagen de la máquina de corte por control numérico disponible en el laboratorio de electrónica de la ESIT de la ULL usada para la fabricación de la “PCB”

#### 2.2.4. Horno de soldadura

Para la soldadura de componentes SMD se ha utilizado un horno de soldadura T962 que se muestra en la figura 2.4. Este dispositivo tiene una potencia de 800W lo cuál le permite llegar a temperaturas de hasta 280 °C para las operaciones de soldadura.

#### 2.2.5. Impresora 3D

Para la creación de las piezas del prototipo funcional, se ha utilizado la tecnología de impresión 3D. El modelo de impresora utilizado ha sido la Ender-3 que se muestra en la figura 2.5. La Creality Ender-3 es una impresora 3D con un volumen de construcción de 220 x 220 x 250 mm, una cama magnética y una ruta de filamento estrecha que facilita el trabajo con materiales flexibles.

#### 2.2.6. Otras herramientas

Por otro lado, es resaltable el hecho de especificar las principales herramientas necesarias para la fabricación del instrumento, enumerándolos y describiendo su función a continuación.



Figura 2.4: Imagen del horno T962 disponible en el laboratorio de electrónica de la ESIT de la ULL usada para la fabricación de la “PCB”

- Soldador eléctrico de precisión: soldar los componentes de la PCB.
- Cortarrentes: cortar y pelar cables.
- Alicates: sujeción de componenetes.
- Guillotina: recortar la PCB.
- bisturí: eleminar partes sobrantes y/o corregir defectos de la PCB.
- Prensa de banco: sujeción de la PBC para montaje o soldadura de componentes.
- Tanque de grabado: atacado ácido de la PCB.
- Rodillo fusor térmico: impresión de fotolitos en las PCB.
- Guantes de seguridad (latex/nitrilo): evitar intoxicación o quemadura química durante la limpieza y el atacado químico de la PCB.
- Gafas protectoras: evitar daños oculares por salpicadura o vapor de productos químicos.
- Desoldador: corregir defectos de soldadura o retirar componentes defectuosos.
- Multímetro: medición de señales eléctricas y diagnóstico de fallos.
- Protoboard: ensayo de los circuitos.
- Sondas de medida y de generación de señal.



Figura 2.5: Imagen de la impresora 3D Ender-3 disponible en el laboratorio de BFmSD

- Ordenador Personal (PC): imprescindible para la automatización informática de los diseños y el almacenamiento de los datos del proyecto, así como para editar y generar documentos relativos al mismo.



## Capítulo 3

### Módulos de detección y aviso

Atendiendo a la metodología seguida durante el desarrollo de este proyecto, se plantea el diseño modular de las diferentes partes del instrumento. En la imagen 3.1 se puede observar el diagrama de bloques del diseño general del instrumento con la relación entre los distintos módulos. Se puede ver que el módulo de alimentación es transversal y sirve para alimentar a todas las partes del instrumento. Los tres módulos principales se conectan de izquierda a derecha para mostrar el flujo de datos en el instrumento.

El primer módulo realiza la detección del ave y comunica al módulo de procesamiento las coordenadas del ave en el mundo. El módulo de procesamiento calcula los retrasos de las diferentes emisiones y los envía al último módulo de aviso que realizará el aviso por medio de una emisión de señales ultrasónicas coordinadas.

En este capítulo se describen los módulos de detección y de aviso y en el siguiente se abordará el módulo de procesamiento así como el de alimentación.

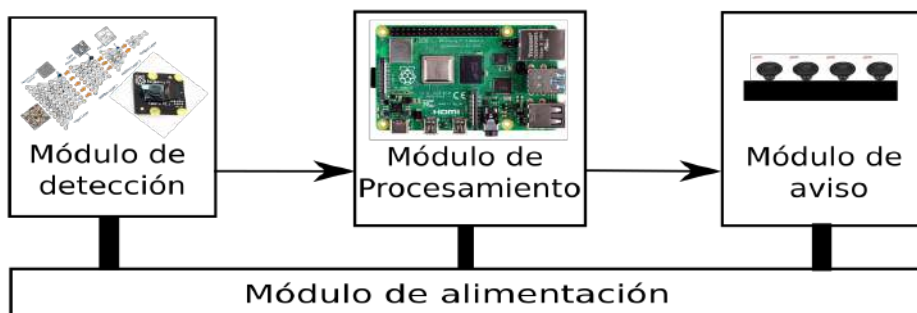


Figura 3.1: Diagrama de bloques del diseño general del instrumento  
Diagrama elaborado en InkScape.

### 3.1. Módulo de detección

Este módulo es el encargado de realizar la captura de imágenes y determinar, mediante el uso de redes neuronales, si existe o no presencia de aves en las mismas. Para el diseño de este módulo se han estudiado las diferentes posibilidades de cámaras a instalar, y se ha decidido que sea un modelo de cámara “NOIR”, cuya información técnica se encuentra en el anexo D, atendiendo a que este tipo de cámaras son capaces de captar imágenes en situaciones de luz desfavorable aumentando con ello el rango de acción diario del instrumento. Aunque en un principio se estudiaron ejemplos donde el sistema de detección se basaba en el uso de sensores “PIR”<sup>1</sup> [20], tras realizar una revisión bibliográfica más extensa, se observó que el empleo de inteligencia artificial, para en este caso detectar la presencia de aves, presentaba un mayor grado de eficiencia. Por ello, se estudiaron diferentes modelos de inteligencia artificial que fuesen capaces de satisfacer este requerimiento de diseño. Además de esto, el sistema implementado en el dispositivo es capaz de determinar el espacio en el que se encuentra el ave detectada dentro del *frame* capturado. Es decir, que permite determinar los píxeles en los que se encuentra el ave detectada, de manera que, instalando dos cámaras con las mismas características se podrían determinar las coordenadas reales de la posición del ave.

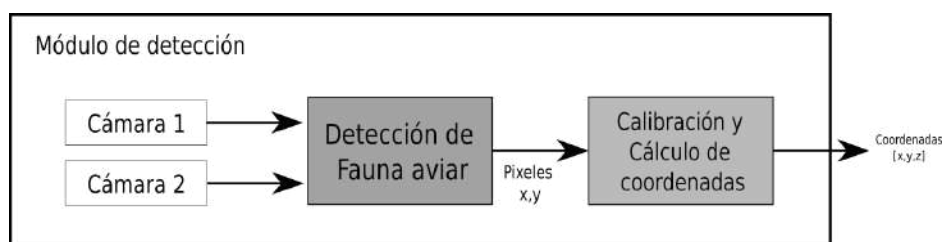


Figura 3.2: Diagrama de bloques que esquematiza el funcionamiento de este módulo

Diagrama elaborado en InkScape.

#### 3.1.1. Red neuronal artificial

Una Red Neuronal Artificial (RNA) es un modelo matemático inspirado en el comportamiento biológico de las neuronas y en cómo se organizan formando la estructura del cerebro. El cerebro puede considerarse un sistema altamente complejo, donde se calcula que hay aproximadamente 100 mil millones ( $10^{11}$ )

<sup>1</sup>El sensor PIR es un tipo de sensor electrónico que mide la luz infrarroja radiada de los objetos situados en su campo de visión. Usado principalmente en los detectores de movimiento

de neuronas en la corteza cerebral de un humano y que forman un entramado de más de 500 billones de conexiones neuronales donde cada neurona puede llegar a tener 100 mil conexiones, aunque la media se sitúa en 5000 [23]. En la figura 3.3 se puede ver de manera esquematizada el comportamiento neuronal descrito anteriormente.

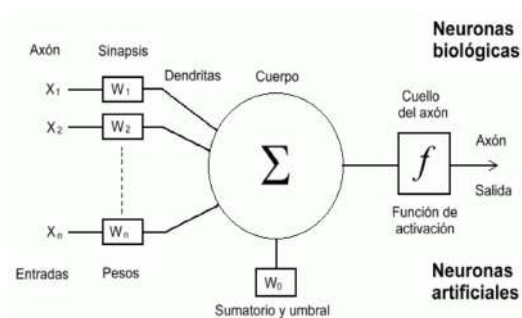


Figura 3.3: Imagen que esquematiza el funcionamiento de una neurona artificial. Imagen obtenida de [23].

Una de las tareas llevadas a cabo con la inteligencia artificial es el reconocimiento de objetos, esto se divide en diferentes etapas. Una primera etapa de extracción de características y una segunda etapa para su clasificación [24].

Los diferentes métodos de extracción de características varían según las características que se pretenden diferenciar y se aplican sobre píxeles<sup>2</sup> y conjuntos de píxeles de la imagen. En el caso del modelo implementado en el dispositivo, el píxel cobra una vital importancia ya que ser conocedor del lugar en el que se encuentra nuestro objeto, en este caso un ave, permite realizar de manera más eficiente la señal de aviso, ya que se podría actuar sobre un punto en concreto.

La etapa de clasificación compara las características que el sistema tiene asociadas para cada clase de objeto con las características extraídas de la imagen, devolviendo la **probabilidad** o *accuracy* para cada objeto que pertenece a una clase.

### 3.1.2. Modelos de redes neuronales artificiales estudiados

Para llevar a cabo el diseño de este módulo, se han estudiado diferentes modelos de reconocimiento de objetos en imágenes: Vgg16, Detectron2 y Birdwatcher. Se implementó en el dispositivo Birdwatcher que demostró cumplir con los

<sup>2</sup>El píxel es la menor unidad espacial de la que esta compuesta la imagen digital y contiene información del color, la luminancia, la forma de los objetos y otras propiedades

requerimientos y se podía adaptar al *hardware* seleccionado. A continuación se describen los modelos estudiados y las simulaciones del modelo seleccionado.

### Vgg16

En un principio se estudió la posibilidad de implementar este modelo preentrenado, ya que en la revisión bibliográfica se pudo observar que se había estudiado su uso en la detección de aves exóticas [21]. Tras estudiar su arquitectura, y realizar algunas pruebas, se pudo observar que este modelo preentrenado era capaz de detectar algunas especies de aves con gran exactitud. Aunque su principal problema para ser implementado en el instrumento que se diseña, era la necesidad de ser “re-entrenada” para poder detectar un mayor número de especies. Esto se podría realizar, e incluso profundizar en el tratamiento de datos con esta red, pero se salía del objetivo y tiempo de este proyecto.

### Detectron2

Atendiendo a lo estudiado en la revisión bibliográfica, se comprobó como se había implementado un modelo preentrenado de inteligencia artificial para la detección de aves en las líneas de alta velocidad del “AVE” [24].

Para esto se utilizó “Detectron2” la biblioteca de “Facebook AI Research” que proporciona algoritmos de detección y segmentación. Es el sucesor de modelos anteriores como “Detectron” y “maskrcnn-benchmark”. Es compatible con una serie de proyectos de investigación de visión artificial y aplicaciones de producción en Facebook. Según su página web, incluye una serie de características como las capacidades de segmentación panóptica, *Densepose*, *Cascade R-CNN*, cuadros delimitadores girados, *PointRend*, *DeepLab* junto con otras. Además se utiliza como una biblioteca para apoyar la construcción de proyectos de investigación. Los modelos se pueden exportar al formato “TorchScript” o al formato “Caffe2” para su uso [29].

Este sistema fue estudiado y se simuló diferentes casos con él, obteniendo valores de *accuracy* bastante elevados, pero presentaba incompatibilidad a la hora de ser adaptado a las necesidades de este proyecto. No sólo en lo que a su desarrollo se refiere, sino también en lo referente a su implementación en el *hardware* seleccionado.

### 3.1.3. Birdwatcher

En la revisión bibliográfica, se encontró un proyecto, llamado *Birdwatcher* que utilizaba la red “MobileNet-SSD” con 21 clases generales para detectar aves. Este es un algoritmo que se utiliza para la detección en tiempo real donde mediante



una cámara web se pueda detectar el objeto en una secuencia de vídeo. Para ello, se utiliza un módulo de detección de objetos que puede detectar lo que hay en la secuencia de vídeo. En este caso se combina MobileNet y el marco SSD para obtener un método rápido y eficiente basado en el aprendizaje profundo de detección de objetos [30].

Al realizar diferentes simulaciones de prueba con este modelo, se pudo observar la eficiencia de sus resultados, aunque estos no eran mejores que los obtenidos al realizar las simulaciones con el sistema de “Detectron2”, su implementación era posible y requería menos recursos de hardware para su funcionamiento, facilitando con ello su implementación. Por ello se decidió que fuese este el modelo implementado en el instrumento.

### Simulaciones referentes a la detección de aves

Para poder determinar cual de los modelos de red neuronal estudiados se implementaba en el sistema, se realizaron una serie de simulaciones para determinar su efectividad a la hora de detectar un ave. En este caso aunque el modelo esta “preentrenado” para poder detectar un determinado número de “objetos”, las simulaciones se centraron en la posibilidad de detectar aves, ya que es lo que realmente cobra importancia en la implementación de este modelo en el instrumento. No obstante también se realizaron pruebas en diferentes situaciones con el fin de tener mayor información sobre el comportamiento del sistema.

Como se puede observar en las figuras 3.5 y 3.7 el modelo de inteligencia artificial implementado otorga una detección con un valor de *accuracy* bastante elevado superando en ambos casos el 99 % de confianza en las detecciones.



Figura 3.4: Imagen de un ejemplar Lagópodo Alpino

Imagen Cedida por SEO/BirdLife.

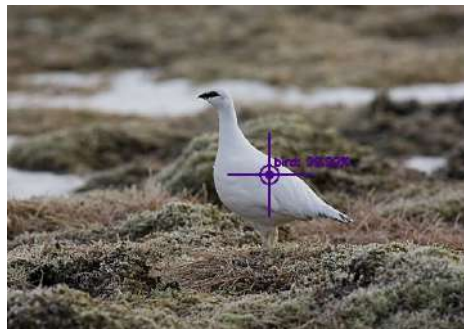


Figura 3.5: Ave detectada en la imagen

Imagen obtenida al hacer uso del modelo “Birdwatcher”.



Figura 3.6: Imagen de una pareja de Gaviota patiamarilla

Imagen Cedida por SEO/BirdLife.

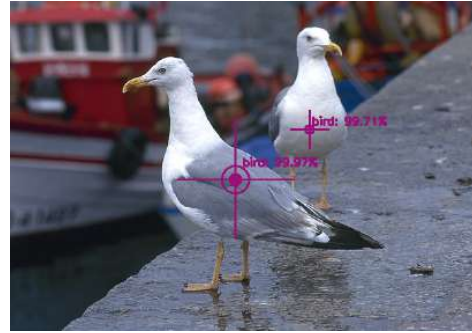


Figura 3.7: Detectada la pareja de aves

Imagen obtenida al hacer uso del modelo "Birdwatcher".

### Simulaciones de detección ante situaciones desfavorables

Con el fin de comprobar la eficiencia del modelo de red neuronal implementado, en situaciones en las que la captación de imágenes se realice en condiciones desfavorables, se ha procedido a hacer un tratamiento a diferentes imágenes de pájaros elegidas de manera estocástica, con una serie de filtros que oscurezcan la imagen para observar el comportamiento de dicho modelo al realizarse la captura de imágenes mediante cámaras sin filtro de infrarrojos. Con el objetivo de validar esto, se ha desarrollado en el Anexo C un mayor conjunto de simulaciones que atienden a diferentes situaciones, donde, se comparan además los resultados obtenidos al utilizar los diferentes modelos de redes neuronales estudiados.



Figura 3.8: Imagen de un pájaro sin especificar

Imagen obtenida de "Pexels".



Figura 3.9: Ave detectada en la imagen

Imagen obtenida al hacer uso del modelo "Birdwatcher".

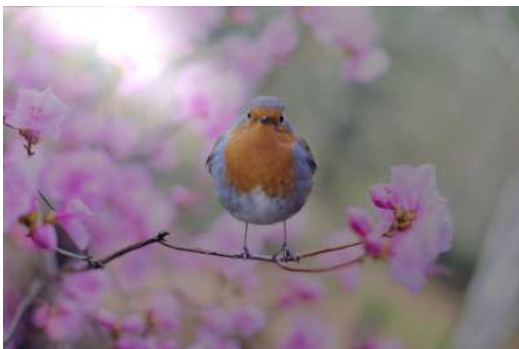


Figura 3.10: Imagen de un pájaro sin especificar

Imagen obtenida de "Pexels".



Figura 3.11: Detectada la pareja de aves

Imagen obtenida al hacer uso del modelo "Birdwatcher".

## 3.2. Módulo de aviso

Este módulo es el encargado de emitir la señal de aviso, depende del módulo de procesamiento, ya que será este el encargado de indicarle los retrasos correspondientes a la presencia de un ave cercana. Dicha señal de aviso se realiza mediante señales ultrasónicas en las que el tiempo de emisión es fundamental para poder hacer un correcto uso de la teoría del *beamforming* y hacer que el receptor reciba las señales de manera potenciada manteniendo los emisores en una posición fija.

### 3.2.1. Generación, filtrado y amplificación de señales ultrasónicas

Al recibir los comandos de activación por parte del módulo de procesamiento, mediante el microcontrolador “ATM328P”, se generan 4 señales cuadradas periódicas de 31250 Hz. Dichas señales pasan por un filtro “RLC” paso bajo de 2<sup>o</sup> orden. Este tipo de filtro como su nombre indica, solo permite el paso de frecuencias bajas y atenúa las frecuencias altas. No se bloquean las frecuencias bajas para poder utilizar este mismo filtro con todas las frecuencias bajas que puede proporcionar el microcontrolador.

Este filtro está compuesto por tres elementos, una resistencia, una bobina y un condensador. Se conoce como pasivo ya que así se caracterizan los elementos que lo componen, y es de segundo orden por que contiene dos elementos reactivos (un condensador y una bobina). En este filtro los valores del factor de calidad “Q” y el valor de la constante “k” dependen de la aproximación que se use [7]. Atendiendo a esto se ha diseñado este filtro en base a los cálculos desarrollados en el Anexo A, para con este convertir la forma de dichas señales a forma senoidal. Una vez se ha obtenido esto, las señales pasan por un proceso de amplificación, donde se ha usado el amplificador operacionales “TCA0372”, tal y como se expone en el Anexo A, para obtener las señales de salida que llegan a los emisores. De manera gráfica se puede observar esto en el diagrama de bloques de la figura 3.12.

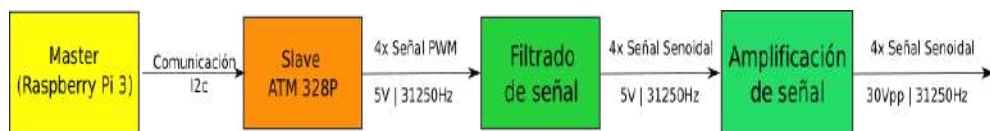


Figura 3.12: Diagrama de bloques de la etapa de generación, filtrado y amplificación de la señal otorgada a los emisores

Diagrama elaborado en InkScape.

Es resaltable el uso en este modulo de una etapa “Step-Up” como la que se puede observar en la imagen 3.13. Se ha recurrido al uso de esta para poder alimentar la parte de amplificación a 24V desde una fuente de alimentación de 12V.



Figura 3.13: Imagen de la etapa “Step-up” seleccionado en el dimensionado de la etapa de amplificación de señal de este módulo

Para comunicar el módulo de aviso con el de detección se ha planteado hacer uso del protocolo de comunicación “I2C” el cual presenta unas características en cuanto a sencillez, fiabilidad y modularidad que le otorgan rigor para ser el sistema implementado. Por otro lado, este modulo se diseña para que cuente con un microcontrolador embebido en una *PCB*, por lo que requiere ser programado mediante programación “ICSP”.

El diseño de esta *PCB* se ha llevado a cabo mediante el uso del *software* KiCad expuesto en detalle anteriormente. El diseño de este circuito impreso se puede observar en las figuras 3.14 y 3.15. Dicho diseño se ha obtenido atendiendo al esquemático adjunto en el Anexo E.

### 3.2.2. Programación ICSP

La programación serial en circuito (ICSP por las siglas del inglés : *In-Circuit Serial Programming*), es la habilidad de algunos dispositivos lógicos programables, microcontroladores y otros circuitos electrónicos, de ser programados mientras están instalados en un sistema completo, en lugar de requerir que el chip sea programado antes de ser instalado dentro del sistema [12]. Por ello se utiliza este método a la hora de programar el microcontrolador que forma parte del módulo de evasión, ya que al conformar un circuito impreso se ha considera esta como la



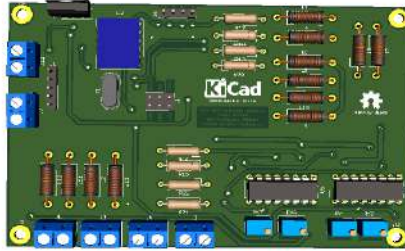


Figura 3.14: Imagen del anverso del diseño de la PCB  
Imagen obtenida mediante el uso de KiCad.

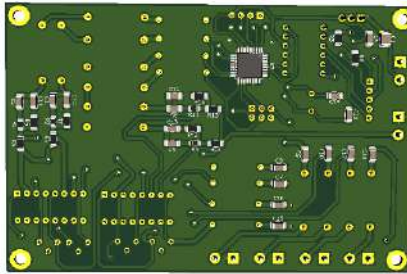


Figura 3.15: Imagen del anverso del diseño de la PCB  
Imagen obtenida mediante el uso de KiCad.

forma más idónea de programación. Para realizar esto es necesario un conector adicional denominado “programador”.

### 3.2.3. Introducción a la teoría del *Beamforming*

Aunque a la hora de realizar la revisión bibliográfica, la literatura estudiada, dictaba en su mayoría las técnicas y procesos usado para la adquisición de señal atendiendo a la teoría del *beamforming*, se ha podido extraer de dicha literatura los conocimientos necesarios para sentar las bases sobre como implementar esta teoría en el instrumento propuesto. En concreto en esta sección se introduce la teoría citada anteriormente, así como al diseño del *array* y al *Dealy and Sum (DAS)*.

El término *beamforming* es un anglicismo, el cual resulta de la unión de las voces inglesas *beam* (traducido al castellano, rayo, haz) y *forming* (cuya traducción es formación). Este término es empleado en ingeniería de comunicaciones, acústica y procesamiento de la señal para designar la transmisión y recepción direccional de señales, siendo también conocido como “método de antena microfónica”, “te-

lescopia acústico” o “cámara acústica”. Este anglicismo deriva del hecho de que los primeros filtros espaciales eran diseñados para formar diagramas de radiación muy directivos, los denominados *pencil beams* [18].

La idea fundamental en la que se basa esta técnica es el empleo de patrones directivos en sistemas receptores, el cual asegura la recepción de señales provenientes de una ubicación dada, atenuando el resto de señales, que a la vista del receptor, no constituyen más que interferencias indeseadas. Por ello, se puede afirmar que el método entra dentro de la categoría de filtrado espacial acompañada de otras técnicas, como filtrado temporal y frecuencial.

En el caso de este proyecto el receptor es el ave detectada y el patrón de radiación se genera desde los emisores instalados en una posición fija.

### **Diseño del *array***

A la hora de diseñar el *array* de emisores para un receptor lejano se deben de tomar en cuenta una serie de aspectos, como la frecuencia, la anchura del lóbulo principal, la resolución espacial, los lóbulos secundarios o la distancia [18].

En base a esto se procede con el diseño y simulación del *array* de emisión. En este capítulo se puede observar como se obtiene el cálculo de los diferentes retrasos a aplicar para que la señal llegue con la máxima potencia al receptor, además de estos cálculos se adjuntan una serie de simulaciones que justifican esto, tanto en la propagación de onda como la suma de las mismas en el punto deseado.

### ***Delay And Sum (DAS)***

Según se pudo estudiar en la literatura encontrada; la localización de fuentes de ruido mediante el método retardo y suma (Delay And Sum, DAS) es, conceptualmente, muy sencilla. Todos los micrófonos registran a la vez, y se aplica un retardo a cada una de las señales registradas en función del posible ángulo de incidencia.

Atendiendo a esto y entendiendo que en el desarrollo de este proyecto no se trabaja con la recepción si no con la emisión de sonido, se propone aplicar un retardo en cada una de las señales emitidas con el fin de potenciar el sonido en el punto en el que se encuentra el receptor. .

### **3.2.4. Aplicación de la teoría del *Beamforming***

En esta sección del capítulo se tratan y exponen tanto las hipótesis planteadas cómo los cálculos y simulaciones realizadas para justificar el empleo de la teoría del *beamforming* en el módulo de aviso del instrumento.

### Disposición de emisores

En primer lugar, se ha seleccionado una localización de emisores para maximizar el efecto de la señal de ultrasonido generada. Se comienza analizando una situación lineal de emisores tal como se muestra en la figura 3.16. Se puede ver en trazo con línea discontinua las trayectorias de propagación de la señal emitida desde los emisores al receptor. Cada emisor estará situado en la posición del espacio tridimensional,  $\vec{p}_n$ , dónde  $n$  indica el número del emisor. Se representa la posición del receptor como,  $\vec{x}$ , que de igual forma, será un punto en el espacio tridimensional. La distancia entre los distintos emisores se supone constante,  $d$ .

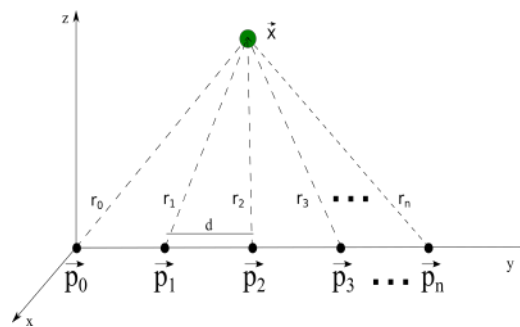


Figura 3.16: Situación de emisores y receptor

Imagen elaborada con Inkscape.

Se puede obtener la distancia entre el emisor  $n$  y el punto de estudio,  $r_n$  mediante la ecuación 3.1.

$$r_n = \sqrt{(x - p_{nx})^2 + (y - p_{ny})^2} \quad (3.1)$$

Comenzaremos planteando la situación en la que solo existen dos emisores. Se conoce que cuándo la superficie de un pulso esférico se propaga de manera radial, este ejerce una fuerza sobre el fluido en contacto, por lo que este es desplazado desde su posición de equilibrio dando como resultado una propagación uniforme en todas las direcciones de ondas esféricas. En la figura 3.17 se observa como se propaga el sonido a través del aire<sup>3</sup> en forma de “circunferencia”. Ya que aunque la onda esférica se desplace con un frente de onda esférico en las tres dimensiones en un medio homogéneo este es unidimensional [26].

<sup>3</sup>A la hora de realizar esta representación se tienen en cuenta condiciones de idealidad como por ejemplo la homogeneidad del aire o la temperatura de  $20^{\circ}\text{C}$  a la que se establece la velocidad de propagación del sonido en  $340 \frac{\text{m}}{\text{s}}$



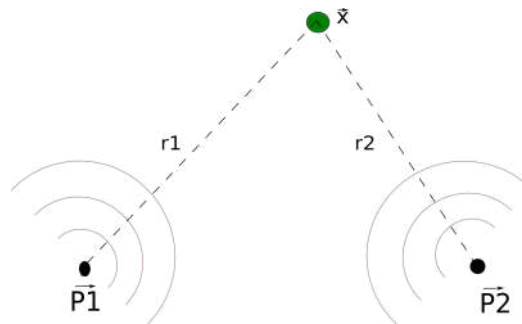


Figura 3.17: Propagación del sonido a través del aire para el caso de dos emisores  
Imagen elaborada con InkScape.

### Desplazamiento temporal del pulso

Una vez es emitida la señal, gráfica e idealmente hablando, ocurrirá algo similar a lo que se muestra en la figura 3.18, en la que se puede observar un desplazamiento temporal de la señal.

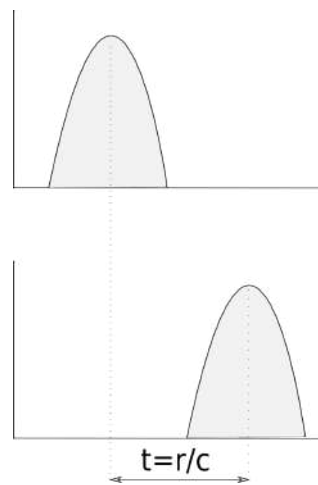


Figura 3.18: Desplazamiento temporal del pulso  
Imagen elaborada con InkScape.

Atendiendo a esto y en base a lo explicado anteriormente podría establecerse que la ecuación 3.2 define la señal.

$$f(t) = f\left(t - \frac{r}{c}\right) \quad (3.2)$$

### Desarrollo analítico en el plano y en el espacio

Como se cita anteriormente en condiciones de idoneidad se puede establecer que la señal se propaga en forma de circunferencia cuándo se habla de su transmisión en el plano, pero se debe tener en cuenta que la señal se propagará de forma esférica en el espacio.

Se conoce que la longitud de una circunferencia se obtiene como se establece en la ecuación 3.3 y que tal y como se muestra en la ecuación 3.4 se puede obtener el valor de la superficie de una esfera.

$$L = 2 \cdot \pi \cdot r \quad (3.3)$$

$$S = \frac{4}{3} \cdot \pi \cdot r^2 \quad (3.4)$$

Entendiendo que para obtener la energía de la señal, se debe partir de la fórmula de la energía que se muestra en la ecuación 3.5. Se podría establecer que la distribución de energía de la señal emitida con simetría circular en el plano se puede escribir con la ecuación 3.6, mientras que para obtener dicha energía en el espacio con simetría esférica se debe atender a la ecuación 3.7.

$$dE(t) = f^2(t)dt \quad (3.5)$$

$$dE(t) = \frac{f^2(t - \frac{r}{c})}{2\pi r} dt d\theta \quad (3.6)$$

$$dE(t) = \frac{f^2(t - \frac{r}{c})}{4\pi r^2} dt d\theta d\phi \quad (3.7)$$

Una vez planteado esto, y sabiendo que lo que se requiere es “sumar” las señales de los emisores de forma que no se interfieran entre ellas, surge el planteamiento de poder obtener el momento “*t*” *perfecto*, con el objetivo de aplicar

el efecto de retraso y suma “DAS”<sup>4</sup> y conseguir el objetivo planteado desde un principio, maximizar el efecto del sonido emitido en el instante en el que llega al receptor.

Para ello se plantea que la posición del receptor puede ser cualquiera, totalmente aleatoria, como se puede observar en la figura 3.19, en la cuál se ha elegido una posición cualquiera. En esta figura se sigue particularizando el caso en el que sólo se poseen dos emisores con el fin de simplificar la explicación.

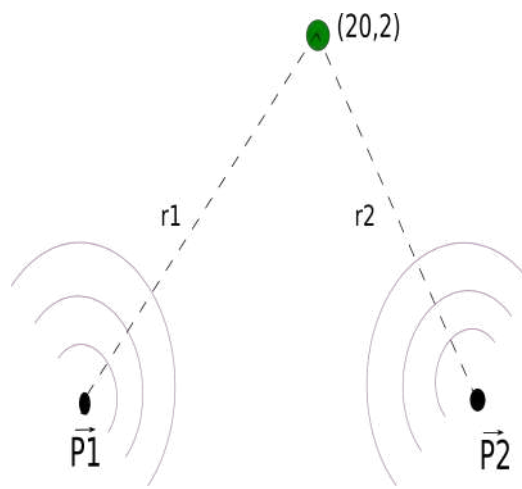


Figura 3.19: Situación de dos emisores y un receptor con una posición dada  
Imagen elaborada con InkScape.

Cómo se plantea anteriormente las ondas de la señal tienen un aspecto muy similar al de una circunferencia, esto implica tener en cuenta que dichas ondas dependiendo de como se establezcan físicamente pueden sumarse o restarse, algo crucial en el desarrollo de este planteamiento. Cómo se observa en la figura 3.20 se muestra una correcta disposición de las ondas para que estas actúen conforme a lo que se requiere.



Figura 3.20: Comportamiento de las ondas  
Imagen elaborada con InkScape.

Planteando que la suma de ambas señales responde a la siguiente ecuación;

<sup>4</sup>Metodo de creación de *beamforming* basado en el *delay and sum*

$$Z_{(x,y,t)} = f_1\left(t - \frac{r_1}{c}\right) + f_2\left(t - \frac{r_2}{c}\right) \quad (3.8)$$

El retardo que debe existir entre un emisor y otro debe estar relacionado con la distancia entre los emisores y el receptor atendiendo a la velocidad de propagación del sonido en el aire.

### Hipótesis de cálculo

Partiendo de la base de que las señales emitidas son iguales, es decir, que responden a la misma gráfica se puede establecer lo siguiente;

$$f_1\left(t_1 - \frac{r}{c}\right) = f_2\left(t_2 - \frac{r}{c}\right) \quad (3.9)$$

Si además, se establece que  $t_1$  corresponde con el instante inicial, es decir que  $t_1 = 0$ , podría desarrollarse lo siguiente;

$$\frac{-r_1}{c} = t_2 - \frac{r_2}{c} \quad (3.10)$$

$$\frac{-r_1}{c} + \frac{r_2}{c} = t_2 \quad (3.11)$$

$$t_2 = \frac{-r_1 + r_2}{c} \quad (3.12)$$

Este desarrollo matemático podría dar un resultado de tiempo negativo, algo que no tiene sentido físicamente hablando, por lo cuál lo que se debe entender al obtener el resultado de este cálculo, es que un resultado negativo, estaría indicando que el emisor que se ha supuesto que emitiría con retraso debe ser el que emita en el instante inicial. Esto es debido a la posición de nuestro receptor, que se podría encontrar más cerca o mas lejos de cualquiera de los emisores. Por lo que esta hipótesis de cálculo no deja de afirmar que el emisor que se encuentra a una

mayor distancia del receptor debe emitir en primera instancia, siendo precedido por el resto de los que se hayan dispuesto en función nuevamente de la distancia a la que se encuentren del receptor.

### 3.2.5. Propagación y suma de señales

Atendiendo a los diferentes cálculos desarrollados anteriormente en las condiciones de idealidad descritas de igual forma en los apartados anteriores. Se ha procedido a realizar una serie de simulaciones en las que se muestra de manera gráfica el comportamiento de las señales emitidas al aplicar en ellas los retrasos temporales correspondientes con respecto a la distancia entre emisores y receptor.

#### Propagación y suma de las ondas

En primer lugar se realizaron una serie de simulaciones, en las que se puede observar como se propaga la onda por el aire desde los emisores (puntos negros), hasta el receptor (estrella roja). Con estas simulaciones se puede observar de manera más detallada lo citado anteriormente.

#### Propagación del frente de onda en el plano de emisión

Por otro lado, se han realizado una serie de simulaciones en las que se puede observar la el comportamiento en la propagación del frente de onda. En dichas imágenes se puede observar como se propaga la onda por el aire desde los emisores (puntos negros), hasta el receptor (estrella roja), punto en el que suman todas las señales emitidas aplicandose la teoría del *Beamforming*.

#### Simulaciones de la variación de la presión acústica

En esta última parte parte, se realizan las simulaciones correspondientes al cambio de presión acústica. En dichas imágenes se puede observar como se propaga la onda de presión por el aire desde los emisores, hasta el receptor (estrella roja), Aunque en este caso se ha centrado el dominio de la imagen alrededor del receptor, con el fin de poder otorgar mayor información, en estas simulaciones se puede observar como la presión acústica de las cuatro señales emitida llega al receptor “sumada” al haberse aplicado una serie de retrasos en su emisión ateniendo a la teoría del *Beamforming* anteriormente expuesta.

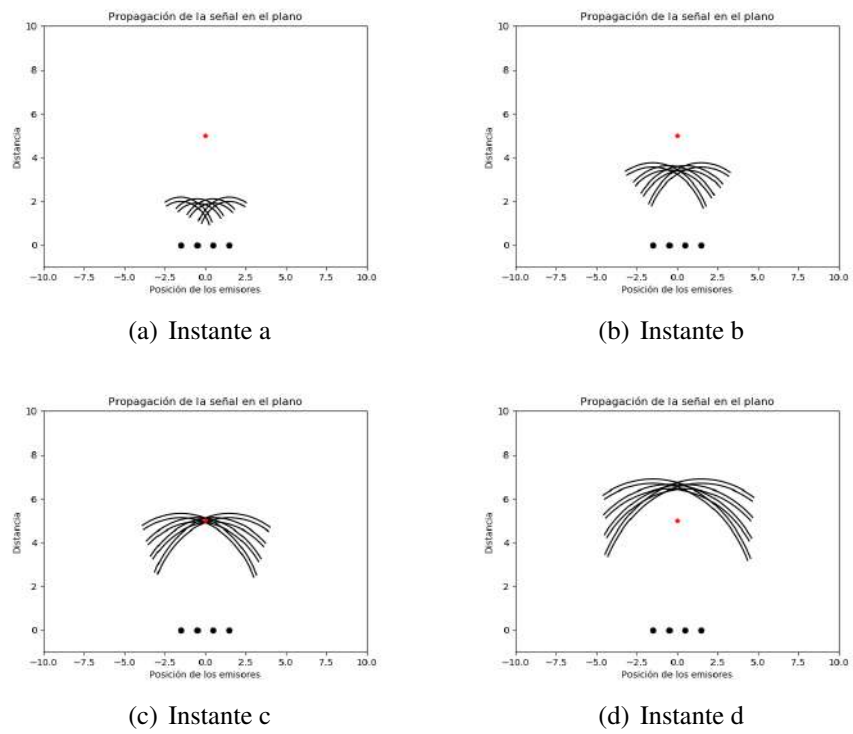


Figura 3.21: Simulaciones de propagación de onda  
Simulaciones realizadas mediante Python.

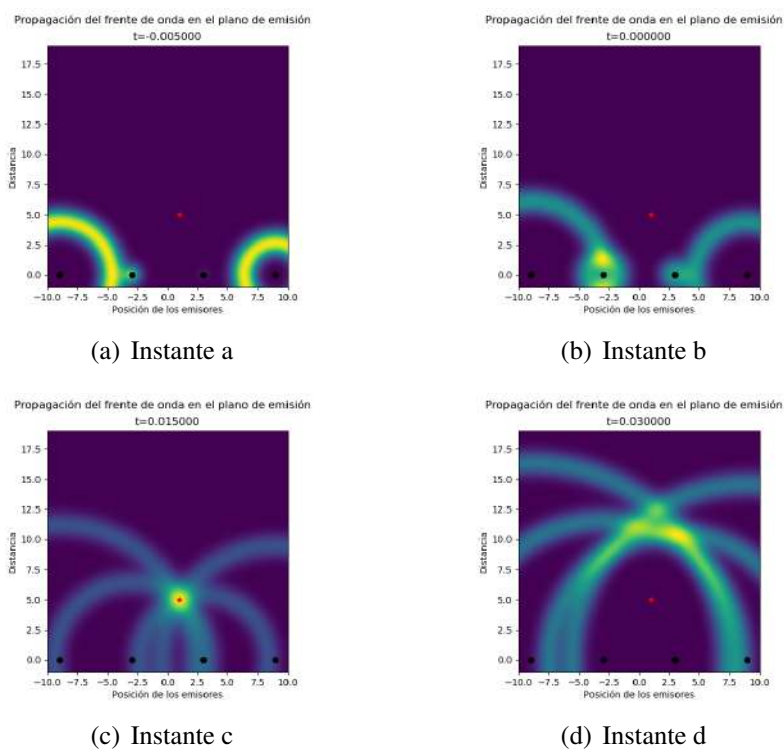


Figura 3.22: Simulaciones de propagación del frente de onda  
Simulaciones realizadas mediante Python.

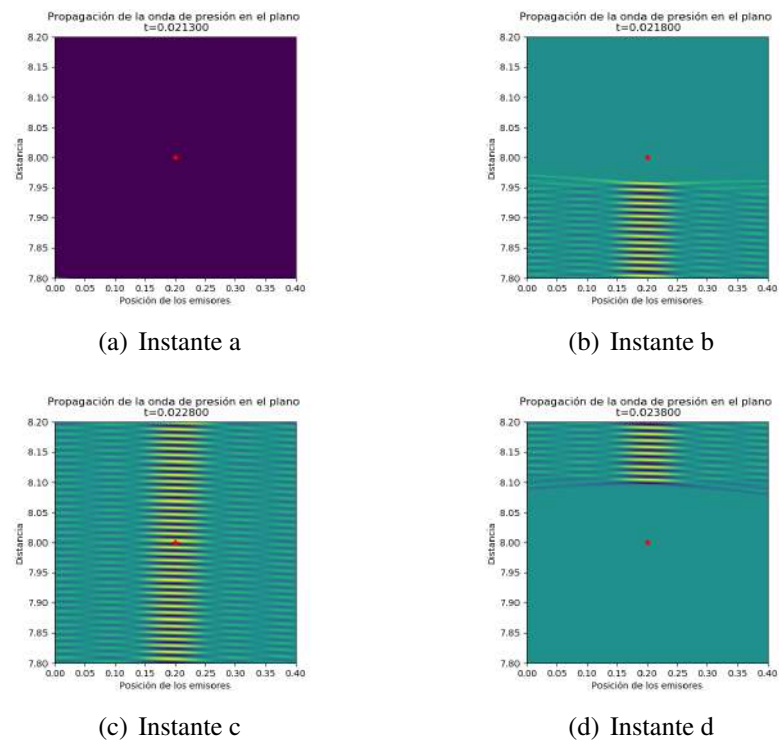


Figura 3.23: Simulaciones de propagación de la presión acústica  
Simulaciones realizadas mediante Python.



# Capítulo 4

## Módulos de procesamiento y alimentación

Trataremos los dos módulos restantes comenzando por el diseño del módulo de procesamiento, dejando para finalizar este capítulo el módulo de alimentación, que como ya se ha comentado alimenta a varias partes de todo el instrumento.

### 4.1. Módulo de procesamiento

Este módulo se encarga de calcular los retrasos de emisión de las diferentes señales y los comunica al módulo de aviso mediante el protocolo de comunicación “I2C”. En la figura 4.1 se muestran los dos bloques que se han incluido en el módulo de procesamiento. El primer bloque calcula los retrasos de tiempo entre los envíos de cada emisor y el segundo bloque se encarga de configurar y de enviar los retrasos al módulo de aviso. También se incluye en este módulo la unidad de procesamiento que se tratará en primer lugar, para luego explicar la comunicación “I2C”, que afecta a este módulo y al módulo de aviso.

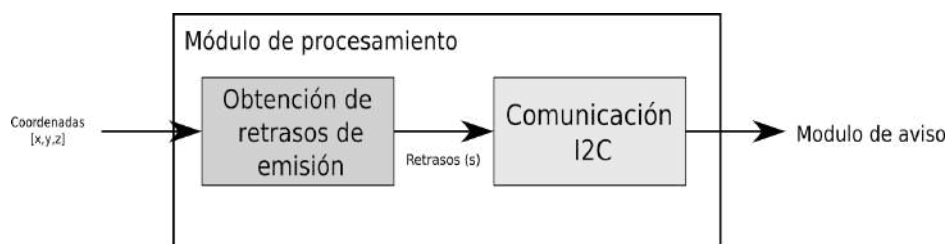


Figura 4.1: Diagrama de bloques que esquematiza el funcionamiento del módulo de procesamiento

Imagen elaborada con InkScape.

### 4.1.1. Unidad de procesamiento

Para la elección de la *CPU* que se integra en este módulo, se ha propuesto usar el microcontrolador “**Raspberry PI 3**” el cual es capaz de una vez procesadas las imágenes correspondientes, comunicarse con el módulo de aviso y ordenarle la emisión de las señales ultrasónicas atendiendo a los diferentes retrasos calculados.

La “**Raspberry PI 3**” es un microcontrolador con diversas *interfaces* de comunicación, alimentación y con puertos de extensión GPIO’s como se muestra en la figura 4.2. Todos los datos técnicos referentes a este elemento, se encuentran detallados en el anexo D. Este tipo de sistema tienen como ventaja la posibilidad de instalar un sistema operativo de propósito general para su utilización. En el caso de este proyecto se eligió Raspbian, un sistema basado en Debian.

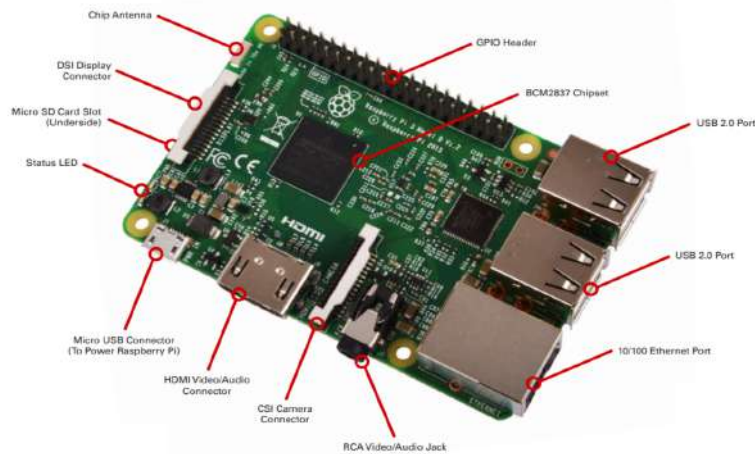


Figura 4.2: Diagrama del modelo de Raspberry PI seleccionado  
Imagen obtenida del datasheet del componente.

Para la programación de las distintas partes del proyecto se utilizó Python versión 3. Los distintos *scripts* utilizados se encuentran en el Anexo G. Especialmente en este módulo se utilizan los *scripts*: “procesar.py”, “detector.py” y “aviso.py”.

La lógica del bucle de procesamiento se realiza en el *script* “procesar.py” en el que primero se configura el sistema periférico del módulo de aviso y se activa la cámara para después entrar en un bucle de procesamiento continuo. Cada ciclo del bucle se realiza una lectura de la cámara que se procesa con la red neuronal. Como el resultado de la red neuronal es una lista de detecciones se realiza un bucle que genera los retrasos para cada detección si se trate de un ave. Estos retrasos se envían al módulo de aviso por I<sup>2</sup>C”.

Los otros dos *scripts* desarrollados contienen el código de cada uno de los módulos descritos en el capítulo anterior.

### 4.1.2. Comunicación - Bus I2C

Abreviatura de Inter-IC (inter integrated circuits), el “I2C” es un tipo de bus diseñado por Philips Semiconductors a principios de los años 80, que se utiliza para conectar circuitos integrados. Es un bus con múltiples maestros, lo que significa que se pueden conectar varios chips al mismo bus y que todos ellos pueden actuar como maestro, sólo con iniciar la transferencia de datos [4]. En el desarrollo de este instrumento se ha implementado este sistema con el fin de comunicar los módulos de procesamiento y aviso donde este último, recibe mediante este protocolo la configuración y la orden de activación de las señales de aviso.

#### Introducción a la comunicación I2C

El bus I2C es un estándar que facilita la comunicación entre microcontroladores, memorias y otros dispositivos utilizando únicamente dos líneas de señal y una de tierra. Permite el intercambio de información entre dispositivos a una velocidad de 100 Kbits por segundo. La metodología de comunicación de datos del bus “I2C” es en serie y sincrónica. Una de las señales del bus marca el tiempo (SCL) y la otra se utiliza para intercambiar datos (SDA) [4].

#### Descripción de las señales

- **SCL** (System Clock) Línea de los pulsos de reloj que sincronizan el sistema.
- **SDA** (System Data) Línea por la que se mueven los datos entre los dispositivos.
- **GND** (Tierra) Línea común de la interconexión entre todos los dispositivos conectados al bus.

Las líneas SDA y SCL son del tipo drenaje abierto, es decir, un estado similar al de colector abierto, pero asociadas a un transistor de efecto de campo (o FET). Se deben polarizar en estado alto<sup>1</sup>, lo que define una estructura de bus que permite conectar en paralelo múltiples entradas y salidas. Esto último es lo que le da una gran importancia a la utilización de este sistema en el desarrollo del instrumento, la posibilidad de poder conectar diferentes *masters* o *slaves* al bus dando una mayor versatilidad a la hora de su puesta en servicio.

---

<sup>1</sup>El estado alto se consigue conectando a la alimentación por medio de resistores “pull-up”

Comando	Tamaño del argumento	Aspecto configurable
SET_FREQ	Intx2	Valor del preescaler
SET_TIME	Intx4	Tiempo de emisión (Milisegundos)
SET_FREQ	4x(Intx4)	Retardo de la emisión (Milisegundos)

Tabla 4.1: Comandos para la comunicación I2C.

### Comandos de activación

Para establecer la comunicación entre el sistema “Master” y el sistema “Slave” mediante el protocolo I2C, se han desarrollado una serie de comandos capaces de realizar el envío de información. Estos comandos de activación, que se pueden observar en la tabla 4.1, tienen unos argumentos asociados que permiten configurar las variables del *slave* en los aspectos relativos a la frecuencia, el tiempo de emisión y los retardos asociados a la activación de cada uno de los pines.

## 4.2. Modulo de Alimentación

Este módulo se ha diseñado siguiendo los principios sobre los que se fundamenta este proyecto, basados en la creación de tecnología ética que contribuya al desarrollo sostenible. Por lo que se plantea un módulo de alimentación capaz de alimentar el sistema mediante el uso de energías renovables. En este caso la fuente de energía limpia utilizada ha sido la energía fotovoltaica, cómo se puede observar, en la figura 4.3, que muestra el diagrama de bloques de este módulo.

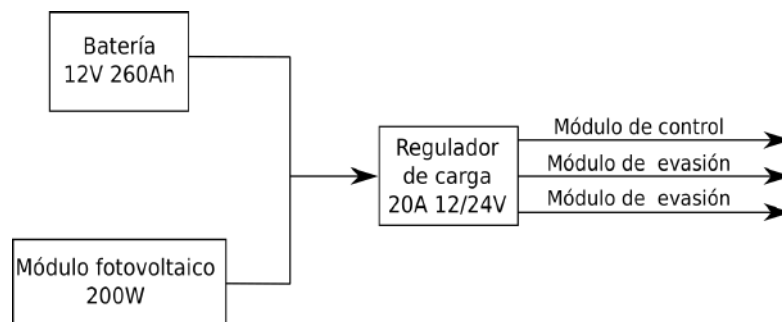


Figura 4.3: Diagrama de bloques del módulo de alimentación

Diagrama elaborado en InkScape.

Para elaborar dicho diseño se ha partido del dato de consumo de los elementos que conforman el instrumento. Para ello se ha recurrido a las diferentes hojas de datos suministradas por los fabricantes adjuntas en el Anexo D, y a las pruebas realizadas en el laboratorio. Atendiendo a esto se ha fijado el consumo para el peor

de los casos en **20W** y en base a esto, se han desarrollado los cálculos necesarios para dimensionar este módulo tal y cómo se expone en el Anexo **A**.

#### 4.2.1. Batería

Atendiendo a los resultados correspondientes a esta sección expuestos en el anexo **A**, se puede observar que la capacidad de la batería elegida debe ser superior a la que se ha obtenido en el cálculo planteado. Por lo que se procede a la elección de la batería “**AGM 12V 260Ah CYM POWER PLUS**” del distribuidor oficial “Grupo Cymasol Renovables S.L.” para un mayor aporte técnico sobre esta, se adjuntan sus datos técnicos en el Anexo **D**. Además de dichos datos se puede observar en la figura 4.4 el aspecto físico de la misma. Por otro lado es resaltable que podría ser más ético en lo que al cuidado del medio ambiente se refiere, hacer uso de una batería de litio, pero atendiendo a las posibles características del emplazamiento donde podría ser instalado el instrumento se cree más conveniente el uso de baterías AGM, ya que además de ser más tolerantes a los golpes, presentan una mayor estabilidad y durabilidad en instalaciones con características de ventilación desfavorable.



Figura 4.4: Imagen de la batería seleccionada

Imagen obtenida de la web del distribuidor oficial.

#### 4.2.2. Módulo fotovoltaico

Para la elección del módulo fotovoltaico se recurrió al sistema de rendimiento fotovoltaico de la comisión Europea (<https://ec.europa.eu/jrc/en/pvgis>). En dicho sistema se realizaron distintas simulaciones atendiendo a las localizaciones de dos lugares de la comunidad autónoma de Canarias con distintas características solares. Uno de ellos situado en un punto de la isla de Gran Canaria y el otro situado en la propia Universidad de La Laguna. Ambas situaciones fueron estudiadas para una orientación optimizada, atendiendo a las coordenadas de la instalación y para un montaje a  $45^\circ$  que facilite la instalación del módulo en lo que acople estructural se refiere, dichos datos se adjuntan en el Anexo **F**.

Atendiendo a los datos citados anteriormente y a los cálculos realizados y expuestos en el Anexo **A**, se ha procedido a la elección del módulo fotovoltaico

“**MÓDULO FOTOVOLTAICO MONO PERC 200Wp**” del distribuidor oficial “Grupo Cymasol Renovables S.L.” del cual se adjuntan sus datos técnicos en el Anexo **D** y que se puede observar con detalle en la figura 4.5.



Figura 4.5: Imagen del modulo fotovoltaico seleccionado  
Imagen obtenida de la web del distribuidor oficial.

### **4.2.3. Regulador de Carga**

Una parte fundamental del diseño de este módulo de alimentación es el regulador de carga, en el caso del diseño propuesto se ha seleccionado el modelo “**PWM PLUS 20A 12V/24V CON DISPLAY CYM POWER**” del distribuidor oficial “Grupo Cymasol Renovables S.L.”, estando sus datos técnicos adjuntos en el Anexo **D**.

El regulador de carga seleccionado se puede observar en la figura 4.6. La elección se ha llevado a cabo en base a los datos anteriormente calculados.



Figura 4.6: Imagen del regulador de carga seleccionado  
Imagen obtenida de la web del distribuidor oficial.

#### 4.2.4. UPS del módulo de procesamiento

Con el fin de garantizar el suministro de energía al sistema de procesamiento y un apagado seguro del mismo, se equipa a este de una fuente de alimentación ininterrumpida o UPS como la que se puede ver en la figura 4.7. Este sistema permite a su vez que el CPU sea alimentado directamente con una fuente de alimentación de entre 7 y 28 Voltios. En el caso de este proyecto se alimentará a 12 voltios.



Figura 4.7: Imagen de la UPS seleccionada  
Imagen obtenida del datasheet del componente.





# Capítulo 5

## Ensayos y montajes de laboratorio

Con el fin de comprobar la efectividad y la puesta en servicio de los diferentes componentes que conforman el instrumento, se han realizado una serie de ensayos y montajes de laboratorio, que se exponen en este capítulo.

### 5.1. Ensayos y montajes del emisor

A continuación se detallan las diferentes pruebas y ensayos que se han llevado a cabo con el fin de poder dar un rigor técnico a la utilización de los emisores elegidos para la fabricación del instrumento.

#### 5.1.1. Frecuencias inferiores a 20KHz

En primer lugar, se llevó a cabo la realización de un ensayo, en el que se procedió a excitar uno de los emisores con señales de distintos valores de amplitud(V) a frecuencias inferiores a los 20KHz (hasta los 17KHz). Esto se realizó con el fin de validar los datos de amplitud y frecuencia de corte suministrados por el vendedor. Para ello se hizo uso de un generador de señales, capaz de enviar las distintas señales al emisor seleccionado, el montaje de este ensayo se puede observar de manera esquemática en la figura 5.1. Al realizar este ensayo, como se muestra en la figura 5.2, se pudo determinar que el emisor era capaz de emitir en las frecuencias audibles para el ser humano en las amplitudes indicadas por el vendedor.

#### 5.1.2. Frecuencias ultrasónicas

En segundo lugar se procedió a la realización de un ensayo de laboratorio que permitiese conocer si el emisor seleccionado era capaz de emitir en frecuencias

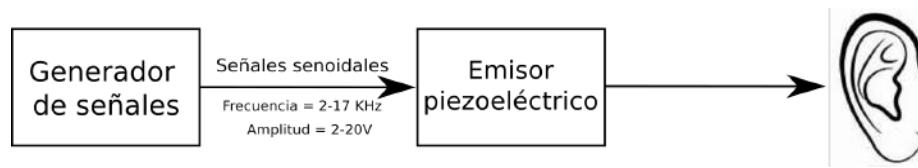


Figura 5.1: Diagrama de bloques del ensayo realizado a frecuencias inferiores a los 20KHz

Imagen realizada en Inkscape.



Figura 5.2: Imagen del montaje del ensayo en el espectro audible  
Prueba realizada en el laboratorio de BFmSD.

superiores a los 20Khz. Esto se realizó atendiendo a los requerimientos de diseño del instrumento, ya que para poder afirmar que se emiten señales ultrasónicas se hace necesario comprobar que el emisor seleccionado es capaz de emitir en estas frecuencias. Para ello se realizó el montaje que de manera esquemática se puede observar en la figura 5.3, el cual consistió en enfrentar dos emisores piezoeléctricos dónde uno actuaba como emisor, estando conectado al generador de señales y el otro como receptor, conectado al osciloscopio. Esto se llevo a cabo atendiendo a las propiedades de los materiales piezoeléctricos expuestas y detalladas en el Anexo B.

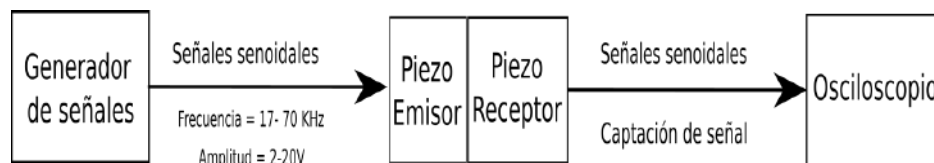


Figura 5.3: Diagrama de bloques del ensayo realizado a frecuencias superiores a los 20KHz con dos emisores

Imagen realizada en Inkscape.

La realización de esta prueba de laboratorio, se puede observar en las figuras 5.4 y 5.5. Dicha prueba dio como resultado, que el piezo receptor era capaz de

captar la señal que se observa en la figura 5.6. Esto permite conocer que el emisor es capaz de ser excitado y actuar a las frecuencias deseadas, no perceptibles en el rango de audición humano. Aunque hay que resaltar que este resultado que se ha obtenido no es del todo veraz, ni concreto. Esto es debido a que en el piezoeléctrico que actúa como receptor, influyen factores como la propia vibración del piezoeléctrico que actúa como emisor, por lo que se podrían estar obteniendo valores que no corresponden íntegramente al sonido emitido.



Figura 5.4: Montaje del ensayo de emisión a frecuencias superiores a 20KHz  
Montaje realizado en el laboratorio de BFmSD.



Figura 5.5: Conexión y ensamblaje de los “piezos” emisor/receptor  
Montaje realizado en el laboratorio de BFmSD.

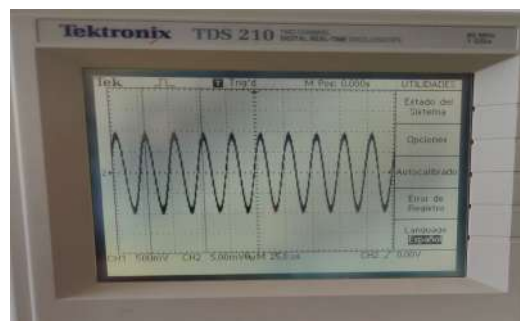


Figura 5.6: Señal obtenida por el “piezo” receptor a 35KHz  
Señal obtenida por el osciloscopio del laboratorio de BFmSD.

Una vez realizados estos ensayos, y siguiendo la misma dinámica, se procedió a realizar un nuevo ensayo en el que se utilizó como receptor un piezoeléctrico comercial como el que se puede observar en la figura 5.7. Tal y como se muestra de manera esquemática en la figura 5.8, se realizó este ensayo, en el que se pretendía

dar mayor veracidad a la comprobación de la emisión a frecuencias superiores a los 20Khz.



Figura 5.7: Imagen de un piezoeléctrico comercial de 12 mm de diámetro.

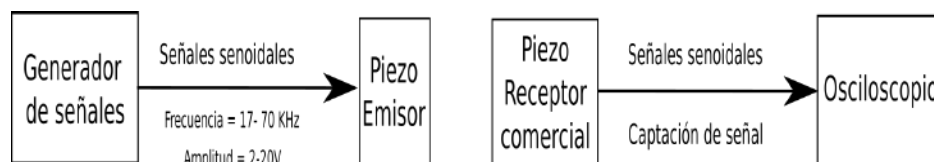


Figura 5.8: Diagrama de bloques del ensayo realizado a frecuencias superiores a los 20KHz con un piezoeléctrico comercial como receptor.

Imagen realizada en Inkscape.

Este montaje se realizó tal y como se puede observar en las figuras 5.9 y 5.10, dando como resultado la señal captada por el osciloscopio que se muestra en la figura 5.11.

## 5.2. Generación de señales ultrasónicas

Tal y como se muestra en la figura 5.12 se realizó un ensayo que sirviese para comprobar que el microcontrolador seleccionado, el ATM328, era capaz generar señales a frecuencias superiores a los 20KHz. Dicho ensayo consistió en poner a prueba el código realizado para la generación de señal *PWM* a 31250Hz, el cual puede ser observado con detalle en el Anexo G.

Mientras que el montaje de este ensayo se puede apreciar en la figura 5.13, el resultado de este se observa en la figura 5.14 donde se puede comprobar lo anteriormente descrito.



Figura 5.9: Montaje del ensayo de emisión a frecuencias superiores a 20KHz  
Montaje realizado en el laboratorio de BFmSD.



Figura 5.10: Conexión emisor/receptor comercial  
Montaje realizado en el laboratorio de BFmSD.



Figura 5.11: Señal obtenida por el piezoeléctrico comercial a 35KHz  
Señal obtenida por el osciloscopio del laboratorio de BFmSD.

### 5.3. Filtrado y amplificación de la señal ultrasónica

Una vez conocido que el emisor era capaz de emitir señales por encima de los 20KHz y que el microcontrolador seleccionado era capaz de generar estas señales. Se procedió a diseñar la etapa de filtrado y amplificación de la señal, la cual se encuentra desarrollada y justificada en el Anexo A. Tras ser diseñada, fue montada en una *protoboard* con el fin de poder comprobar su correcto funcionamiento. En la figura 5.15, se puede observar de manera esquemática la composición de este montaje.

Mientras que en la figura 5.16 se puede observar la totalidad del montaje realizado en este ensayo. En la figura 5.17 se puede observar con mayor detalle el montaje que se ha realizado en la *protoboard*, además de poder observar en el Anexo E el esquemático realizado para este diseño. Por otro lado, se puede observar en la figura 5.18, como esta etapa de filtrado y amplificación tiene una etapa *Step-Up* previa, que permite alimentar el circuito a 24V, desde una fuente de 12V.

En la figura 5.19 se muestra la comparativa entre la señal generada por el microcontrolador ATM328 y la señal de salida del circuito de filtrado y amplifica-

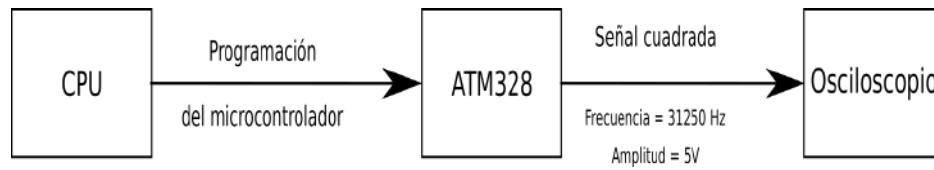


Figura 5.12: Diagrama de bloques del ensayo realizado para la comprobación de la generación de señales superiores a 20KHz

Imagen realizada en Inkscape.

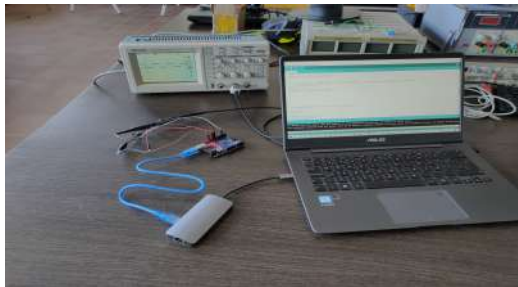


Figura 5.13: Montaje del ensayo de generación de señal con el microcontrolador ATM328

Montaje realizado en el laboratorio de BFmSD.

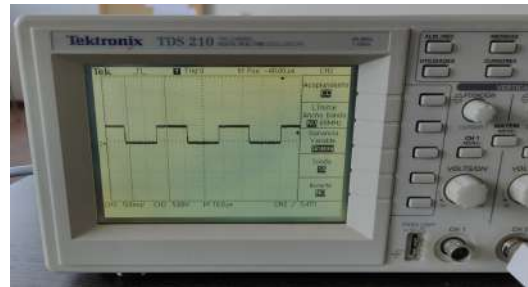


Figura 5.14: Señal cuadrada obtenida a la salida del microcontrolador ATM328

Señal obtenida por el osciloscopio del laboratorio de BFmSD.

ción. En ella se puede observar con mayor detalle como la etapa diseñada cumple con su objetivo de diseño, pudiendo medir a su salida una señal senoidal de 30Vpp desde una señal cuadrada de 5v de amplitud. Esta medida, se ha obtenido directamente del osciloscopio usando el mismo procedimiento que lleva a cabo para obtener los datos del Anexo B.

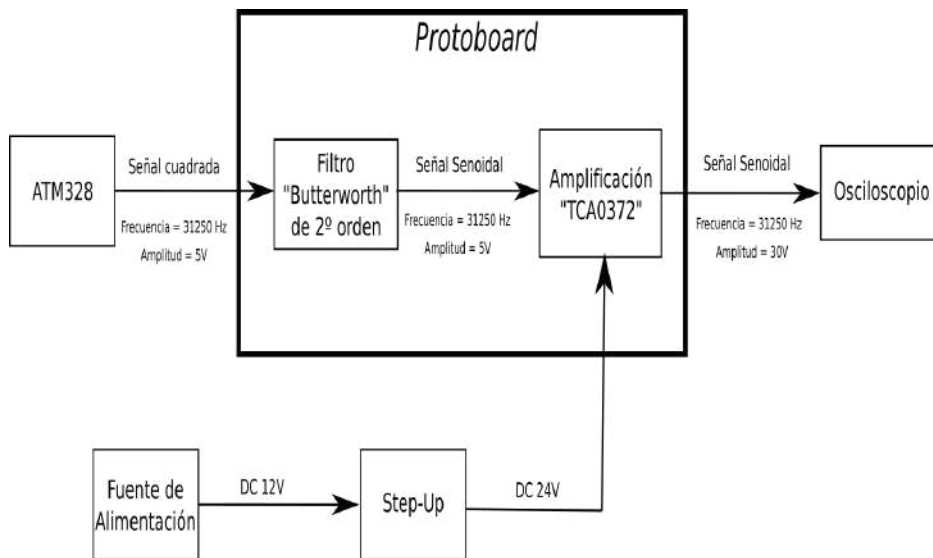


Figura 5.15: Diagrama de bloques del ensayo realizado para comprobar el funcionamiento de la etapa de filtrado y amplificación

Imagen realizada en Inkscape.



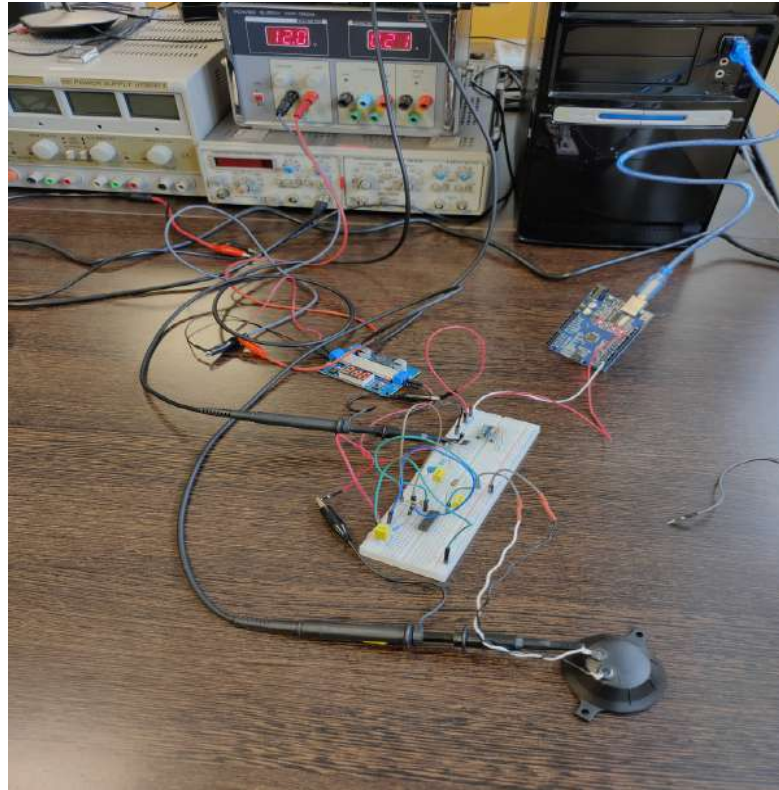


Figura 5.16: Montaje en *protoboard* del sistema de filtrado y amplificación de señal

Montaje realizado en el laboratorio de BFmSD.

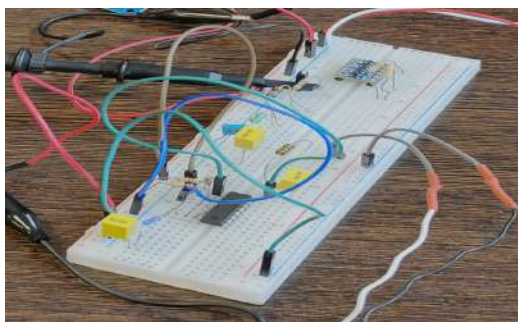


Figura 5.17: Detalle del montaje en *protoboard* del sistema de filtrado y amplificación de la señal.

Montaje realizado en el laboratorio de BFmSD.

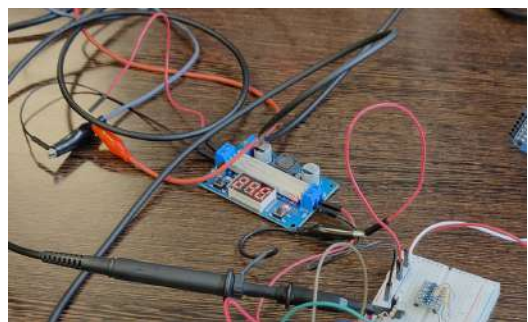


Figura 5.18: Detalle de la etapa *Step-Up* previa al montaje en *protoboard* del sistema de filtrado y amplificación de señal

Montaje realizado en el laboratorio de BFmSD.



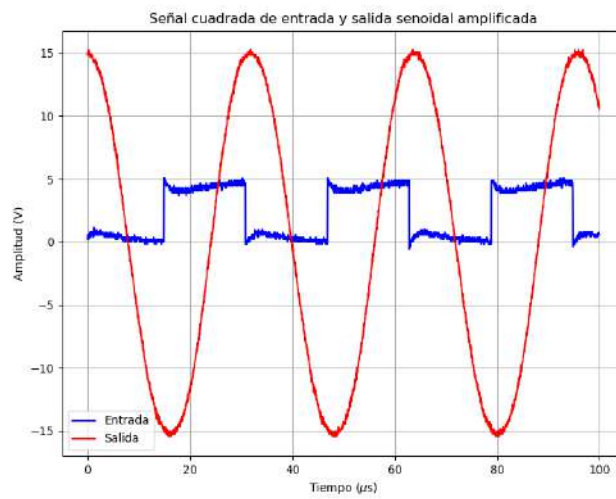


Figura 5.19: Comparativa de señales de entrada y de salida en la etapa de filtrado y amplificación



# Capítulo 6

## Fabricación del prototipo funcional

En el presente capítulo se describen los procedimientos necesarios para la fabricación del prototipo funcional que permita realizar una prueba de concepto. Esta fabricación sigue la idea fundamental de diseño del instrumento.

### 6.1. Fabricación de la “PCB” del módulo de aviso

En primer lugar, se fabricó la *PCB* diseñada y expuesta en el capítulo 3 en ella se encuentra la etapa de generación, filtrado y amplificación de señal. Para llevar a cabo dicha fabricación se ha procedido en orden con el corte y realizado de pasantes, la transferencia de fotolitos, el ataque químico y la soldadura y fijación de componentes que se describen en las siguientes sub-secciones.

#### 6.1.1. Corte y realizado de pasantes

En esta primera etapa haciendo uso del sistema de corte por control numérico expuesto y detallado en el capítulo 2, se realizaron los diferentes cortes y orificios pasantes en la placa de cobre necesarios para comenzar con la fabricación de la *PCB*. En la figura 6.1 se puede observar el resultado obtenido tras llevar esto a cabo.

#### 6.1.2. Transferencia de fotolitos

Una vez realizado el paso anterior y haciendo uso del rodillo fusor térmico, se procede a la transferencia de los fotolitos por ambas caras de la placa, obteniendo el resultado que se puede observar en las figuras 6.2 y 6.3.



Figura 6.1: Imagen de la *PCB* tras realizarle los orificios y cortes necesarios  
Imagen realizada en el laboratorio de electrónica de la ULL.

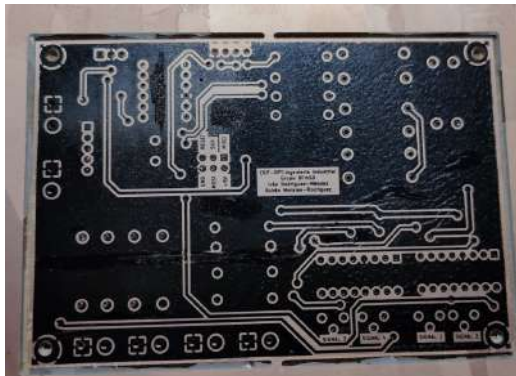


Figura 6.2: Imagen del anverso de la *PCB* tras realizarle la transferencia de fotolitos

Imagen realizada en el laboratorio de electrónica de la ULL.

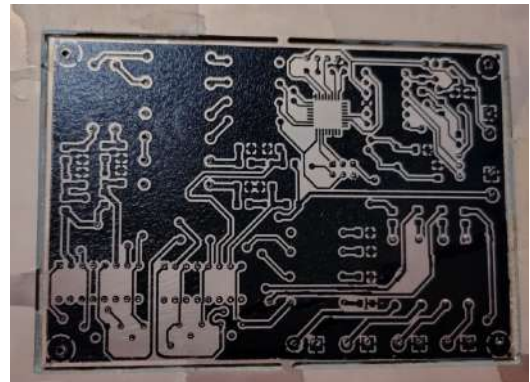


Figura 6.3: Imagen del reverso de la *PCB* tras realizarle la transferencia de fotolitos

Imagen realizada en el laboratorio de electrónica de la ULL.

### 6.1.3. Ataque químico

Tras haber realizado la transferencia de los fotolitos, se realiza el ataque químico a la placa de cobre con el fin de eliminar de esta el cobre que no es útil. Tras realizar dicho ataque se obtiene como resultado lo que se puede observar en las figuras 6.4 y 6.5.

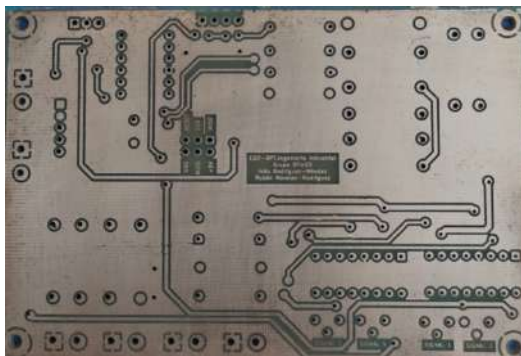


Figura 6.4: Imagen del anverso de la *PCB* tras ser atacada químicamente  
Imagen realizada en el laboratorio de electrónica de la ULL.

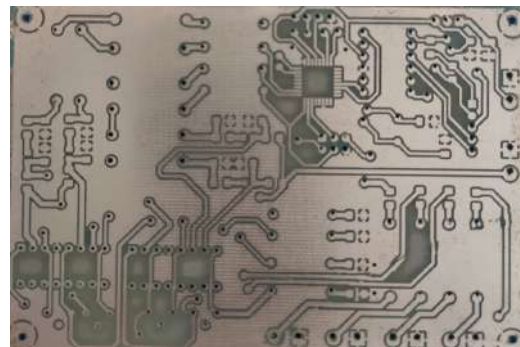


Figura 6.5: Imagen del reverso de la *PCB* tras ser atacada químicamente  
Imagen realizada en el laboratorio de electrónica de la ULL.

#### 6.1.4. Fijación y conexionado de los componentes

Una vez llevado a cabo el ataque químico se procede con la última etapa de la fabricación de la *PCB*, la cual consiste en la fijación y conexionado de los diferentes componentes que conforman el diseño mediante soldadura de estaño. El resultado de la realización de esta última etapa se puede observar en las figuras 6.6 y 6.7

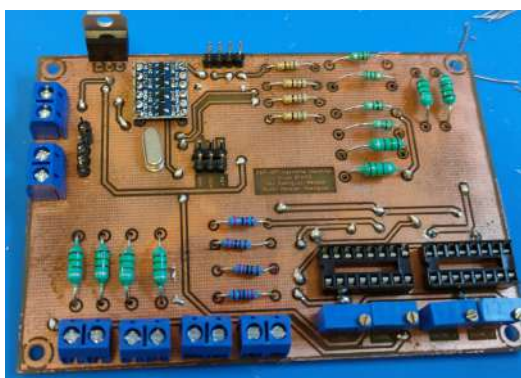


Figura 6.6: Imagen del anverso de la *PCB* tras ser soldados sus componentes  
Imagen realizada en el laboratorio de electrónica de la ULL.

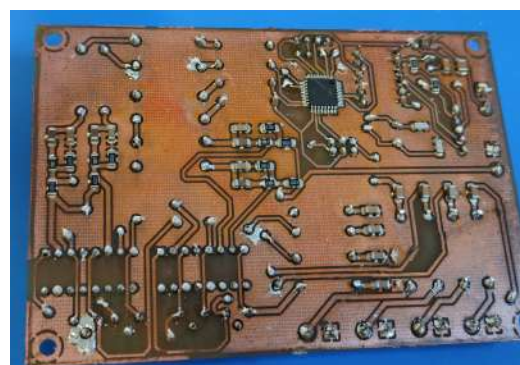


Figura 6.7: Imagen del reverso de la *PCB* tras ser soldados sus componentes  
Imagen realizada en el laboratorio de electrónica de la ULL.

## 6.2. Conexionado y encapsulado de los Emisores implementados

Los emisores implementados son emisores piezoeléctricos, los cuales fueron sometidos a una serie de ensayos con el fin de ser caracterizados, tal y como se puede observar en el Anexo B. Estos emisores fueron seleccionados atendiendo a sus características las cuales fueron facilitadas por el vendedor y se basaban en lo siguiente;

- Rango de frecuencia de emisión: 2 - 60 (KHz)
- Voltaje mínimo de excitación: 5V
- Voltaje Máximo de excitación: 30V

En las figuras 6.8 y 6.9 se puede observar el aspecto físico de los emisores seleccionados.



Figura 6.8: Imagen de la parte delantera de uno de los emisores.



Figura 6.9: Imagen de la parte posterior de uno de los emisores.

### 6.2.1. Conexionado de los emisores

Para llevar a cabo el conexionado entre los emisores y las *PCB* se ha utilizado cable de audio de dos hilos, como el que se muestra en la figura 6.10, cuya hoja de datos se encuentra adjunta en el apartado correspondiente del Anexo D.

Atendiendo a la disposición de los emisores en el instrumento, se procedió a la realización de los cortes de cable en función de las distancias desde los emisores al punto de conexionado, siendo estas de 1.30m, 1m, 0.70m y 0.40m desde el emisor más alejado hasta el más cercano al punto de conexionado respectivamente (estas distancias atienden a holguras de conexionado).

## 6.2. CONEXIONADO Y ENCAPSULADO DE LOS EMISORES IMPLEMENTADOS 91



Figura 6.10: Imagen del cable de audio seleccionado  
Imagen obtenida de la web del vendedor.

Una vez obtenidos estos cortes se realizó un trenzado en cada par de cables con el fin de evitar interferencias entre los campos magnéticos que podría generar cada uno de los pares. En la tabla 6.1 se refleja de manera detallada el par de vueltas por unidad de longitud de cada uno de los pares, atendiendo a que las longitudes eran distintas se le dio el mismo número de vueltas a cada par de cables pudiendo observar el resultado obtenido en la figura 6.11.

Longitud de línea	Nº de vueltas	Vueltas por unidad de longitud (Nºv/m)
1.30	25	19.23
1	25	25
0.70	25	35.71
0.40	25	62.5

Tabla 6.1: Relación del par de trenzado del cableado de los emisores.





Figura 6.11: Imagen del trenzado de cable fabricado  
Imagen realizada en el laboratorio de BFmSD.

Por último para concluir con el conexionado de los emisores al cable seleccionado, se procedió a fijar mediante soldadura de estaño los terminales del piezoeléctrico a una de las puntas de cada par trenzado. Además de esto, se protegieron dichas soldaduras mediante el uso de material termoretráctil. El resultado de llevar esto a cabo se puede observar en las figuras 6.12 y 6.13.



Figura 6.12: Imagen de la parte delantera de uno de los emisores tras ser conectado.



Figura 6.13: Imagen de la parte posterior de uno de los emisores tras ser conectado.



### 6.2.2. Encapsulado de los emisores

Una vez se realizaron los correspondientes conexiones de los emisores, se diseñaron y fabricaron mediante impresión 3D las cápsulas que permitiesen además de proteger dichos emisores frente a leves inclemencias meteorológicas, fijarlos a la estructura portante. En las figuras 6.14 y 6.15 se puede observar con detalle el resultado obtenido en esta etapa.



Figura 6.14: Imagen detallada del frontal de la cápsula fabricada para los emisores.

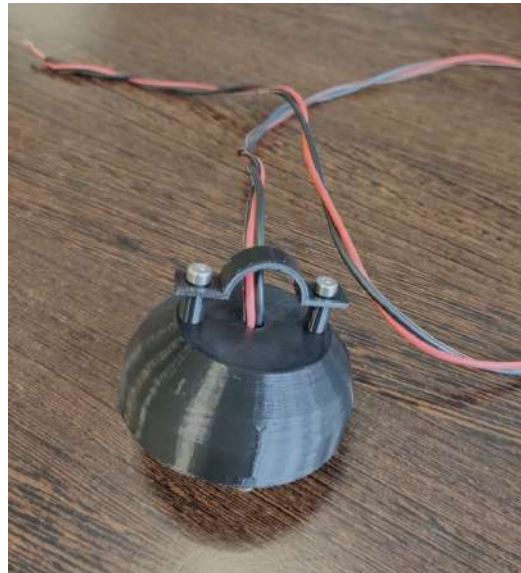


Figura 6.15: Imagen detallada de la sujeción de la cápsula

### 6.2.3. Fijación de los encapsulados de los emisores

Estas cápsulas con su emisor correspondiente fueron fijadas al tubo portante de la forma que se puede observar en las figuras 6.16 y 6.17. Dando como resultado lo que se puede observar en las figuras 6.18 y 6.19.



Figura 6.16: Imagen detallada del lateral de la fijación de la cápsula.



Figura 6.17: Imagen detallada del reverso de la fijación de la cápsula.



Figura 6.18: Imagen detallada de la fijación de las cápsulas de los emisores al tubo portante



Figura 6.19: Imagen detallada de la fijación de las cápsulas cableadas al tubo portante

### 6.3. Diseño y fabricación del soporte de la cámara

Además de esto se realizó el diseño de un soporte para la cámara que permitiese, igual que en el caso anterior, protegerla ante leves inclemencias meteorológicas y su integración en el sistema. Dicho soporte se puede observar con detalle en las figuras 6.20 y 6.21.



Figura 6.20: Imagen detallada del frontal del soporte de la cámara.



Figura 6.21: Imagen detallada del lateral del soporte de la cámara.

## 6.4. Integración mecánica del sistema

Una vez realizado esto y utilizando la misma técnica de diseño 3D se diseñó y fabricó una caja que albergase los diferentes sistemas electrónicos que constituyen el prototipo. Esta se fabricó de manera que además de proteger al sistema ante leves inclemencias meteorológicas, permitiese integrar todas las partes que se había diseñado y permitiese la conexión de las diferentes *interfaces* de alimentación. En las figuras 6.22 y 6.23 se puede observar el resultado obtenido en cuanto a la caja se refiere, mientras que los detalles de integración e *interface* de conexionado se pueden observar en las figuras 6.24 y 6.25.

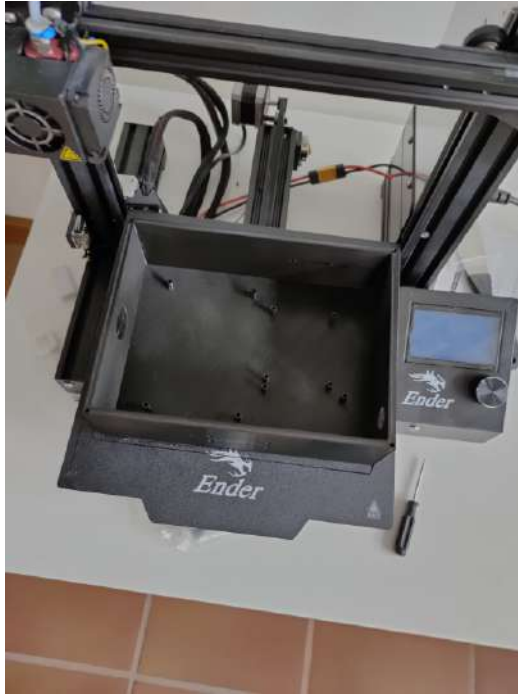


Figura 6.22: Imagen de la caja antes de ser retirada de la impresora 3D.  
Imagen realizada en el laboratorio de BFmSD.



Figura 6.23: Imagen de la caja antes de ser retirada de la impresora 3D.  
Imagen realizada en el laboratorio de BFmSD.



Figura 6.24: Imagen detallada de la fijación del soporte de la cámara la caja.

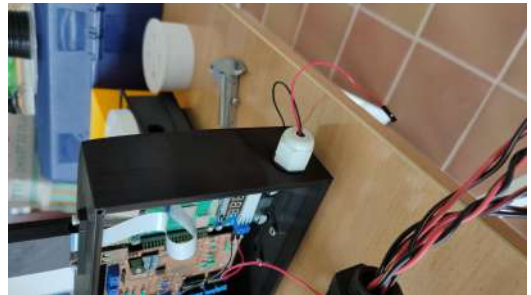


Figura 6.25: Imagen detallada de la *interface* de alimentación de la caja.

# Capítulo 7

## Discusión y mejoras futuras

### 7.1. Módulo de detección

Para poder determinar las coordenadas del mundo en las que se encuentra el ave detectada, es necesario poder conocer las coordenadas a las que corresponde cada píxel de la imagen obtenida. Para ello se plantea como mejora futura hacer uso de dos cámaras con las mismas características, ya que conociendo las coordenadas de cada una de las cámaras, se podrían obtener las coordenadas de posición de cada ave (x,y,z) en el mundo.

Además de esto, en el desarrollo del dispositivo funcional, no se tuvieron en cuenta aspectos como la distorsión de la imagen, realizando la calibración de la cámara para un caso ideal. Por ello además de plantear el uso de dos cámaras se destaca como línea futura hacer una correcta corrección de la calibración de la cámara. Otra posible mejora podría ser la acomodación de la red Detectron2 aunque eso implicaría el cambio de la unidad de procesamiento.

### 7.2. Módulo de procesamiento

Una de las mejoras que se proponen para este módulo, es la creación de un sistema *boot* en la raspberry que actúa como *master*. La idea es que este *boot* sea capaz de interactuar con el usuario y que además de enviar información relativa al funcionamiento del propio sistema, sea capaz de informar de las veces que ha sido activado por la cercanía de un ave, e incluso si el modelo de detección lo permitiese, informar de que aves se han podido detectar.

### 7.3. Módulo de aviso

Como posible mejora en el módulo de aviso se plantea recalcular el filtro de señal diseñado. Este se calculó atendiendo a las aproximaciones de Butterworth pero podría plantearse obtener este diseño mediante las aproximaciones de Chevysev y/o Besel con el fin de estudiar cual presenta mejor señal de salida.

#### 7.3.1. Módulo de alimentación

A medida que se ha desarrollado el proyecto se ha llegado a la conclusión de que una de las medidas a implementar como mejora futura de este sistema, es realizar un nuevo cálculo más preciso atendiendo al consumo del dispositivo. Para ello se propone hacer uso de herramientas de medida de consumo, capaces de determinar de manera fiable el consumo diario del instrumento. Esto permitiría diseñar un módulo de alimentación más eficiente capaz de ser reducido tanto en costes como en tamaño.

### 7.4. Uso del instrumento en otros casos

Tras haber realizado, de manera previa al proyecto, una extensa revisión bibliográfica, se han podido observar diferentes situaciones en las que este instrumento podría otorgar un gran aporte a la sociedad.

Situaciones como la conservación de diferentes edificaciones o esculturas, que forman parte de la sociedad y en ocasiones parte del patrimonio cultural, Nacional o Mundial de los excrementos o anidaciones de aves, podrían ser solventadas por el instrumento que se propone.

Por otro lado, la presencia cercana de aves a aeropuertos y pistas de aterrizaje/despegue, es un problema comúnmente conocido. Por lo general la presencia de aves cerca a estas zonas, suele evitarse con el uso de la cetrería<sup>1</sup>, situación que aunque salvaguarde el despegue o aterrizaje de una aeronave va en detrimento de la conservación animal. Es por ello por lo que se podría considerar la implementación de sistemas como este.

---

<sup>1</sup>La cetrería se define como la actividad de cazar con aves rapaces entrenadas, especialmente con halcones, azores y otras aves de presa para la captura de especies de volatería o de tierra.

# Capítulo 8

## Discussion and future improvement

### 8.1. Detection module

In order to determine the coordinates of the world in which the detected bird is located, it is necessary to know the coordinates to which each pixel of the image obtained corresponds. In order to do this, it is proposed as a future improvement to make use of two cameras with the same characteristics, since knowing the coordinates of each of the cameras, the position coordinates of each bird  $(x,y,z)$  in the world could be obtained.

In addition to this, in the development of the functional device, aspects such as image distortion were not taken into account, carrying out the calibration of the camera for an ideal case. Therefore, in addition to considering the use of two cameras, a correct correction of the camera calibration is highlighted as a future line of action. Another possible improvement could be the accommodation of the Detectron2 network, although this would imply changing the processing unit.

### 8.2. Processing module

One of the improvements proposed for this module is the creation of a boot system in the raspberry that acts as the master. The idea is for this boot to be able to interact with the user and, in addition to sending information on the operation of the system itself, to be able to report how many times it has been activated by the proximity of a bird, and even, if the detection model allows it, to report which birds have been detected.

### **8.3. Warning module**

As a possible improvement in the warning module, it is proposed to recalculate the designed signal filter. This was calculated using the Butterworth approximations, but it could be considered to obtain this design using the Chebyshev and/or Bessel approximations in order to study which has the best output signal.

### **8.4. Power supply module**

As the project has developed, it has been concluded that one of the measures to be implemented as a future improvement of this system is to make a new, more precise calculation based on the consumption of the device. To this end, it is proposed to make use of consumption measurement tools capable of reliably determining the daily consumption of the instrument. This would allow the design of a more efficient power supply module capable of being reduced in both cost and size.

### **8.5. Use of the instrument in other cases**

After having carried out, prior to the project, an extensive bibliographic review, it has been possible to observe different situations in which this instrument could make a great contribution to society.

Situations such as the conservation of different buildings or sculptures, which are part of society and sometimes part of the cultural, national or world heritage of bird droppings or nesting sites, could be solved by the proposed instrument.

On the other hand, the presence of birds near airports and runways is a commonly known problem. Falconry is defined as the activity of hunting with trained birds of prey, especially hawks, goshawks and other birds of prey, for the capture of birds of prey or ground species, a situation which, although it safeguards the take-off or landing of an aircraft, is detrimental to animal conservation. It is for this reason that the implementation of such systems could be considered.



# Bibliografía

- [1] *¿Qué es el Software Libre? - Proyecto GNU - Free Software Foundation.* URL: <https://www.gnu.org/philosophy/free-sw.es.html> (visitado 19-11-2021).
- [2] Isaac Pérez Borrero y col. *La inteligencia artificial nos ayuda a estudiar la fauna de Doñana para mejorar su conservación.* en. URL: <http://theconversation.com/la-inteligencia-artificial-nos-ayuda-a-estudiar-la-fauna-de-donana-para-mejorar-su-conservacion-162815> (visitado 19-11-2021).
- [3] DTBird Product Brochure. *A Self-Working System to Reduce Bird and Bat Mortality at Wind Farms.* 2013.
- [4] E Carletti. *Comunicación–Bus I2C. Descripción y funcionamiento.* 2014.
- [5] Inkscape Website Developers. *Draw Freely | Inkscape.* es. URL: <https://inkscape.org/es/> (visitado 19-11-2021).
- [6] Alfredo García Fernández. *¿Por qué mueren las aves en los tendidos eléctricos?* en. URL: <http://theconversation.com/por-que-mueren-las-aves-en-los-tendidos-electricos-160032> (visitado 19-11-2021).
- [7] *Filtrado de señal. Filtro RLC paso bajo de segundo orden.* URL: <https://wilaebaelectronica.blogspot.com/2017/01/filtro-pasa-bajos-pasivo-de-2do-orden-rlc.html> (visitado 04-03-2022).
- [8] *FreeCad - Diseño en 3D.* URL: <https://www.freecadweb.org/> (visitado 29-01-2022).
- [9] *Git Distribución de licencia de Git.* URL: <https://www.gnu.org/licenses/old-licenses/gpl-2.0.html> (visitado 29-01-2022).
- [10] *Git Software de control de versiones diseñado por Linus Torvalds.* URL: <https://git-scm.com/> (visitado 29-01-2022).
- [11] *History and License — Python 3.10.0 documentation.* URL: <https://docs.python.org/3/license.html> (visitado 19-11-2021).

- [12] jecrespom. *ICSP*. es. Nov. de 2016. URL: <https://aprendiendoarduino.wordpress.com/2016/11/06/icsp/> (visitado 02-12-2021).
- [13] *Kate - Get an Edge in Editing*. en. URL: <https://kate-editor.org/> (visitado 19-11-2021).
- [14] *KiCad EDA*. en-us. URL: <https://www.kicad.org/> (visitado 19-11-2021).
- [15] *Kile - an Integrated LaTeX Editing Environment*. URL: <https://kile.sourceforge.io/> (visitado 19-11-2021).
- [16] *LaTeX - A document preparation system*. URL: <https://www.latex-project.org/> (visitado 19-11-2021).
- [17] *Latex. Distribución de licencia LPPL de Latex*. URL: <https://www.latex-project.org/lppl/> (visitado 29-01-2022).
- [18] Jorge Machín Mindán. «Beamforming, localización de fuentes sonoras mediante arrays de micrófonos.» Tesis doct. Universitat Politècnica de València, 2016.
- [19] Eremita Miranda. «Animal intrusion in substations and transmission towers». En: *1999 IEEE Power Engineering Society Summer Meeting. Conference Proceedings (Cat. No.99CH36364)*. Vol. 1. 1999. DOI: [10.1109/PSS.1999.784365](https://doi.org/10.1109/PSS.1999.784365).
- [20] Azamjon Muminov y col. «Muminovetal2017.Pdf». En: *IEEE Xplore* (2017).
- [21] Guillermo-Eduardo Narváez. «Red neuronal convolucional para la detección de aves exóticas en peligro de extinción». En: (2020).
- [22] *PrusaSlicer - Impesión en 3D*. URL: [https://www.prusa3d.com/es/pagina/prusaslicer\\_424/](https://www.prusa3d.com/es/pagina/prusaslicer_424/) (visitado 29-01-2022).
- [23] *Redes Neuronales: una visión superficial - Fernando Sancho Caparrini*. URL: <http://www.cs.us.es/~fsancho/?e=72> (visitado 19-11-2021).
- [24] Julia Ruiz Cardón. «Detección automática de aves en las vías del AVE mediante análisis de vídeo». B.S. thesis. 2020.
- [25] *SEOBirdLife Tercera versión de las directrices para la evaluación del impacto de los parques eólicos en aves y murciélago*. URL: [https://www.seo.org/wp-content/uploads/2012/05/MANUAL-MOLINOS-VERSION-31\\_WEB.pdf](https://www.seo.org/wp-content/uploads/2012/05/MANUAL-MOLINOS-VERSION-31_WEB.pdf) (visitado 29-01-2022).
- [26] William W Seto. *Teoría y problemas de Acústica*. Inf. téc. 1973.
- [27] Yahot Siahaan, Bheta Agus Wardijono y Yulisdin Mukhlis. «Design of Birds Detector and Repellent Using Frequency Based Arduino Uno with Android System». En: *2 International Conferences on Information Technology, Information System and Electrical Engineering 06* (2017).

- [28] Varios Autores. *FlatCAM*. URL: <http://flatcam.org/>. (visitado 04-03-2022).
- [29] Yuxin Wu y col. *Detectron2*. 2019. URL: <https://github.com/facebookresearch/detectron2>.
- [30] Ayesha Younis y col. «Real-time object detection using pre-trained deep learning models MobileNet-SSD». En: *Proceedings of 2020 the 6th International Conference on Computing and Data Engineering*. 2020, págs. 44-48.





**Universidad  
de La Laguna**

**Escuela Superior  
de Ingeniería y Tecnología**

**Departamento de Ingeniería Industrial**

## **Trabajo de Fin de Grado**

**Protección a la Avifauna de la colisión contra  
estructuras peligrosas mediante señales ultrasónicas.**

### **TOMO II**

#### **Medición y presupuesto**

**Titulación:** Grado en Ingeniería Electrónica Industrial y  
Automática

#### **Estudiante:**

Rubén Morales Rodríguez

**Tutor:** Fernando Luis Rosa González

**Co-Tutor:** Iván Rodríguez Méndez

11 de marzo de 2022



# Capítulo 9

## Mediciones

En este capítulo se establecen las diferentes mediciones realizadas para llevar a cabo el diseño y fabricación del instrumento. Para realizar esto se tienen en cuenta tanto la medición de componentes necesarios para la obtención del dispositivo funcional, cómo el cálculo de las horas y tipo de las mismas necesarias para la obtención de dicho dispositivo.

### 9.1. Medición de componentes

La medición de los componentes se realiza ateniendo a los cálculos expuestos en el anexo **A**, en base a los descrito aquí y a las diferentes necesidades expuestas y justificadas a lo largo de la presente memoria, se seleccionan tanto la cantidad cómo la tipología de los componentes a usar.

### 9.2. Medición de horas de diseño y programación

Para el cálculo económico de las horas de diseño y programación se ha establecido un valor de 30 €, por hora dedicada a estas actividades. Por lo tanto ateniendo a esto, se puede establecer tal y cómo se hace en la tabla **9.1** el coste total de las horas de diseño y programación dedicadas al desarrollo de este proyecto.

<b>Medición presupuestaria relativa a las horas de diseño y programación</b>			
Concepto	Horas de dedicación	Precio (€/ Ud)	Precio total (€)
Diseño	100	30.00	3000.00
Programación	50	30.00	1500.00
<b>Subtotal</b>			4500.00

Tabla 9.1: Medición presupuestaria de las horas de diseño y programación

### 9.3. Medición de horas de fabricación y pruebas

En cuanto a las horas dedicadas a fabricación y realización de pruebas, se establece un precio de 25 €, por hora dedicada a este concepto. Atendiendo a esto se establece, tal y cómo se observa en la tabla 9.2 el coste total de la actividad realizada.

<b>Medición presupuestaria relativa a las de horas de fabricación y pruebas</b>			
Concepto	Horas de dedicación	Precio (€/ Ud)	Precio total (€)
Fabricación/pruebas	60	25.00	1500.00
<b>Subtotal</b>			1500.00

Tabla 9.2: Medición presupuestaria de las horas de fabricación y pruebas

### 9.4. Medición de horas de redacción de documentación

Por último, para el cálculo económico de las horas dedicadas a la redacción de la documentación, se ha establecido un valor de 20 €, por hora dedicada a este concepto. Por lo que en la tabla 9.3 se puede observar el coste total de la actividad realizada.



<b>Medición presupuestaria relativa a las horas de redacción de documentación</b>			
Concepto	Horas de dedicación	Precio (€/ Ud)	Precio total (€)
Redacción	90	20.00	1800.00
<b>Subtotal</b>			1800.00

Tabla 9.3: Medición presupuestaria de las horas de redacción de documentación

## 9.5. Total de horas medidas

En esta última sección se establece el computo total de las horas dedicadas al proyecto atendiendo a su tipología y con su precio correspondiente asociado. Tal y como se puede observar en la tabla 9.4 se detalla cada concepto y se establece un “Subtotal” del coste de las horas de dedicación.

<b>Medición presupuestaria relativa al total de horas</b>			
Concepto	Horas de dedicación	Precio (€/ Ud)	Precio total (€)
Horas de diseño	100	30.00	3000.00
Horas de Programación	50	30.00	1500.00
Horas de fabricación/pruebas	60	25.00	1500.00
Horas de redacción	90	20.00	1000.00
<b>Total de horas</b>	300.00	–	–
<b>Subtotal</b>			7000.00

Tabla 9.4: Medición presupuestaria de horas totales de dedicación



# Capítulo 10

## Presupuesto

En este capítulo se establece el presupuesto total del proyecto, donde además de las diferentes mediciones presupuestarias establecidas en el capítulo 9, se desglosan los costes materiales del mismo <sup>1</sup>.

### 10.1. Presupuesto de los componentes del módulo de alimentación

En la tabla 10.1 se establece el valor de los componentes necesarios para la fabricación del **módulo de alimentación**.

Componentes del módulo de alimentación				
Componente	Modelo	Precio (€/ Ud)	Unidad/es	Precio total (€)
Batería	AGM 12V 260Ah	322.00	1	322.00
Módulo Fotovoltaico	Mono Perc 200wp	129.00	1	129.00
Regulador de carga	PWM plus 20A 12/24V	39.00	1	39.00
<b>Subtotal</b>				490.00

Tabla 10.1: Presupuesto de los componentes del módulo de alimentación

---

<sup>1</sup>En los distintos costes se han tenido en cuenta los diferentes impuestos y cargas que proceden, siendo el valor que aparece en las tablas el “PVP” resultante de aplicar los correspondientes gravámenes.

## 10.2. Presupuesto de los componentes del módulo de procesamiento

En esta sección se establecen los costes de los componentes necesarios para la fabricación del **módulo de procesamiento**, dichos costes pueden observarse de manera detallada en la tabla 10.2.

Componentes del módulo de detección				
Componente	Modelo	Precio (€/ Ud)	Unidad/es	Precio total (€)
Raspberry	PI3	39.00	1	39.00
UPS	Raspberry PI3	20.50	1	20.50
<b>Subtotal</b>				59.50

Tabla 10.2: Presupuesto de los componentes del módulo de procesamiento

## 10.3. Presupuesto de los componentes del módulo de detección

En esta sección se establecen los costes de los componentes necesarios para la fabricación del **módulo de detección**, dichos costes pueden observarse de manera detallada en la tabla 10.3.

Componentes del módulo de detección				
Componente	Modelo	Precio (€/ Ud)	Unidad/es	Precio total (€)
Modulo Camara	Raspberry Pi V2	28.00	2	56.00
<b>Subtotal</b>				56.00

Tabla 10.3: Presupuesto de los componentes del módulo de detección

## 10.4. Presupuesto de los componentes del módulo de aviso

En la tabla 10.4 se establecen los costes de los componentes necesarios para la fabricación del **módulo de aviso**.

Componentes del módulo de evasión				
Componente	Modelo	Precio (€/ Ud)	Unidad/es	Precio total (€)
Condensador	1uF	0.25	12	3.00
Condensador	0.33uF	0.20	1	0.20
Condensador	22pF	0.37	2	0.37
Condensador	100nF	0.15	4	0.6
Condensador	0.1uF	0.2	1	0.2
Conector male	Bloque de 16 conectores	1.40	1	1.40
Screw terminal	2x terminal	0.45	6	2.70
Bobina	150uH	0.23	8	1.84
Bobina	100uH	0.20	4	0.80
Resistencia	1K	0.16	4	0.64
Resistencia	4.7K	0.13	2	0.26
Resistencia	100K	0.31	8	2.48
Resistencia	2.2K	0.17	4	0.68
Resistencia	68 $\Omega$	0.10	4	0.40
Resistencia	10K	0.13	4	0.52
Regulador de tensión	7805	2.52	1	2.52
Convertor	Convertor de nivel lógico	1.60	1	1.60
Microcontrolador	ATM328	2.88	1	2.88
Amplificador	Amplificador operacional	0.89	2	1.78
Tornillería	M3	12.50	1	12.50
Tornillería	M4	12.50	1	12.50
Puntas de conexión	0.5-1.5mm	1.89	2	3.78
Oscilador	Oscilador de cristal	1.85	1	1.85
ELevador de tensión	Step-Up (12-24)v	15.47	1	15.47
Placa de cobre	Placa para PCB	4.23	1	4.23
Cables de conexionado	Cables dupont	3.60	1	3.60
Filamento para impresión en 3D	20m de PLA	20.00	1	20.00
Cable	1m x Cable de 2 hilos	1.28	5	6.44
Emisores	8 Emisores piezoeléctricos	22.46	1	22.46
<b>Subtotal</b>				127.70

Tabla 10.4: Presupuesto de los componentes del módulo de aviso

## 10.5. Presupuesto del total de los componentes

En esta sección se establece el computo total de los costes de los componentes necesarios para la fabricación del instrumento, el cual se muestra en la tabla 10.5.

<b>Presupuesto total de los componentes</b>			
Módulo	Precio (€/ Ud)	Unidad/es	Precio total (€)
Módulo de alimentación	490.00	1	490.00
Módulo de procesamiento	59.50	1	59.50
Módulo de detección	67.00	1	56
Módulo de aviso	91.09	1	127.70
<b>Subtotal</b>			733.50
Uso de pequeña herramienta	3 %	1	21.99
Uso de maquinaria	5 %	1	36.67
<b>Total de fabricación</b>			791.86

Tabla 10.5: Presupuesto del total de los componentes

## 10.6. Presupuesto total del proyecto

En esta sección se establece el presupuesto total del proyecto donde se tienen en cuenta los diferentes conceptos, como horas de investigación, ejecución, redacción de documentación y programación así como el propio precio de los componentes obtenidos anteriormente. En la tabla 10.6 se puede observar con detalle el desglose y el coste total del proyecto.

<b>Presupuesto total del proyecto por conceptos</b>			
Concepto	Precio (€/ Ud)	Cantidad	Precio total (€)
Total de horas	7000.00	1	7000.00
Total de fabricación	791.86	1	791.86
<b>Total</b>			7791.86

Tabla 10.6: Presupuesto total del proyecto

Esto podría resumirse tal y como se hace en la siguiente tabla 10.7 donde se observa el coste final del instrumento diseñado.

<b>Presupuesto total del Instrumento</b>				
Concepto	Proveedor	Precio (€/ Ud)	Cantidad	Precio total (€)
Instrumento Avifauna	ULL (BioFiMSenDis)	7791.86	1	7791.86
<b>Total</b>				7791.86

Tabla 10.7: Presupuesto total del instrumento



**Universidad  
de La Laguna**

**Escuela Superior  
de Ingeniería y Tecnología**

**Departamento de Ingeniería Industrial**

## **Trabajo de Fin de Grado**

**Protección a la Avifauna de la colisión contra  
estructuras peligrosas mediante señales ultrasónicas.**

### **TOMO III**

#### **Anexos**

**Titulación:** Grado en Ingeniería Electrónica Industrial y  
Automática

#### **Estudiante:**

Rubén Morales Rodríguez

**Tutor:** Fernando Luis Rosa González

**Co-Tutor:** Iván Rodríguez Méndez

11 de marzo de 2022





# Anexo A

## Anexo de cálculos

En este anexo se establecen todos los cálculos necesarios para poder determinar los componentes a usar y así, poder establecer el diseño del dispositivo de manera justificada.

### A.1. Cálculos del módulo de aviso

En esta sección se establecen todos aquellos cálculos referentes al diseño electrónico necesario para llevar a cabo el módulo de aviso.

#### A.1.1. Filtro “RLC”

Al generar la señal de aviso en frecuencia ultrasónica mediante un microcontrolador “ATM328”, se obtiene la señal a la frecuencia deseada con una amplitud de 5V de forma cuadrada. Es por ello, por lo que se procede al filtrado de la misma, para obtener una señal a la frecuencia establecida lo más parecido a una señal senoidal que permita excitar el emisor. En este aspecto, es destacable el uso de la señal senoidal para la excitación del emisor ya que el estudio de este emisor, expuesto en el anexo B, se realizó utilizando este tipo de señales.

Para llevar a cabo el filtrado de esta señal, se diseña un filtro pasivo pasa baja de 2<sup>o</sup> orden (“RLC”) con el fin de no bloquear las frecuencias bajas para poder utilizar este mismo filtro con todas las frecuencias bajas que puede proporcionar el microcontrolador. Este filtro fue calculado atendiendo a las siguientes ecuaciones;

$$C = \frac{1}{2 \cdot \pi \cdot f_c \cdot k \cdot Q \cdot R} \quad (\text{A.1})$$

$$L = \frac{Q \cdot R}{2 \cdot \pi \cdot f_c \cdot k} \quad (\text{A.2})$$

Donde  $f_c = 35000\text{Hz}$ ,  $Q$  es el factor de calidad del filtro el cual siguiendo las aproximaciones de Butterworth se fija en  $Q = 0,7071$  mientras que  $K$  es una constante de cuyo valor es de 1.

Para realizar los cálculos correspondientes, atendiendo a lo anteriormente descrito se han fijado en primer lugar las condiciones de contorno. En dichas condiciones se fija el valor de la resistencia, atendiendo a los valores de las series de las que se dispone, siendo el valor fijado el de  $68\Omega$ . Habiendo fijado este valor se obtiene el valor del condensador, el cual es de  $95\text{ nF}$ , un valor muy proximo al valor comercial de  $100\text{ nF}$ . Por último se obtuvo el valor de la bobina “L”, dicho valor dió como resultado el valor de  $220\text{ }\mu\text{H}$ , este valor en la práctica ha de ser sustituido por el valor de  $250\text{ }\mu\text{H}$  debido a la carencia de valores comerciales que permitan establecer tal configuración. Por lo que atendiendo a esto se podría establecer que los valores de los componentes necesarios para llevar a cabo este filtro son los siguientes;

- $R = 68\Omega$
- $L = 250\text{ }\mu\text{H}$  (Compuesta por dos bobinas en serie)
  1.  $L = 100\text{ }\mu\text{H}$
  2.  $L = 150\text{ }\mu\text{H}$
- $C = 100\text{ nF}$

Esto se puede observar de una manera más gráfica en la figura A.1 donde se muestra el esquemático que representa el filtro diseñado.

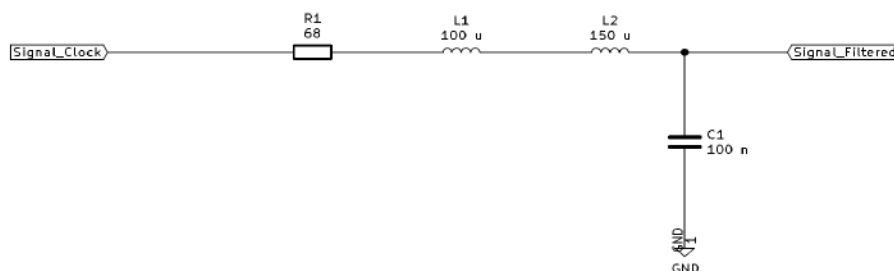


Figura A.1: Esquemático del filtro “RLC”

Imagen elaborada con KiCad.

### A.1.2. Amplificador de señal

Una vez filtrada la señal, se obtiene una forma de señal similar a la buscada, a una frecuencia deseada pero con una amplitud insuficiente para excitar a los emisores piezoeléctricos. Es por ello por lo que se procede a amplificar la señal mediante un “Amplificador operacional” en configuración no inversora. Dicha configuración se puede observar en la imagen A.2.

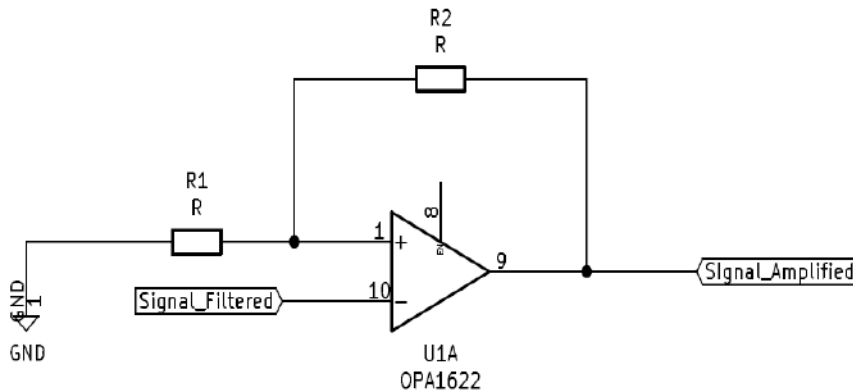


Figura A.2: Esquema de un amplificador operacional en configuración no inversora  
Imagen elaborada con KiCad.

Esta configuración de uso de un amplificador operacional otorga su ganancia en base a la siguiente ecuación A.3.

$$A = \left(1 + \frac{R_2}{R_1}\right) \quad (\text{A.3})$$

Atendiendo al estudio que se había realizado sobre el comportamiento eléctrico del piezoeléctrico se eligió el amplificado operacional “TCA0372”, el cual tiene su datasheet adjunto en el anexo D. En este se puede observar que es posible alimentar dicho “operacional” con  $\pm 12\text{V}$  lo cual se puede simplificar en una alimentación entre  $24\text{V}$  y  $GND$ . Sabiendo esto y atendiendo a que existe un voltaje de caída con respecto al de alimentación se fija una amplitud máxima de salida en  $20\text{V}$ . Lo cual se traduce en lo siguiente;

$$A = \frac{V_{out}}{V_{in}} \quad (\text{A.4})$$

Siendo el valor de  $V_{out} = 20$  y  $V_{in} = 5$ ;

$$A = 4$$

Por lo tanto si se fija  $R_1$  en  $2.2 \text{ K}\Omega$  se podría decir lo siguiente;

$$4 = \left(1 + \frac{R_2}{2.2}\right) \quad (\text{A.5})$$

$$R_2 = 6.6 \text{ k}\Omega$$

Este valor es sustituido en la práctica por un potenciómetro de  $10 \text{ K}\Omega$  que facilite el ajuste de la ganancia y permita obtener una mejor señal de salida.

Además de lo anteriormente desarrollado, en el circuito de amplificación se hacen uso de una serie de condensadores de valores típicos con el fin de eliminar cualquier componente de continua que pueda interferir en proceso de amplificación. Además de esto, se establece un partidor de tensión con el fin de “encuadrar” la señal y terminar obteniendo una señal amplificada con forma senoidal cuyo centro “pase por 0”. Esto se puede observar con mayor detalle en la figura A.3 en la cual se puede observar además, como a la salida del amplificador se ha conectado una bobina de  $150 \mu\text{H}$ . Esto último es debido a que en las pruebas realizadas en el laboratorio, se pudo comprobar que al conectar esta bobina en la salida del circuito, además de “suavizar” la señal aumentaba su amplitud, favoreciendo por tanto al resultado final obtenido.

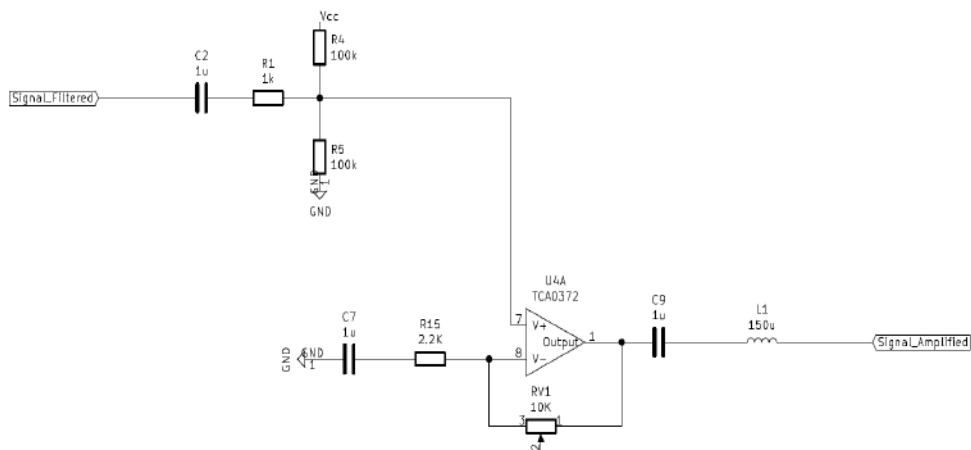


Figura A.3: Esquemático del circuito de amplificación

Imagen elaborada con KiCad.

## A.2. Cálculos del módulo de alimentación

En esta sección se establecen todos aquellos cálculos referentes al módulo de alimentación del instrumento a fabricar. Dicho módulo en la práctica es sustituido por una fuente de alimentación de 12V, pero por condiciones éticas, tal y como se cita en el apartado correspondiente de la memoria, se ha planteado el dimensionado de un sistema de alimentación basado en energías renovables.

Para hacer un correcto dimensionamiento se ha planteado el consumo máximo por parte del sistema atendiendo a las hojas de características de cada uno de los componentes, adjuntas en el anexo D y a los resultados de consumo obtenidos en las distintas pruebas de laboratorio. Además se plantea que el sistema de alimentación sea capaz en un solo día de satisfacer las necesidades energéticas del dispositivo durante 3 días. Esto se plantea, atendiendo a los estudios que afirman que en Canarias al menos 1 de cada 3 días se puede utilizar el sol como fuente de energía renovable.

$$Potencia = P \cdot H \quad (A.6)$$

Donde  $P = 20 W$  ,  $H = 24 Horas$

$$Potencia = 480 Wh$$

Resultado que al multiplicar por 3 días, se obtiene que;

$$Potencia_{requerida} = Potencia_{(Wh)} \cdot D \quad (A.7)$$

Donde  $Potencia_{(Wh)} = 480 Wh$  ,  $D = 3 Dias$

$$Potencia_{requerida} = 1440 Wh$$

### A.2.1. Cálculos justificativos de la batería usada

Una vez habiendo obtenido el dato de la potencia total requerida, en primer lugar se procede con el dimensionado de la batería. Para ello se fija que la profundidad de descarga de la batería no supere 50% lo que obliga a establecer el siguiente cálculo;

$$Energia_{(Wh)} = Potencia_{requerida} \cdot Prf \quad (A.8)$$

Donde  $Potencia_{requerida} = 1440 W$  ,  $Prf = 50\%$

$$Potencia_{total} = 2880 Wh$$

Esto se puede traducir también en lo siguiente;

$$\boxed{Potencia_{total} = 2.88 \text{ KWh}}$$

Habiendo obtenido este resultado y atendiendo a que el voltaje de salida de la batería que requiere el sistema propuesto es de 12V, podemos definir la capacidad de la batería en Ah de la siguiente manera;

$$CapacidadMinima = \frac{Potencia_{total}}{V} \quad (A.9)$$

Donde  $Potencia_{total} = 2880 \text{ W}$  ,  $V = 12 \text{ v}$

$$\boxed{CapacidadMinima = 240 \text{ Ah}}$$

### A.2.2. Cálculos justificativos del módulo fotovoltaico elegido

Partiendo de los datos obtenidos en las hojas de insolación expuestas en el anexo F, se puede observar como para la peor de las situaciones propuestas, el módulo seleccionado es capaz de otorgar una producción eléctrica media mensual de 21.81 Kwh en el peor mes del año. Esto se traduce en que para este caso la suma media mensual de la irradiación global recibida por metro cuadrado, por el número de módulos fotovoltaicos instalados es de  $132.9 \frac{KWh}{m^2}$ . Si se atiende a que esto es una producción mensual y a las dimensiones del módulo fotovoltaico se puede determinar de la siguiente manera la cantidad de energía que se podría obtener;

$$IrradiacinMediaDiaria = \frac{IrradiacinGlobalMedia \frac{Kwh}{m^2}}{D} \quad (A.10)$$

Donde  $IrradiacinGlobalMedia = 132,9 \frac{Kwh}{m^2}$  ,  $D = 31 \text{ Dias}$

$$\boxed{IrradiacinMediaDiaria = 4.29 \frac{KWh}{m^2}}$$

Atendiendo a las dimensiones del modulo seleccionado;

$$EnergiaObtenida = IrradiacinMediaDiaria \frac{Kwh}{m^2} \cdot Superficie \quad (A.11)$$

Donde  $IrradiacinMediaDiaria = 4,29 \frac{KWh}{m^2}$  ,  $S = 1,02 \text{ m}^2$

$$\boxed{EnergiaObtenida = 4.37 \text{ KWh}}$$

Se puede determinar por tanto, que dicho módulo fotovoltaico es capaz de otorgar una potencia diaria superior a la requerida en el peor mes del año, y en el peor, energeticamente hablando, de los emplazamientos elegidos. Por lo que atendiendo a esto, y a la capacidad mínima que debe tener la batería se puede determinar que ambos elementos elegidos y cuyos datos técnicos están adjuntos en el anexo **D**, cumplen las condiciones de dimensionado. Por otro lado y atendiendo nuevamente a dichas fichas técnicas se elige el regulador de carga teniendo en cuenta los valores de corriente y tensión necesarios para la correcta elección de este.





## Anexo B

# Caracterización del Piezoeléctrico usado

En este anexo se adjunta toda aquella información relativa a los materiales piezoeléctricos que han sido usados en el presente proyecto, así como los resultados obtenidos al realizar diferentes ensayos con estos.

### B.1. Analisis de un material piezoeléctrico

Desde 1987 las relaciones constitutivas de un material piezoeléctrico, para pequeñas condiciones de campo, fueron establecidas de la siguiente manera por el Instituto de Electricidad y Electrónica (ieee) en términos de constantes piezoeléctricas, de permitividad y de deformación elástica.

$$\begin{pmatrix} D_1 \\ D_2 \\ D_3 \\ \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \varepsilon_4 \\ \varepsilon_5 \\ \varepsilon_6 \end{pmatrix} = \begin{pmatrix} e_{11}^\sigma & 0 & 0 & 0 & 0 & 0 & 0 & d_{15}^d & 0 \\ 0 & e_{22}^\sigma & 0 & 0 & 0 & 0 & d_{24}^d & 0 & 0 \\ 0 & 0 & e_{33}^\sigma & d_{31}^d & d_{32}^d & d_{33}^d & 0 & 0 & 0 \\ 0 & 0 & d_{31}^c & \delta_{11} & \delta_{12} & \delta_{13} & 0 & 0 & 0 \\ 0 & 0 & d_{32}^c & \delta_{12} & \delta_{22} & \delta_{23} & 0 & 0 & 0 \\ 0 & 0 & d_{33}^c & \delta_{13} & \delta_{23} & \delta_{33} & 0 & 0 & 0 \\ 0 & d_{24}^c & 0 & 0 & 0 & 0 & \delta_{44} & 0 & 0 \\ d_{15}^c & 0 & 0 & 0 & 0 & 0 & 0 & \delta_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \delta_{66} \end{pmatrix} \cdot \begin{pmatrix} E_1 \\ E_2 \\ E_3 \\ \sigma_{11} \\ \sigma_{22} \\ \sigma_{33} \\ \sigma_{23} \\ \sigma_{31} \\ \sigma_{12} \end{pmatrix} \quad (\text{B.1})$$

Como se desarrolla anteriormente en el modelo matemático, el efecto piezoeléctrico describe la propiedad de determinados sólidos para generar una carga eléctrica bajo tensión mecánica. La fuerza de actuación modifica la estructura microscópica del cuerpo creando dipolos entre los que se acumula una tensión. Por

otro lado es posible realizar esto de manera inversa, al aplicar una tensión, los materiales piezoeléctricos (PE) pueden deformarse elásticamente. Esto último se realizó en los ensayos de laboratorio y que justifica el los procedimientos realizados en el capítulo 5.

Atendiendo a esto se podría que un material tiene características piezoeléctricas cuando al ser sometido a compresiones o deformaciones mecánicas (además de variar sus dimensiones físicas de acuerdo con las leyes de la elasticidad), produce una polarización eléctrica. si un material piezoeléctrico, es sometido por ejemplo, a vibraciones acústicas, generará entre sus caras un campo eléctrico de la misma frecuencia que la vibración (así funcionan los micrófonos piezoeléctricos, y a la inversa los parlantes de agudos o tweeters). La respuesta en frecuencia de estos materiales no es plana, es decir, hay ciertas frecuencias para las cuales la transferencia de energía “eléctricomecánica” es máxima, y otras para las cuales ésta es mínima, situación que se pudo demostrar en el laboratorio observando como la señal del emisor aumentaba o disminuía dependiendo de la frecuencia a la que se emitía. Esto depende en si del “piezo” en cuestión, cada piezo tiene su propias características que difieren en función de su tamaño y espesor. Por lo tanto un piezo podría tener una respuesta en frecuencia muy debil ante la entrada de una señal mientras que otro podría tener una gran respuesta ante el mismo estímulo. Cada piezoeléctrico tiene un factor de calidad denominado “ $Q$ ” diferente, el cual es inversamente proporcional al ancho de la resonancia. Un valor grande este factor de calidad significa que la resonancia es muy estrecha y la estabilidad del resonador es bastante grande. En este sentido, el piezoeléctrico se comporta de manera análoga a un circuito eléctrico RLC(figura B.1) sometido a una excitación de corriente alterna.

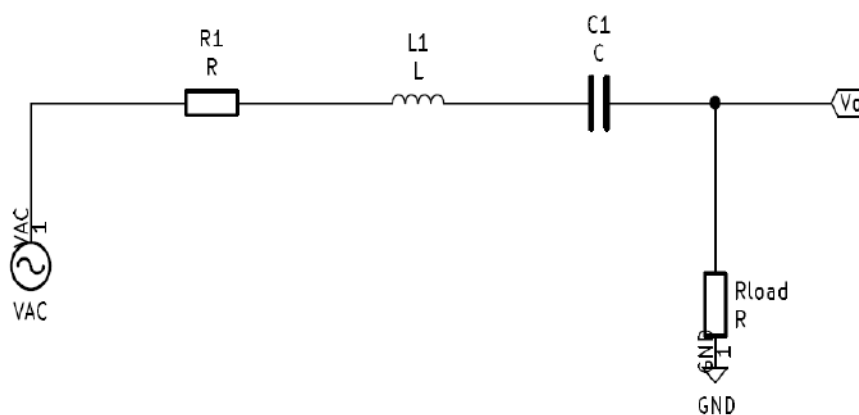


Figura B.1: Circuito equivalente de un piezoeléctrico  
Imagen elaborada con KiCad.

La analogía es clara si se recuerda que la ecuación de movimiento de un cuerpo vibrando debido a la acción de una fuerza externa dependiente del tiempo es;

$$F(t) = m \cdot \frac{d^2x}{dt^2} + b \cdot \frac{dx}{dt} + k \cdot x \quad (\text{B.2})$$

### B.1.1. El efecto de los terminales

Además de lo anteriormente citado y los datos que se podrían extraer de dicho circuito cabe destacar el efecto de los terminales en el cristal. El circuito eléctrico equivalente a tener en cuenta es un poco más complicado que el de la figura B.1, ya que los dos electrodos que están sobre el piezoeléctrico funcionan como una capacidad adicional, en paralelo. Es por esto que en el circuito eléctrico equivalente hay que incluir este efecto, como una capacidad  $C_2$  en paralelo con el piezoeléctrico siendo el circuito equivalente final como el que se muestra en la figura B.2.

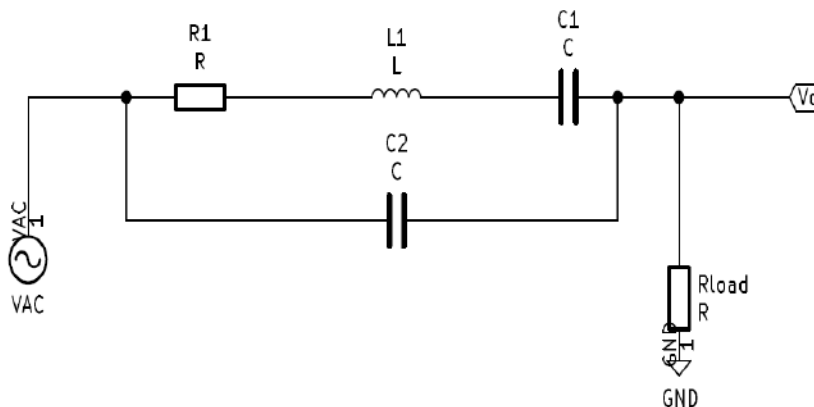


Figura B.2: Circuito equivalente de un piezoeléctrico atendiendo a los efectos de sus terminales

Imagen elaborada con KiCad.

## B.2. Experimento Realizado

Teniendo en cuenta lo anteriormente citado se procedió a realizar un ensayo de laboratorio con el fin de determinar diversas características del piezoeléctrico y obtener una mayor información del mismo. En este aspecto, determinar la

impedancia se consideró fundamental, ya que es un valor que depende de la frecuencia y obteniendo esta a la frecuencia de puesta en servicio se podría conocer la información necesaria para llevar a cabo la etapa de amplificación de señal.

Para llevar a cabo este ensayo se reprodujo el montaje que se puede observar en la figura B.3. Donde los valores de  $R_1$  y  $R_2$  eran valores conocidos ( $10\text{K}\Omega$ ) y el generador de funciones otorgaba una señal senoidal que variaba en frecuencia.

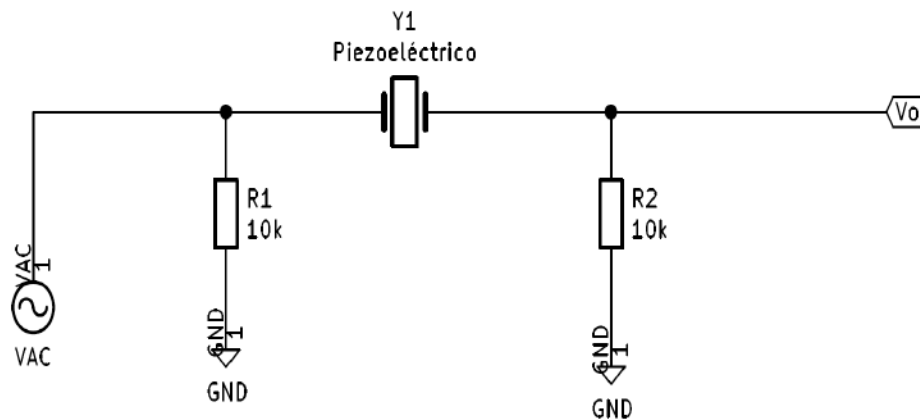


Figura B.3: Esquema del montaje del experimento realizado  
Imagen elaborada con KiCad.

### B.3. Tratamiento de los datos

Según se obtenían los datos de cada medida con el osciloscopio se extraían dichos datos, mediante un *script* de python que se adjunta en el anexo G, en un fichero “.csv” para ser tratados posteriormente. Una vez tratados todos los datos se generó de nuevo un fichero “.csv” con los cálculos realizados sobre cada medición y las diferentes gráficas representativas de los datos tratados. Para el tratamiento de los datos se utilizó el *script* que se adjunta en el anexo G, con el cual se realizan diferentes acciones para obtener el resultado final. Y otro *script* también adjunto en el anexo citado, con el cual se obtienen de manera gráfica estos datos, permitiendo al lector tener una mejor visión sobre los resultados obtenidos en el ensayo.

#### B.3.1. Operaciones realizadas con los datos extraídos del ensayo

En primer lugar se realizó la media de las señales obtenidas y se operó con los datos de cada señal y su correspondientes medias con el fin de obtener ambas

señales centradas en el valor “0” del eje  $x$ . Esto permitió realizar el tratamiento de manera fiable, en lo que a la elección de las amplitudes obtenidas se refería.

Posteriormente se trataron ambas señales con sus medias para eliminar el ruido y terminar obteniendo las amplitudes de las señales sabiendo que;

$$A_1 = \sqrt{\int (x^2) \cdot \frac{2}{N}} \quad (\text{B.3})$$

$$A_2 = \sqrt{\int (y^2) \cdot \frac{2}{N}} \quad (\text{B.4})$$

Siendo  $N$  = Número de medidas.

Teniendo en cuenta esto y operando con las amplitudes obtenidas y las señales tratadas se pudo determinar que la fase del sistema representado mediante las curva de *Lissajous* respondía a lo siguiente;

$$\phi = \arccos\left(\int (x \cdot y) \cdot \frac{2}{N}\right) \quad (\text{B.5})$$

Siendo  $N$  = Número de medidas.

Por lo que habiendo tratado los datos de esta forma, se pudo proceder a operar con ellos habiendo generado previamente un fichero de datos que permitiese tener estos ordenados de una manera correcta.

### B.3.2. Resultados gráficos del tratamiento de los datos

Una vez realizado lo anteriormente explicado, se obtienen los resultados de manera gráfica mediante el *script* adjunto en el anexo **G** de la presente memoria. En la imagen **B.4** podemos observar la relación existente entre la impedancia del piezo y la frecuencia, según aumenta esta última disminuye la primera. Por otro lado en la figura **B.5** se observa de manera logarítmica la ganancia del sistema a las diferentes frecuencias de estudio, o lo que es lo mismo, se puede observar el diagrama de bode del sistema. Y por último, en la última gráfica, que se observa en la figura **B.6**, se observa como cambia la fase del sistema a las diferentes frecuencias de estudio.

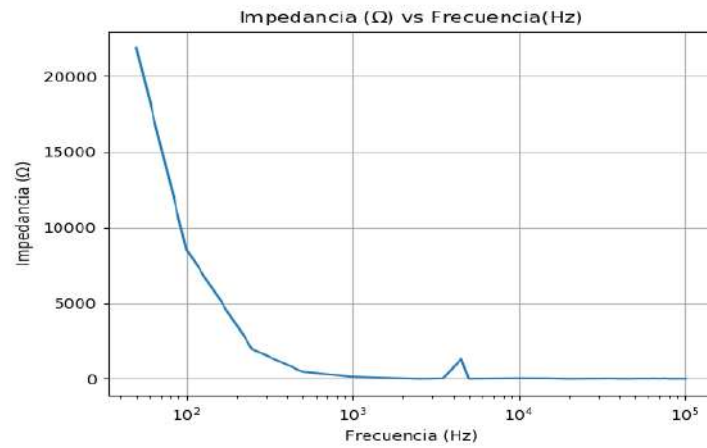


Figura B.4: Grafica de la impedancia del piezoeléctrico frente a la frecuencia de la señal de entrada

Imagen elaborada mediante pyton

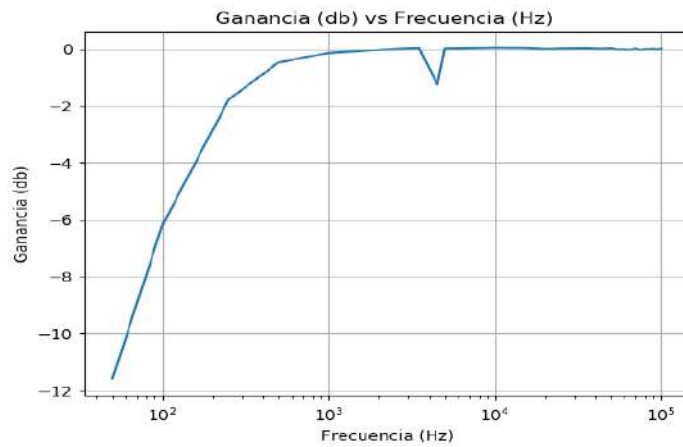


Figura B.5: Grafica de la ganancia del piezoeléctrico frente a la frecuencia de la señal de entrada

Imagen elaborada mediante pyton

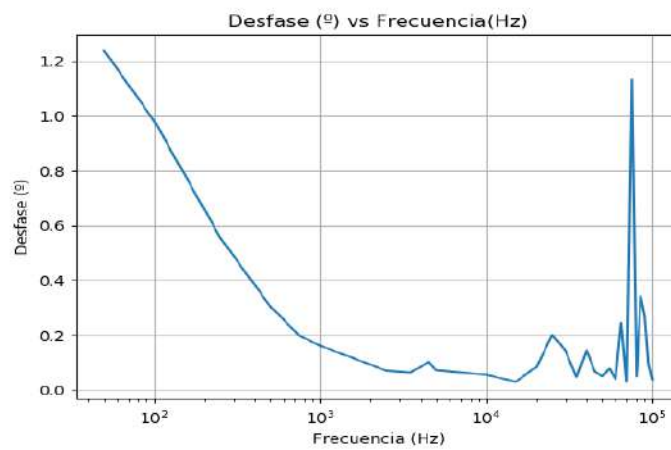


Figura B.6: Grafica de la fase frente a la frecuencia de la señal de entrada  
Imagen elaborada mediante python





# Anexo C

## Simulaciones realizadas

En este anexo se establecen las comparativas entre los diferentes modelos pre-entrenados de redes neuronales que se han estudiado, con el fin de poder comparar los niveles de confianza de cada uno, pudiendo validar con ello el modelo de red neuronal implementado en el sistema.

### C.1. Simulaciones gráficas de imágenes reales de aves

En esta sección del anexo se adjuntan los resultados gráficos obtenidos tras realizar las simulaciones pertinentes usando imágenes reales de aves en sus habitats naturales. La totalidad y veracidad de estas imágenes han sido cedidas por la sociedad española de ornitología, SEO/BirdLife.



Figura C.1: Ejemplar de Agteador Euroasiático  
Imagen Cedida por SEO/BirdLife.



Figura C.2: Detección como “Pájaro” en una imagen de un Agateador Euroasiático

Imagen obtenida al hacer uso del modelo “Birdwatcher”.



Figura C.3: Detección como “Pájaro” en una imagen de un Agateador Euroasiático

Imagen obtenida al hacer uso del modelo “Detectron2”.



Figura C.4: Ejemplar de Aguila Real volando  
Imagen Cedida por SEO/BirdLife.



Figura C.5: Detección como “Pájaro” en una imagen de un Aguila real

Imagen obtenida al hacer uso del modelo “Birdwatcher”.



Figura C.6: Detección como “Pájaro” en una imagen de un Aguila real

Imagen obtenida al hacer uso del modelo “Detectron2”.



Figura C.7: Ejemplar de un Alcaudón chico

Imagen Cedida por SEO/BirdLife.



Figura C.8: Detección como “Pájaro” en una imagen de un Alcaudón chico  
Imagen obtenida al hacer uso del modelo “Birdwatcher”.

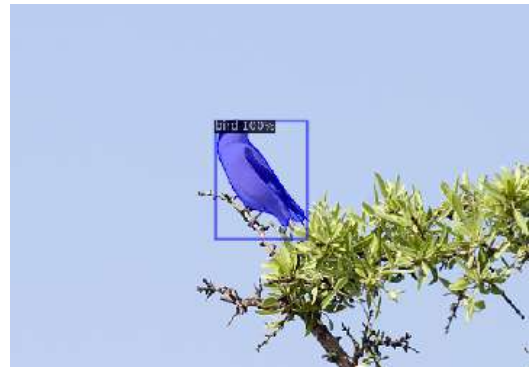


Figura C.9: Detección como “Pájaro” en una imagen de un Alcaudón chico  
Imagen obtenida al hacer uso del modelo “Detectron2”.



Figura C.10: Ejemplar de un Avefría Europea volando  
Imagen Cedida por SEO/BirdLife.



Figura C.11: Detección como “Pájaro” en una imagen de un Avefría Europea volando

Imagen obtenida al hacer uso del modelo “Birdwatcher”.



Figura C.12: Detección como “Pájaro” en una imagen de un Avefría Europea volando

Imagen obtenida al hacer uso del modelo “Detectron2”.



Figura C.13: Ejemplar de un Alcaudón dosirrojo

Imagen Cedida por SEO/BirdLife.



Figura C.14: Detección como “Pájaro”  
en una imagen de un Alcaudón dosirrojo  
Imagen obtenida al hacer uso del modelo  
“Birdwatcher”.

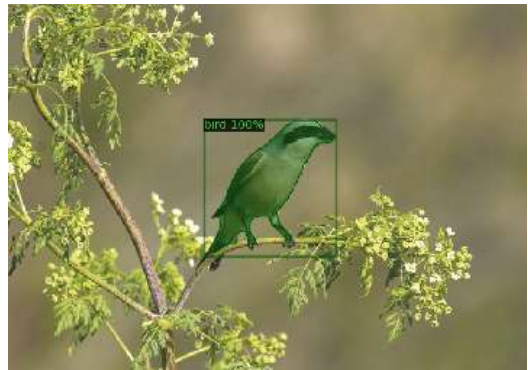


Figura C.15: Detección como “Pájaro”  
en una imagen de un Alcaudón dosirrojo  
Imagen obtenida al hacer uso del modelo  
“Detectron2”.



Figura C.16: Ejemplar de un Cernícalo vulgar volando  
Imagen Cedida por SEO/BirdLife.



Figura C.17: Detección como “Pájaro” en una imagen de un Cernícalo vulgar volando

Imagen obtenida al hacer uso del modelo “Birdwatcher”.

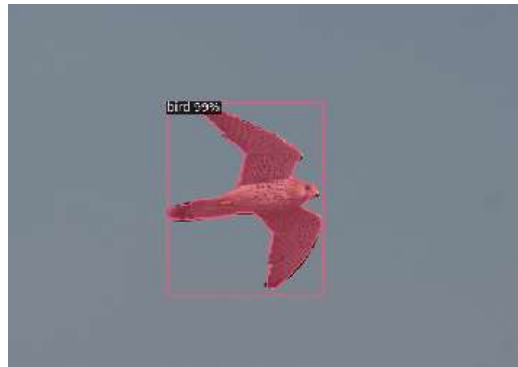


Figura C.18: Detección como “Pájaro” en una imagen de un Cernícalo vulgar volando

Imagen obtenida al hacer uso del modelo “Detectron2”.



Figura C.19: Ejemplar de un Cernícalo vulgar

Imagen Cedida por SEO/BirdLife.



Figura C.20: Detección como “Pájaro”  
en una imagen de un Cernícalo vulgar  
Imagen obtenida al hacer uso del modelo  
“Birdwatcher”.



Figura C.21: Detección como “Pájaro”  
en una imagen de un Cernícalo vulgar  
Imagen obtenida al hacer uso del modelo  
“Detectron2”.



Figura C.22: Ejemplar de un Charrán común volando  
Imagen Cedida por SEO/BirdLife.





Figura C.23: Detección como “Pájaro” en una imagen de un Charrán común volando

Imagen obtenida al hacer uso del modelo “Birdwatcher”.

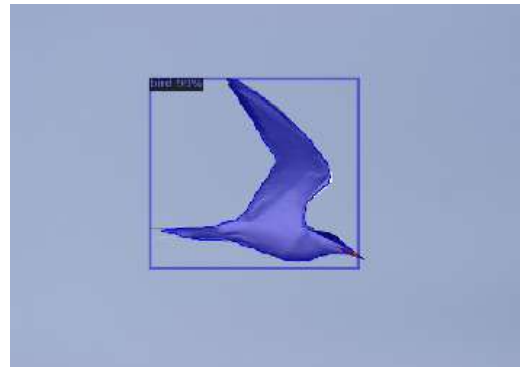


Figura C.24: Detección como “Pájaro” en una imagen de un Charrán común volando

Imagen obtenida al hacer uso del modelo “Detectron2”.



Figura C.25: Ejemplar de un Camachuelo trompetero

Imagen Cedida por SEO/BirdLife.



Figura C.26: Detección como “Pájaro” en una imagen de un Camachuelo trompetero

Imagen obtenida al hacer uso del modelo “Birdwatcher”.

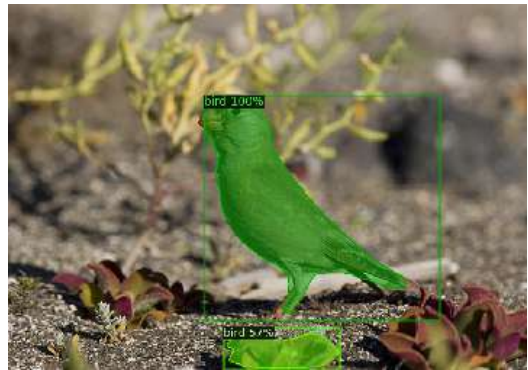


Figura C.27: Detección como “Pájaro” en una imagen de un Camachuelo trompetero

Imagen obtenida al hacer uso del modelo “Detectron2”.



Figura C.28: Ejemplar de un Chorlito gris volando

Imagen Cedida por SEO/BirdLife.



Figura C.29: Detección como “Pájaro” en una imagen de un Chorlito gris volando

Imagen obtenida al hacer uso del modelo “Birdwatcher”.

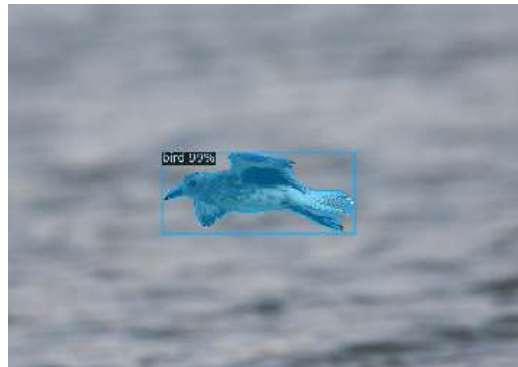


Figura C.30: Detección como “Pájaro” en una imagen de un Chorlito gris volando

Imagen obtenida al hacer uso del modelo “Detectron2”.



Figura C.31: Ejemplar de un Correlimos gordo

Imagen Cedida por SEO/BirdLife.



Figura C.32: Detección como “Pájaro” en una imagen de un Correlimos gordo  
Imagen obtenida al hacer uso del modelo “Birdwatcher”.

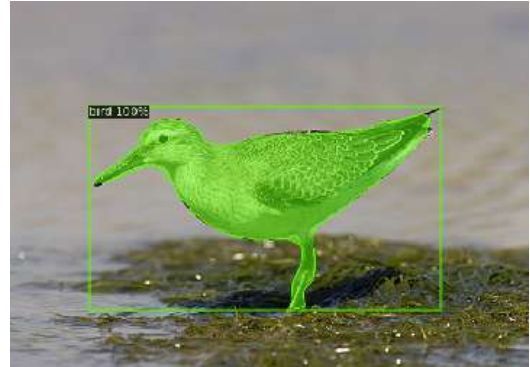


Figura C.33: Detección como “Pájaro” en una imagen de un Correlimos gordo  
Imagen obtenida al hacer uso del modelo “Detectron2”.



Figura C.34: Ejemplar de un Correlimos de temminck volando  
Imagen Cedida por SEO/BirdLife.



Figura C.35: Detección como “Pájaro” en una imagen de un Correlimos de teminck volando

Imagen obtenida al hacer uso del modelo “Birdwatcher”.



Figura C.36: Detección como “Pájaro” en una imagen de un Correlimos de teminck volando

Imagen obtenida al hacer uso del modelo “Detectron2”.



Figura C.37: Ejemplar de Escribano hortelano

Imagen Cedida por SEO/BirdLife.



Figura C.38: Detección como “Pájaro”  
en una imagen de un Escribano hortelano  
Imagen obtenida al hacer uso del modelo  
“Birdwatcher”.



Figura C.39: Detección como “Pájaro”  
en una imagen de un Escribano hortelano  
Imagen obtenida al hacer uso del modelo  
“Detectron2”.



Figura C.40: Ejemplar de Culebrera Europea volando  
Imagen Cedida por SEO/BirdLife.



Figura C.41: Detección como “Pájaro” en una imagen de una Culebrera Europea volando

Imagen obtenida al hacer uso del modelo “Birdwatcher”.

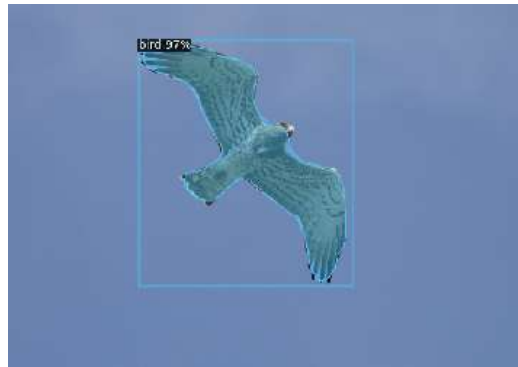


Figura C.42: Detección como “Pájaro” en una imagen de una Culebrera Europea volando

Imagen obtenida al hacer uso del modelo “Detectron2”.



Figura C.43: Ejemplar de Falaropo picofino

Imagen Cedida por SEO/BirdLife.



Figura C.44: Detección como “Pájaro”  
en una imagen de un Falaropo picofino  
Imagen obtenida al hacer uso del modelo  
“Birdwatcher”.



Figura C.45: Detección como “Pájaro”  
en una imagen de un Falaropo picofino  
Imagen obtenida al hacer uso del modelo  
“Detectron2”.



Figura C.46: Ejemplar de Ostrero Euroasiático volando  
Imagen Cedida por SEO/BirdLife.





Figura C.47: Detección como “Pájaro” en una imagen de un Ostrero Euroasiático volando

Imagen obtenida al hacer uso del modelo “Birdwatcher”.

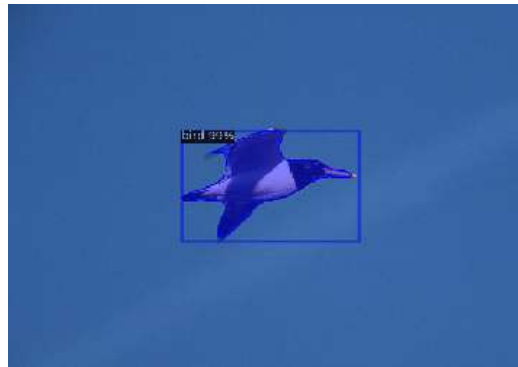


Figura C.48: Detección como “Pájaro” en una imagen de un Ostrero Euroasiático volando

Imagen obtenida al hacer uso del modelo “Detectron2”.



Figura C.49: Ejemplar de Torcecuellos Euroasiático

Imagen Cedida por SEO/BirdLife.



Figura C.50: Detección como “Pájaro” en una imagen de un Torcecuellos Euroasiático

Imagen obtenida al hacer uso del modelo “Birdwatcher”.



Figura C.51: Detección como “Pájaro” en una imagen de un Torcecuellos Euroasiático

Imagen obtenida al hacer uso del modelo “Detectron2”.



Figura C.52: Ejemplar de Pagalo pomarino volando

Imagen Cedida por SEO/BirdLife.



Figura C.53: Detección como “Pájaro” en una imagen de un Pagalo pomarino volando

Imagen obtenida al hacer uso del modelo “Birdwatcher”.

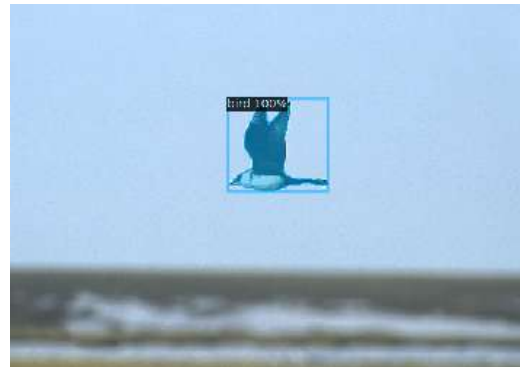


Figura C.54: Detección como “Pájaro” en una imagen de un Pagalo pomarino volando

Imagen obtenida al hacer uso del modelo “Detectron2”.



Figura C.55: Ejemplar de Vuelvepiedras común

Imagen Cedida por SEO/BirdLife.



Figura C.56: Detección como “Pájaro” en una imagen de un Vuelvepiedras común

Imagen obtenida al hacer uso del modelo “Birdwatcher”.

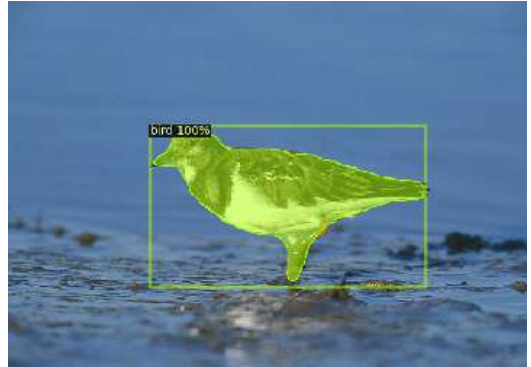


Figura C.57: Detección como “Pájaro” en una imagen de un Vuelvepiedras común

Imagen obtenida al hacer uso del modelo “Detectron2”.



Figura C.58: Ejemplar de Quebrantahuesos volando

Imagen Cedida por SEO/BirdLife.



Figura C.59: Detección como “Pájaro” en una imagen de un Quebrantahuesos volando

Imagen obtenida al hacer uso del modelo “Birdwatcher”.



Figura C.60: Detección como “Pájaro” en una imagen de un Quebrantahuesos volando

Imagen obtenida al hacer uso del modelo “Detectron2”.

## C.2. Simulaciones gráficas de imágenes de diferentes animales

En esta sección del anexo se adjuntan las simulaciones gráficas realizadas sobre imágenes en las que no aparecen aves pero sí otros animales en diferentes situaciones. Esto se lleva a cabo con el fin de comprobar la eficacia del modelo implementado en el sistema a la hora de diferenciar diferentes especies de animales. A su vez se analizan varios paisajes con el fin de comprobar si es capaz de diferenciar si hay presencia o no de algún animal en el.



Figura C.61: Imagen de un gato sin especificar  
Imagen obtenida de “Pexels”.



Figura C.62: Detección como “gato” en una imagen de un gato sin especificar  
Imagen obtenida al hacer uso del modelo “Birdwatcher”.



Figura C.63: Detección como “gato” en una imagen de un gato sin especificar  
Imagen obtenida al hacer uso del modelo “Detectron2”.

C.2. SIMULACIONES GRÁFICAS DE IMÁGENES DE DIFERENTES ANIMALES 155



Figura C.64: Imagen de un gato sin especificar  
Imagen obtenida de "Pexels".



Figura C.65: Detección como "gato" en  
una imagen de un gato sin especificar  
Imagen obtenida al hacer uso del modelo  
"Birdwatcher".

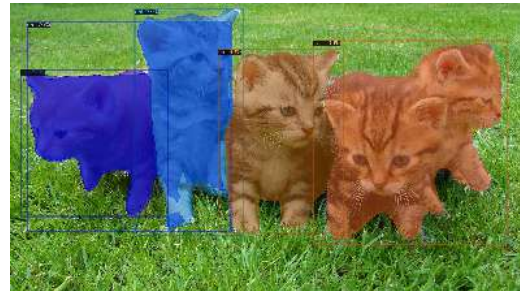


Figura C.66: Detección como "gato" en  
una imagen de un gato sin especificar  
Imagen obtenida al hacer uso del modelo  
"Detectron2".



Figura C.67: Imagen de un gato sin especificar  
Imagen obtenida de "Pexels".





Figura C.68: Detección como “gato” en una imagen de un gato sin especificar  
Imagen obtenida al hacer uso del modelo “Birdwatcher”.



Figura C.69: Detección como “gato” en una imagen de un gato sin especificar  
Imagen obtenida al hacer uso del modelo “Detectron2”.



Figura C.70: Imagen de un perro sin especificar  
Imagen obtenida de “Pexels”.



C.2. SIMULACIONES GRÁFICAS DE IMÁGENES DE DIFERENTES ANIMALES 157



Figura C.71: Detección como “perro” en una imagen de un perro sin especificar  
Imagen obtenida al hacer uso del modelo “Birdwatcher”.



Figura C.72: Detección como “perro” en una imagen de un perro sin especificar  
Imagen obtenida al hacer uso del modelo “Detectron2”.



Figura C.73: Imagen de un perro sin especificar  
Imagen obtenida de “Pexels”.



Figura C.74: Detección como “perro” en una imagen de un perro sin especificar  
Imagen obtenida al hacer uso del modelo “Birdwatcher”.



Figura C.75: Detección como “perro” en una imagen de un perro sin especificar  
Imagen obtenida al hacer uso del modelo “Detectron2”.



Figura C.76: Imagen de un perro sin especificar  
Imagen obtenida de “Pexels”.

C.2. SIMULACIONES GRÁFICAS DE IMÁGENES DE DIFERENTES ANIMALES 159



Figura C.77: Detección como “perro” en una imagen de un perro sin especificar  
Imagen obtenida al hacer uso del modelo “Birdwatcher”.



Figura C.78: Detección como “perro” en una imagen de un perro sin especificar  
Imagen obtenida al hacer uso del modelo “Detectron2”.



Figura C.79: Imagen de tortugas sin especificar  
Imagen obtenida de “Pexels”.



Figura C.80: Detección como “tortuga”  
en una imagen de tortugas sin especificar  
Imagen obtenida al hacer uso del modelo  
“Birdwatcher”.

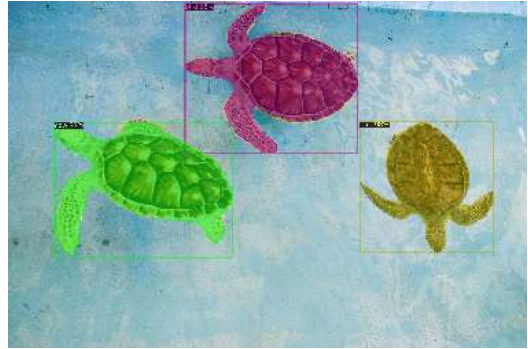


Figura C.81: Detección como “tortuga”  
en una imagen de tortugas sin especificar  
Imagen obtenida al hacer uso del modelo  
“Detectron2”.



Figura C.82: Imagen de tortugas sin especificar  
Imagen obtenida de “Pexels”.

C.2. SIMULACIONES GRÁFICAS DE IMÁGENES DE DIFERENTES ANIMALES 161



Figura C.83: Detección como “tortuga”  
en una imagen de tortugas sin especificar  
Imagen obtenida al hacer uso del modelo  
“Birdwatcher”.

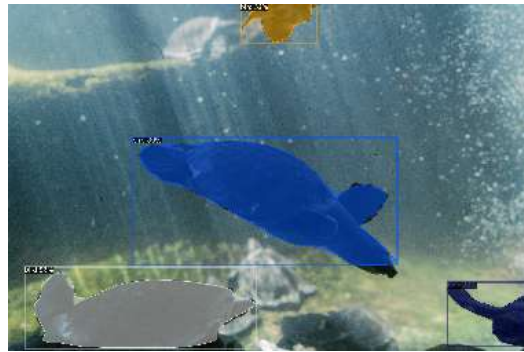


Figura C.84: Detección como “tortuga”  
en una imagen de tortugas sin especificar  
Imagen obtenida al hacer uso del modelo  
“Detectron2”.



Figura C.85: Imagen de un paisaje sin especificar  
Imagen obtenida de “Pexels”.





Figura C.86: Detección en un paisaje sin especificar

Imagen obtenida al hacer uso del modelo "Birdwatcher".



Figura C.87: Detección en un paisaje sin especificar

Imagen obtenida al hacer uso del modelo "Detectron2".



Figura C.88: Imagen de un paisaje sin especificar

Imagen obtenida de "Pexels".

C.2. SIMULACIONES GRÁFICAS DE IMÁGENES DE DIFERENTES ANIMALES163



Figura C.89: Detección en un paisaje sin especificar

Imagen obtenida al hacer uso del modelo  
“Birdwatcher”.



Figura C.90: Detección en un paisaje sin especificar

Imagen obtenida al hacer uso del modelo  
“Detectron2”.

### C.3. Simulaciones gráficas de imágenes tratadas

En esta sección del anexo se adjuntan una serie de imágenes de aves obtenidas de internet, de las cuales no se puede afirmar su veracidad ni que muestren a las aves en su habitat natural. Dichas imágenes fueron sometidas a distintos filtros de distorsión, cubismo, pixelado, zoom, sombreado, cambio de color y efecto solar entre otros y luego analizadas por los modelos de inteligencia artificial estudiados en este proyecto. Esto se realiza con el fin de comprobar la eficiencia del sistema ante las posibles perturbaciones que pudiese sufrir la captación de imágenes, como podrían ser las inclemencias meteorológicas.



Figura C.91: Imagen de un pajaros sin especificar  
Imagen obtenida de "Pexels" y tratada con "Gimp".



Figura C.92: Detección como "Pájaro" en una imagen de un pajaros sin especificar

Imagen obtenida al hacer uso del modelo "Birdwatcher".

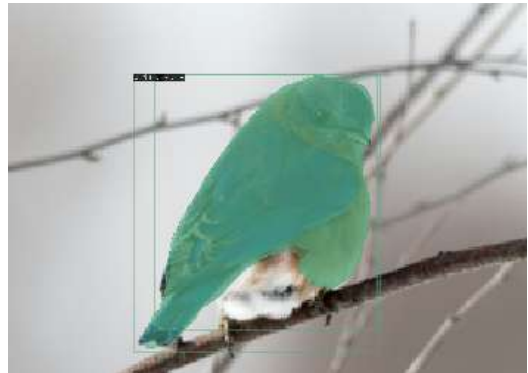


Figura C.93: Detección como "Pájaro" en una imagen de un pajaros sin especificar

Imagen obtenida al hacer uso del modelo "Detectron2".





Figura C.94: Imagen de un pájaro sin especificar  
Imagen obtenida de “Pexels” y tratada con “Gimp”.



Figura C.95: Detección como “Pájaro”  
en una imagen de un pájaro sin especificar

Imagen obtenida al hacer uso del modelo  
“Birdwatcher”.

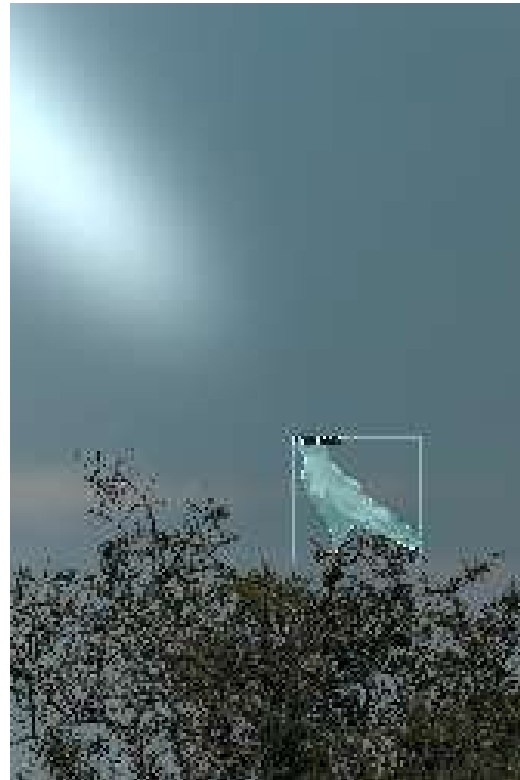


Figura C.96: Detección como “Pájaro”  
en una imagen de un pájaro sin especificar

Imagen obtenida al hacer uso del modelo  
“Detectron2”.



Figura C.97: Imagen de un pajaros sin especificar  
Imagen obtenida de "Pexels" y tratada con "Gimp".



Figura C.98: Detección como "Pájaro"  
en una imagen de un pajaros sin especificar

Imagen obtenida al hacer uso del modelo  
"Birdwatcher".



Figura C.99: Detección como "Pájaro"  
en una imagen de un pajaros sin especificar

Imagen obtenida al hacer uso del modelo  
"Detectron2".

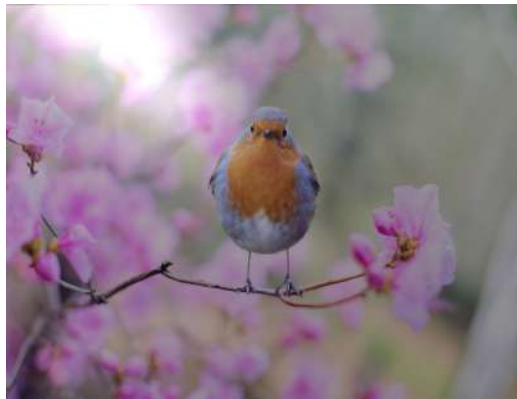


Figura C.100: Imagen de un pájaro sin especificar  
Imagen obtenida de “Pexels” y tratada con “Gimp”.

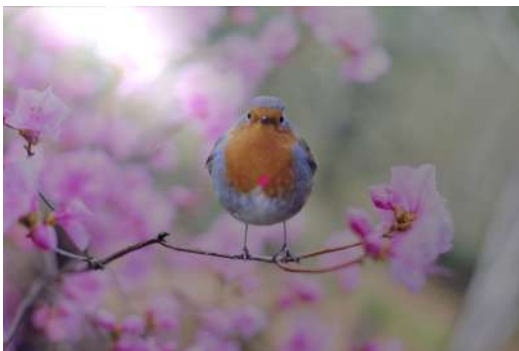


Figura C.101: Detección como “Pájaro”  
en una imagen de un pájaro sin especificar

Imagen obtenida al hacer uso del modelo  
“Birdwatcher”.

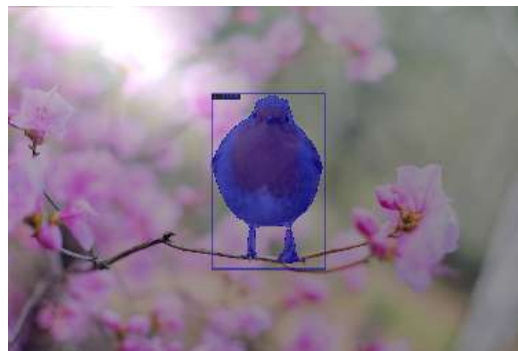


Figura C.102: Detección como “Pájaro”  
en una imagen de un pájaro sin especificar

Imagen obtenida al hacer uso del modelo  
“Detectron2”.



Figura C.103: Imagen de un pajaros sin especificar  
Imagen obtenida de "Pexels" y tratada con "Gimp".



Figura C.104: Detección como "Pájaro"  
en una imagen de un pajaros sin especificar

Imagen obtenida al hacer uso del modelo  
"Birdwatcher".



Figura C.105: Detección como "Pájaro"  
en una imagen de un pajaros sin especificar

Imagen obtenida al hacer uso del modelo  
"Detectron2".



Figura C.106: Imagen de un pajaros sin especificar  
Imagen obtenida de “Pexels” y tratada con “Gimp”.



Figura C.107: Detección como “Pájaro”  
en una imagen de un pajaros sin especificar

Imagen obtenida al hacer uso del modelo  
“Birdwatcher”.



Figura C.108: Detección como “Pájaro”  
en una imagen de un pajaros sin especificar

Imagen obtenida al hacer uso del modelo  
“Detectron2”.



Figura C.109: Imagen de un pajaros sin especificar  
Imagen obtenida de “Pexels” y tratada con “Gimp”.

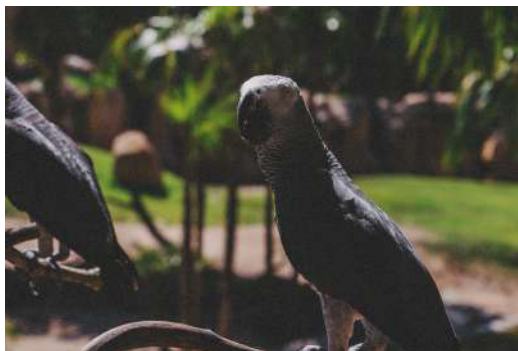


Figura C.110: Detección como “Pájaro”  
en una imagen de un pajaros sin especificar

Imagen obtenida al hacer uso del modelo  
“Birdwatcher”.



Figura C.111: Detección como “Pájaro”  
en una imagen de un pajaros sin especificar

Imagen obtenida al hacer uso del modelo  
“Detectron2”.





Figura C.112: Imagen de un pajaros sin especificar  
Imagen obtenida de “Pexels” y tratada con “Gimp”.

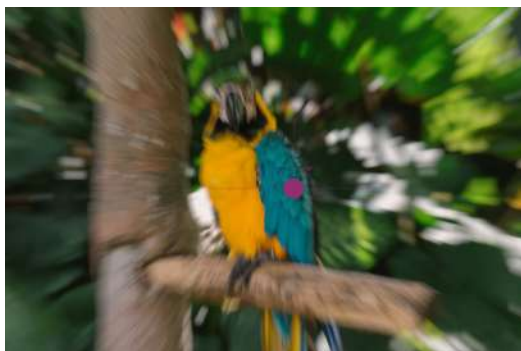


Figura C.113: Detección como “Pájaro”  
en una imagen de un pajaros sin especificar

Imagen obtenida al hacer uso del modelo  
“Birdwatcher”.



Figura C.114: Detección como “Pájaro”  
en una imagen de un pajaros sin especificar

Imagen obtenida al hacer uso del modelo  
“Detectron2”.

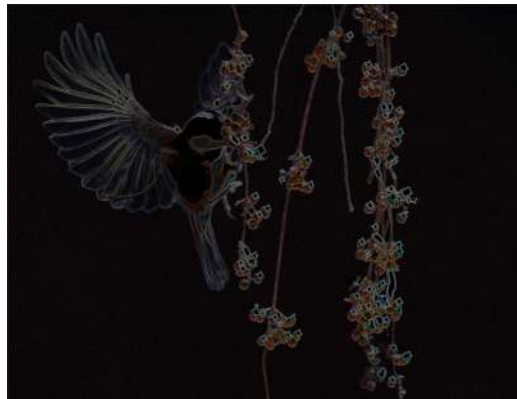


Figura C.115: Imagen de un pajaros sin especificar  
Imagen obtenida de "Pexels" y tratada con "Gimp".

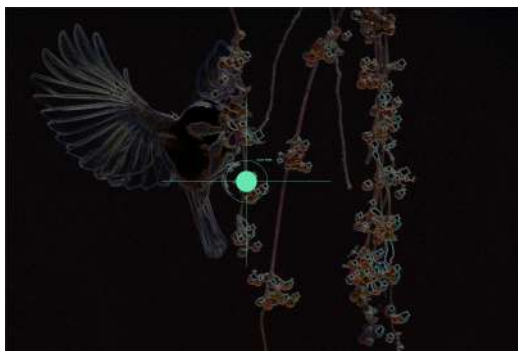


Figura C.116: Detección como "Pájaro"  
en una imagen de un pajaros sin especificar

Imagen obtenida al hacer uso del modelo  
"Birdwatcher".

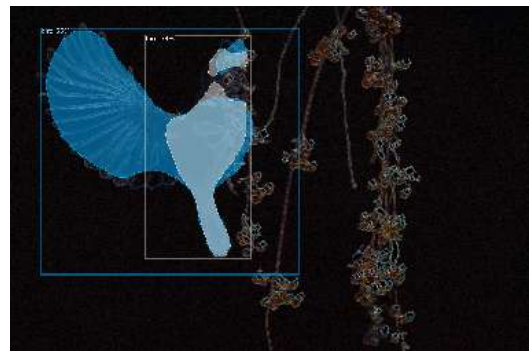


Figura C.117: Detección como "Pájaro"  
en una imagen de un pajaros sin especificar

Imagen obtenida al hacer uso del modelo  
"Detectron2".





Figura C.118: Imagen de un pajaros sin especificar  
Imagen obtenida de "Pexels" y tratada con "Gimp".



Figura C.119: Detección como "Pájaro"  
en una imagen de varios pajaros sin especificar

Imagen obtenida al hacer uso del modelo  
"Birdwatcher".



Figura C.120: Detección como "Pájaro"  
en una imagen de varios pajaros sin especificar

Imagen obtenida al hacer uso del modelo  
"Detectron2".

## C.4. Simulaciones gráficas de imágenes complejas

Para concluir con las simulaciones que se adjuntan en este anexo, se ha procedido a realizar una serie de simulaciones con imágenes en las que encontramos, bandadas de aves, parejas o grupos. Estas simulaciones se llevan a cabo con el fin de validar el sistema ante la posibilidad de tener que detectar varios pájaros a la vez.



Figura C.121: Imagen de una bandada de Aguja colinegra  
Imagen Cedida por SEO/BirdLife.



Figura C.122: Detección como “Pájaro”  
en una imagen de una bandada de Aguja  
colinegra

Imagen obtenida al hacer uso del modelo  
“Birdwatcher”.

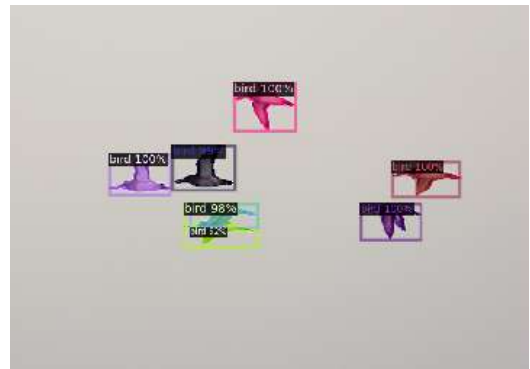


Figura C.123: Detección como “Pájaro”  
en una imagen de una bandada de Aguja  
colinegra

Imagen obtenida al hacer uso del modelo  
“Detectron2”.



Figura C.124: Imagen de una bandada de Avefría Europea  
 Imagen Cedida por SEO/BirdLife.

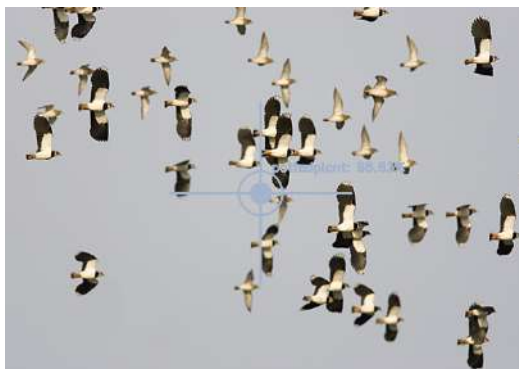


Figura C.125: Detección como “Pájaro”  
 en una imagen de una bandada de Avefría Europea

Imagen obtenida al hacer uso del modelo  
 “Birdwatcher”.



Figura C.126: Detección como “Pájaro”  
 en una imagen de una bandada de Avefría Europea

Imagen obtenida al hacer uso del modelo  
 “Detectron2”.



Figura C.127: Imagen de una bandada de Combatiente  
Imagen Cedida por SEO/BirdLife.



Figura C.128: Detección como “Pájaro”  
en una imagen de una bandada de Com-  
batiente

Imagen obtenida al hacer uso del modelo  
“Birdwatcher”.

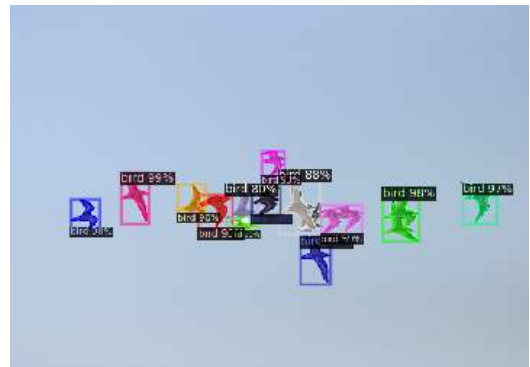


Figura C.129: Detección como “Pájaro”  
en una imagen de una bandada de Com-  
batiente

Imagen obtenida al hacer uso del modelo  
“Detectron2”.



Figura C.130: Imagen de un grupo de Aguja colinegra  
Imagen Cedida por SEO/BirdLife.



Figura C.131: Detección como “Pájaro”  
en una imagen de un grupo de Aguja colinegra

Imagen obtenida al hacer uso del modelo  
“Birdwatcher”.

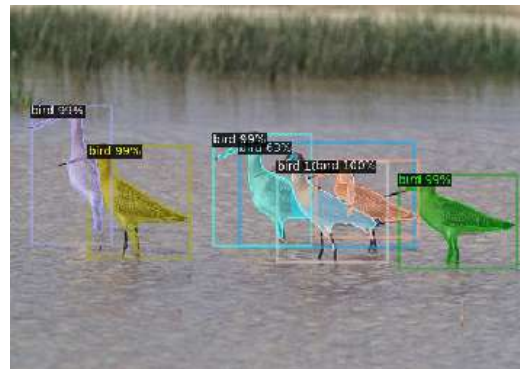


Figura C.132: Detección como “Pájaro”  
en una imagen de un grupo de Aguja colinegra

Imagen obtenida al hacer uso del modelo  
“Detectron2”.



Figura C.133: Imagen de un grupo de Charrán patinegro  
Imagen Cedida por SEO/BirdLife.



Figura C.134: Detección como “Pájaro”  
en una imagen de un grupo de Charrán  
patinegro

Imagen obtenida al hacer uso del modelo  
“Birdwatcher”.

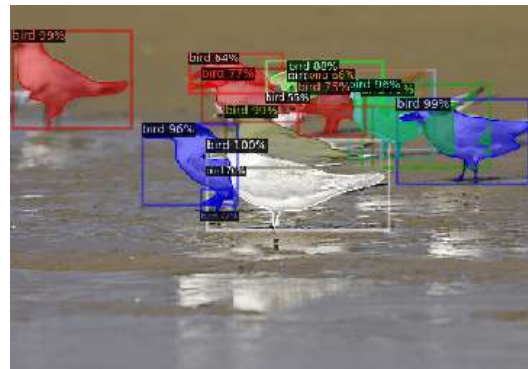


Figura C.135: Detección como “Pájaro”  
en una imagen de un grupo de Charrán  
patinegro

Imagen obtenida al hacer uso del modelo  
“Detectron2”.



Figura C.136: Imagen de un grupo de Gaviota de Auodin  
Imagen Cedida por SEO/BirdLife.

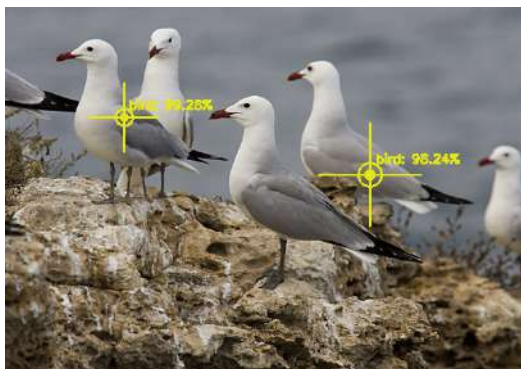


Figura C.137: Detección como “Pájaro”  
en una imagen de un grupo de Gaviota  
de Auodin

Imagen obtenida al hacer uso del modelo  
“Birdwatcher”.

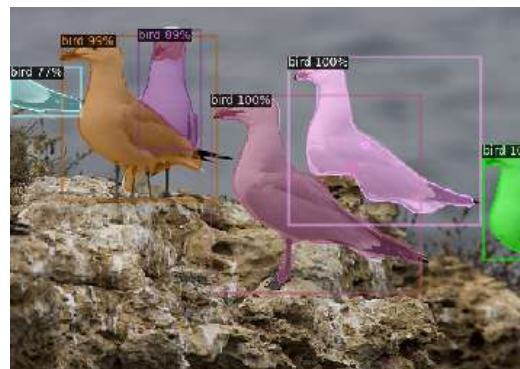


Figura C.138: Detección como “Pájaro”  
en una imagen de un grupo de Gaviota  
de Auodin

Imagen obtenida al hacer uso del modelo  
“Detectron2”.





Figura C.139: Imagen de un grupo de Gaviota de Auodin  
Imagen Cedida por SEO/BirdLife.



Figura C.140: Detección como “Pájaro”  
en una imagen de de un grupo de Gaviota  
de Auodin

Imagen obtenida al hacer uso del modelo  
“Birdwatcher”.

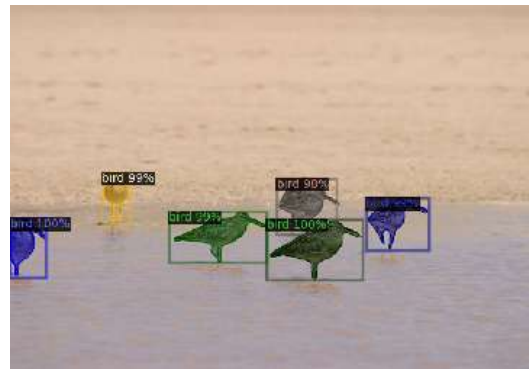


Figura C.141: Detección como “Pájaro”  
en una imagen de un grupo de Gaviota  
de Auodin

Imagen obtenida al hacer uso del modelo  
“Detectron2”.





Figura C.142: Imagen de una pareja de Charrán bengalí  
Imagen Cedida por SEO/BirdLife.



Figura C.143: Detección como “Pájaro”  
en una imagen de una pareja de Charrán  
bengalí

Imagen obtenida al hacer uso del modelo  
“Birdwatcher”.

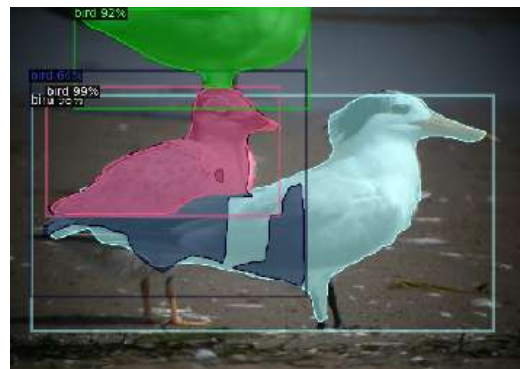


Figura C.144: Detección como “Pájaro”  
en una imagen de una pareja de Charrán  
bengalí

Imagen obtenida al hacer uso del modelo  
“Detectron2”.



Figura C.145: Imagen de una pareja de Gaviota del Atlántico  
Imagen Cedida por SEO/BirdLife.

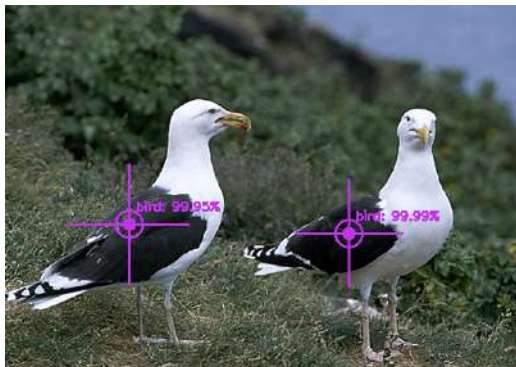


Figura C.146: Detección como “Pájaro”  
en una imagen de una pareja de Gaviota  
del Atlántico

Imagen obtenida al hacer uso del modelo  
“Birdwatcher”.

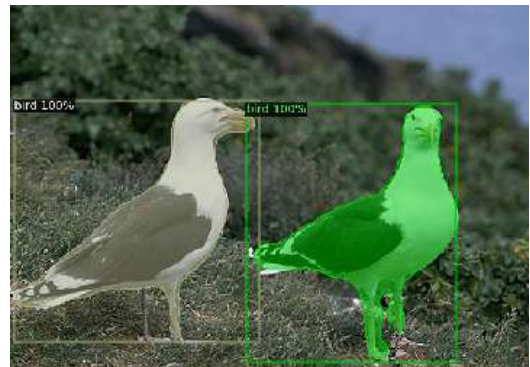


Figura C.147: Detección como “Pájaro”  
en una imagen de una pareja de Gaviota  
del Atlántico

Imagen obtenida al hacer uso del modelo  
“Detectron2”.



Figura C.148: Imagen de una pareja de Gaviota patiamarilla  
Imagen Cedida por SEO/BirdLife.

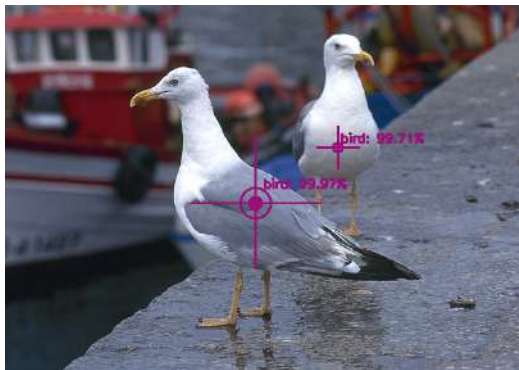


Figura C.149: Detección como “Pájaro”  
en una imagen de una pareja de Gaviota  
patiamarilla

Imagen obtenida al hacer uso del modelo  
“Birdwatcher”.

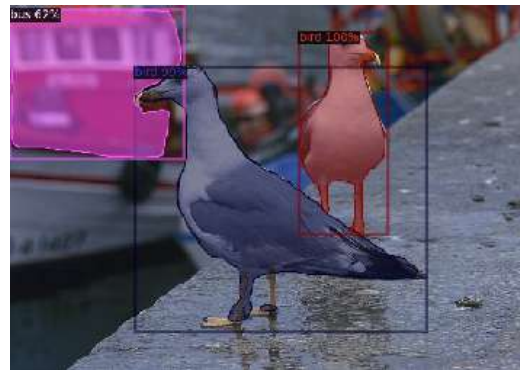


Figura C.150: Detección como “Pájaro”  
en una imagen de una pareja de Gaviota  
patiamarilla

Imagen obtenida al hacer uso del modelo  
“Detectron2”.

## C.5. Comparativa de resultados obtenidos en las simulaciones realizadas con imágenes reales

En la tabla C.1 se establecen los diferentes estados de las aves en la imágenes reales y a su vez el nivel de “accuracy” <sup>1</sup> obtenido. Esto se realiza con el fin de establecer una comparativa que sea más legible y que permita interpretar los resultados obtenidos en las diferentes simulaciones, permitiendo obtener con ello conclusiones sobre el uso de los modelos de inteligencia artificial estudiados.

<b>Resultados de Accuracy obtenidos al analizar imágenes reales</b>			
Imagen	Estado	Accuracy (%) “Birdwatcher”	Accuracy (%) “Detectron2”
C.1	Posada	0 %	99 %
C.4	Volando	97.31 %	100 %
C.7	Posada	90.31 %	100 %
C.10	Volando	71.24 %	100 %
C.13	Posada	97.70 %	100 %
C.16	Volando	99.97 %	99 %
C.19	Posada	99.58 %	100 %
C.22	Volando	99.75 %	99 %
C.25	Posada	99.91 %	100 %
C.28	Volando	99.16 %	99 %
C.31	Posada	100 %	100 %
C.34	Volando	99.63 %	100 %
C.37	Posada	99.94 %	100 %
C.40	Volando	99.93 %	97 %
C.43	Posada	99.96 %	100 %
C.46	Volando	99.77 %	99 %
C.49	Posada	87.76 %	100 %
C.52	Volando	70.55 %	100 %
C.55	Posada	99.97 %	100 %
C.58	Volando	99.98 %	100 %
<b>Media (Posada)</b>		<b>87.52 %</b>	<b>99.9 %</b>
<b>Media (Volando)</b>		<b>93.73 %</b>	<b>99.5 %</b>
<b>Media (Media total)</b>		<b>90.62 %</b>	<b>99.7 %</b>

Tabla C.1: Estado de las aves y niveles de accuracy obtenidos al hacer uso de los modelos preentrenados de inteligencia artificial estudiados en imágenes reales.

<sup>1</sup>En los campos de la ciencia, la ingeniería, la industria y las estadísticas, se define como la exactitud de un sistema de medición.

## C.6. COMPARATIVA DE RESULTADOS OBTENIDOS EN LAS SIMULACIONES REALIZADAS CON

Como se puede observar en la tabla C.1 el nivel de confianza del modelo preentrenado de “Detectron2” es mayor que el de “Birdwatcher”. Cabe destacar que el modelo preentrenado de “Detectron2” presenta valores de detección bastante altos y similares independientemente del estado del ave, ya se encuentre esta posada o volando. Mientras que por lo que se ha podido simular el modelo de “Birdwatcher” detecta mejor las aves cuando se encuentran volando. Este último modelo, presenta un valor de confianza total cercano al 90 %, valor que se puede considerar aceptable para la implementación de este modelo en el dispositivo. Aunque existe un caso en el que el modelo no es capaz de detectar nada en la imagen y en la que claramente se puede observar la presencia de un ave en ella, en el resto de imágenes presenta un gran nivel de confianza, no estando en ningún caso (a excepción del citado) por debajo del 70 %.

### C.6. Comparativa de resultados obtenidos en las simulaciones realizadas con imágenes de diferentes animales y paisajes

En la tabla C.2 se establecen los diferentes resultados obtenidos al utilizar los modelos de inteligencia artificial estudiados para detectar la presencia de distintos animales en una imagen. Esto se realiza con el fin de establecer una comparativa que sea más legible y que permita interpretar los resultados obtenidos en las diferentes simulaciones, permitiendo obtener con ello conclusiones sobre el uso del modelo de IA a implementar en el sistema.

Resultados de Accuracy obtenidos al analizar imágenes de distintos animales		
Imagen	Accuracy (%) “Birdwatcher”	Accuracy (%) “Detectron2”
C.61	99.99 %	100 %
C.64	81.90 %	93 %
C.67	99.79 %	100 %
C.70	84.11 %	100 %
C.73	66.86 %	99 %
C.76	99.75 %	100 %
C.79	0 %	98 %
C.82	Bird 99.87 %	100 %
C.85	Bird 89.36 %	0 %
C.88	0 %	0 %

Tabla C.2: Niveles de accuracy obtenidos al hacer uso de los modelos preentrenados de inteligencia artificial estudiados en imágenes con distintos animales.

En la tabla anterior (C.2) se puede observar como el modelo preentrenado de “Detectron2” sigue presentando mejores valores de confianza que el modelo preentrenado “Birdwatcher” aunque este presenta valores bastante aceptables para su implementa en el dispositivo. Este último modelo preentrenado además de mostrar valores de confianza más bajos en todas las imagenes analizadas, en algunos casos donde existen grupos de animales no detecta la totalidad de los mismos, llegando incluso a confundir la presencia de aves en algunas de las imagenes analizadas.

## C.7. Comparativa de resultados obtenidos en las simulaciones realizadas con imágenes de aves tratadas

En la tabla C.3 se establecen los diferentes resultados obtenidos al utilizar los modelos de inteligencia artificial estudiados en imágenes de las que se desconoce su procedencia y que además han sido tratadas mediante filtros. Esto se realiza con el fin de interpretar los resultados que se podrían obtener si el sistema de adquisición de imagenes estuviese sometido a alguna perturbación.

Resultados de Accuracy obtenidos al analizar imagenes de pajaros tratadas con distintos filtros		
Imagen	Accuracy (%) “Birdwatcher”	Accuracy (%) “Detectron2”
C.91	99.58 %	100 %
C.94	99.60 %	100 %
C.97	99.94 %	100 %
C.100	99.40 %	100 %
C.103	0 %	0 %
C.106	0 %	35 %
C.109	0 %	22 %
C.112	99.57 %	74 %
C.115	99.52 %	100 %
C.118	98.95 %	64 %
<b>Media</b>	<b>69.65 %</b>	<b>69.5 %</b>

Tabla C.3: Niveles de accuracy obtenidos al hacer uso de los modelos preentrenados de inteligencia artificial estudiados con imagenes tratadas mediante filtros.

Al analizar imagenes tratadas mediante filtros, los cuales dificultan la detección en las mismas, es esperable obtener un nivel de confianza más bajo que en el caso en el que se analizan imagenes reales. Como se puede observar en la tabla C.3, al analizar este tipo de imagenes, ambos modelos presentan un nivel medio de confianza muy similar con lo cual se podría decir que ante esta situación ambos son igual de validos.

## C.8. COMPARATIVA DE RESULTADOS OBTENIDOS EN LAS SIMULACIONES REALIZADAS CON

### C.8. Comparativa de resultados obtenidos en las simulaciones realizadas con imágenes complejas

En la tabla C.4 se establecen los diferentes resultados obtenidos al utilizar los modelos preentrenados de inteligencia artificial en imágenes en las que aparece un gran número de aves, ya sea en bandada, en grupo o en parejas. Esto se realiza con el fin de interpretar los resultados que se podrían obtener si el sistema visualizara más de un ave a la vez. Todas estas imágenes son reales y muestran el comportamiento de las aves en su hábitat natural.

Resultados de Accuracy obtenidos al analizar imágenes de donde se muestran varias aves				
Imagen	Número de aves detectadas	Accuracy (%) "Birdwatcher"	Número de aves detectadas	Accuracy (%) "Detectron2"
C.121	0/7	0%	7/7	52%
C.124	1/36	98.63%	32/36	69%
C.127	1/16	57.02%	16/16	58%
C.130	3/6	68.41%	6/6	63%
C.133	3/9	59.60%	9/9	64%
C.136	2/6	98.24%	6/6	77%
C.139	2/7	74.92%	6/7	90%
C.142	2/3	97.75%	3/3	64%
C.145	2/2	99.95%	2/2	100%
C.148	2/2	99.71%	2/2	100%
	<b>Media</b>	<b>76.42%</b>	<b>Media</b>	<b>73.7</b>

Tabla C.4: Niveles de accuracy obtenidos al hacer uso de los modelos preentrenados de inteligencia artificial estudiados con imágenes donde se muestran varias aves.

En este último caso de estudio se puede comprobar como nuevamente ambos modelos presentan valores de confianza muy similares. Es cierto que el modelo preentrenado de "Detectron2" es capaz de detectar un mayor número de aves en la imagen que el modelo de "Birdwatcher" debido a su proceso de segmentación de imágenes que lo hace un sistema más "robusto" pero a su vez más complejo. Por otro lado durante todas las simulaciones pero en especial en las mostradas en la tabla C.4, se observa como el modelo preentrenado "Birdwatcher" pierde efectividad a medida que las aves se encuentran más alejadas de la cámara, situación importante en la implementación del modelo en el dispositivo ya que se pierde efectividad a distancias largas.







## Anexo D

# Datasheets de los componentes utilizados

Las siguientes hojas de características pertenecen a los distintos componentes que se han empleado en el desarrollo del dispositivo. Dichas hojas se han dividido en base a los diferentes módulos que se han implementado, siendo por tanto divididas en las diferentes secciones de este anexo.

### D.1. Módulo de alimentación

En esta sección se adjuntan todas las fichas técnicas de los diferentes componentes que conforman el **Módulo de Alimentación** del instrumento.

## D.1.1. Datasheet de la batería seleccionada



## BATERÍAS MONOBLOCK AGM 12V CYM POWER PLUS

Esta batería de súper ciclo profundo (1200 ciclos al 50% de descarga) tiene una alta capacidad de almacenamiento. Admite una gran descarga profunda, por lo que dispone también de una alta capacidad de recuperación de la misma. El proceso de fabricación de esta batería se realiza manualmente realizando no sólo el ensamble de ella, sino que también, todo el proceso de soldaduras. De esta forma, se realiza un control exhaustivo del producto ya que la fabricación NO ES EN SERIE. Estas baterías se recomiendan para sistemas solares, caravanas, embarcaciones de recreo, carritos de golf, sillas de ruedas eléctricas,...

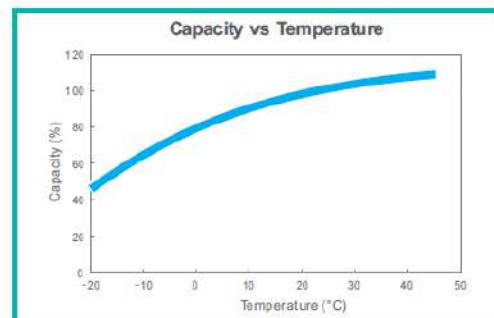
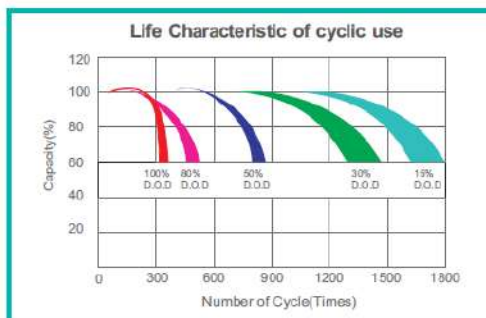
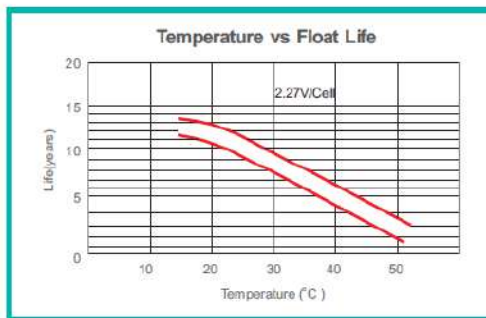
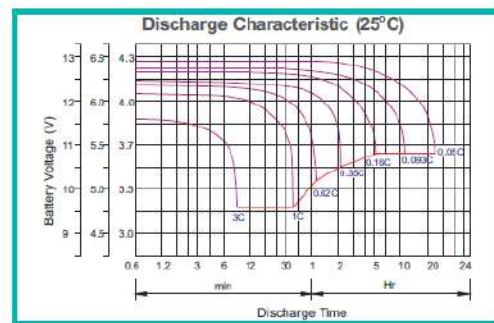
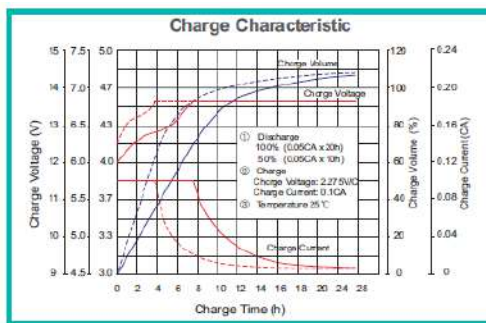
### ESPECIFICACIONES TÉCNICAS:

Ref:	0259	1130	0260	1056	0266
Voltaje nominal	12V (6 celdas por la unidad)	12V (6 celdas por la unidad)	12V (6 celdas por la unidad)	12V (6 celdas por la unidad)	12V (6 celdas por la unidad)
Vida de flotación	12 años	12 años	12 años	12 años	12 años
Capacidad nominal	100Ah 100Hr-rate (to 1.80Vpc) 80Ah C10	130Ah 100Hr-rate (to 1.80Vpc) 100Ah C10	160Ah 100Hr-rate (to 1.80Vpc) 130Ah C10	260Ah 100Hr-rate (to 1.80Vpc) 220Ah C10	325Ah 100Hr-rate (to 1.80Vpc) 250Ah C10
Dimensiones (mm) (total + bornes + asas)	370x178x220 350 + 20 con asas 168 + 10 de ancho 210 + 10 con bornes	347x178x220 327 + 20 con asas 168 + 10 de ancho 210 + 10 con bornes	429x186x235 409 + 20 con asas 176 + 10 de ancho 225 + 10 con bornes	500x235x230 480 + 20 con asas 225 + 10 de ancho 220 + 10 con bornes	510x265x220 490 + 20 con asas 255 + 10 de ancho 210 + 10 con bornes
Peso (kg)	23,5 Kg	28,6 Kg	37Kg	58Kg	78Kg
Tipo de terminal	Hembra M8	Hembra M8	Hembra M8	Hembra M8	Hembra M8
Resistencia interna	Aprox. 0.0054 Ohm (totalmente cargado 20°C)	Aprox. 0.0045 Ohm (totalmente cargado 20°C)	Aprox. 0.0038 Ohm (totalmente cargado 20°C)	Aprox. 0.0026 Ohm (totalmente cargado 20°C)	Aprox. 0.0026 Ohm (totalmente cargado 20°C)
Max. Corriente de carga	24A	30A	40A	75A	100A
Max. Corriente de descarga (SS)	24A	30A	40A	75A	130A
Corriente de cortocircuito	2200A	2600A	3150A	4600A	4600A
Autodescarga	Aprox. 3% /mes 20°C	Aprox. 3% /mes 20°C	Aprox. 3% /mes 20°C	Aprox. 3% /mes 20°C	Aprox. 3% /mes 20°C
Temperatura Ambiente	Descarga: -15-50°C Carga: -15-40°C Almacenamiento: -15-40°C	Descarga: -15-50°C Carga: -15-40°C Almacenamiento: -15-40°C	Descarga: -15-50°C Carga: -15-40°C Almacenamiento: -15-40°C	Descarga: -15-50°C Carga: -15-40°C Almacenamiento: -15-40°C	Descarga: -15-50°C Carga: -15-40°C Almacenamiento: -15-40°C
Flotación (20-25°C)	13.6-13.8V (-3MV/cell/°C)	13.6-13.8V (-3MV/cell/°C)	13.6-13.8V (-3MV/cell/°C)	13.6-13.8V (-3MV/cell/°C)	13.6-13.8V (-3MV/cell/°C)
Absorción (20-25°C)	14.4-14.8V (-5MV/cell/°C)	14.4-14.8V (-5MV/cell/°C)	14.4-14.8V (-5MV/cell/°C)	14.4-14.8V (-5MV/cell/°C)	14.4-14.8V (-5MV/cell/°C)
Material	ABS (UL94-V0 opcional)	ABS (UL94-V0 opcional)	ABS (UL94-V0 opcional)	ABS (UL94-V0 opcional)	ABS (UL94-V0 opcional)




**GRUPO CYMASOL RENOVABLES S.L.**  
VALLE DE GUERRA- LA LAGUNA. SANTA CRUZ DE TENERIFE  
TLF/FAX: 922 158 309 – 646 455 823  
[info@cymasol.com](mailto:info@cymasol.com) – [www.cymasol.com](http://www.cymasol.com)

**CARACTERÍSTICAS TÉCNICAS:**




**GRUPO CYMASOL RENOVABLES S.L.**  
 VALLE DE GUERRA- LA LAGUNA. SANTA CRUZ DE TENERIFE  
 TLF/FAX: 922 158 309 – 646 455 823  
[info@cymasol.com](mailto:info@cymasol.com) – [www.cymasol.com](http://www.cymasol.com)


### D.1.2. Datasheet del módulo fotovoltaico seleccionado




**MONO MODULE**

JST200M(36) 200W







High conversion efficiency  
High module efficiency to guarantee power output.




Self-cleaning glass  
Coating glass for self-cleaning, reduce surface dust.




Outstanding low irradiation performance  
Excellent module efficiency even in the weak light conditions, such as morning or cloudy.



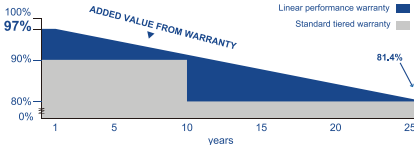
Excellent loading capability  
2400Pa wind loads, 5400Pa snow loads.



0 to +5W positive tolerance  
Detailed information in Electrical Specifications.




48-hour response service




Linear performance warranty  
Standard tiered warranty


Year	Standard Tiered Warranty (%)	Linear Performance Warranty (%)
1	97%	97%
10	88%	90%
25	81.4%	81.4%



25-year performance warranty




10-year warranty on materials and workmanship



**GRUPO CYMASOL RENOVABLES S.L.**

www.cymasol.com    venta@cymasol.com



**ELECTRICAL DATA**

Model Type	JST200M
Peak Power (Pmax)	200W
Module Efficiency	19.61%
Maximum Power Voltage (Vmp)	18.70V
Maximum Power Current (Imp)	10.70A
Open Circuit Voltage (Voc)	22.80V
Short Circuit Current (Isc)	11.24A
Power Tolerance	0 to +5W
Maximum System Voltage	1000V
Nominal Operating Cell Temperature	44.4±2°C
Maximum Series Fuse Rating	10A

**MECHANICAL DATA**

Cell Type	158.75×158.75mm
Number of Cells	36 (9×4)
Weight	11.5kg
Dimension	1500×680×35mm
Max Load	5400 Pascals
Junction Box	IP67 rated
Connector	MC4 Compatible
Wire Type	PV Wire

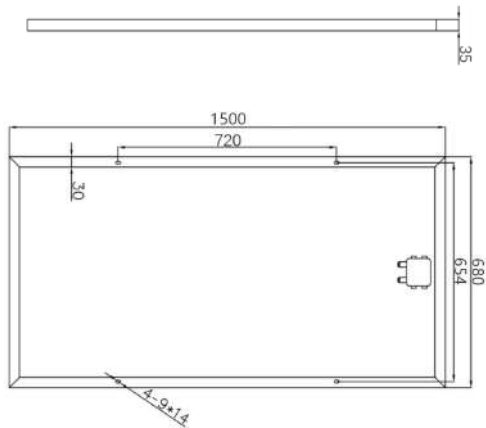
**TEMPERATURE CHARACTERISTICS**

Temp. Coeff. of Isc (TK Isc)	0.04% /°C
Temp. Coeff. of Voc (TK Voc)	-0.28% /°C
Temp. Coeff. of Pmax (TK Pmax)	-0.37% /°C

**PACKING MANNER**

Container	20' GP	40' HQ
Pieces per Pallet	30	30
Pieces per Container	500	1080

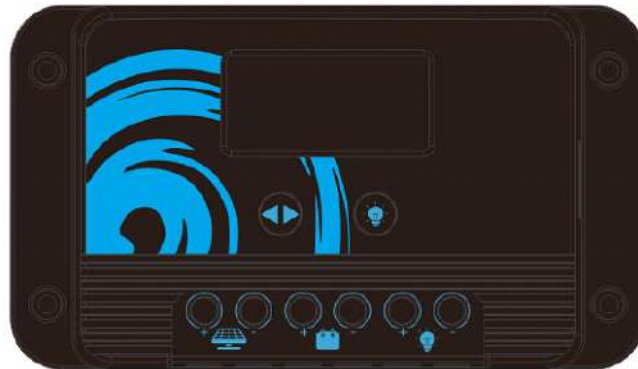
**PHYSICAL CHARACTERISTICS**






**D.1.3. Datasheet del controlador de carga seleccionado**

# Manual de programación

## Regulador Solar PWM 12/24V 10A/20A



¡Gracias por utilizar nuestro regulador solar!

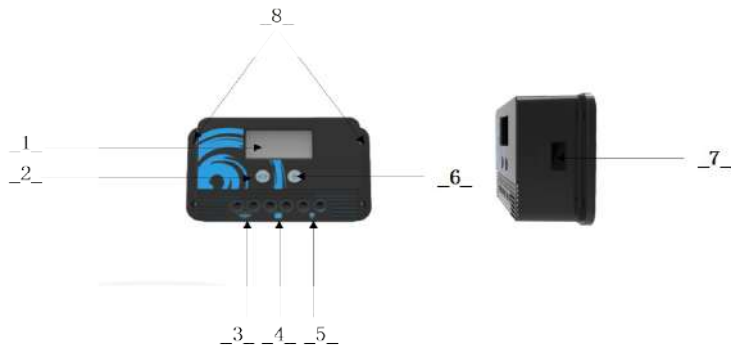
Marcas y Herramientas	Nombre	Descripción
	Marca de peligro de alto voltaje	Puede existir alto voltaje en el controlador y todas las operaciones deben ser realizadas por profesionales eléctricos.
	Precaución de calefacción	Mantenga la distancia con el controlador, debido al posible calentamiento cuando está funcionando.
	Marca WEEE de la UE	No tire el controlador como basura.
	Pelacables	Para pelar cables.
	Multiplicar metro	Para verificar la conexión positiva / negativa y para verificar la corriente u otro valor eléctrico.
	Guante antiestático	Para evitar daños al controlador causados por la electricidad estática del cuerpo humano.
	Cinta eléctrica	Colocar cinta adhesiva en la unión del cableado por motivos de seguridad.
	Destornillador	Para fijar los tornillos.

**I. CARACTERÍSTICAS DEL REGULADOR**

Gracias por usar nuestro producto. Este controlador de carga solar PWM suele ser un dispositivo para la regulación de la carga solar y el control de salida de descarga, con pantalla LCD y USB adicional de 5V, utilizado principalmente en sistemas de energía solar CC de tamaño pequeño.

- a) La mayoría de los tipos de batería se pueden admitir y seleccionar, como AGM (u otro tipo sellado), GEL, inundada y batería de litio (con varios ajustes de voltaje), mediante la configuración de la tecla en el controlador.
- b) Reconocimiento automático del sistema de batería de 12V / 24V para batería de tipo plomo-ácido o batería de tipo no litio.
- c) Carga PWM trifásica: ecualización - impulso - flotación (para batería de tipo plomo-ácido inundada, AGM, GEL)
- d) Pantalla LCD, que muestra el estado de funcionamiento del sistema y los parámetros de configuración.
- e) Operación de pulsación de teclas fácil de usar, simple y fácil.
- f) Salida USB adicional de 5V, adecuada para carga de CC móvil.
- g) Multiplique la selección del modo de control de salida: modo de control de luz, modo de control de luz + tiempo, modo de prueba y depuración, modo manual y modo siempre activo.
- h) Diseño de grado industrial, para un mejor funcionamiento en condiciones ambientales extremas.
- i) Gama completa de protecciones eléctricas, como anti-conexión en PV y cableado de batería, cortocircuito de carga, sobredescarga de batería, sobrevoltaje del sistema, etc.

**II. ILUSTRACIÓN DEL REGULADOR**



1	Pantalla LCD	5	Terminal de cableado de carga
2	Tecla del menú	6	Tecla de función
3	Terminal de cableado de entrada solar	7	Puerto USB
4	Terminal de cableado de batería	8	Agujeros de instalación

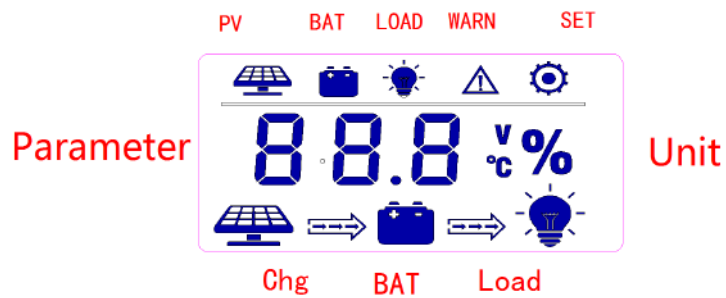


III. SECUENCIAS DE CABLEADO











Primero: conecte la batería primero, elija el cable en consecuencia.  
 Segundo: conecta el panel solar.  
 Último: conecte el cableado de carga a la carga (si es necesario).




IV. ILUSTRACIÓN DE LA PANTALLA LCD



1. SECCIÓN DE PANTALLA LCD

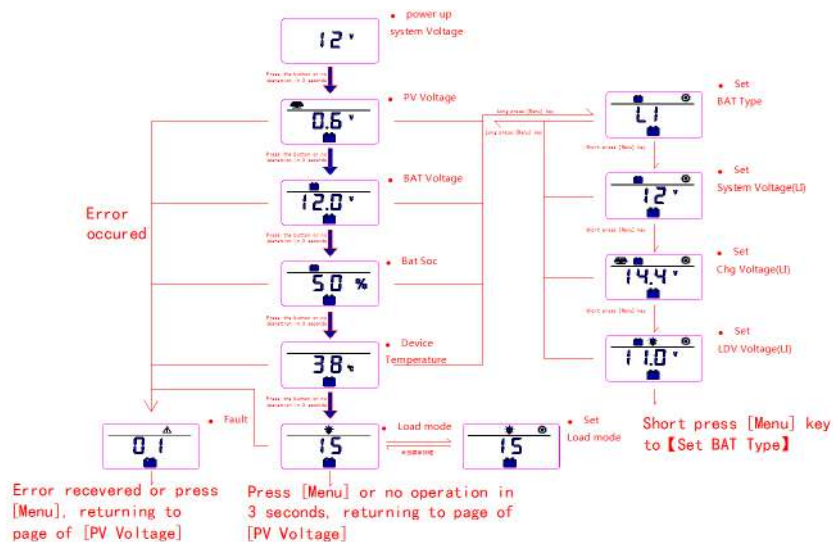
NOMBRE	DESCRIPCIÓN	ICONOS
Estado	Estado de funcionamiento actual del sistema	 $\Rightarrow$  $\Rightarrow$ 
Parámetros	Valor de parámetro para el elemento seleccionado	<b>88.8</b> v %
Objeto seleccionado	Elemento seleccionado actual	    

2. INDICACIONES DE ENERGÍA SOLAR (PV), BATERÍA Y CARGA



ICONO	NOMBRE	ESTADO	INDICACIÓN OBSERVACIÓN
	Indicación PV	Constante ON	Voltio fotovoltaico más alto que el voltaje de control de luz
		OFF	Voltio PV más bajo que el voltaje de control de luz
		Flash lento ON	Cargando
		Parpadeo rápido ON	PV sobre voltaje
	Indicación batería	Constante ON	La batería está bien.
		OFF	La batería está anormal
		Parpadeo rápido ON	Batería sobre descarga
	Indicación carga	Flotante	Cargando
		No flotación	No carga

3. OBSERVACIONES

- \* La página pasará a la siguiente si no se realiza ninguna operación en 3 segundos.
- \* El sistema ingresará automáticamente a la página de "error" cuando se detecte un error. Esta página permanecerá quieta hasta que el usuario opere en el controlador para ingresar a las otras páginas.



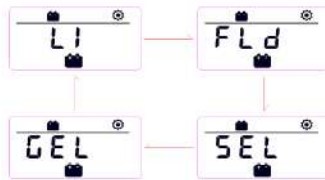
**V. INFORMACIÓN DE AJUSTES DE BOTONES**

BOTÓN	ESTADO DEL AJUSTE	PRESIÓN	FUNCIÓN
	En Configuración	Presión larga	Entrar en la página no para la configuración
		Presión corta	Ingresa a la página siguiente para configuraciones
	No en Configuración	Presión larga	Ingresa a la página para configuraciones
		Presión corta	Ingresa a la página siguiente, no para configuraciones
	En Configuración	Presión larga	Sin función
		Presión corta	Para ajustar parámetros
	No en Configuración	Presión corta	Interruptor de carga (en modo manual)

Observación: “En Configuración” significa que el usuario está en la página para configurar los parámetros.

**VI. AJUSTE DE PARÁMETROS Y TIPO DE BATERÍAS**

1. CONFIGURACIÓN TIPOS DE BATERÍAS

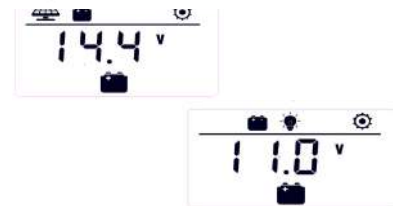


NOMBRE	TIPO DE BATERÍA	DESCRIPCIÓN
FLD	Batería inundada	Reconocimiento automático de voltaje del sistema de batería; parámetros predeterminados establecidos.
SEL	Batería sellada/AGM	
GEL	Batería de GEL	Voltaje del sistema, parámetros de carga/descarga ajustables
LI	Batería de Litio	

1. Sistema de voltaje (Solo para baterías de litio)



2. Ajuste del voltaje de carga (Solo para baterías de litio)



3. Configuración del voltaje de sobredescarga (Solo para baterías de litio)

4. Load Mode Settings



#### VII. LISTA DE ERRORES DEL CONTROLADOR Y RECUPERACIÓN



CÓDIGO	ERROR	ANÁLISIS	SOLUCIÓN DE RECUPERACIÓN
E00	No error	--	--
E01	Sobredescarga	El voltaje de la batería se ha descargado a un nivel bajo, la carga se corta	Recupere una vez que el voltaje de la batería vuelva al nivel normal. Entonces se permite que la carga se encienda.
E02	Batería sobre voltaje	El voltaje de la batería ha superado el nivel máximo.	Recupere una vez que el voltaje de la batería vuelva al nivel normal.
E04	Cortocircuito de carga	La carga se cortocircuita	Verifique el cableado y la condición de carga.
E05	Sobrecarga de carga	La potencia de carga ha superado el valor nominal	Compruebe y reduzca el requisito de potencia de carga.
E06	Dispositivo sobrecalentamiento	El regulador se calienta demasiado a alta temperatura, la carga se corta	Haz que el dispositivo se enfríe para disminuir la temperatura
E08	Carga de potencia sobre nominal	La potencia de entrada ha superado el valor nominal máximo	Para disminuir la potencia de entrada
E10	PV sobre voltaje	La tensión de entrada fotovoltaica es demasiado alta	Para disminuir el voltaje de entrada
E13	Anti-conexión fotovoltaica	El lado fotovoltaico tiene anti-conexión	Verifique y vuelva a conectar los cables fotovoltaicos en la posición correcta
E14	Anti-conexión de la batería	El lado de la batería tiene anti-conexión	Verifique y vuelva a conectar los cables de la batería en la posición correcta

**VIII. ESPECIFICACIONES DEL REGULADOR**

\* Observación: 12V = n°1 24V = n°2

NOMBRE	PARÁMETROS			
	PWM 10A		PWM 20A	
Model N°	12V/24V Auto (FLD/GEL/SLD) - 12V/24V Manual (LI)			
Voltaje del Sistema	12V/24V Auto (FLD/GEL/SLD) - 12V/24V Manual (LI)			
Pérdida sin carga	8ma(12V), 12ma (24V)			
Voltaje máximo de entrada PV	< 55V			
Corriente de carga nominal	10A		20A	
Potencia máxima de entrada fotovoltaica	170W/12V 340W/24V		340W/12V 680W/24V	
Selección del tipo de batería	FLD	SEL	GEL	LI
Ecualizar el voltaje de carga	14.8V*n	14.6V*n	--	--
Aumente el voltaje de carga	14.6V*n	14.4V*n	14.2V*n	14.4V*n (ajustable)
Voltaje de carga flotante	13.8V*n			--
Aumente el voltaje de recuperación de carga	13.2V*n			--
Voltaje de recuperación de descarga excesiva	12.6V*n			12.6V*n (ajuste automático al voltaje de descarga excesiva)
Sobre voltaje de descarga	11.1V*n			11.1V*n
Voltaje de control de luz	5V (12V system) , 10V(24V system)			
Tiempo de retardo del control de luz	10s			
Modos de carga	control de luz (del anochecer al amanecer), control de luz + tiempo, modo de depuración, control manual, modo fijo.			
Temperatura de operación	-35°C ~ +45°C			
Protección IP	IP30			
Peso neto	0.20 KG			
Comunicación	NA			
Pantalla	LCD			
Altitud de operación	≤ 3000m			

## **D.2. Módulo de procesamiento**

En esta sección se adjuntan todas las fichas técnicas de los diferentes componentes que conforman el **Módulo de procesamiento** del instrumento.

### D.2.1. Datasheet Raspberry Pi 3 Model B



Raspberry Pi

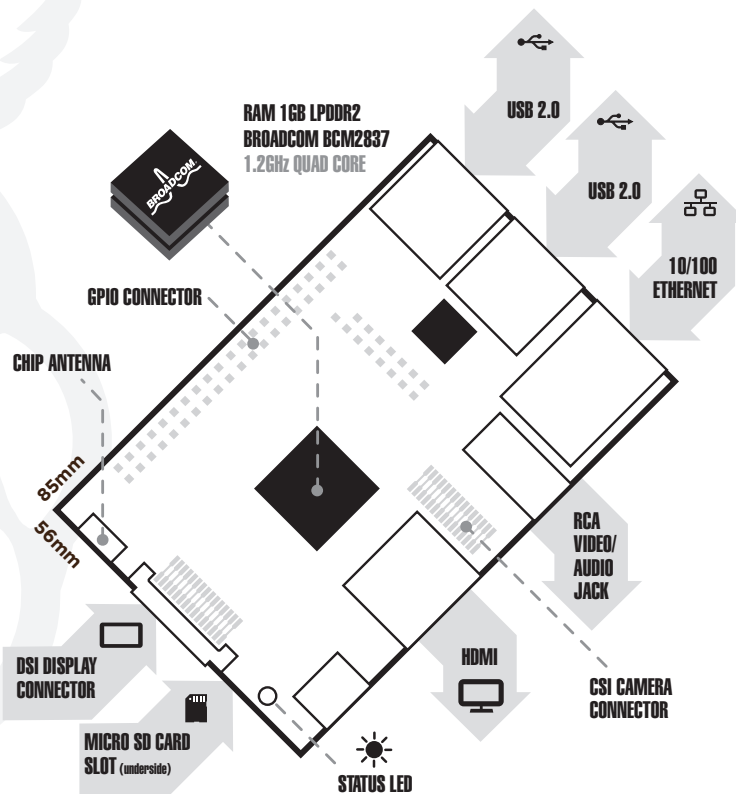


## Raspberry Pi 3 Model B

**Product Name** Raspberry Pi 3

**Product Description** The Raspberry Pi 3 Model B is the third generation Raspberry Pi. This powerful credit-card sized single board computer can be used for many applications and supersedes the original Raspberry Pi Model B+ and Raspberry Pi 2 Model B. Whilst maintaining the popular board format the Raspberry Pi 3 Model B brings you a more powerful processor, 10x faster than the first generation Raspberry Pi. Additionally it adds wireless LAN & Bluetooth connectivity making it the ideal solution for powerful connected designs.

**RS Part Number** 896-8660





# Raspberry Pi

## Raspberry Pi 3 Model B

### Specifications

<b>Processor</b>	Broadcom BCM2387 chipset. 1.2GHz Quad-Core ARM Cortex-A53 802.11 b/g/n Wireless LAN and Bluetooth 4.1 (Bluetooth Classic and LE)
<b>GPU</b>	Dual Core VideoCore IV® Multimedia Co-Processor. Provides Open GL ES 2.0, hardware-accelerated OpenVG, and 1080p30 H.264 high-profile decode.  Capable of 1Gpixel/s, 1.5Gtexel/s or 24GFLOPs with texture filtering and DMA infrastructure
<b>Memory</b>	1GB LPDDR2
<b>Operating System</b>	Boots from Micro SD card, running a version of the Linux operating system or Windows 10 IoT
<b>Dimensions</b>	85 x 56 x 17mm
<b>Power</b>	Micro USB socket 5V1, 2.5A

### Connectors:

<b>Ethernet</b>	10/100 BaseT Ethernet socket
<b>Video Output</b>	HDMI (rev 1.3 & 1.4) Composite RCA (PAL and NTSC)
<b>Audio Output</b>	Audio Output 3.5mm jack, HDMI USB 4 x USB 2.0 Connector
<b>GPIO Connector</b>	40-pin 2.54 mm (100 mil) expansion header: 2x20 strip Providing 27 GPIO pins as well as +3.3 V, +5 V and GND supply lines
<b>Camera Connector</b>	15-pin MIPI Camera Serial Interface (CSI-2)
<b>Display Connector</b>	Display Serial Interface (DSI) 15 way flat flex cable connector with two data lanes and a clock lane
<b>Memory Card Slot</b>	Push/pull Micro SDIO

### Key Benefits

- Low cost
- 10x faster processing
- Consistent board format
- Added connectivity

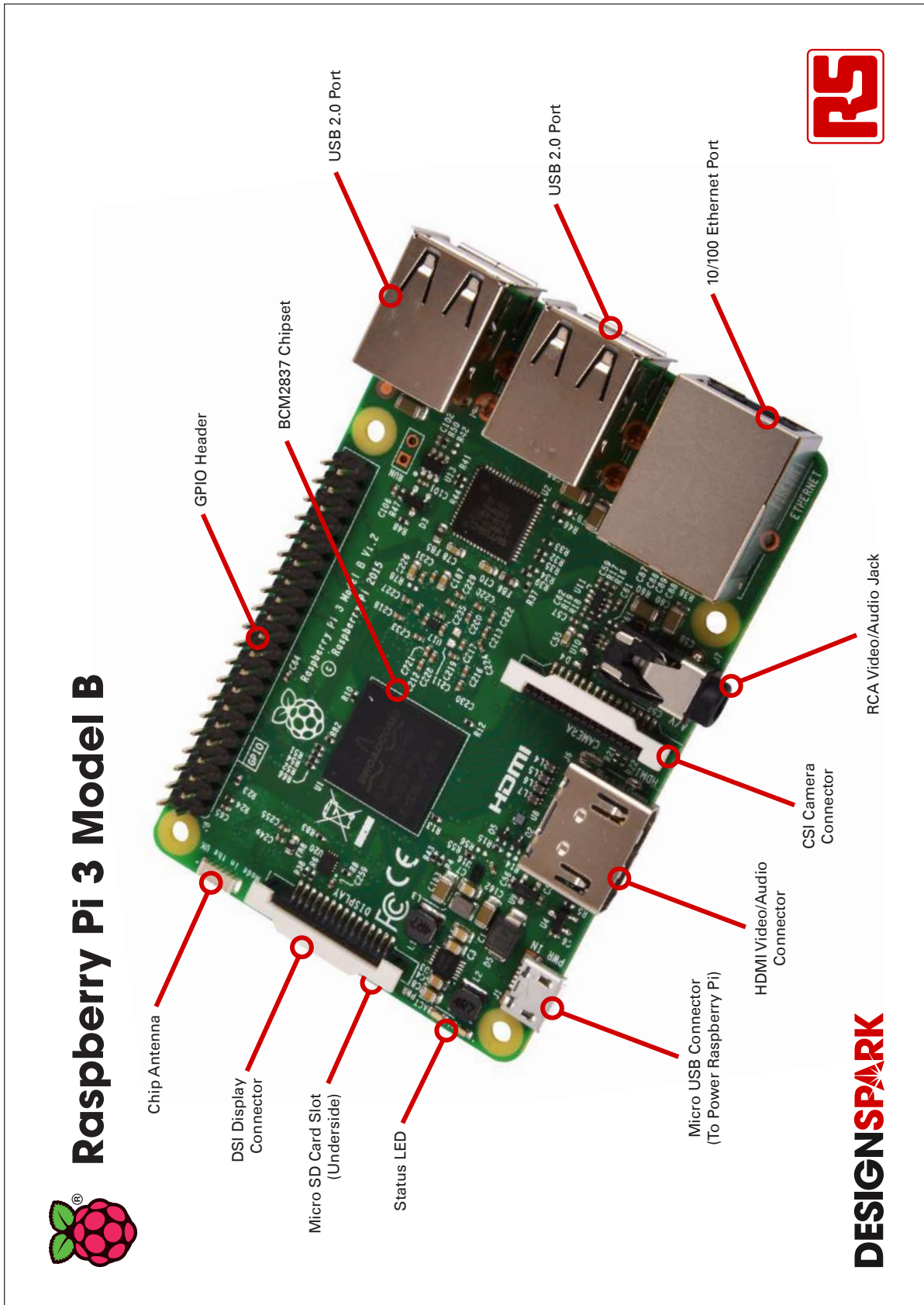
### Key Applications

- Low cost PC/tablet/laptop
- Media centre
- Industrial/Home automation
- Print server
- Web camera
- Wireless access point
- Environmental sensing/monitoring (e.g. weather station)
- IoT applications
- Robotics
- Server/cloud server
- Security monitoring
- Gaming





### D.2.2. Diagrama Raspberry Pi 3 Model B



### D.2.3. Datasheet UPS Raspberry Pi



INDUSTRIAL SHIELDS

#### Raspberry Pi UPS & RTC Shield



- Plug & Play UPS Smart Shield and Real Time Clock (RTC) for Raspberry Pi
- Avoid any SD corruption issue
- 12Vdc or 24Vdc
- With RS485 protocol available
- Compatible with Tinkerboard. All Raspberry Pi 2, 3, 4 & all Raspberry Pi B models
- Recharge Time: Less than 1 minute
- Reconnecting time: Less than 20 seconds
- With reboot button (or the possibility to add an external one)

# Table of Contents

1	General Description .....	5
1.1.	General Specifications.....	5
1.2.	Available Communications & Accessories.....	5
1.3.	UPS & RTC Shield Pinout .....	6
1.4.	Mechanical dimension .....	7
1.5.	Main parameters.....	7
1.6.	Raspberry PI 3 Pinout Connector .....	8
1.7.	Raspberry Pi Connections: .....	10
1.8.	Connectors: .....	11
2.	Raspberry Pi Connections.....	13
2.1.	Enable the controlled shutdown on your Raspberry Pi .....	13
2.2.	Enable the RTC functionality on your Raspberry Pi.....	14
2.3.	Enable the RS-485 functionality on your Raspberry Pi.....	14



INDUSTRIAL SHIELDS

## 1 General Description

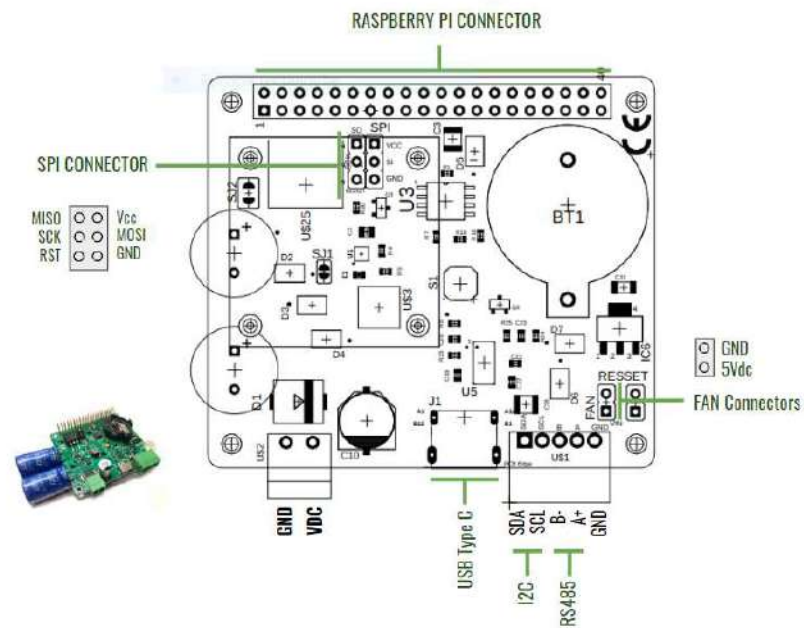
### 1.1. General Specifications

Operational Voltage	Current	Max. Current	Super Capacitors	Certificate	Possible MCU Connections	RTC
<b>12 or 24Vdc</b> (Antipolarity + Filter)	<b>3 A</b> (Autoprotected by chip)	<b>3,5 A</b>	x2 ( <b>25F</b> )	CE, RoHs	-Raspberry Pi 2 & 2B -Raspberry Pi 3 & 3B -Raspberry Pi 4B -Tinkerboard (ASUS)	<b>DS3231</b> Coin cell : CR2032

### 1.2. Available Communications & Accessories

- RS-485 Port
- ICSP Connector
- I2C (3,3Vdc)
- USB Type-C (Power Only)
- RTC
- Raspberry Pi Connectors
- Possibility of connecting a 5Vdc fan

1.3. UPS & RTC Shield Pinout



1.4. Mechanical dimension

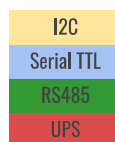


1.5. Main parameters

Parameter	Value	Conditions
<b>Input parameters</b>		
Input voltage (screw terminal)	7V DC ... 28V DC	At screw terminal
Input voltage (USB terminal)	5V DC	USB terminal
Average input current	0.5A	Charge mode + RPi3 Booting at $V_{in} = 24V$
<b>Output parameters</b>		
Output voltage range	4.75V DC ... 5.5V DC	
Output current range	0Aavg ... 2Aavg	
Output ripple	20mV pp	Raspberry Pi4 as a web server
Fan voltage	4.75V – 5.5V DC	
<b>Control parameters</b>		
Pin 16 (GPIO23)	3.3V – 5V (Inputs maximum voltage)	
Pin 13 (GPIO27)	3.3V – 5V (Outputs maximum voltage)	

1.6. Raspberry PI 3 Pinout Connector

	NC	<b>1</b>	<b>2</b>	Vin	
SDA	GPIO2	<b>3</b>	<b>4</b>	Vin	
SCL	GPIO3	<b>5</b>	<b>6</b>	GND	
	NC	<b>7</b>	<b>8</b>	GPIO14	TXD
	GND	<b>9</b>	<b>10</b>	GPIO15	RXD
RE	GPIO17	<b>11</b>	<b>12</b>	NC	
DE	GPIO27	<b>13</b>	<b>14</b>	GND	
	NC	<b>15</b>	<b>16</b>	GPIO23	UPS Control from RASPERRY
	NC	<b>17</b>	<b>18</b>	GPIO24	UPS Control to RASPERRY
	NC	<b>19</b>	<b>20</b>	GND	
	NC	<b>21</b>	<b>22</b>	NC	
	NC	<b>23</b>	<b>24</b>	NC	
	GND	<b>25</b>	<b>26</b>	NC	
	NC	<b>27</b>	<b>28</b>	NC	
	NC	<b>29</b>	<b>30</b>	GND	
	NC	<b>31</b>	<b>32</b>	NC	
	NC	<b>33</b>	<b>34</b>	GND	
	NC	<b>35</b>	<b>36</b>	NC	
	NC	<b>37</b>	<b>38</b>	NC	
	GND	<b>39</b>	<b>40</b>	NC	



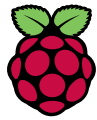
- GPIO24 (Raspberry's 18 pin) is used by UPS to report if an emergency power-off is needed. If the UPS shield detects that the external power supply has been lost, this pin is going to be connected to GND. In the opposite case, this pin is not going to be connected. Because of this, it is recommended to configure this pin with a pull-up software.
- GPIO23 (Raspberry's 16 pin) is used to report a finished saving process. If system fails, the UPS shield will maintain the power supply till it receives a low logic value from this pin. If

### **D.3. Módulo de detección**

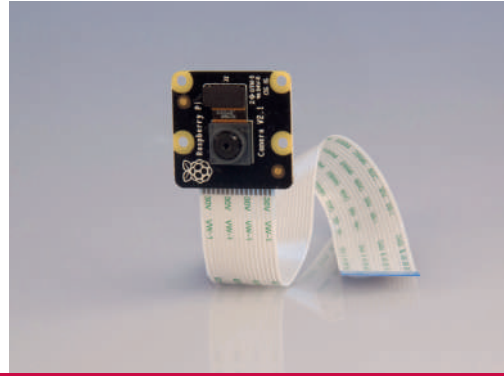
En esta sección se adjuntan todas las fichas técnicas de los diferentes componentes que conforman el **Módulo de detección** del instrumento.



### D.3.1. Datasheet Módulo cámara Raspberry Pi V2



Raspberry Pi



## Pi NoIR Camera Module

<b>Product Name</b>	<b>Pi NoIR - Raspberry Pi Infrared Camera Module</b>
<b>Product Description</b>	Featuring the same 8 megapixel image sensor as the standard Raspberry Pi camera with the infrared cut-off filter removed to increase IR light sensitivity. The Pi NoIR is compatible with all Raspberry Pi models and provides high definition, high sensitivity, low crosstalk and low noise image capture in an ultra small and lightweight design. The camera module connects to the Raspberry Pi board via the CSI connector which is capable of extremely high data rates and it exclusively carries pixel data to the processor.
<b>RS Part Numer</b>	<b>913-2673</b>
<b>Specifications</b>	
<b>Image Sensor</b>	Sony IMX 219 PQ CMOS image sensor in a fixed-focus module with IR blocking filter removed.
<b>Resolution</b>	8-megapixel
<b>Still picture resolution</b>	3280 x 2464
<b>Max image transfer rate</b>	1080p: 30fps (encode and decode) 720p: 60fps
<b>Connection to Raspberry Pi</b>	15-pin ribbon cable, to the dedicated 15-pin MIPI Camera Serial Interface (CSI-2).
<b>Image control functions</b>	Automatic exposure control Automatic white balance Automatic band filter Automatic 50/60 Hz luminance detection Automatic black level calibration
<b>Temp range</b>	Operating: -20° to 60° Stable image: -20° to 60°
<b>Lens size</b>	1/4"
<b>Dimensions</b>	23.86 x 25 x 9mm
<b>Weight</b>	3g
<b>Note</b>	The Pi NoIR camera is sensitive to short-wavelength IR radiation (around 880nm) and requires IR illumination to see in the dark.

## **D.4. Módulo de aviso**

En esta sección se adjuntan las fichas técnicas de los componentes más importantes que conforman el **Módulo de aviso** del instrumento.

### D.4.1. Amplificador operacional TCA0372

## TCA0372, TCA0372B, NCV0372B

### 1.0 A Output Current, Dual Power Operational Amplifiers

The TCA0372 is a monolithic circuit intended for use as a power operational amplifier in a wide range of applications, including servo amplifiers and power supplies. No deadband crossover distortion provides better performance for driving coils.

**Features**

- Output Current to 1.0 A
- Slew Rate of 1.3 V/μs
- Wide Bandwidth of 1.1 MHz
- Internal Thermal Shutdown
- Single or Split Supply Operation
- Excellent Gain and Phase Margins
- Common Mode Input Includes Ground
- Zero Deadband Crossover Distortion
- NCV devices are AEC-Q100 Qualified and PPAP Capable
- These Devices are Pb-Free, Halogen Free/BFR Free and are RoHS Compliant

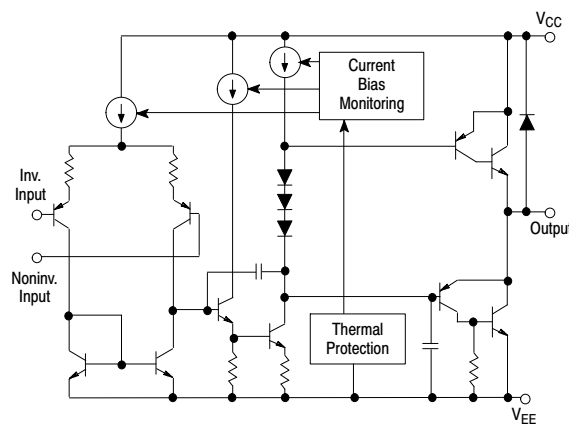
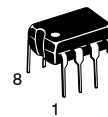


Figure 1. Representative Block Diagram

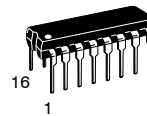


ON Semiconductor®

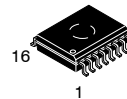
<http://onsemi.com>



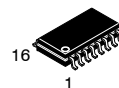
PDIP-8  
DP1 SUFFIX  
CASE 626



PDIP-16  
DP2 SUFFIX  
CASE 648



SOIC-16W  
DW SUFFIX  
CASE 751G



SOEIAJ-16  
DM2 SUFFIX  
CASE 966

**ORDERING INFORMATION**

See detailed ordering and shipping information in the package dimensions section on page 6 of this data sheet.

**DEVICE MARKING INFORMATION**

See general marking information in the device marking section on page 6 of this data sheet.

**TCA0372, TCA0372B, NCV0372B****MAXIMUM RATINGS**

Rating	Symbol	Value	Unit
Supply Voltage (from $V_{CC}$ to $V_{EE}$ )	$V_S$	40	V
Input Differential Voltage Range	$V_{IDR}$	Note 1	V
Input Voltage Range	$V_{IR}$	Note 1	V
Junction Temperature (Note 2)	$T_J$	+150	°C
Operating Temperature Range	$T_A$	-40 to +125	°C
Storage Temperature Range	$T_{stg}$	-55 to +150	°C
DC Output Current	$I_O$	1.0	A
Peak Output Current (Nonrepetitive)	$I_{(max)}$	1.5	A
Thermal Resistance, Junction-to-Air Case 626 Case 648 Case 751G	$R_{\theta JA}$	137 72 80	°C/W
Thermal Resistance, Junction-to-Case Case 626 Case 648 Case 751G	$R_{\theta JC}$	23 10 12	°C/W

Stresses exceeding Maximum Ratings may damage the device. Maximum Ratings are stress ratings only. Functional operation above the Recommended Operating Conditions is not implied. Extended exposure to stresses above the Recommended Operating Conditions may affect device reliability.

1. Either or both input voltages should not exceed the magnitude of  $V_{CC}$  or  $V_{EE}$ .
2. Power dissipation must be considered to ensure maximum junction temperature ( $T_J$ ) is not exceeded.

## TCA0372, TCA0372B, NCV0372B

DC ELECTRICAL CHARACTERISTICS ( $V_{CC} = +15\text{ V}$ ,  $V_{EE} = -15\text{ V}$ ,  $R_L$  connected to ground,  $T_A = -40^\circ$  to  $+125^\circ\text{C}$ .)

Characteristics	Symbol	Min	Typ	Max	Unit
Input Offset Voltage ( $V_{CM} = 0$ ) $T_A = +25^\circ\text{C}$ $T_A, T_{low}$ to $T_{high}$	$V_{IO}$	-	1.0	15	mV
Average Temperature Coefficient of Offset Voltage	$\Delta V_{IO}/\Delta T$	-	20	-	$\mu\text{V}/^\circ\text{C}$
Input Bias Current ( $V_{CM} = 0$ )	$I_{IB}$	-	100	500	nA
Input Offset Current ( $V_{CM} = 0$ )	$I_{IO}$	-	10	50	nA
Large Signal Voltage Gain $V_O = \pm 10\text{ V}$ , $R_L = 2.0\text{ k}$	$A_{VOL}$	30	100	-	V/mV
Output Voltage Swing ( $I_L = 100\text{ mA}$ ) $T_A = +25^\circ\text{C}$ $T_A = T_{low}$ to $T_{high}$ $T_A = +25^\circ\text{C}$ $T_A = T_{low}$ to $T_{high}$	$V_{OH}$ $V_{OL}$	14.0 13.9	14.2 -	- -14.0	V
Output Voltage Swing ( $I_L = 1.0\text{ A}$ ) $V_{CC} = +24\text{ V}$ , $V_{EE} = 0\text{ V}$ , $T_A = +25^\circ\text{C}$ $V_{CC} = +24\text{ V}$ , $V_{EE} = 0\text{ V}$ , $T_A = T_{low}$ to $T_{high}$ $V_{CC} = +24\text{ V}$ , $V_{EE} = 0\text{ V}$ , $T_A = +25^\circ\text{C}$ $V_{CC} = +24\text{ V}$ , $V_{EE} = 0\text{ V}$ , $T_A = T_{low}$ to $T_{high}$	$V_{OH}$ $V_{OL}$	22.5 22.5	22.7 -	- 1.5	V
Input Common Mode Voltage Range $T_A = +25^\circ\text{C}$ $T_A = T_{low}$ to $T_{high}$	$V_{ICR}$	$V_{EE}$ to $(V_{CC} - 1.0)$ $V_{EE}$ to $(V_{CC} - 1.3)$			V
Common Mode Rejection Ratio ( $R_S = 10\text{ k}$ )	CMRR	70	90	-	dB
Power Supply Rejection Ratio ( $R_S = 100\ \Omega$ )	PSRR	70	90	-	dB
Power Supply Current $T_A = +25^\circ\text{C}$ $T_A = T_{low}$ to $T_{high}$	$I_D$	-	5.0 8.0	10 10 14	mA
				14	

AC ELECTRICAL CHARACTERISTICS ( $V_{CC} = +15\text{ V}$ ,  $V_{EE} = -15\text{ V}$ ,  $R_L$  connected to ground,  $T_A = +25^\circ\text{C}$ , unless otherwise noted.)

Characteristics	Symbol	Min	Typ	Max	Unit
Slew Rate ( $V_{in} = -10\text{ V}$ to $+10\text{ V}$ , $R_L = 2.0\text{ k}$ , $C_L = 100\text{ pF}$ ) $A_V = -1.0$ , $T_A = T_{low}$ to $T_{high}$	SR	1.0	1.4	-	V/ $\mu\text{s}$
Gain Bandwidth Product ( $f = 100\text{ kHz}$ , $C_L = 100\text{ pF}$ , $R_L = 2.0\text{ k}$ ) $T_A = 25^\circ\text{C}$ $T_A = T_{low}$ to $T_{high}$	GBW	0.9 0.7	1.4 -	- -	MHz
Phase Margin $T_J = T_{low}$ to $T_{high}$ $R_L = 2.0\text{ k}$ , $C_L = 100\text{ pF}$	$\phi_m$	-	65	-	Degrees
Gain Margin $R_L = 2.0\text{ k}$ , $C_L = 100\text{ pF}$	$A_m$	-	15	-	dB
Equivalent Input Noise Voltage $R_S = 100\ \Omega$ , $f = 1.0$ to $100\text{ kHz}$	$e_n$	-	22	-	$\text{nV}/\sqrt{\text{Hz}}$
Total Harmonic Distortion $A_V = -1.0$ , $R_L = 50\ \Omega$ , $V_O = 0.5\text{ VRMS}$ , $f = 1.0\text{ kHz}$	THD	-	0.02	-	%

NOTE: In case  $V_{EE}$  is disconnected before  $V_{CC}$ , a diode between  $V_{EE}$  and Ground is recommended to avoid damaging the device.

TCA0372, TCA0372B, NCV0372B

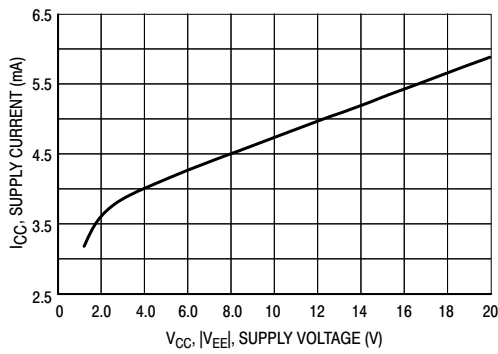


Figure 2. Supply Current versus Supply Voltage with No Load

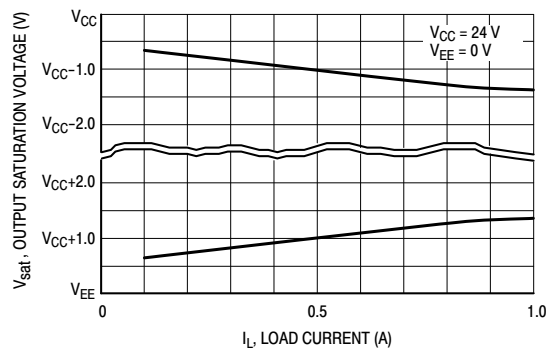


Figure 3. Output Saturation Voltage versus Load Current

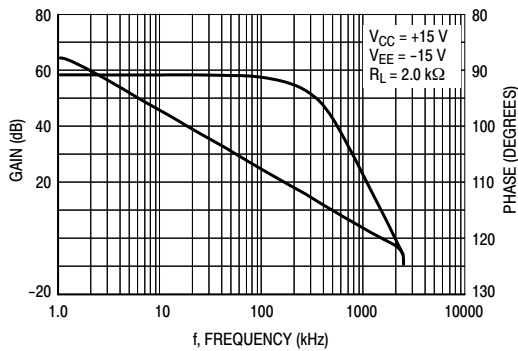


Figure 4. Voltage Gain and Phase versus Frequency

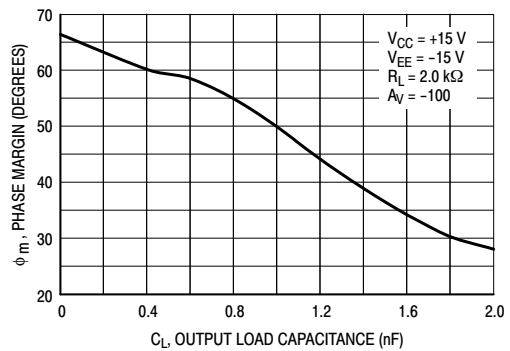


Figure 5. Phase Margin versus Output Load Capacitance

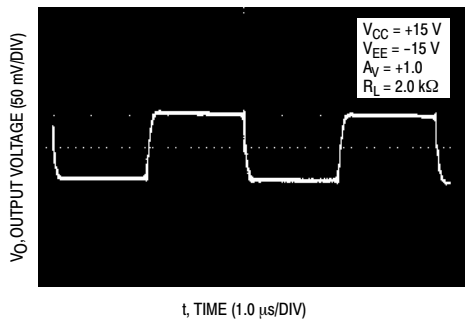


Figure 6. Small Signal Transient Response

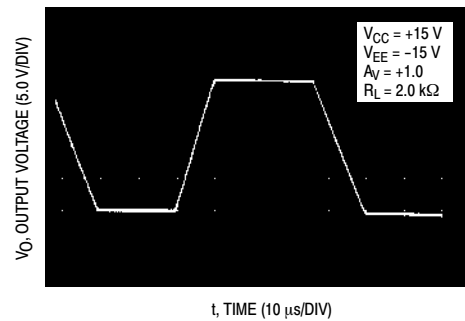


Figure 7. Large Signal Transient Response

TCA0372, TCA0372B, NCV0372B

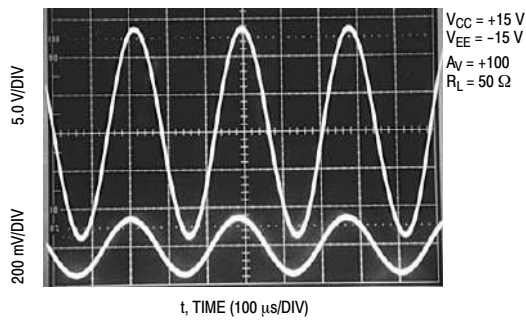


Figure 8. Sine Wave Response

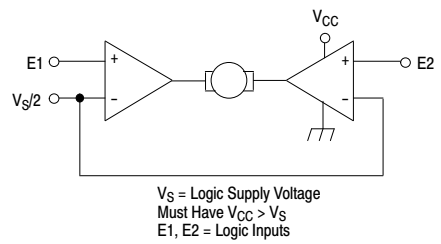
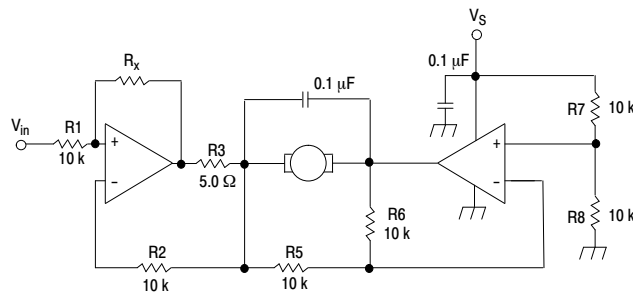


Figure 9. Bidirectional DC Motor Control with Microprocessor-Compatible Inputs  
 $V_S$  = Logic Supply Voltage  
 Must Have  $V_{CC} > V_S$   
 E1, E2 = Logic Inputs



For circuit stability, ensure that  $R_x > \frac{2R_3 \cdot R_1}{R_M}$  where,  $R_M$  = internal resistance of motor.  
 The voltage available at the terminals of the motor is:  $V_M = 2(V_1 - \frac{V_S}{2}) + |R_0| \cdot I_M$   
 where,  $|R_0| = \frac{2R_3 \cdot R_1}{R_x}$  and  $I_M$  is the motor current.

Figure 10. Bidirectional Speed Control of DC Motors

**TCA0372, TCA0372B, NCV0372B**

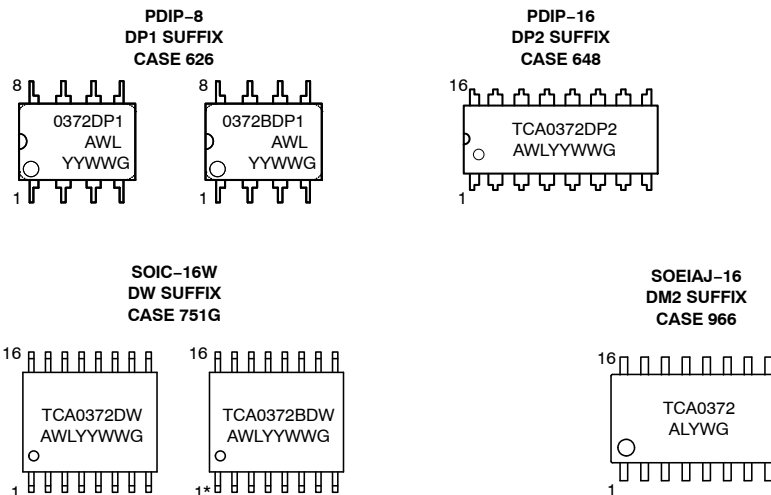
**ORDERING INFORMATION**

Device	Package	Shipping†
TCA0372DWG	SOIC-16W (Pb-Free)	47 Units / Rail
TCA0372DWR2G	SOIC-16W (Pb-Free)	1000 / Tape & Reel
TCA0372BDWR2G	SOIC-16W (Pb-Free)	1000 / Tape & Reel
NCV0372BDWR2G*	SOIC-16W (Pb-Free)	1000 / Tape & Reel
TCA0372DP1G	PDIP-8 (Pb-Free)	50 Units / Rail
TCA0372BDP1G	PDIP-8 (Pb-Free)	50 Units / Rail
TCA0372DP2G	PDIP-16 (Pb-Free)	25 Units / Rail
TCA0372DM2ELG	SOEIAJ-16 (Pb-Free)	2500 / Tape & Reel

†For information on tape and reel specifications, including part orientation and tape sizes, please refer to our Tape and Reel Packaging Specifications Brochure, BRD8011/D.

\*AEC-Q100 Qualified and PPAP Capable

**MARKING DIAGRAMS**



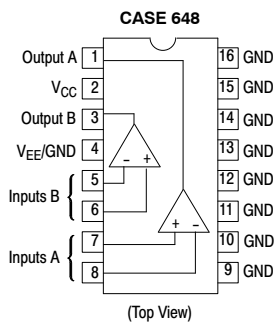
\*Also applies to NCV0372BDWR2G.

- A = Assembly Location
- WL, L = Wafer Lot
- YY, Y = Year
- WW, W = Work Week
- G = Pb-Free Package

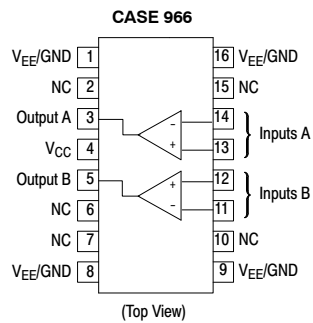
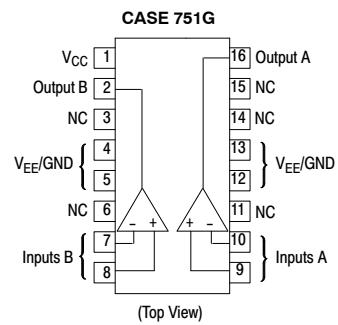
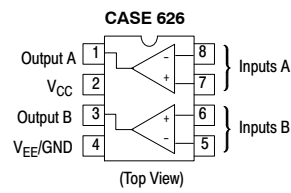


**TCA0372, TCA0372B, NCV0372B**

**PIN CONNECTIONS**



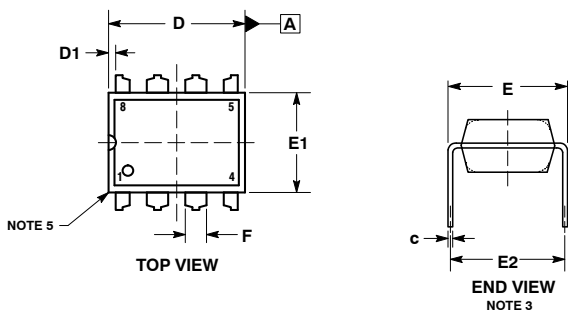
\*Pins 4 and 9 to 16 are internally connected.



TCA0372, TCA0372B, NCV0372B

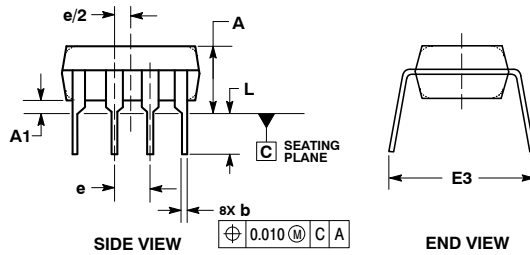
PACKAGE DIMENSIONS

**PDIP-8**  
**DP1 SUFFIX**  
 CASE 626-05  
 ISSUE M

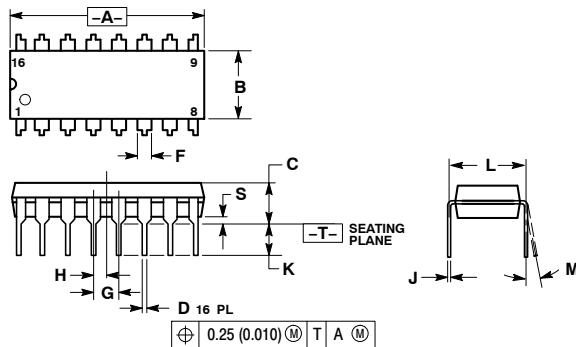


- NOTES:
1. DIMENSIONING AND TOLERANCING PER ASME Y14.5M, 1994.
  2. CONTROLLING DIMENSION: INCHES.
  3. DIMENSION E IS MEASURED WITH THE LEADS RESTRAINED PARALLEL AT WIDTH E2.
  4. DIMENSION E1 DOES NOT INCLUDE MOLD FLASH.
  5. ROUNDED CORNERS OPTIONAL.

DIM	INCHES			MILLIMETERS		
	MIN	NOM	MAX	MIN	NOM	MAX
A	---	---	0.210	---	---	5.33
A1	0.015	---	---	0.38	---	---
b	0.014	0.018	0.022	0.35	0.46	0.56
C	0.008	0.010	0.014	0.20	0.25	0.36
D	0.355	0.365	0.400	9.02	9.27	10.02
D1	0.005	---	---	0.13	---	---
E	0.300	0.310	0.325	7.62	7.87	8.26
E1	0.240	0.250	0.280	6.10	6.35	7.11
E2	---	0.300 BSC	---	7.62 BSC	---	---
E3	---	---	0.430	---	---	10.92
e	---	0.100 BSC	---	2.54 BSC	---	---
L	0.115	0.130	0.150	2.92	3.30	3.81



**PDIP-16**  
**DP2 SUFFIX**  
 CASE 648-08  
 ISSUE T



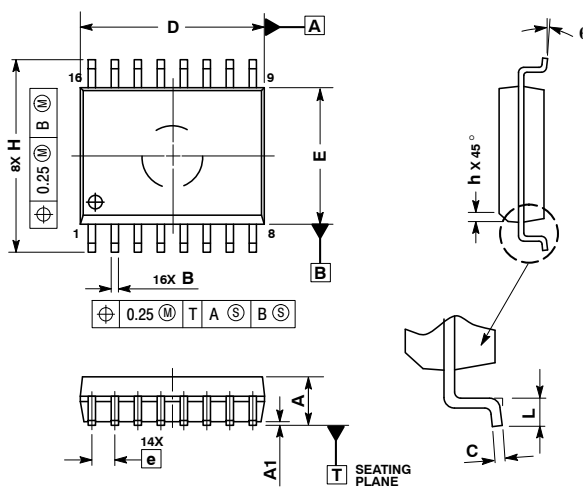
- NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
  2. CONTROLLING DIMENSION: INCH.
  3. DIMENSION L TO CENTER OF LEADS WHEN FORMED PARALLEL.
  4. DIMENSION B DOES NOT INCLUDE MOLD FLASH.
  5. ROUNDED CORNERS OPTIONAL.

DIM	INCHES		MILLIMETERS	
	MIN	MAX	MIN	MAX
A	0.740	0.770	18.80	19.55
B	0.250	0.270	6.35	6.85
C	0.145	0.175	3.69	4.44
D	0.015	0.021	0.39	0.53
F	0.040	0.70	1.02	1.77
G	---	0.100 BSC	---	2.54 BSC
H	---	0.050 BSC	---	1.27 BSC
J	0.008	0.015	0.21	0.38
K	0.110	0.130	2.80	3.30
L	0.295	0.305	7.50	7.74
M	0°	10°	0°	10°
S	0.020	0.040	0.51	1.01

TCA0372, TCA0372B, NCV0372B

PACKAGE DIMENSIONS

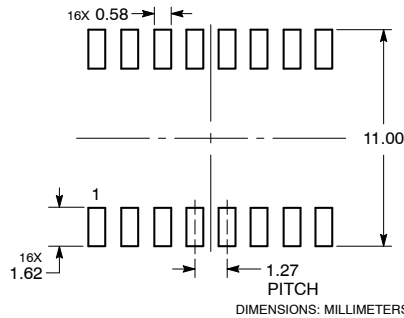
SOIC-16 WB  
CASE 751G-03  
ISSUE D



- NOTES:
1. DIMENSIONS ARE IN MILLIMETERS.
  2. INTERPRET DIMENSIONS AND TOLERANCES PER ASME Y14.5M, 1994.
  3. DIMENSIONS D AND E DO NOT INCLUDE MOLD PROTRUSION.
  4. MAXIMUM MOLD PROTRUSION 0.15 PER SIDE.
  5. DIMENSION B DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL BE 0.13 TOTAL IN EXCESS OF THE B DIMENSION AT MAXIMUM MATERIAL CONDITION.

MILLIMETERS		
DIM	MIN	MAX
A	2.35	2.65
A1	0.10	0.25
B	0.35	0.49
C	0.23	0.32
D	10.15	10.45
E	7.40	7.60
e	1.27 BSC	
H	10.05	10.55
h	0.25	0.75
L	0.50	0.90
q	0 <sup>o</sup>	7 <sup>o</sup>

SOLDERING FOOTPRINT\*

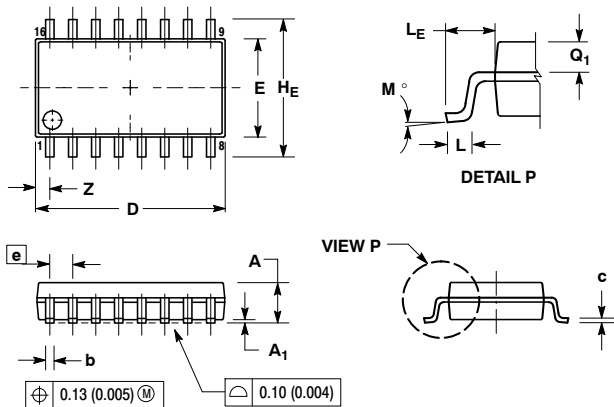


DIMENSIONS: MILLIMETERS

\*For additional information on our Pb-Free strategy and soldering details, please download the ON Semiconductor Soldering and Mounting Techniques Reference Manual, SOLDERRM/D.

**TCA0372, TCA0372B, NCV0372B**

**SOEIAJ-16  
DM2 SUFFIX  
CASE 966  
ISSUE A**



- NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
  2. CONTROLLING DIMENSION: MILLIMETER.
  3. DIMENSIONS D AND E DO NOT INCLUDE MOLD FLASH OR PROTRUSIONS AND ARE MEASURED AT THE PARTING LINE. MOLD FLASH OR PROTRUSIONS SHALL NOT EXCEED 0.15 (0.006) PER SIDE.
  4. TERMINAL NUMBERS ARE SHOWN FOR REFERENCE ONLY.
  5. THE LEAD WIDTH DIMENSION (b) DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL BE 0.08 (0.003) TOTAL IN EXCESS OF THE LEAD WIDTH DIMENSION AT MAXIMUM MATERIAL CONDITION. DAMBAR CANNOT BE LOCATED ON THE LOWER RADIUS OR THE FOOT. MINIMUM SPACE BETWEEN PROTRUSIONS AND ADJACENT LEAD TO BE 0.46 (0.018).

DIM	MILLIMETERS		INCHES	
	MIN	MAX	MIN	MAX
A	---	2.05	---	0.081
A <sub>1</sub>	0.05	0.20	0.002	0.008
b	0.35	0.50	0.014	0.020
c	0.10	0.20	0.007	0.011
D	9.90	10.50	0.390	0.413
E	5.10	5.45	0.201	0.215
e	1.27 BSC		0.050 BSC	
HE	7.40	8.20	0.291	0.323
L	0.50	0.85	0.020	0.033
L <sub>c</sub>	1.10	1.50	0.043	0.059
M	0°	10°	0°	10°
Q <sub>1</sub>	0.70	0.90	0.028	0.035
Z	---	0.78	---	0.031

ON Semiconductor and are registered trademarks of Semiconductor Components Industries, LLC (SCILLC). SCILLC owns the rights to a number of patents, trademarks, copyrights, trade secrets, and other intellectual property. A listing of SCILLC's product/patent coverage may be accessed at [www.onsemi.com/site/pdf/Patent-Marketing.pdf](http://www.onsemi.com/site/pdf/Patent-Marketing.pdf). SCILLC reserves the right to make changes without further notice to any products herein. SCILLC makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does SCILLC assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation special, consequential or incidental damages. "Typical" parameters which may be provided in SCILLC data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts. SCILLC does not convey any license under its patent rights nor the rights of others. SCILLC products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the SCILLC product could create a situation where personal injury or death may occur. Should Buyer purchase or use SCILLC products for any such unintended or unauthorized application, Buyer shall indemnify and hold SCILLC and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that SCILLC was negligent regarding the design or manufacture of the part. SCILLC is an Equal Opportunity/Affirmative Action Employer. This literature is subject to all applicable copyright laws and is not for resale in any manner.

**PUBLICATION ORDERING INFORMATION**

**LITERATURE FULFILLMENT:**  
Literature Distribution Center for ON Semiconductor  
P.O. Box 5163, Denver, Colorado 80217 USA  
Phone: 303-675-2175 or 800-344-3860 Toll Free USA/Canada  
Fax: 303-675-2176 or 800-344-3867 Toll Free USA/Canada  
Email: [orderlit@onsemi.com](mailto:orderlit@onsemi.com)

**N. American Technical Support:** 800-282-9855 Toll Free  
USA/Canada  
**Europe, Middle East and Africa Technical Support:**  
Phone: 421 33 790 2910  
**Japan Customer Focus Center**  
Phone: 81-3-5817-1050

**ON Semiconductor Website:** [www.onsemi.com](http://www.onsemi.com)  
**Order Literature:** <http://www.onsemi.com/orderlit>

For additional information, please contact your local Sales Representative

## D.4.2. Regulador de tensión 7805

**FAIRCHILD**  
SEMICONDUCTOR®

www.fairchildsemi.com

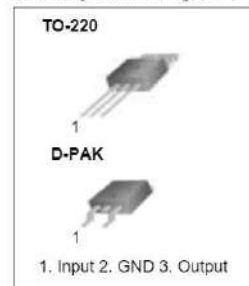
### MC78XX/LM78XX/MC78XXA 3-Terminal 1A Positive Voltage Regulator

#### Features

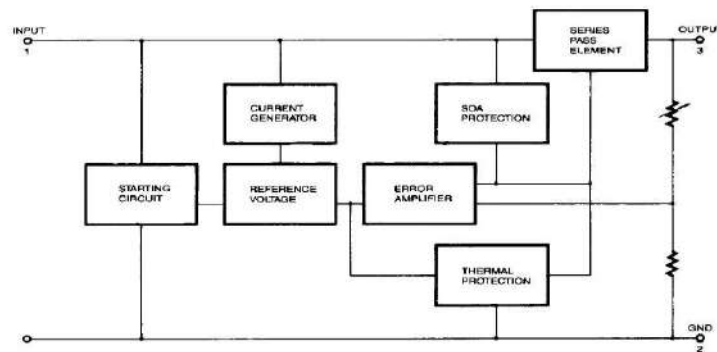
- Output Current up to 1A
- Output Voltages of 5, 6, 8, 9, 10, 12, 15, 18, 24V
- Thermal Overload Protection
- Short Circuit Protection
- Output Transistor Safe Operating Area Protection

#### Description

The MC78XX/LM78XX/MC78XXA series of three terminal positive regulators are available in the TO-220/D-PAK package and with several fixed output voltages, making them useful in a wide range of applications. Each type employs internal current limiting, thermal shut down and safe operating area protection, making it essentially indestructible. If adequate heat sinking is provided, they can deliver over 1A output current. Although designed primarily as fixed voltage regulators, these devices can be used with external components to obtain adjustable voltages and currents.



#### Internal Block Diagram



Rev. 1.0.1

MC78XX/LM78XX/MC78XXA

**Absolute Maximum Ratings**

Parameter	Symbol	Value	Unit
Input Voltage (for $V_O = 5V$ to $18V$ ) (for $V_O = 24V$ )	$V_I$ $V_I$	35 40	V V
Thermal Resistance Junction-Cases (TO-220)	$R_{\theta JC}$	5	$^{\circ}C/W$
Thermal Resistance Junction-Air (TO-220)	$R_{\theta JA}$	65	$^{\circ}C/W$
Operating Temperature Range	$T_{OPR}$	0 ~ +125	$^{\circ}C$
Storage Temperature Range	$T_{STG}$	-65 ~ +150	$^{\circ}C$

**Electrical Characteristics (MC7805/LM7805)**(Refer to test circuit,  $0^{\circ}C < T_J < 125^{\circ}C$ ,  $I_O = 500mA$ ,  $V_I = 10V$ ,  $C_I = 0.33\mu F$ ,  $C_O = 0.1\mu F$ , unless otherwise specified)

Parameter	Symbol	Conditions	MC7805/LM7805			Unit	
			Min.	Typ.	Max.		
Output Voltage	$V_O$	$T_J = +25^{\circ}C$	4.8	5.0	5.2	V	
		$5.0mA \leq I_O \leq 1.0A$ , $P_O \leq 15W$ $V_I = 7V$ to $20V$	4.75	5.0	5.25		
Line Regulation (Note1)	Regline	$T_J = +25^{\circ}C$	$V_O = 7V$ to $25V$	-	4.0	100	mV
			$V_I = 8V$ to $12V$	-	1.6	50	
Load Regulation (Note1)	Regload	$T_J = +25^{\circ}C$	$I_O = 5.0mA$ to $1.5A$	-	9	100	mV
			$I_O = 250mA$ to $750mA$	-	4	50	
Quiescent Current	$I_Q$	$T_J = +25^{\circ}C$	-	5.0	8.0	mA	
Quiescent Current Change	$\Delta I_Q$	$I_O = 5mA$ to $1.0A$	-	0.03	0.5	mA	
		$V_I = 7V$ to $25V$	-	0.3	1.3		
Output Voltage Drift	$\Delta V_O / \Delta T$	$I_O = 5mA$	-	-0.8	-	mV/ $^{\circ}C$	
Output Noise Voltage	$V_N$	$f = 10Hz$ to $100KHz$ , $T_A = +25^{\circ}C$	-	42	-	$\mu V/V_O$	
Ripple Rejection	RR	$f = 120Hz$ $V_O = 8V$ to $18V$	62	73	-	dB	
Dropout Voltage	$V_{Drop}$	$I_O = 1A$ , $T_J = +25^{\circ}C$	-	2	-	V	
Output Resistance	$r_O$	$f = 1KHz$	-	15	-	m $\Omega$	
Short Circuit Current	$I_{SC}$	$V_I = 35V$ , $T_A = +25^{\circ}C$	-	230	-	mA	
Peak Current	$I_{PK}$	$T_J = +25^{\circ}C$	-	2.2	-	A	

**Note:**

1. Load and line regulation are specified at constant junction temperature. Changes in  $V_O$  due to heating effects must be taken into account separately. Pulse testing with low duty is used.

### D.4.3. Microcontrolador ATM328

#### Features

- High Performance, Low Power AVR<sup>®</sup> 8-Bit Microcontroller
- Advanced RISC Architecture
  - 131 Powerful Instructions – Most Single Clock Cycle Execution
  - 32 x 8 General Purpose Working Registers
  - Fully Static Operation
  - Up to 20 MIPS Throughput at 20 MHz
  - On-chip 2-cycle Multiplier
- High Endurance Non-volatile Memory Segments
  - 4/8/16/32K Bytes of In-System Self-Programmable Flash program memory (ATmega48P/88P/168P/328P)
  - 256/512/512/1K Bytes EEPROM (ATmega48P/88P/168P/328P)
  - 512/1K/1K/2K Bytes Internal SRAM (ATmega48P/88P/168P/328P)
  - Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
  - Data retention: 20 years at 85°C/100 years at 25°C<sup>(1)</sup>
  - Optional Boot Code Section with Independent Lock Bits
    - In-System Programming by On-chip Boot Program
    - True Read-While-Write Operation
  - Programming Lock for Software Security
- Peripheral Features
  - Two 8-bit Timer/Counters with Separate Prescaler and Compare Mode
  - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
  - Real Time Counter with Separate Oscillator
  - Six PWM Channels
  - 8-channel 10-bit ADC in TQFP and QFN/MLF package
    - Temperature Measurement
  - 6-channel 10-bit ADC in PDIP Package
    - Temperature Measurement
  - Programmable Serial USART
  - Master/Slave SPI Serial Interface
  - Byte-oriented 2-wire Serial Interface (Philips I<sup>2</sup>C compatible)
  - Programmable Watchdog Timer with Separate On-chip Oscillator
  - On-chip Analog Comparator
  - Interrupt and Wake-up on Pin Change
- Special Microcontroller Features
  - Power-on Reset and Programmable Brown-out Detection
  - Internal Calibrated Oscillator
  - External and Internal Interrupt Sources
  - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby, and Extended Standby
- I/O and Packages
  - 23 Programmable I/O Lines
  - 28-pin PDIP, 32-lead TQFP, 28-pad QFN/MLF and 32-pad QFN/MLF
- Operating Voltage:
  - 1.8 - 5.5V for ATmega48P/88P/168PV
  - 2.7 - 5.5V for ATmega48P/88P/168P
  - 1.8 - 5.5V for ATmega328P
- Temperature Range:
  - -40°C to 85°C
- Speed Grade:
  - ATmega48P/88P/168PV: 0 - 4 MHz @ 1.8 - 5.5V, 0 - 10 MHz @ 2.7 - 5.5V
  - ATmega48P/88P/168P: 0 - 10 MHz @ 2.7 - 5.5V, 0 - 20 MHz @ 4.5 - 5.5V
  - ATmega328P: 0 - 4 MHz @ 1.8 - 5.5V, 0 - 10 MHz @ 2.7 - 5.5V, 0 - 20 MHz @ 4.5 - 5.5V
- Low Power Consumption at 1 MHz, 1.8V, 25°C for ATmega48P/88P/168P:
  - Active Mode: 0.3 mA
  - Power-down Mode: 0.1 µA
  - Power-save Mode: 0.8 µA (Including 32 kHz RTC)



**8-bit AVR<sup>®</sup>  
Microcontroller  
with 4/8/16/32K  
Bytes In-System  
Programmable  
Flash**

**ATmega48P/V  
ATmega88P/V  
ATmega168P/V  
ATmega328P**

**Preliminary**

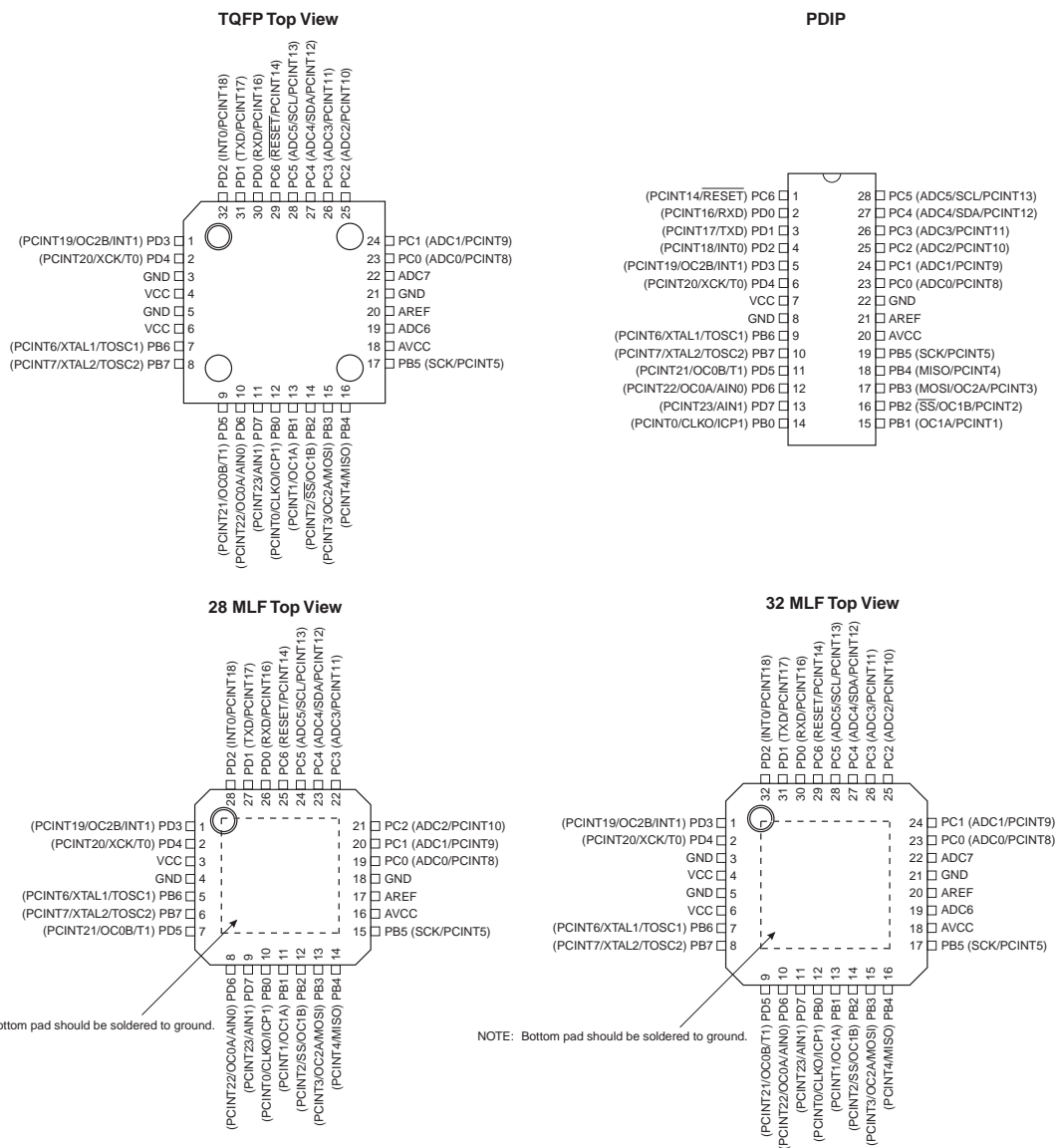
**Summary**





### 1. Pin Configurations

Figure 1-1. Pinout ATmega48P/88P/168P/328P





## ATmega48P/88P/168P/328P

### 1.1 Pin Descriptions

#### 1.1.1 VCC

Digital supply voltage.

#### 1.1.2 GND

Ground.

#### 1.1.3 Port B (PB7:0) XTAL1/XTAL2/TOSC1/TOSC2

Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running.

Depending on the clock selection fuse settings, PB6 can be used as input to the inverting Oscillator amplifier and input to the internal clock operating circuit.

Depending on the clock selection fuse settings, PB7 can be used as output from the inverting Oscillator amplifier.

If the Internal Calibrated RC Oscillator is used as chip clock source, PB7..6 is used as TOSC2..1 input for the Asynchronous Timer/Counter2 if the AS2 bit in ASSR is set.

The various special features of Port B are elaborated in ["Alternate Functions of Port B" on page 82](#) and ["System Clock and Clock Options" on page 26](#).

#### 1.1.4 Port C (PC5:0)

Port C is a 7-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The PC5..0 output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not running.

#### 1.1.5 PC6/ $\overline{\text{RESET}}$

If the RSTDISBL Fuse is programmed, PC6 is used as an I/O pin. Note that the electrical characteristics of PC6 differ from those of the other pins of Port C.

If the RSTDISBL Fuse is unprogrammed, PC6 is used as a Reset input. A low level on this pin for longer than the minimum pulse length will generate a Reset, even if the clock is not running. The minimum pulse length is given in [Table 28-3 on page 320](#). Shorter pulses are not guaranteed to generate a Reset.

The various special features of Port C are elaborated in ["Alternate Functions of Port C" on page 85](#).

#### 1.1.6 Port D (PD7:0)

Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running.



The various special features of Port D are elaborated in "[Alternate Functions of Port D](#)" on page 88.

**1.1.7 AV<sub>CC</sub>**

AV<sub>CC</sub> is the supply voltage pin for the A/D Converter, PC3:0, and ADC7:6. It should be externally connected to V<sub>CC</sub>, even if the ADC is not used. If the ADC is used, it should be connected to V<sub>CC</sub> through a low-pass filter. Note that PC6..4 use digital supply voltage, V<sub>CC</sub>.

**1.1.8 AREF**

AREF is the analog reference pin for the A/D Converter.

**1.1.9 ADC7:6 (TQFP and QFN/MLF Package Only)**

In the TQFP and QFN/MLF package, ADC7:6 serve as analog inputs to the A/D converter. These pins are powered from the analog supply and serve as 10-bit ADC channels.

**1.2 Disclaimer**

Typical values contained in this datasheet are based on simulations and characterization of other AVR microcontrollers manufactured on the same process technology. Min and Max values will be available after the device is characterized.

## D.4.4. Cable de audio



## Datasheet

**4030.7126RN – Flat speaker cable 2 x 1.50 mm<sup>2</sup>**

EN

**Conductor :**

- Stranded
- Section : 1.50 mm<sup>2</sup>

**Outer Jacket :**

- Insulation : PVC
- Colours coding : red / black
- Ø : 2,60 x 5,20 mm

**PRODUCT INFORMATION**
**Application**

- Wiring all types of sound systems
- For fixed and mobile installations
- Conductor identification by colour

**Electrical Properties**

- Operating tension : < **900 V**
- Test tension : > **2500 V**
- Electrical resistance at 20°C : **18.1 Ω/km**

**Characteristics**

- Temperature range : **Static use** : - 25°C → + 70°C
  - **Mobile use** : - 5°C → + 70°C
- Minimum bending radius : **8 x Ø**
- Conform to **RoHS directive**
- Packaging : **100 m**

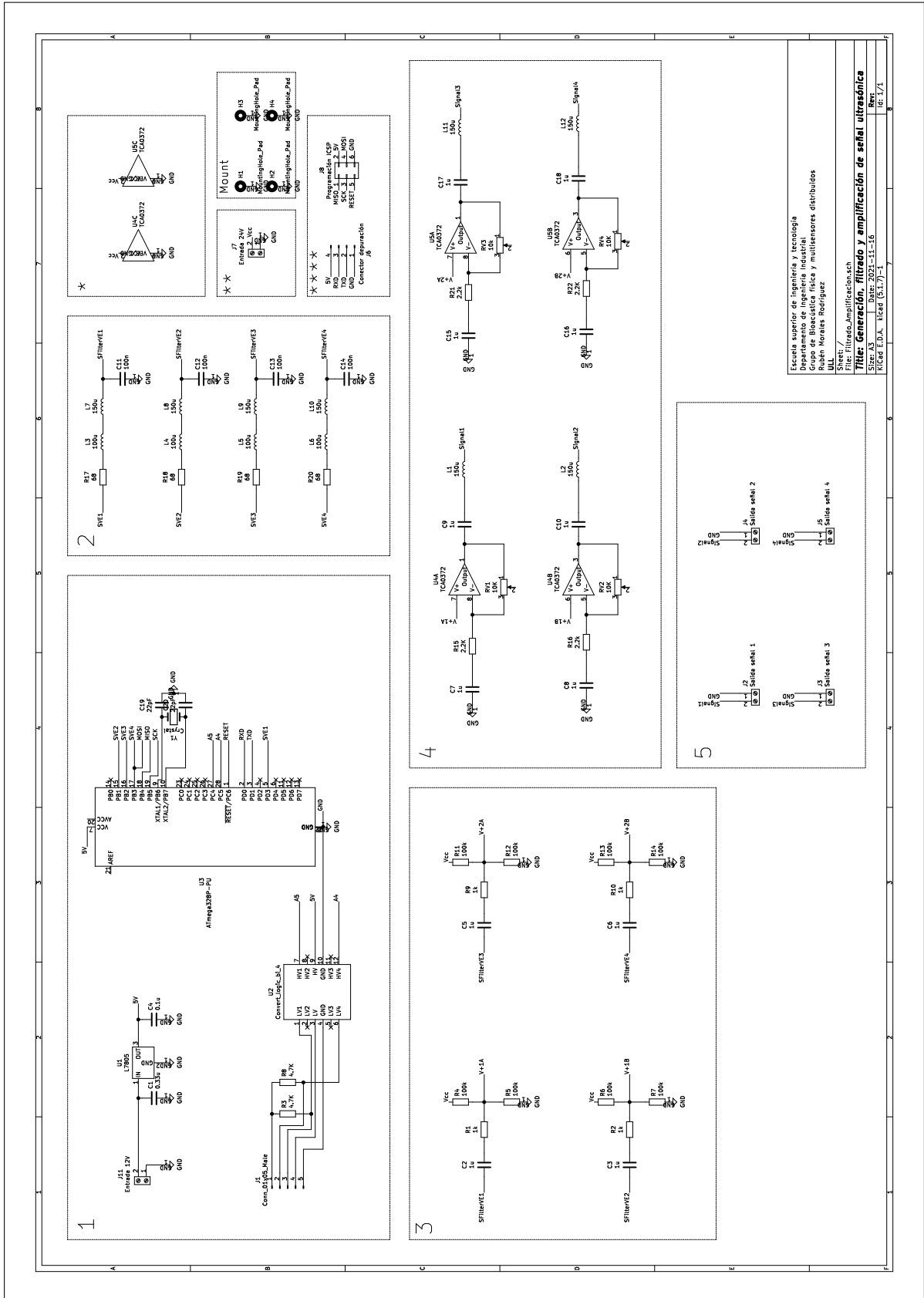


## **Anexo E**

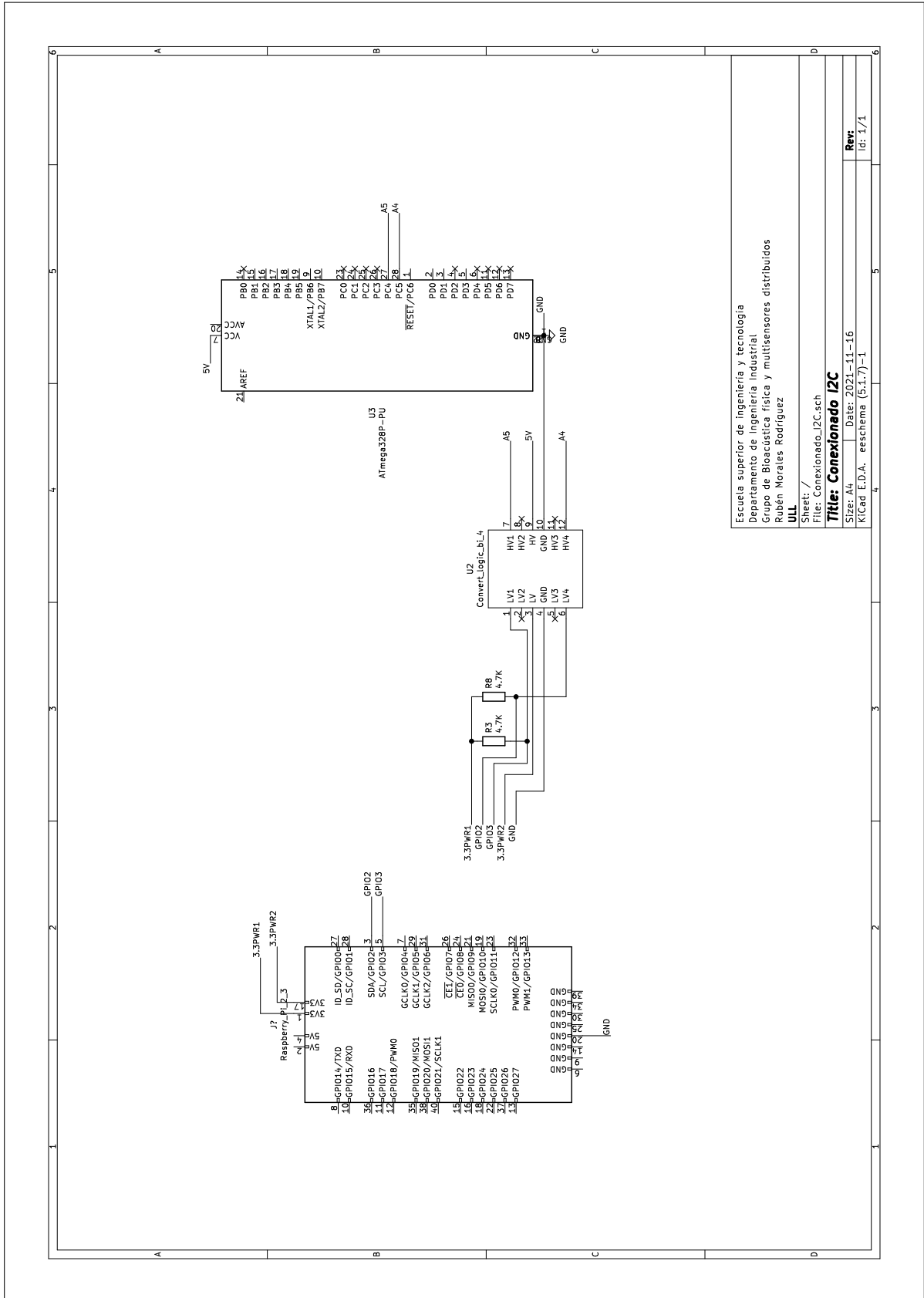
# **Esquematicos utilizados para el desarrollo del dispositivo**

En este capítulo se adjuntan todos aquellos esquematicos que se han desarrollado o usado para llevar a cabo el dispositivo funcional objeto de este proyecto.

E.0.1. Etapa de generación, filtrado y amplificación de señal



### E.0.2. Esquemático para la comunicación I2C







# **Anexo F**

## **Información geográfica**

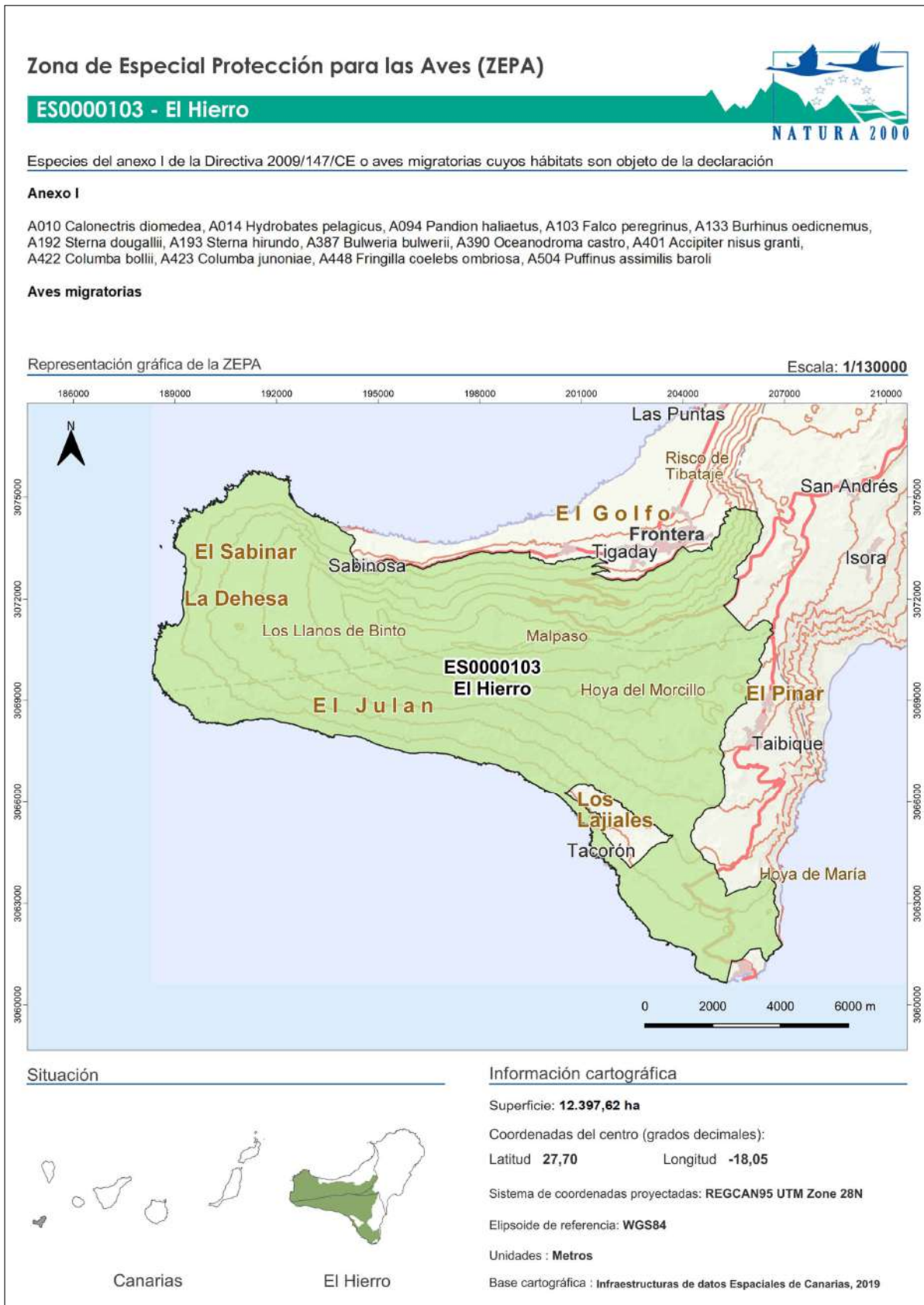
En este anexo se adjunta toda aquella información geográfica referente a las distintas zonas de especial protección avifauna (ZEPA) y a los cálculos de rendimiento fotovoltaico del módulo de alimentación.

### **F.1. Zonas de especial protección avifauna**

En esta sección se adjunta toda la información referente a las distintas zonas de especial protección avifauna del archipiélago Canario. Aunque es cierto que a la hora de desarrollar el módulo de alimentación del proyecto sólo se han tenido en cuenta dos localizaciones de las Islas Canarias, para dar un mayor aporte al lector, e intentar justificar de manera detallada la necesidad de hacer uso de sistemas como el planteado en este proyecto, se adjunta la información relevante a este tipo de zonas para todo el archipiélago.

### F.1.1. Zonas de especial protección avifauna de El Hierro







## Zona de Especial Protección para las Aves (ZEPA)

### ES0000104 - Gorreta y Salmor



Especies del anexo I de la Directiva 2009/147/CE o aves migratorias cuyos hábitats son objeto de la declaración

#### Anexo I

A010 Calonectris diomedea, A014 Hydrobates pelagicus, A094 Pandion haliaetus, A103 Falco peregrinus, A133 Burhinus oedicnemus, A192 Sterna dougallii, A193 Sterna hirundo, A387 Bulweria bulwerii, A390 Oceanodroma castro, A401 Accipiter nisus granti, A422 Columba bollii, A423 Columba junoniae, A448 Fringilla coelebs ombriosa, A504 Puffinus assimilis baroli

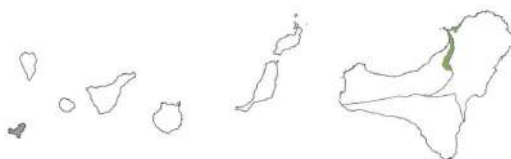
#### Aves migratorias

Representación gráfica de la ZEPA

Escala: 1/60000



#### Situación



Canarias

El Hierro

#### Información cartográfica

Superficie: **601,73 ha**

Coordenadas del centro (grados decimales):

Latitud **27,79**                      Longitud **-17,98**

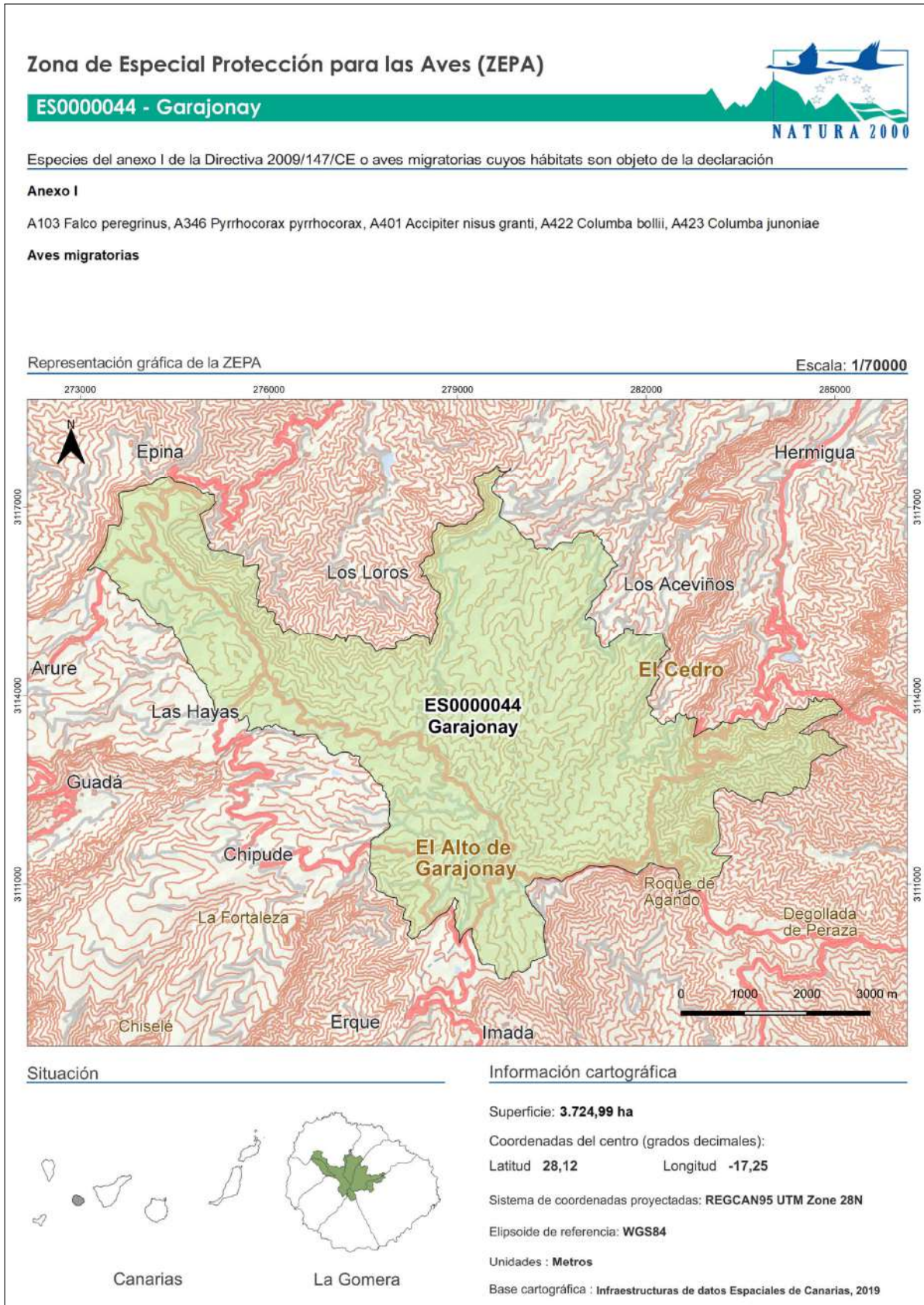
Sistema de coordenadas proyectadas: **REGCAN95 UTM Zone 28N**

Elipsoide de referencia: **WGS84**

Unidades: **Metros**

Base cartográfica: **Infraestructuras de datos Espaciales de Canarias, 2019**

**F.1.2. Zonas de especial protección avifauna de La Gomera**





## Zona de Especial Protección para las Aves (ZEPA)

**ES0000105 - Acanilados de Alajeró, la Dama y Valle Gran Rey**



Especies del anexo I de la Directiva 2009/147/CE o aves migratorias cuyos hábitats son objeto de la declaración

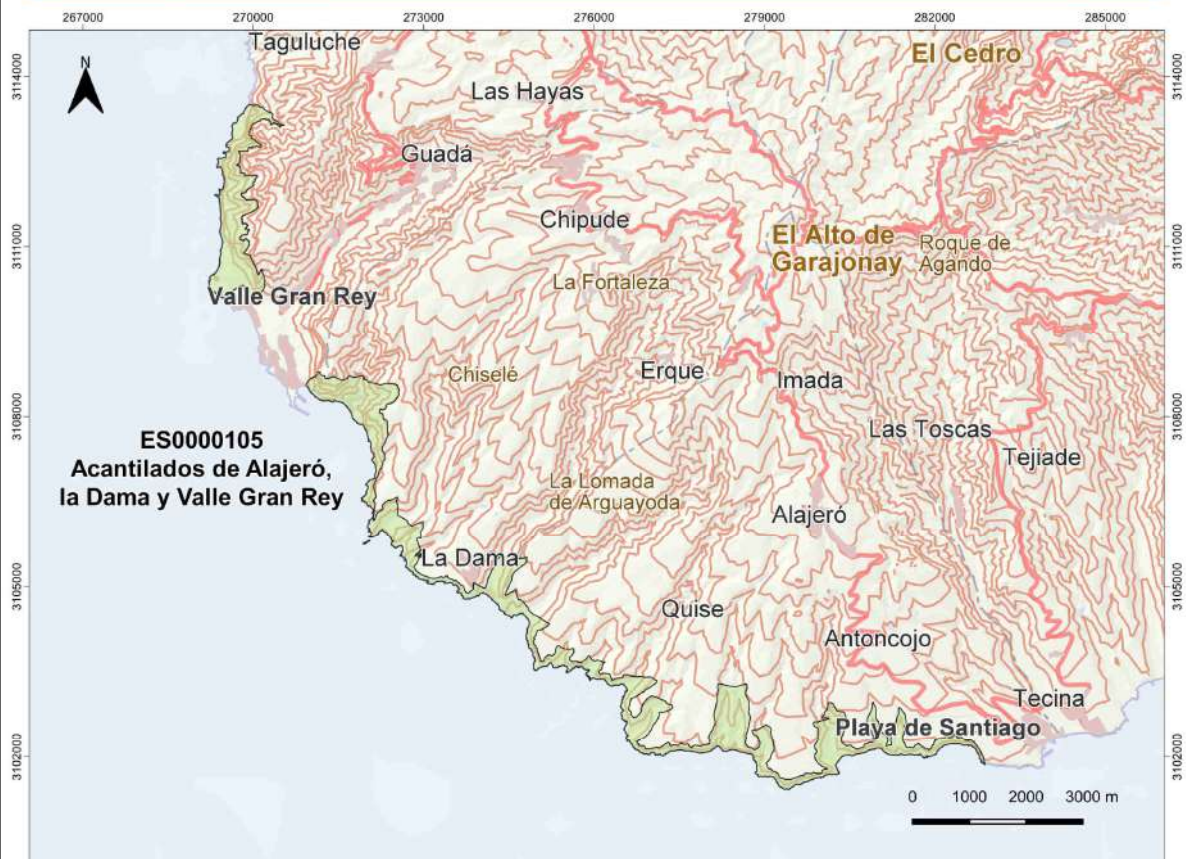
### Anexo I

A010 Calonectris diomedea, A094 Pandion haliaetus, A104 Hydrobates pelagicus, A103 Falco peregrinus, A133 Burhinus oedicnemus, A193 Sterna hirundo, A387 Bulweria bulwerii, A452 Bucanetes githagineus, A504 Puffinus assimilis baroli

### Aves migratorias

Representación gráfica de la ZEPA

Escala: 1/100000



### Situación



Canarias

La Gomera

### Información cartográfica

Superficie: **722,89 ha**

Coordenadas del centro (grados decimales):

Latitud **28,11** Longitud **-17,34**

Sistema de coordenadas proyectadas: **REGCAN95 UTM Zone 28N**

Elipsoide de referencia: **WGS84**

Unidades : **Metros**

Base cartográfica : **Infraestructuras de datos Espaciales de Canarias, 2019**





## Zona de Especial Protección para las Aves (ZEPA)

**ES0000342 - Costa de Majona, El Águila y Avalo**



Especies del anexo I de la Directiva 2009/147/CE o aves migratorias cuyos hábitats son objeto de la declaración

### Anexo I

A010 *Colonectris diomedea*, A094 *Pandion haliaetus*, A103 *Falco peregrinus*, A193 *Sterna hirundo*, A452 *Bucanetes githagineus*, A504 *Puffinus assimilis baroli*

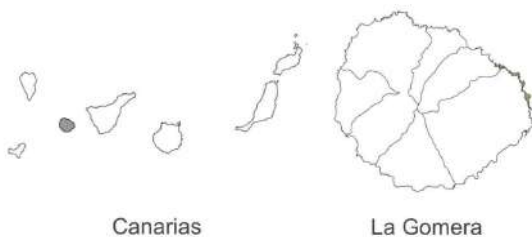
### Aves migratorias

Representación gráfica de la ZEPA

Escala: 1/45000



### Situación



Canarias

La Gomera

### Información cartográfica

Superficie: **174,60 ha**

Coordenadas del centro (grados decimales):

Latitud **28,13** Longitud **-17,11**

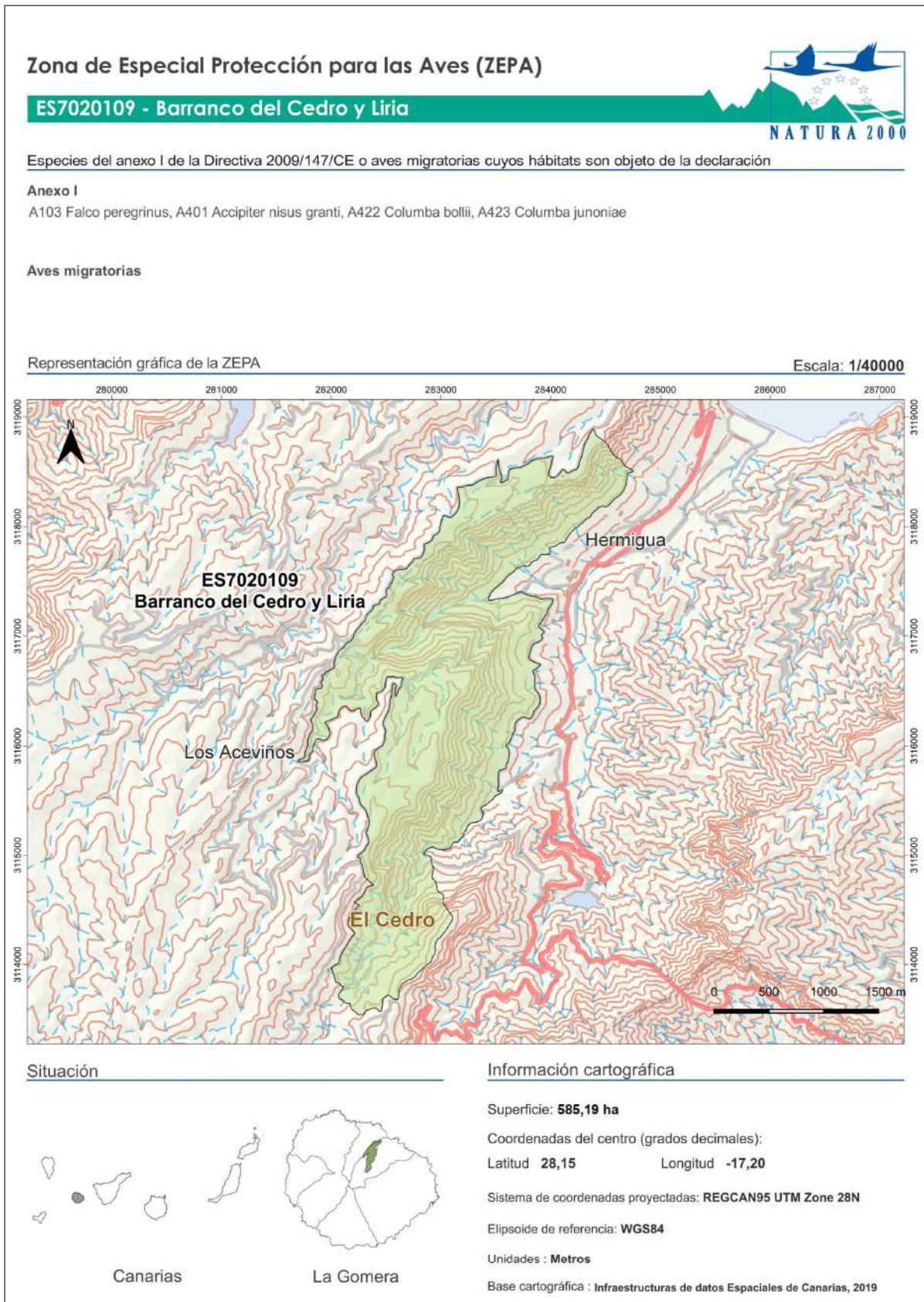
Sistema de coordenadas proyectadas: **REGCAN95 UTM Zone 28N**

Elipsoide de referencia: **WGS84**

Unidades : **Metros**

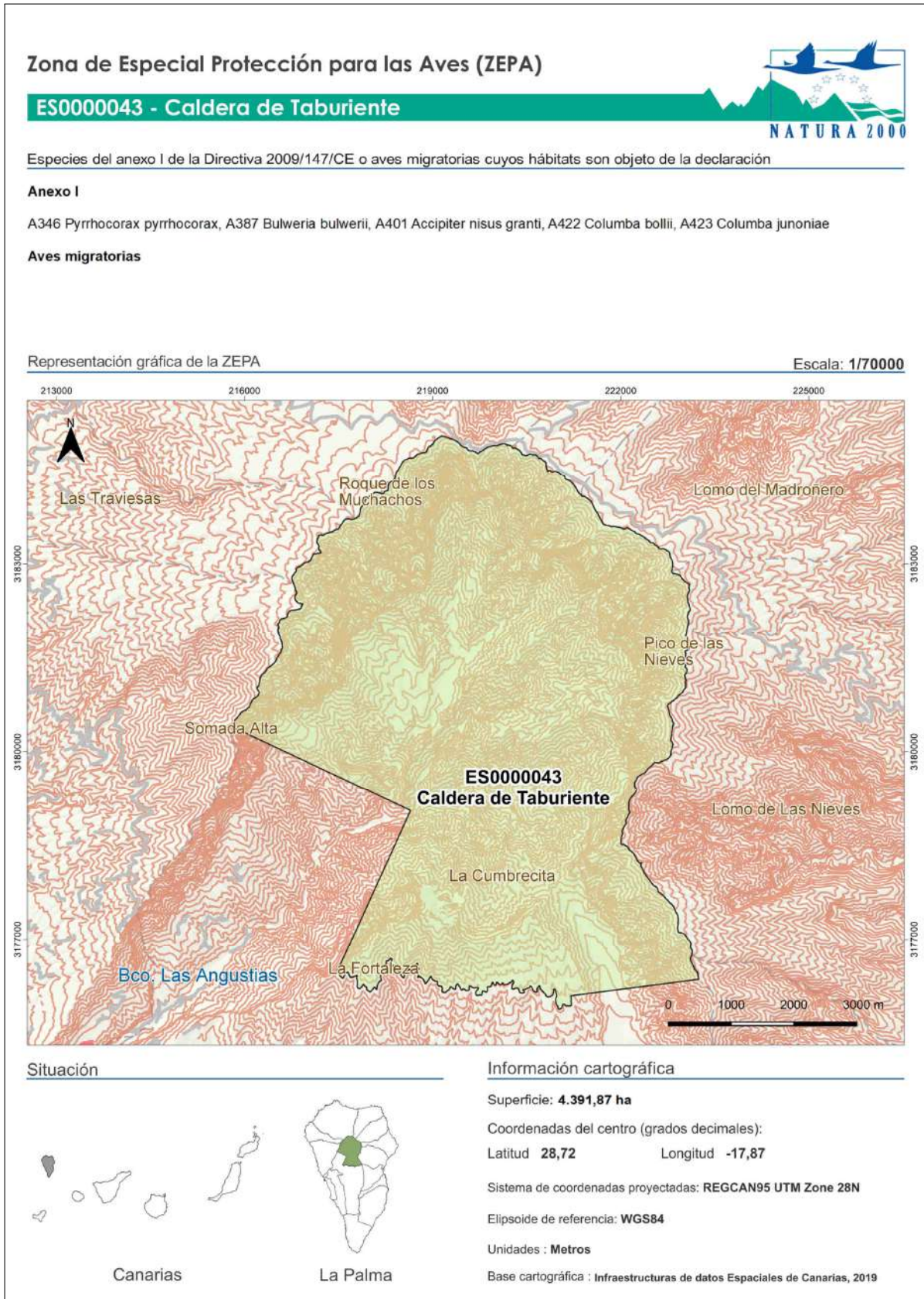
Base cartográfica : **Infraestructuras de datos Espaciales de Canarias, 2019**

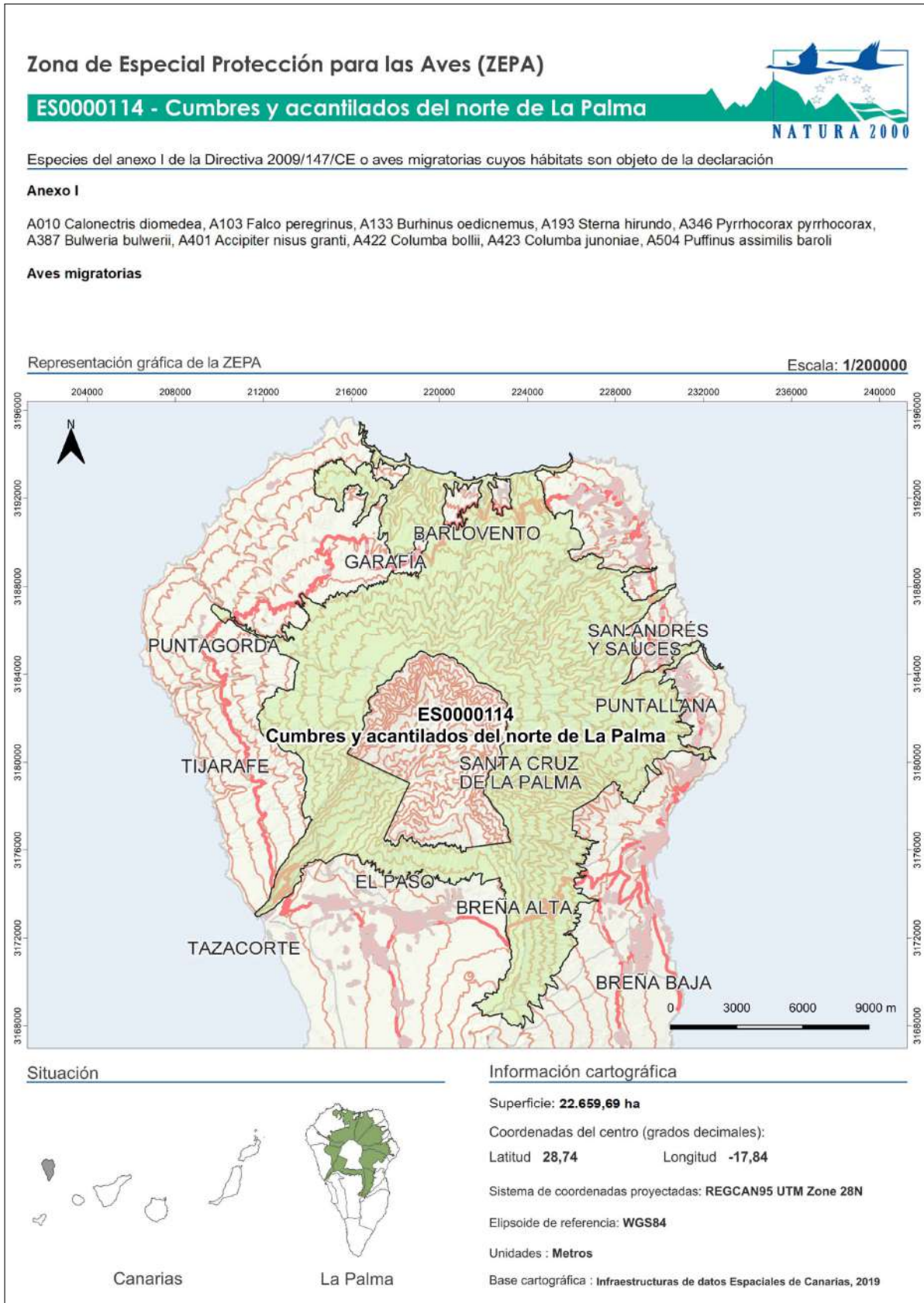




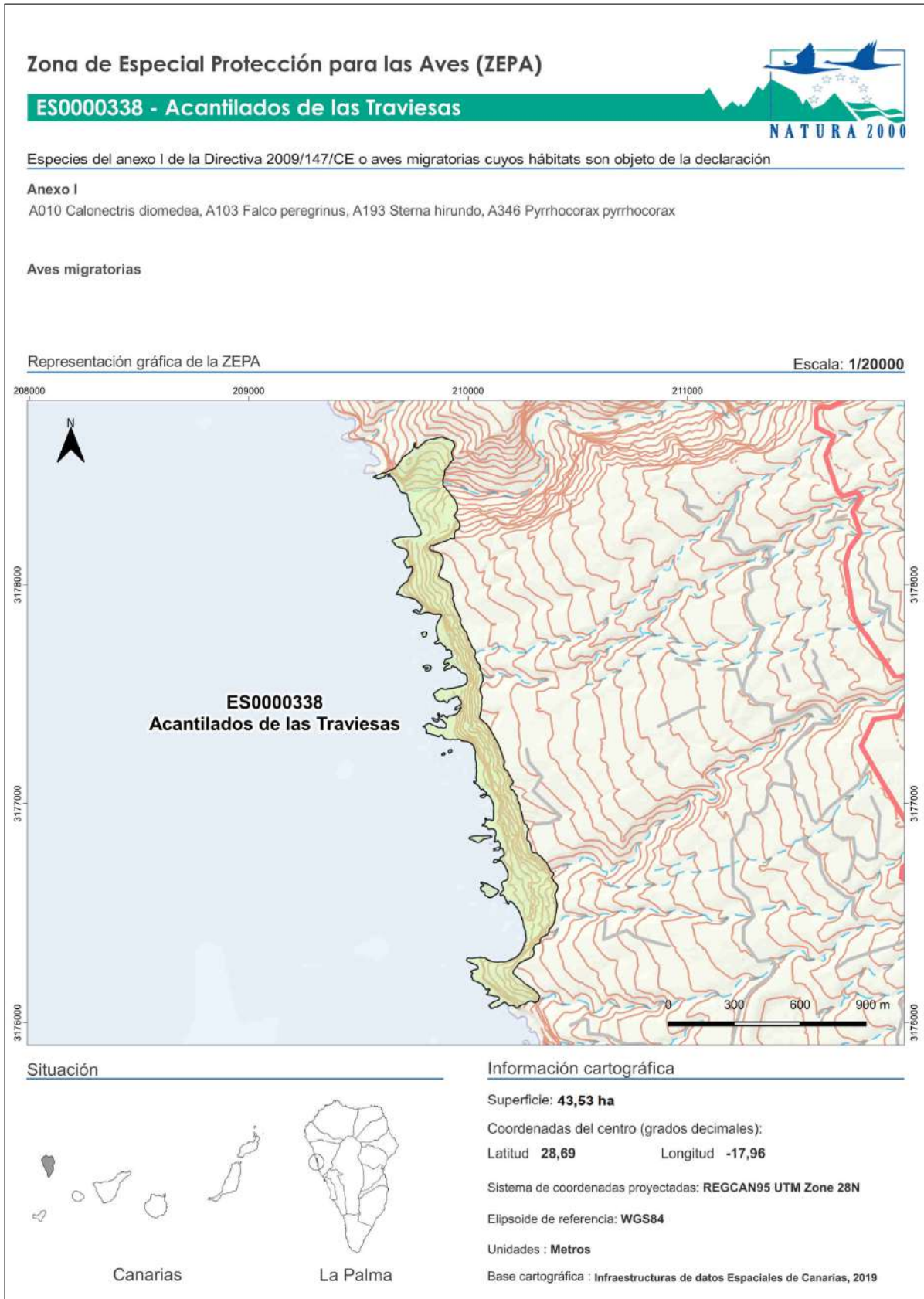


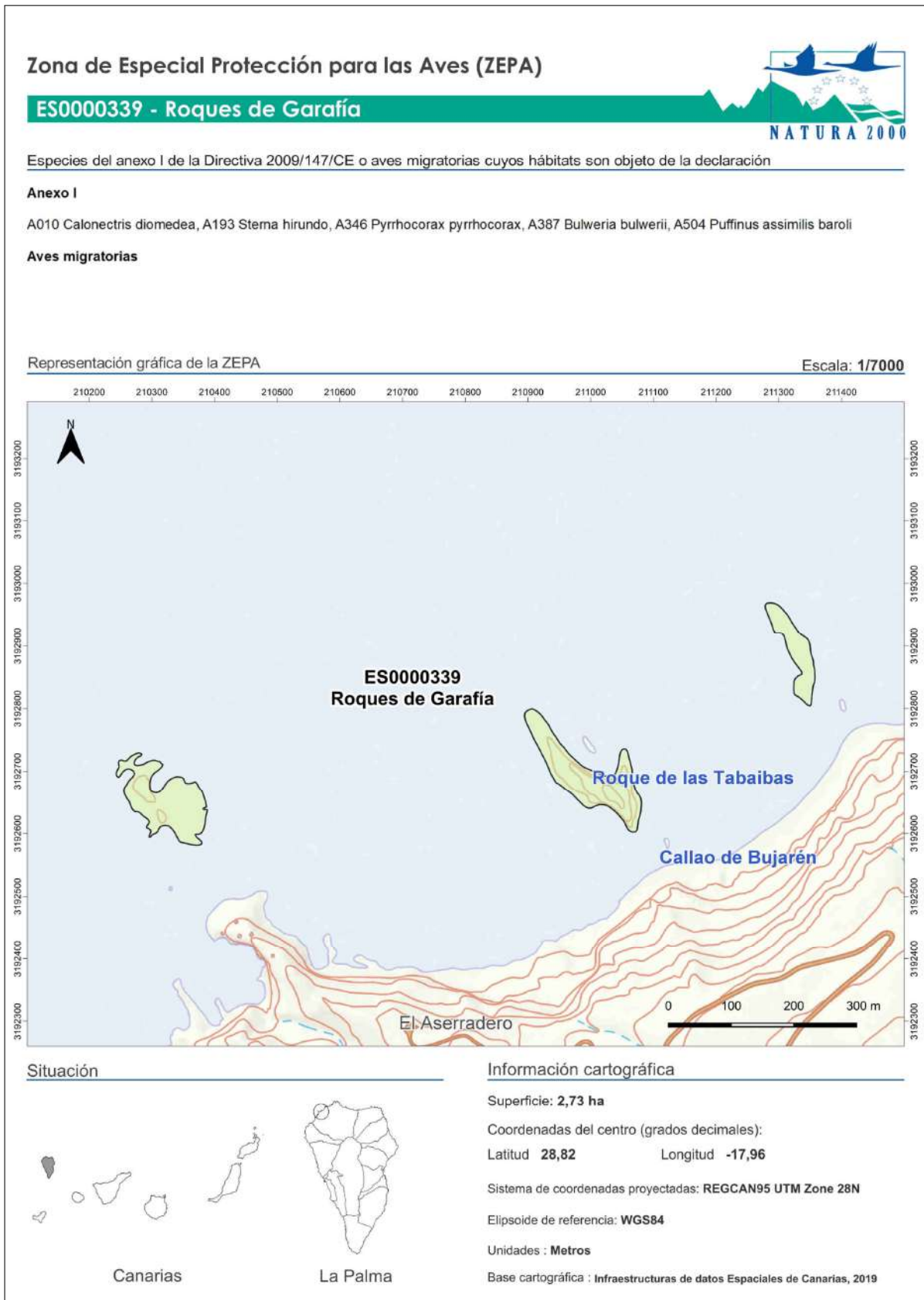
### F.1.3. Zonas de especial protección avifauna de La Palma











## Zona de Especial Protección para las Aves (ZEPA)

### ES0000340 - Roque Negro



Especies del anexo I de la Directiva 2009/147/CE o aves migratorias cuyos hábitats son objeto de la declaración

#### Anexo I

A010 Calonectris diomedea, A193 Sterna hirundo, A387 Bulweria bulwerii, A423 Columba junoniae, A504 Puffinus assimilis baroli

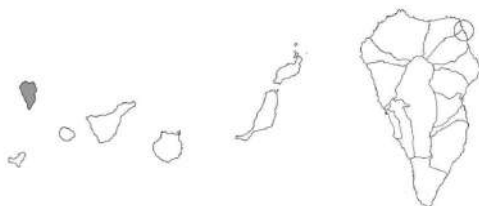
#### Aves migratorias

Representación gráfica de la ZEPA

Escala: 1/2000



#### Situación



Canarias

La Palma

#### Información cartográfica

Superficie: **1,71 ha**

Coordenadas del centro (grados decimales):

Latitud **28,81**      Longitud **-17,76**

Sistema de coordenadas proyectadas: **REGCAN95 UTM Zone 28N**

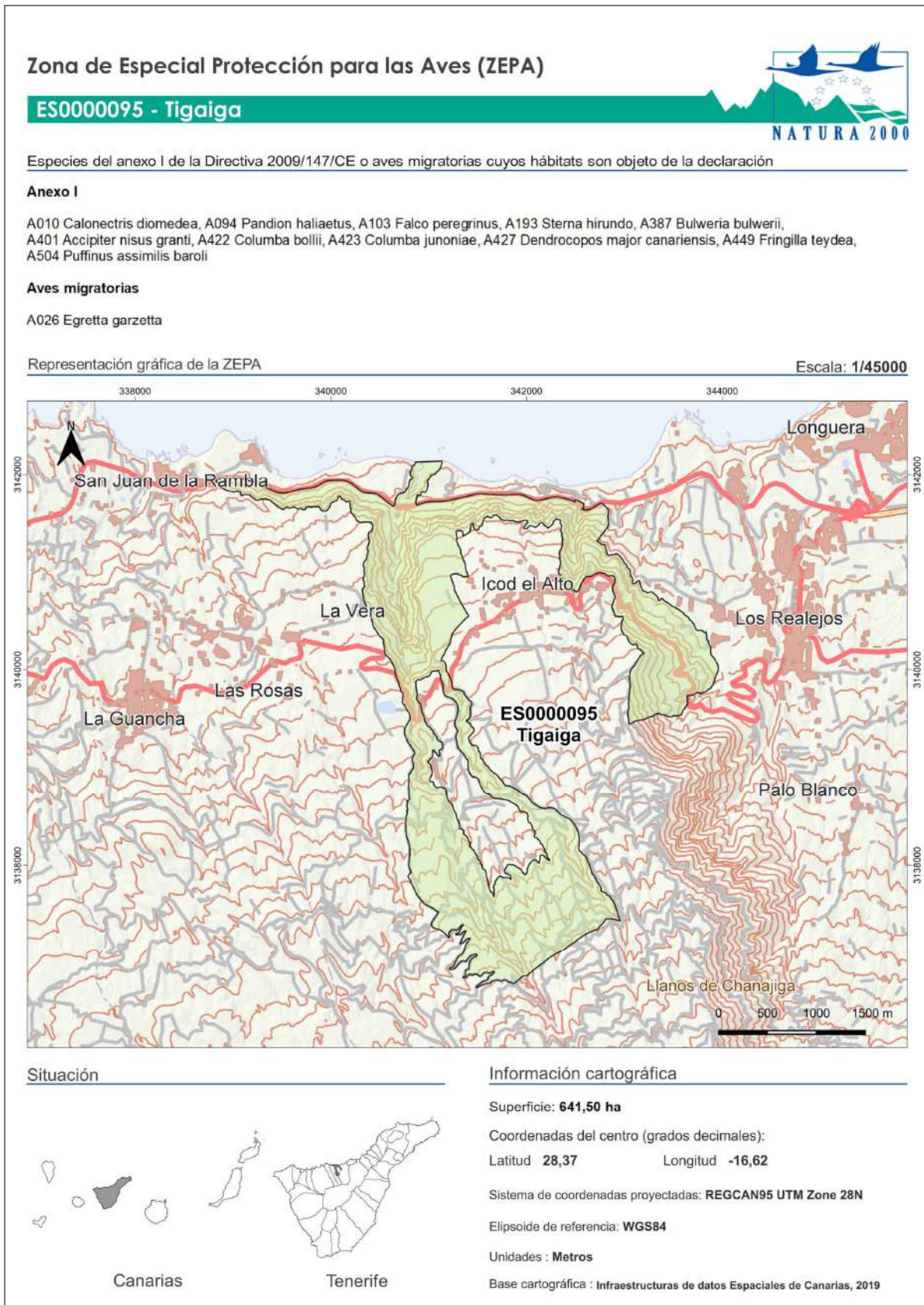
Elipsoide de referencia: **WGS84**

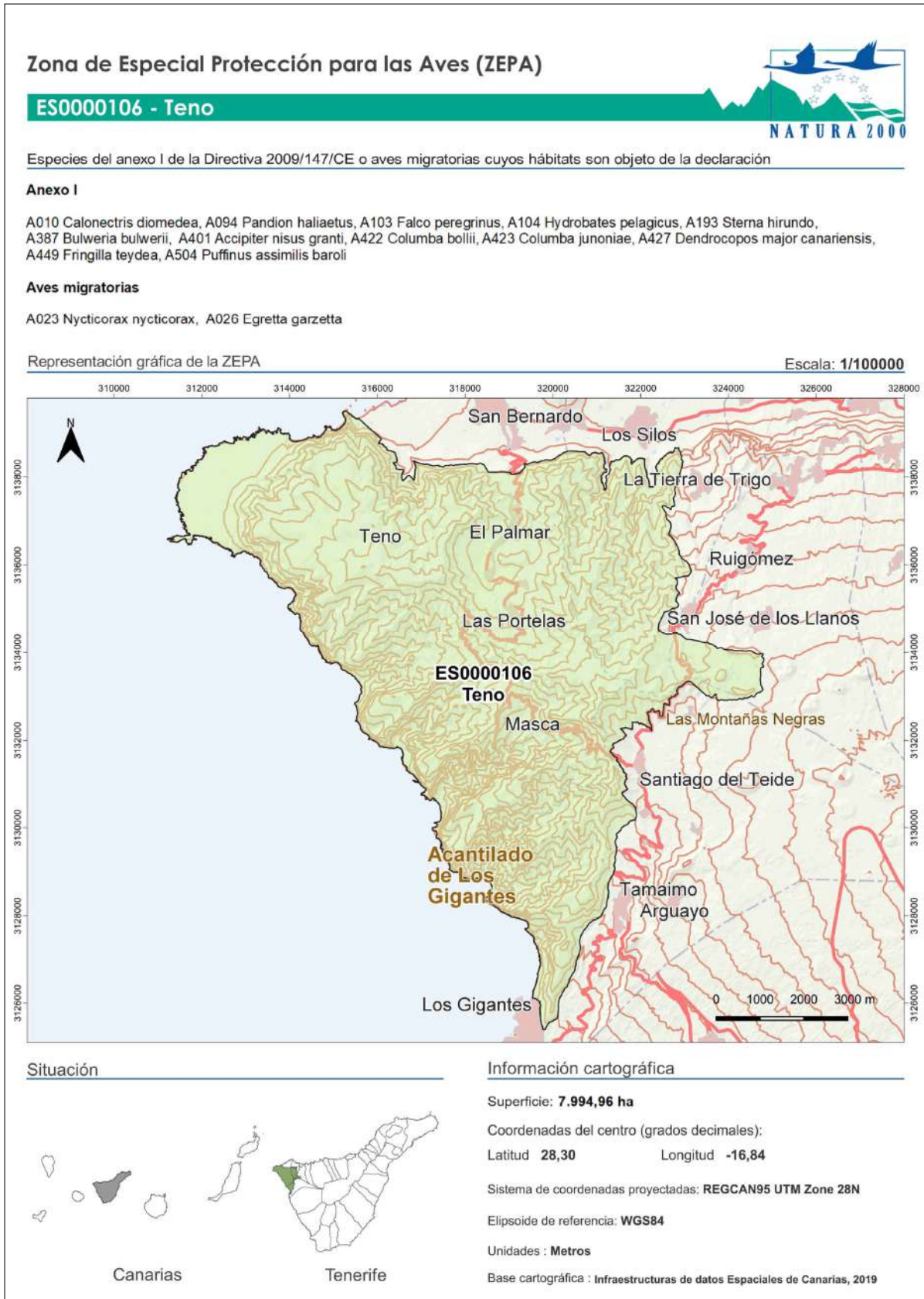
Unidades: **Metros**

Base cartográfica: **Infraestructuras de datos Espaciales de Canarias, 2019**



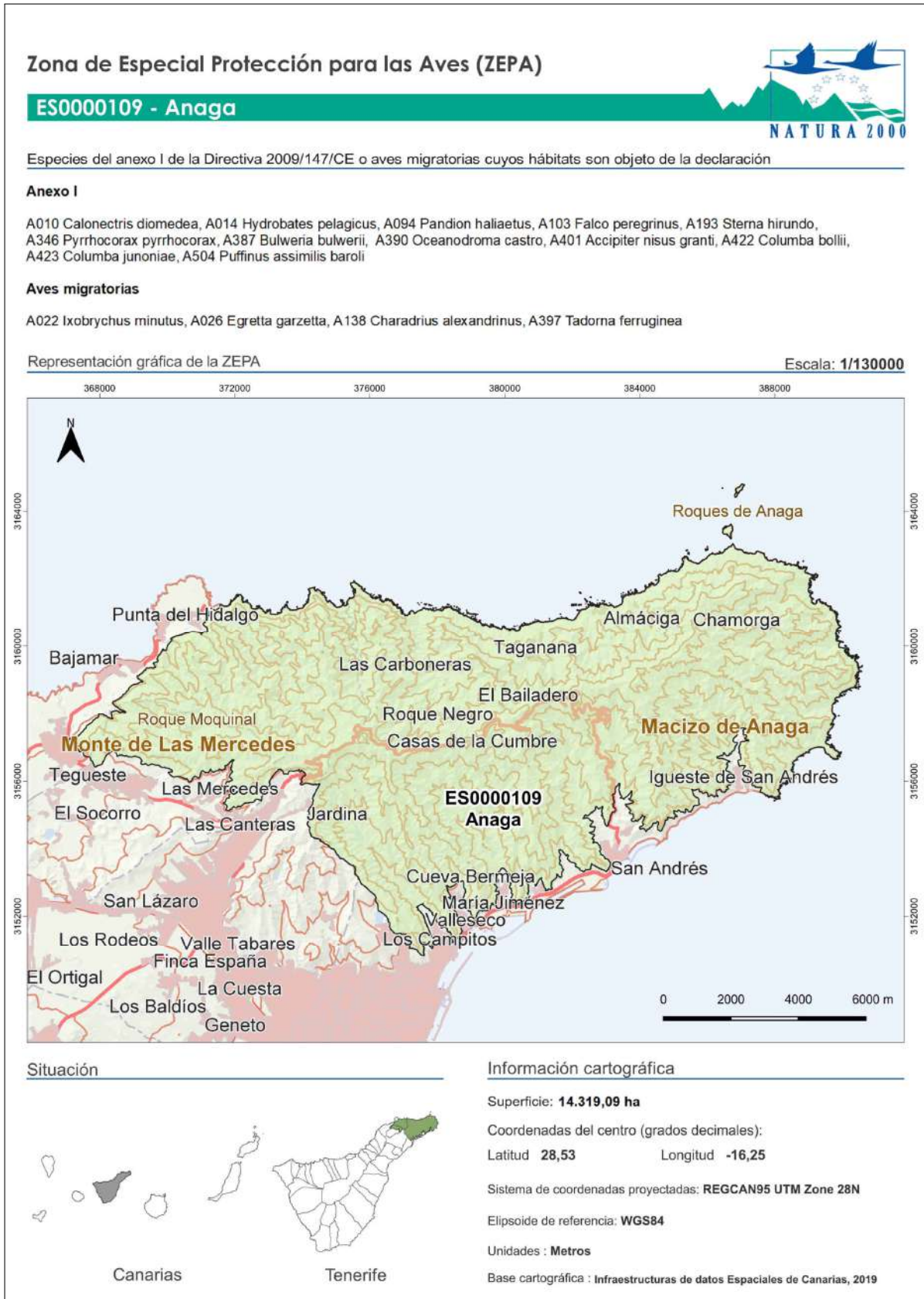
### F.1.4. Zonas de especial protección avifauna de Tenerife





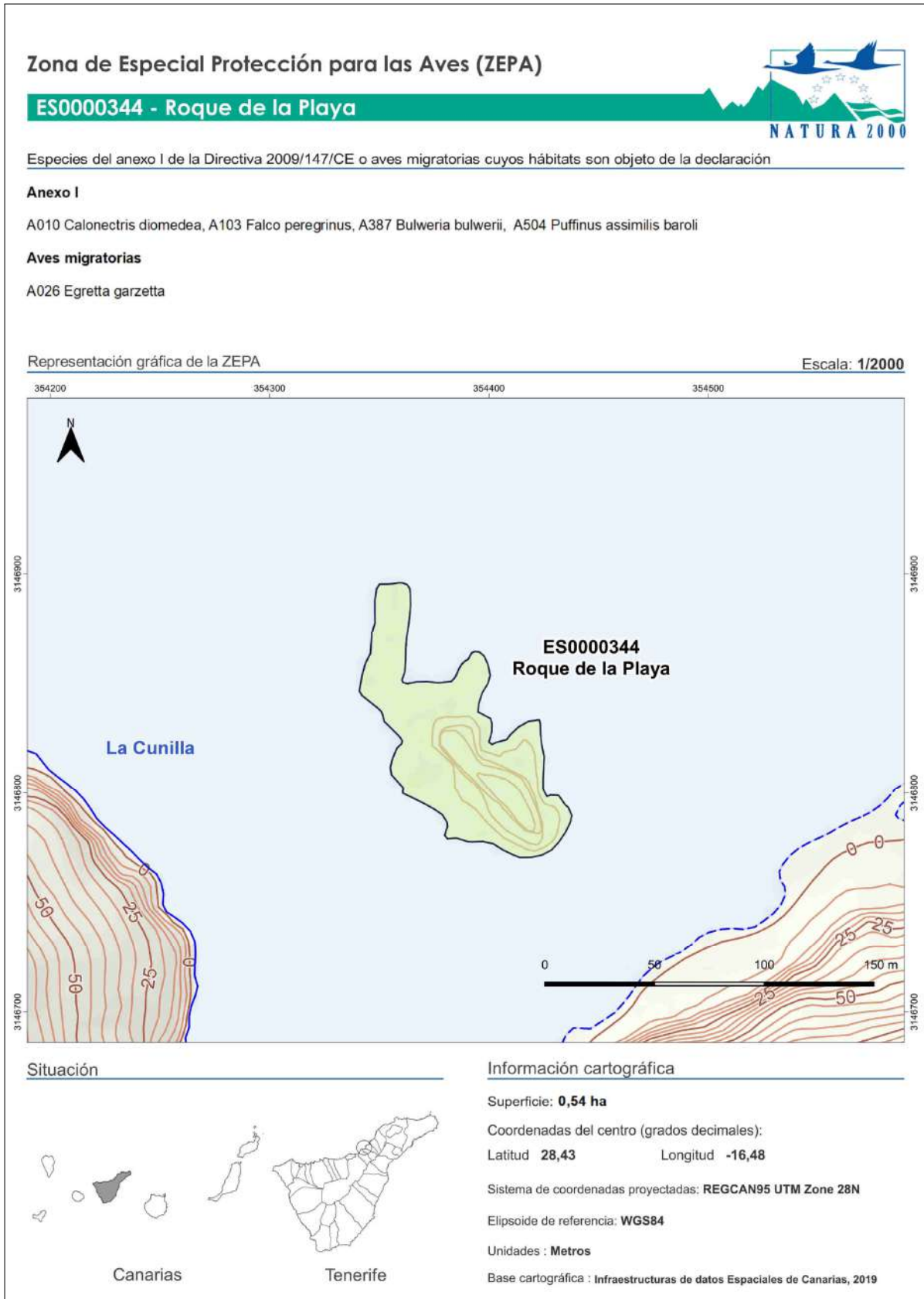


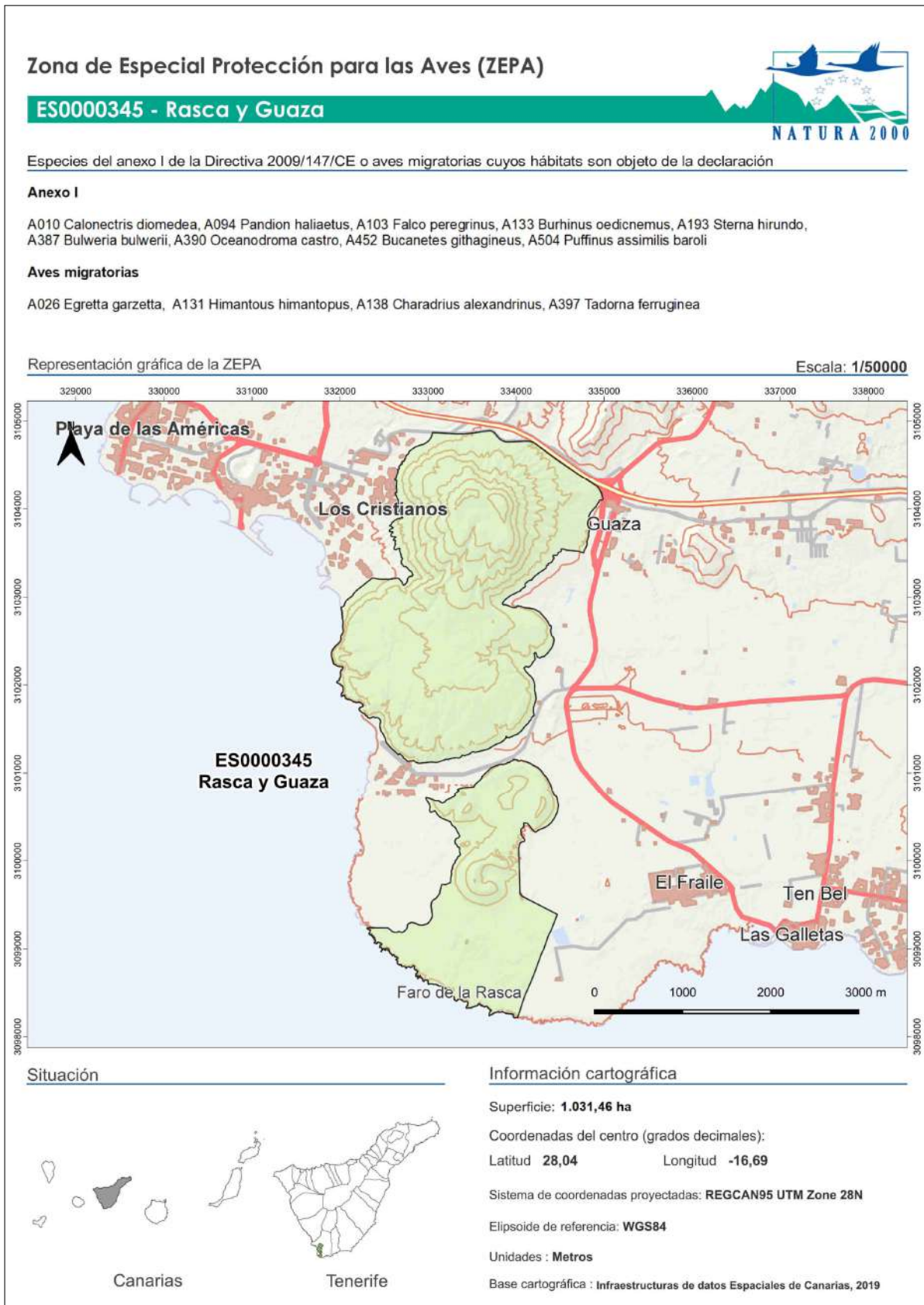




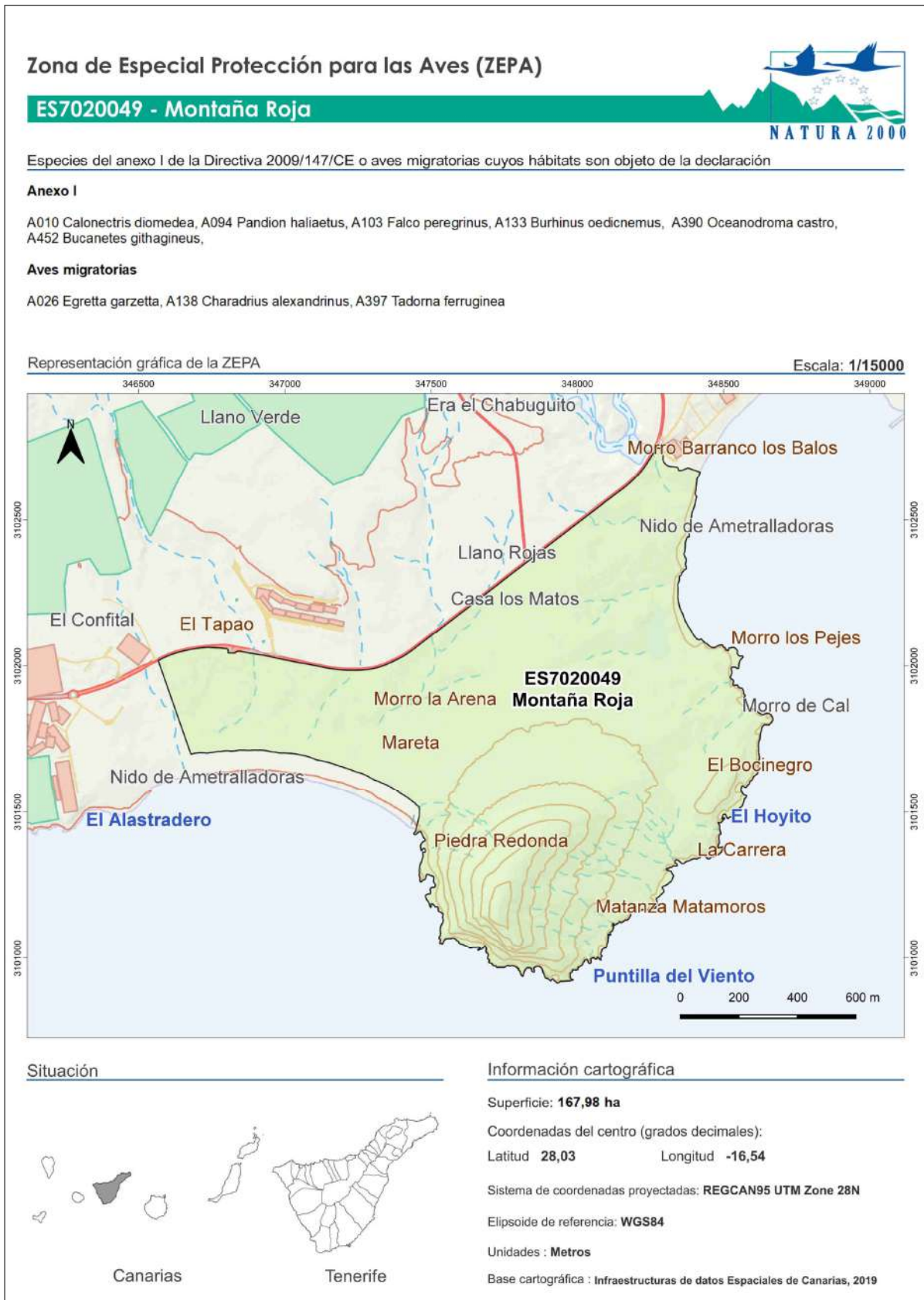


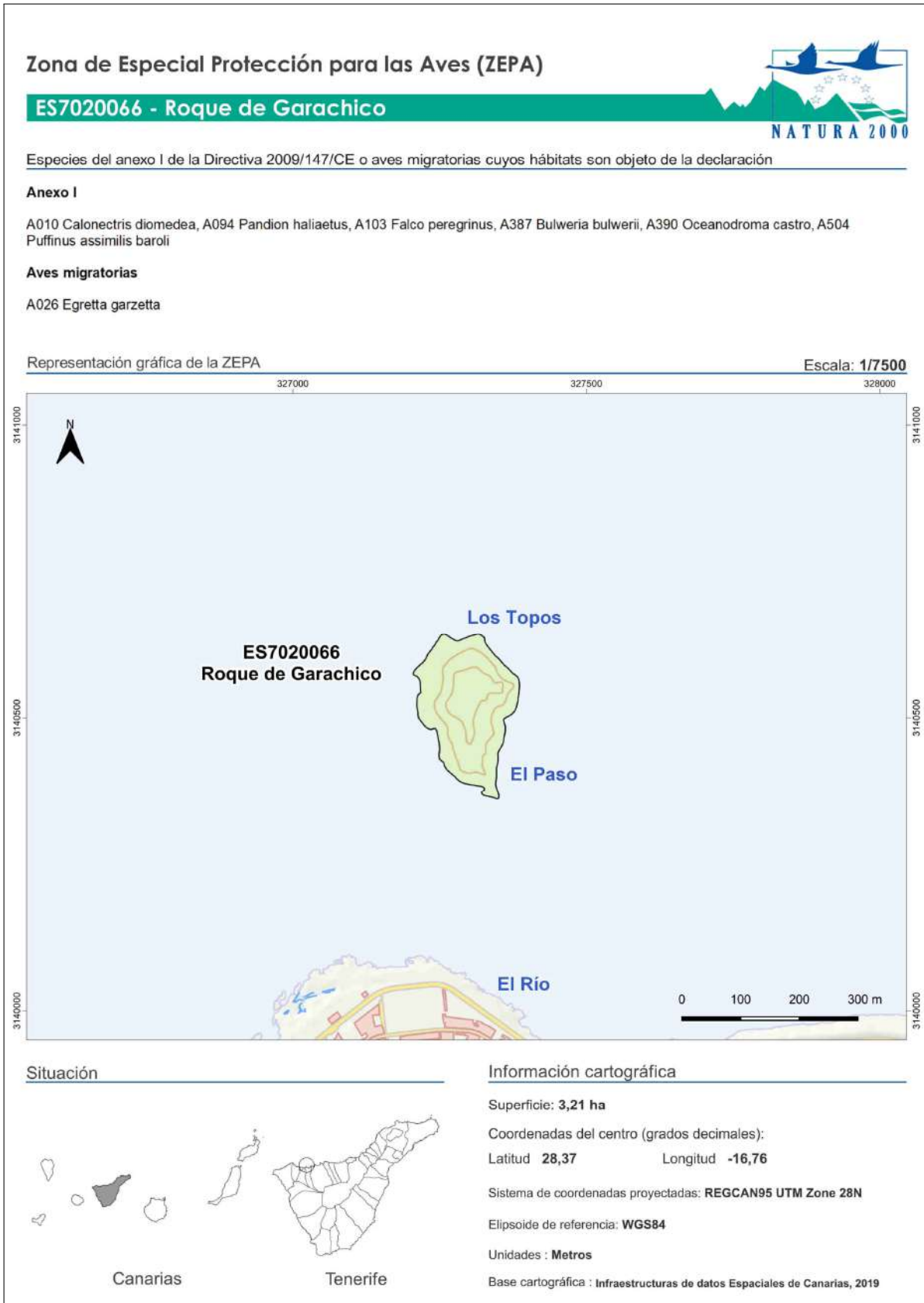










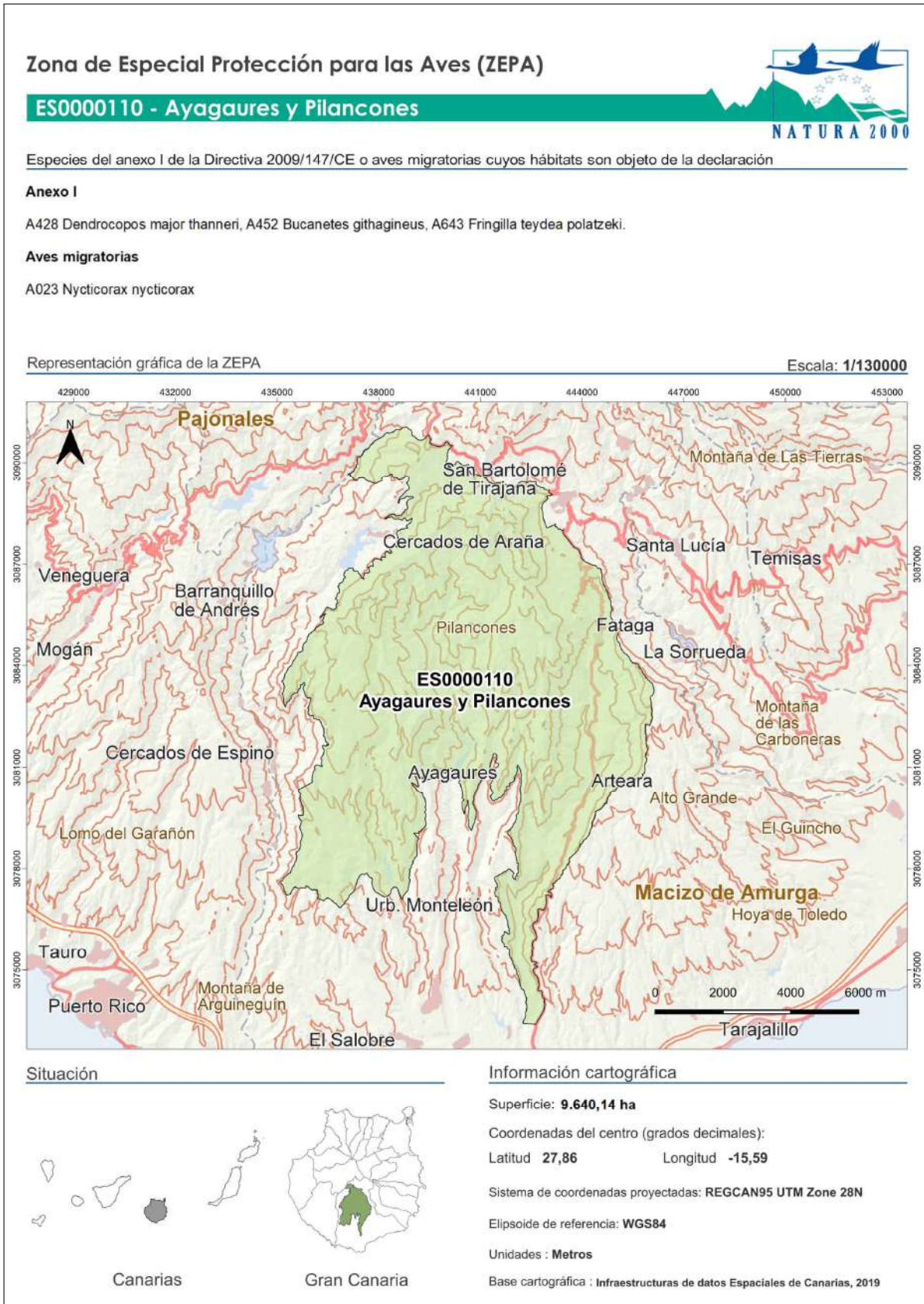




### F.1.5. Zonas de especial protección avifauna de Gran Canaria







## Zona de Especial Protección para las Aves (ZEPA)

**ES0000112 - Juncalillo del Sur**



Especies del anexo I de la Directiva 2009/147/CE o aves migratorias cuyos hábitats son objeto de la declaración

### Anexo I

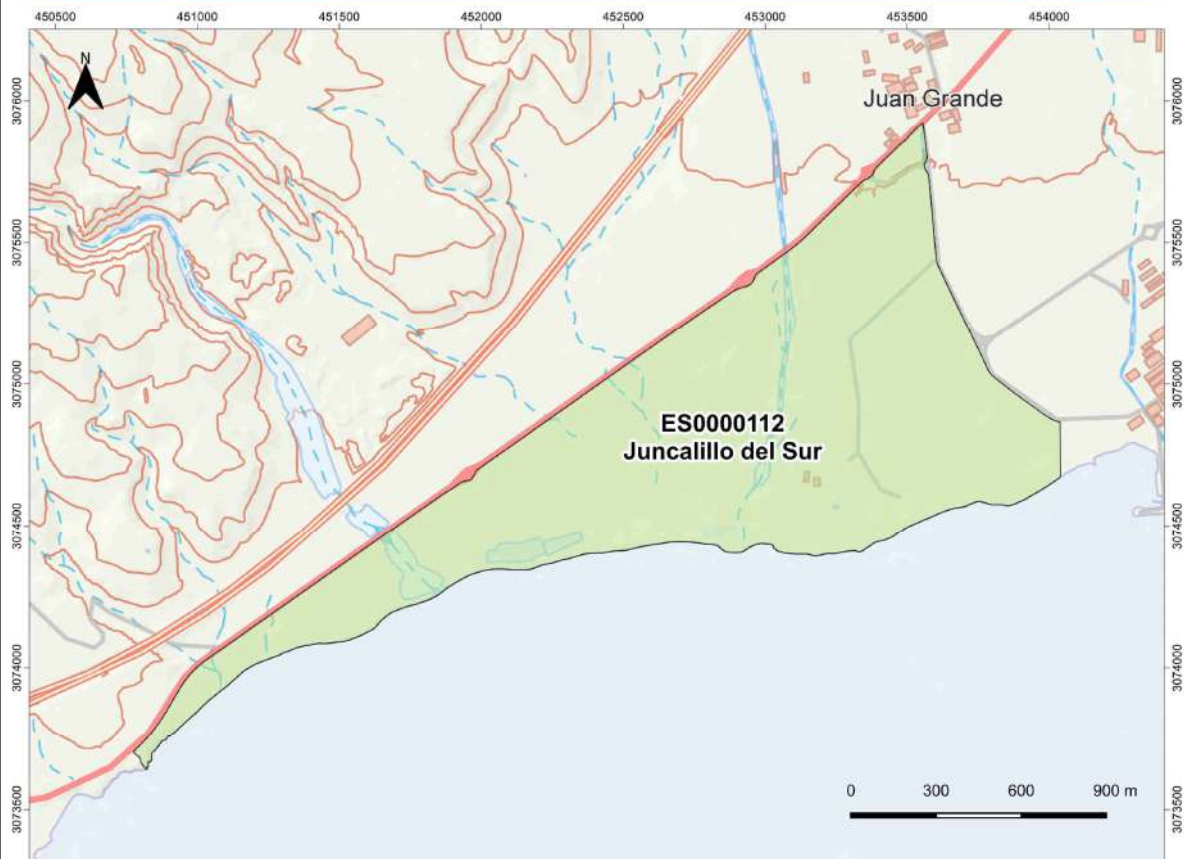
A010 Calonectris diomedea, A103 Falco peregrinus, A133 Burhinus oedicnemus, A452 Bucanetes githagineus

### Aves migratorias

A131 Himantopus himantopus, A138 Charadrius alexandrinus

Representación gráfica de la ZEPA

Escala: 1/20000



### Situación



Canarias

Gran Canaria

### Información cartográfica

Superficie: **187,22 ha**

Coordenadas del centro (grados decimales):

Latitud **27,79** Longitud **-15,47**

Sistema de coordenadas proyectadas: **REGCAN95 UTM Zone 28N**

Elipsoide de referencia: **WGS84**

Unidades: **Metros**

Base cartográfica: **Infraestructuras de datos Espaciales de Canarias, 2019**



Zona de Especial Protección para las Aves (ZEPA)

**ES0000113 - Macizo de Tauro**



Especies del anexo I de la Directiva 2009/147/CE o aves migratorias cuyos hábitats son objeto de la declaración

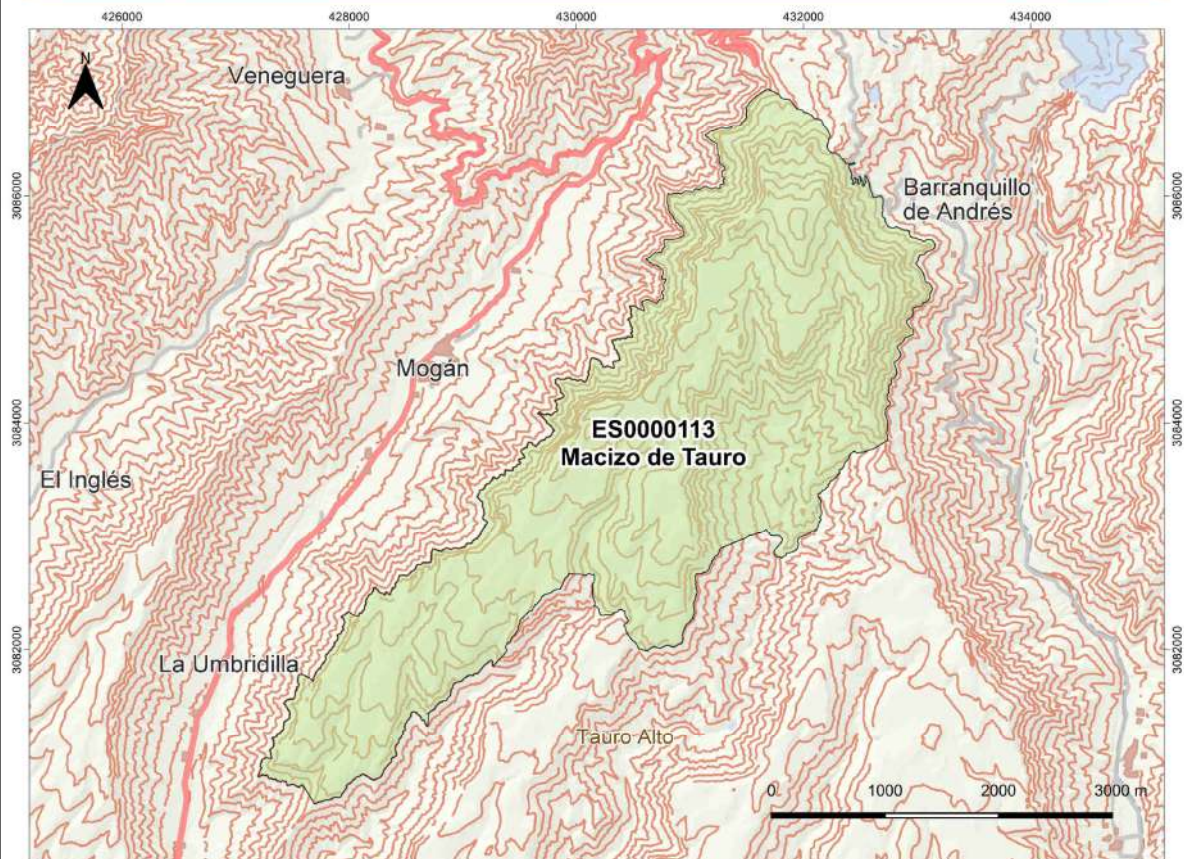
**Anexo I**

A428 *Dendrocopos major thanneri*, A452 *Bucanetes githagineus*, A643 *Fringilla teydea polatzeki*.

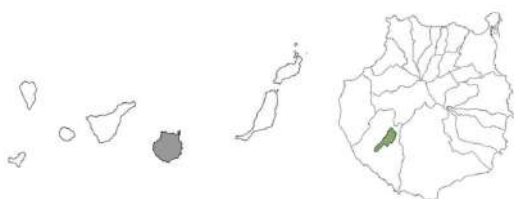
**Aves migratorias**

Representación gráfica de la ZEPA

Escala: **1/50000**



**Situación**



Canarias

Gran Canaria

**Información cartográfica**

Superficie: **1.248,82 ha**

Coordenadas del centro (grados decimales):

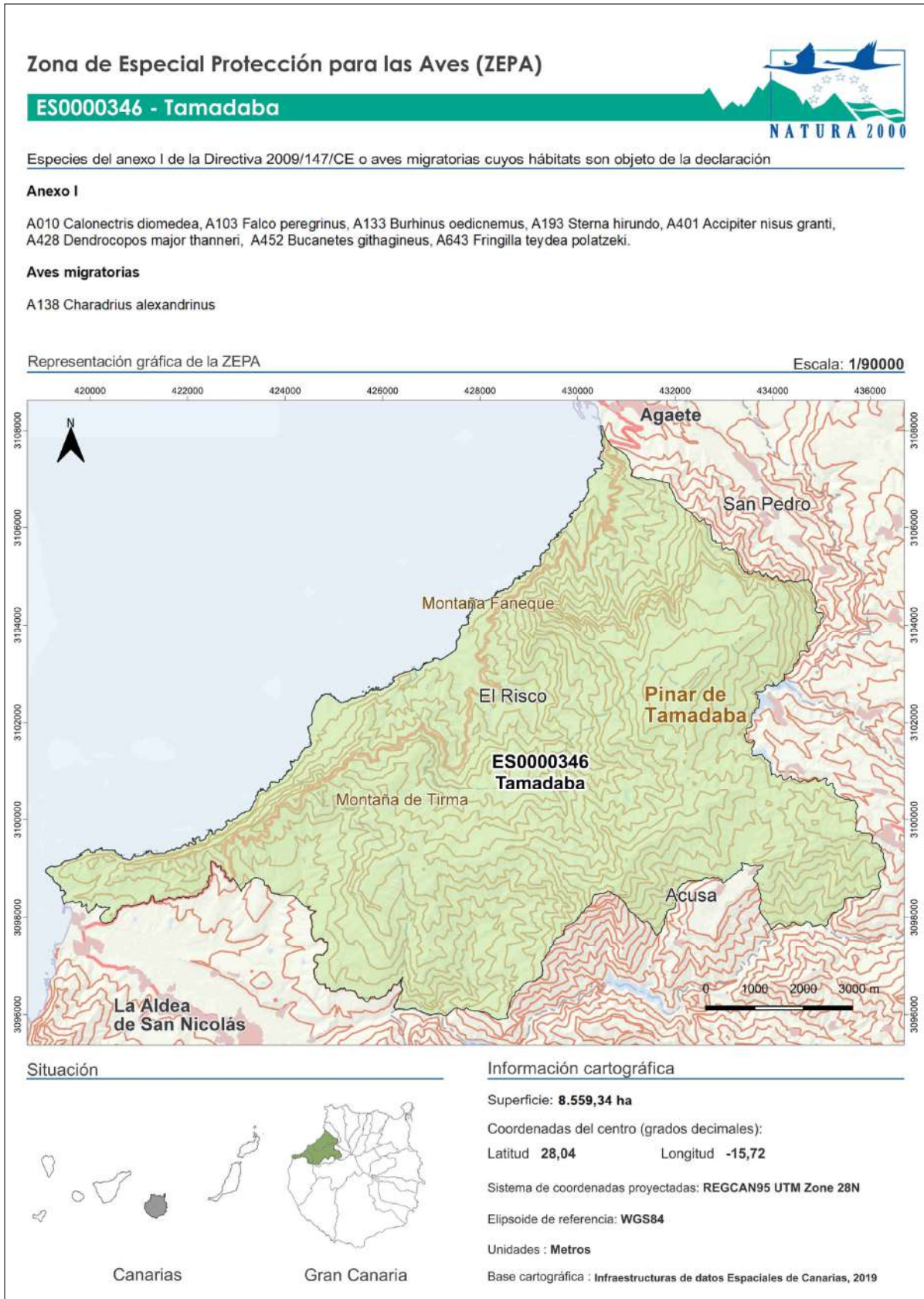
Latitud **27,87** Longitud **-15,69**

Sistema de coordenadas proyectadas: **REGCAN95 UTM Zone 28N**

Elipsoide de referencia: **WGS84**

Unidades : **Metros**

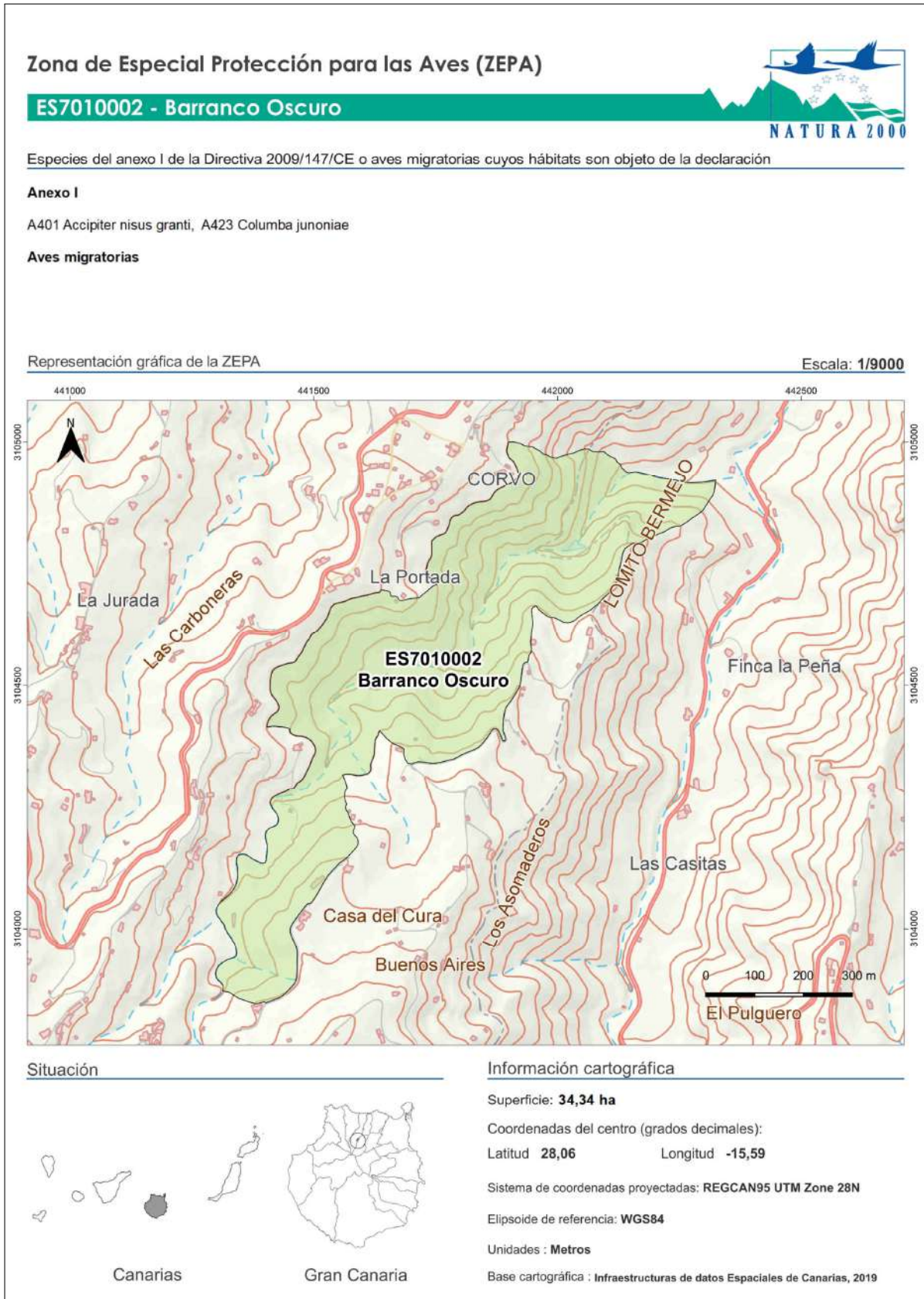
Base cartográfica : **Infraestructuras de datos Espaciales de Canarias, 2019**











Zona de Especial Protección para las Aves (ZEPA)

**ES7010004 - Azuaje**



Especies del anexo I de la Directiva 2009/147/CE o aves migratorias cuyos hábitats son objeto de la declaración

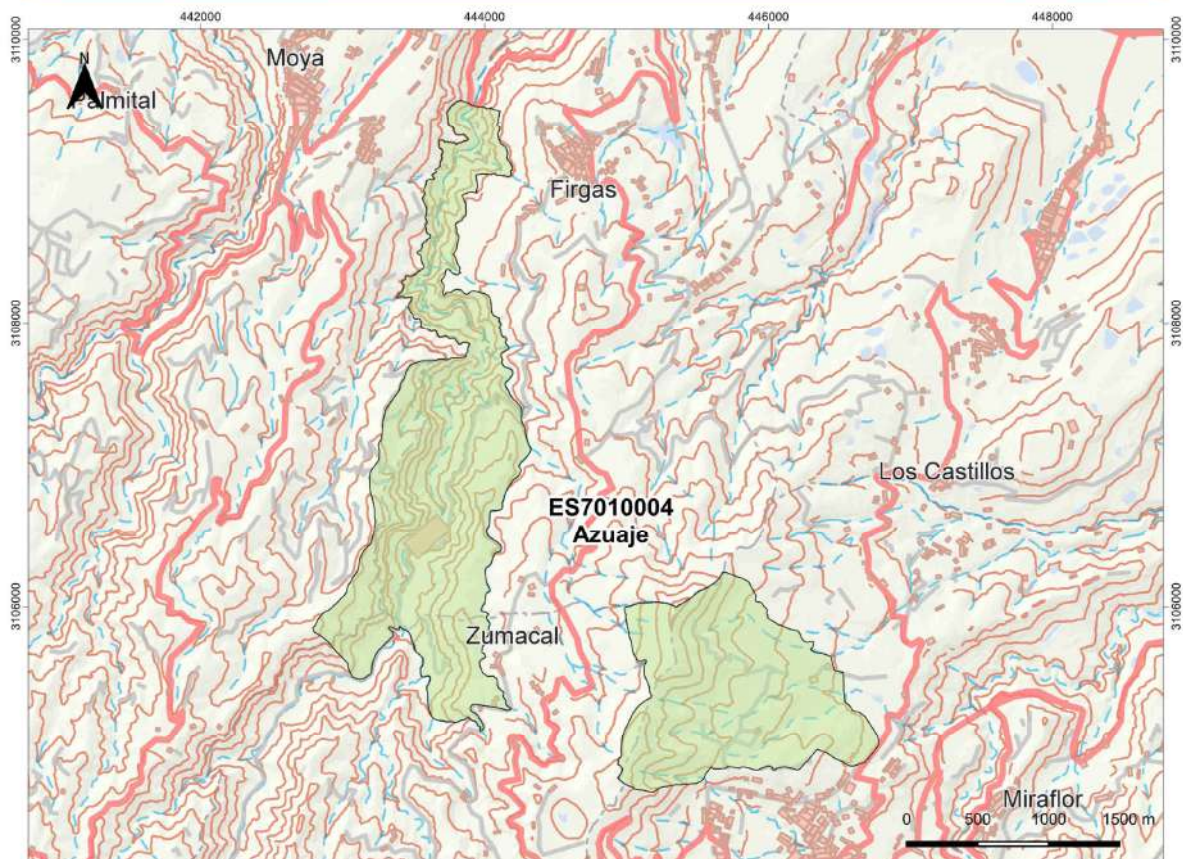
**Anexo I**

A133 *Burhinus oedicephalus*, A401 *Accipiter nisus granti*, A423 *Columba junoniae*

**Aves migratorias**

Representación gráfica de la ZEPA

Escala: 1/40000



**Situación**



Canarias

Gran Canaria

**Información cartográfica**

Superficie: **460,85 ha**

Coordenadas del centro (grados decimales):

Latitud **28,07** Longitud **-15,55**

Sistema de coordenadas proyectadas: **REGCAN95 UTM Zone 28N**

Elipsoide de referencia: **WGS84**

Unidades: **Metros**

Base cartográfica: **Infraestructuras de datos Espaciales de Canarias, 2019**



## Zona de Especial Protección para las Aves (ZEPA)

### ES7010038 - Barranco de la Virgen



Especies del anexo I de la Directiva 2009/147/CE o aves migratorias cuyos hábitats son objeto de la declaración

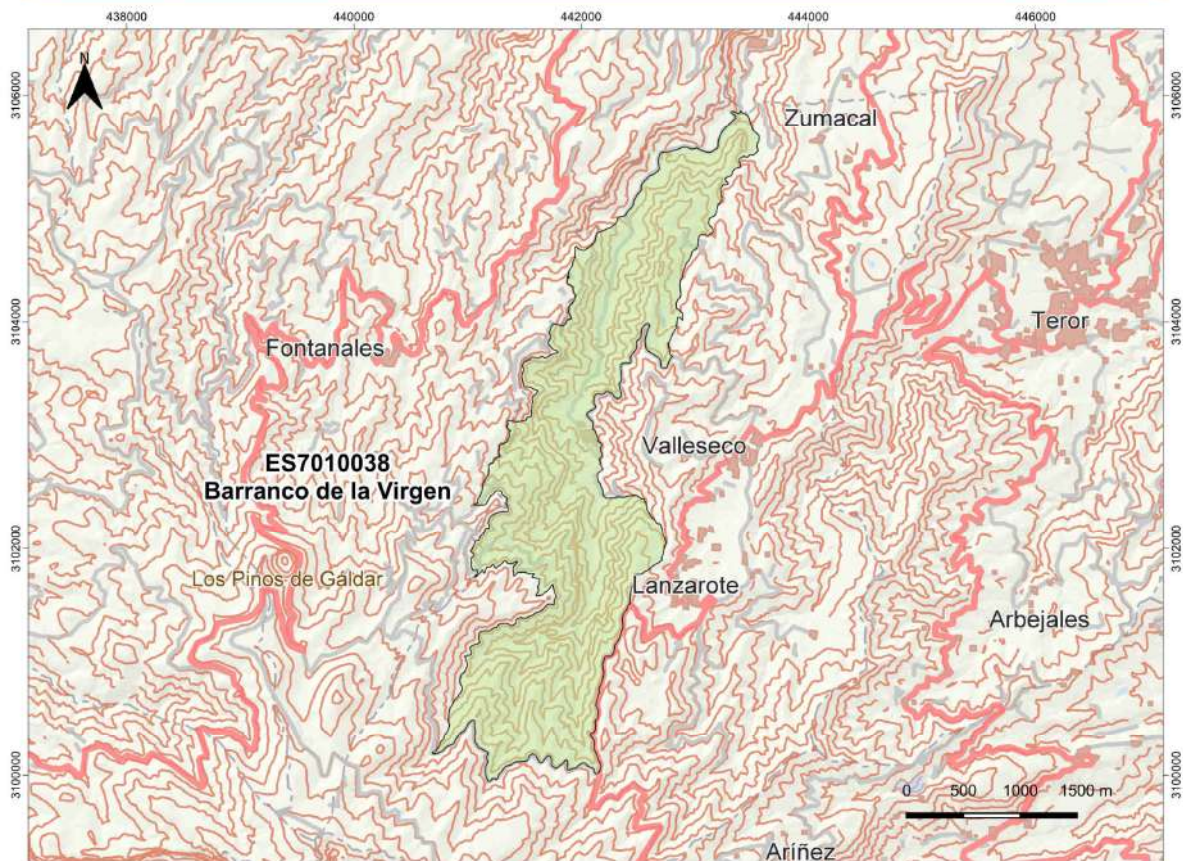
#### Anexo I

A401 *Accipiter nisus granti*, A423 *Columba junoniae*

#### Aves migratorias

Representación gráfica de la ZEPA

Escala: 1/50000



#### Situación



Canarias

Gran Canaria

#### Información cartográfica

Superficie: **558,66 ha**

Coordenadas del centro (grados decimales):

Latitud **28,05** Longitud **-15,59**

Sistema de coordenadas proyectadas: **REGCAN95 UTM Zone 28N**

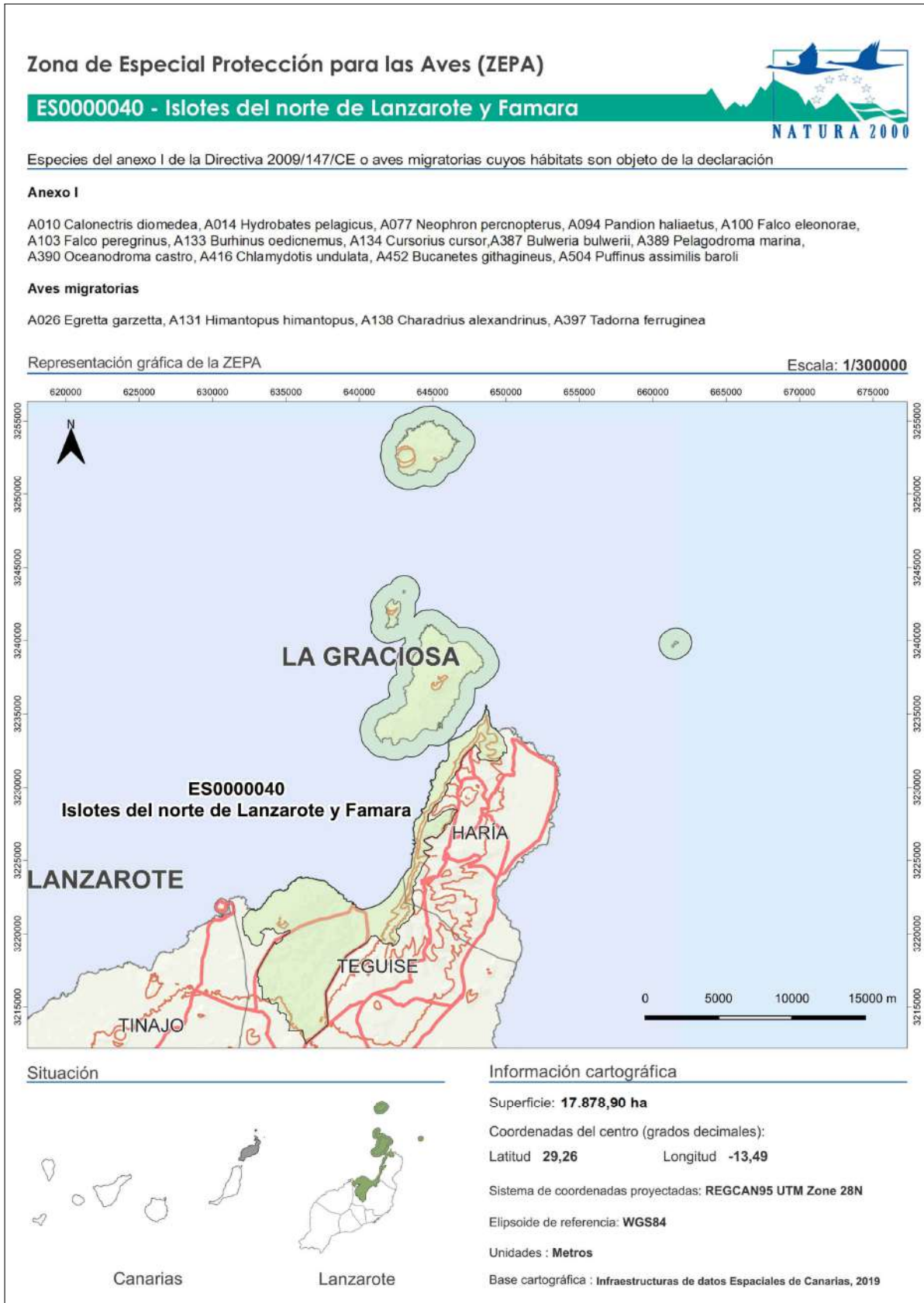
Elipsoide de referencia: **WGS84**

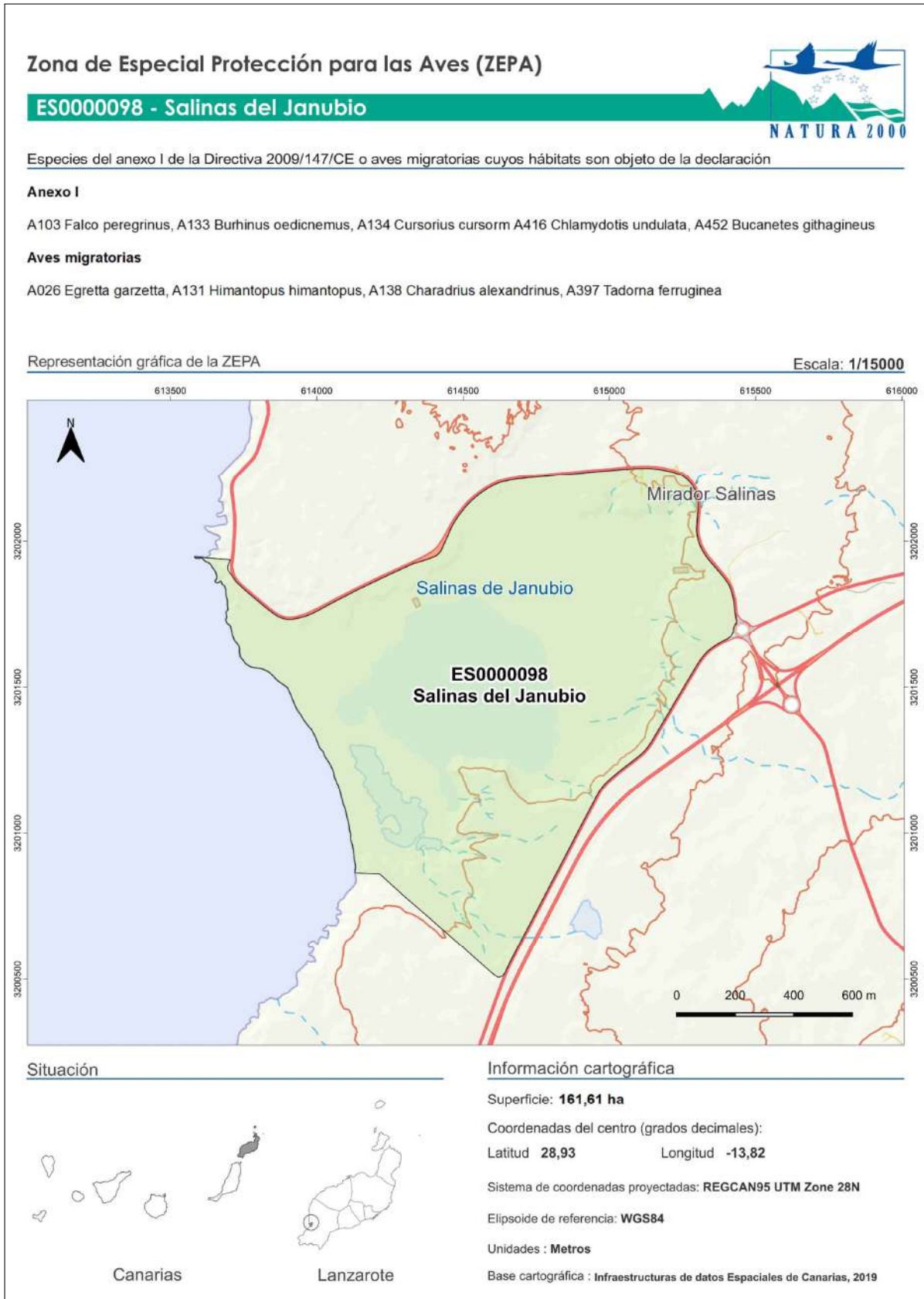
Unidades: **Metros**

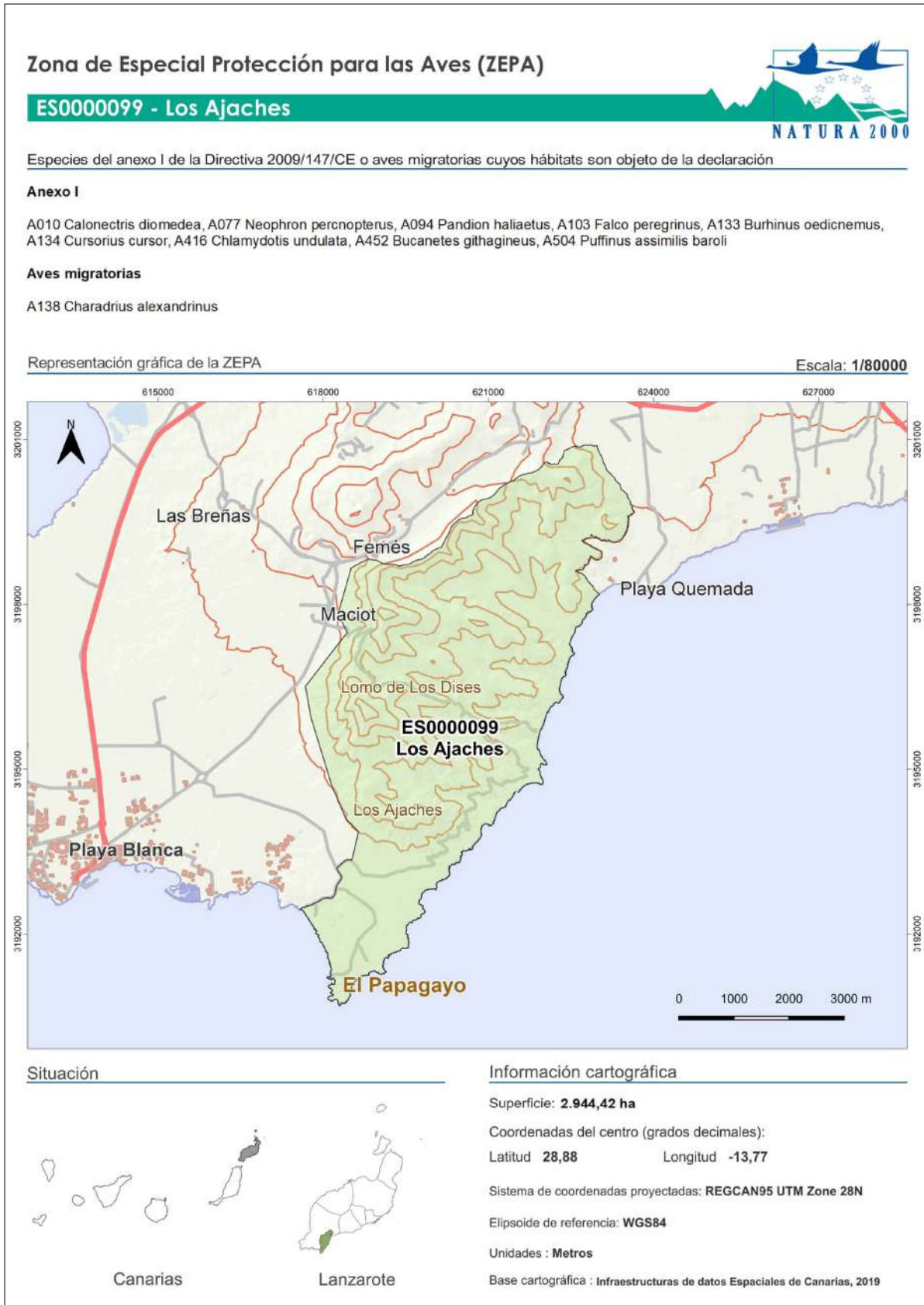
Base cartográfica: **Infraestructuras de datos Espaciales de Canarias, 2019**



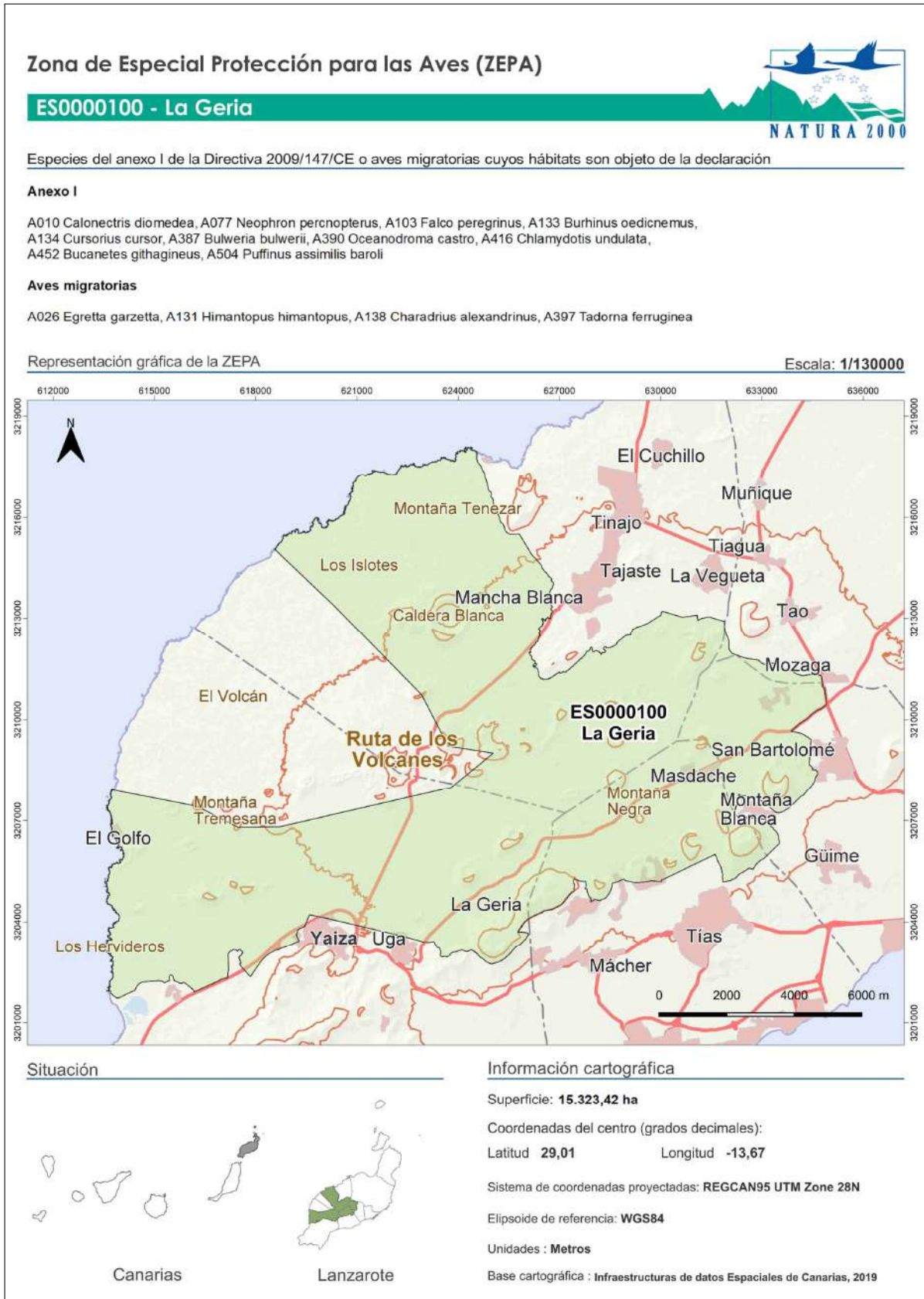
**F.1.6. Zonas de especial protección avifauna de Lanzarote**

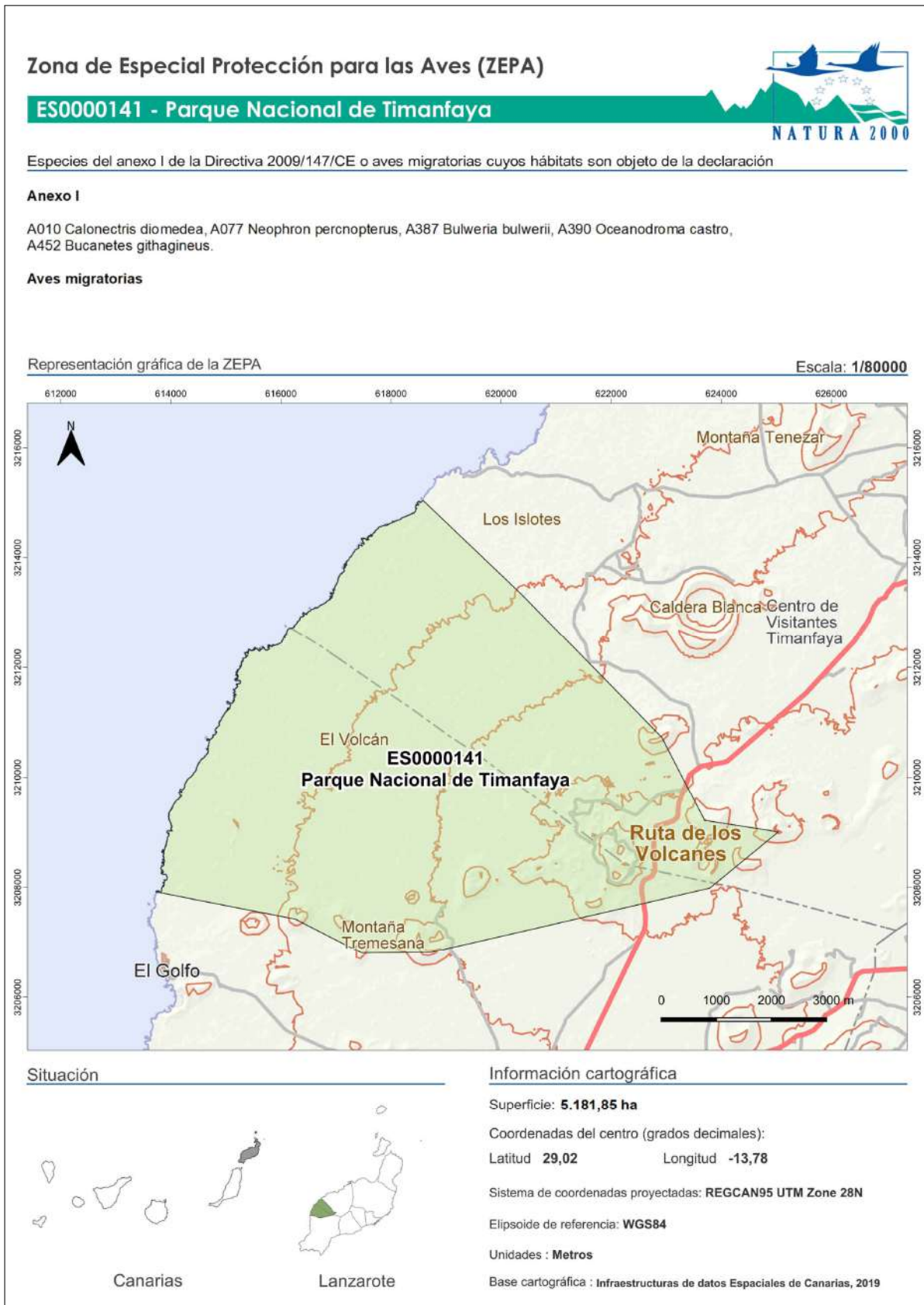












## Zona de Especial Protección para las Aves (ZEPA)

### ES0000350 - Llanos de la Corona y Tegala Grande



Especies del anexo I de la Directiva 2009/147/CE o aves migratorias cuyos hábitats son objeto de la declaración

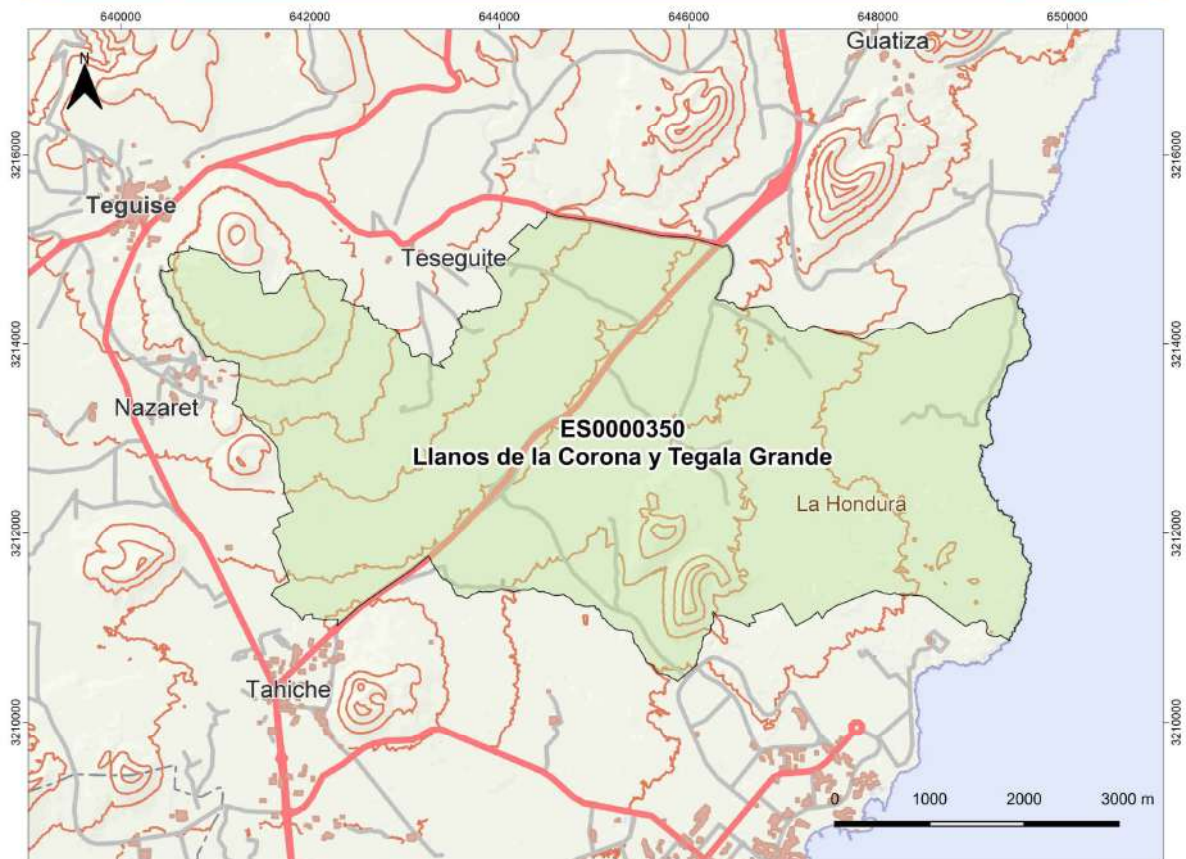
#### Anexo I

A010 Calonectris diomedea, A077 Neophron percnopterus, A103 Falco peregrinus, A133 Burhinus oedicnemus, A134 Cursorius cursor, A387 Bulweria bulwerii, A416 Chlamydotis undulata, A452 Bucanetes githagineus

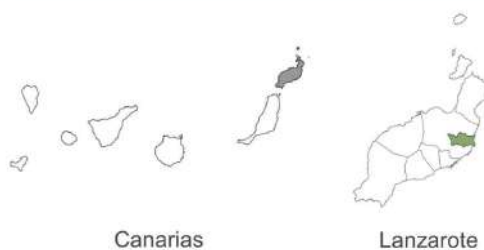
#### Aves migratorias

Representación gráfica de la ZEPA

Escala: 1/60000



#### Situación



#### Información cartográfica

Superficie: **2.747,94 ha**

Coordenadas del centro (grados decimales):

Latitud **29,03**      Longitud **-13,50**

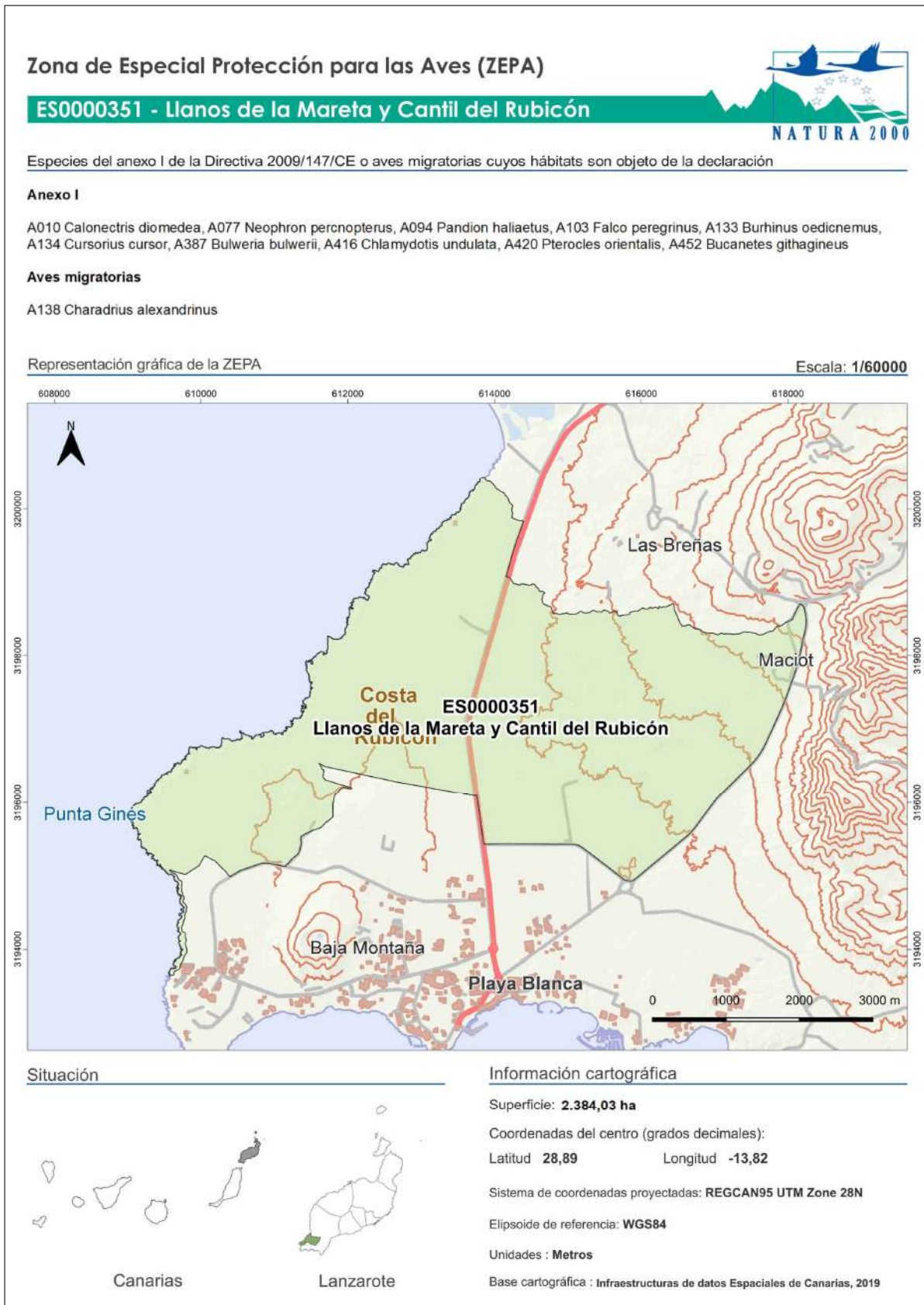
Sistema de coordenadas proyectadas: **REGCAN95 UTM Zone 28N**

Eliipsoide de referencia: **WGS84**

Unidades : **Metros**

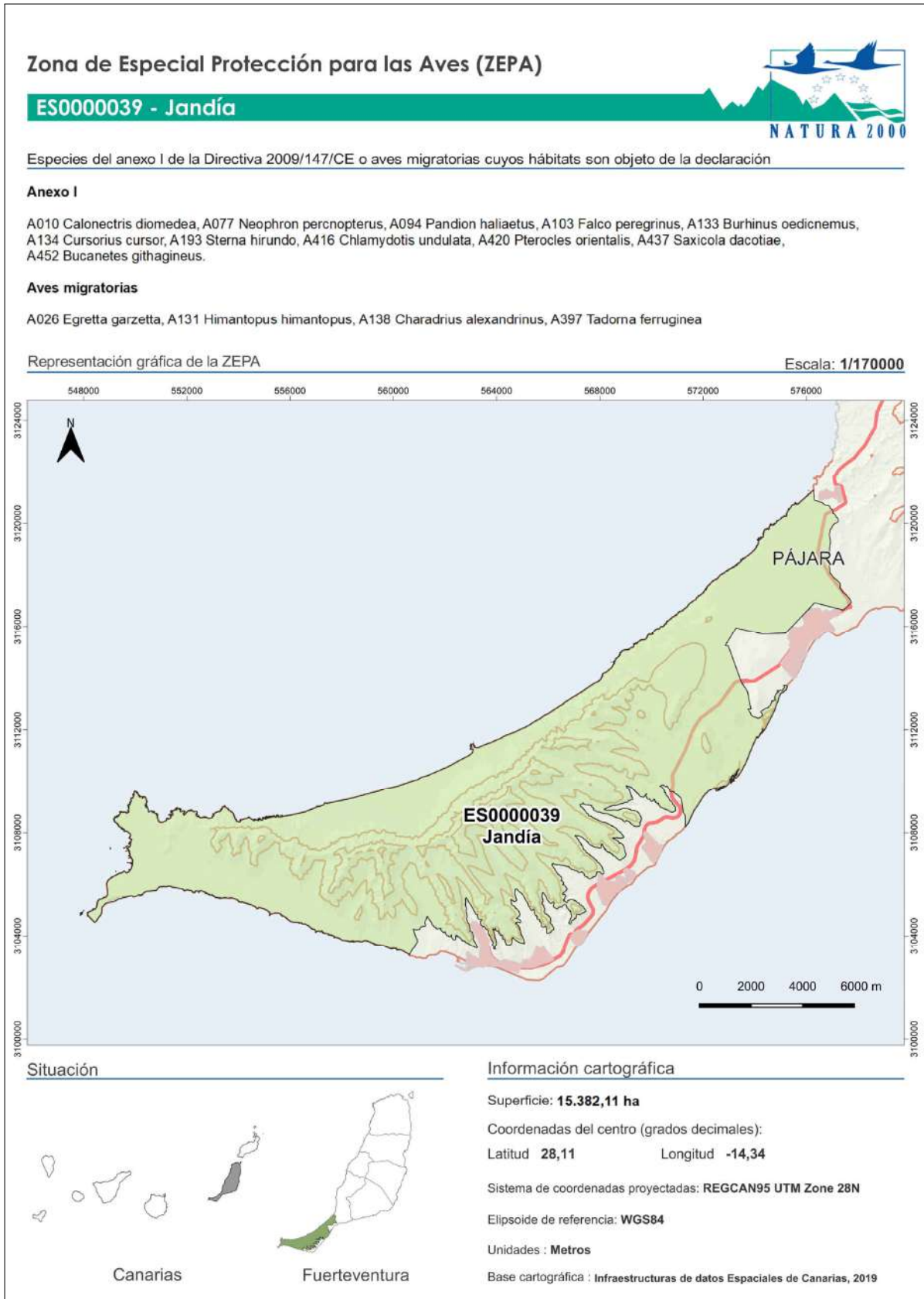
Base cartográfica : **Infraestructuras de datos Espaciales de Canarias, 2019**

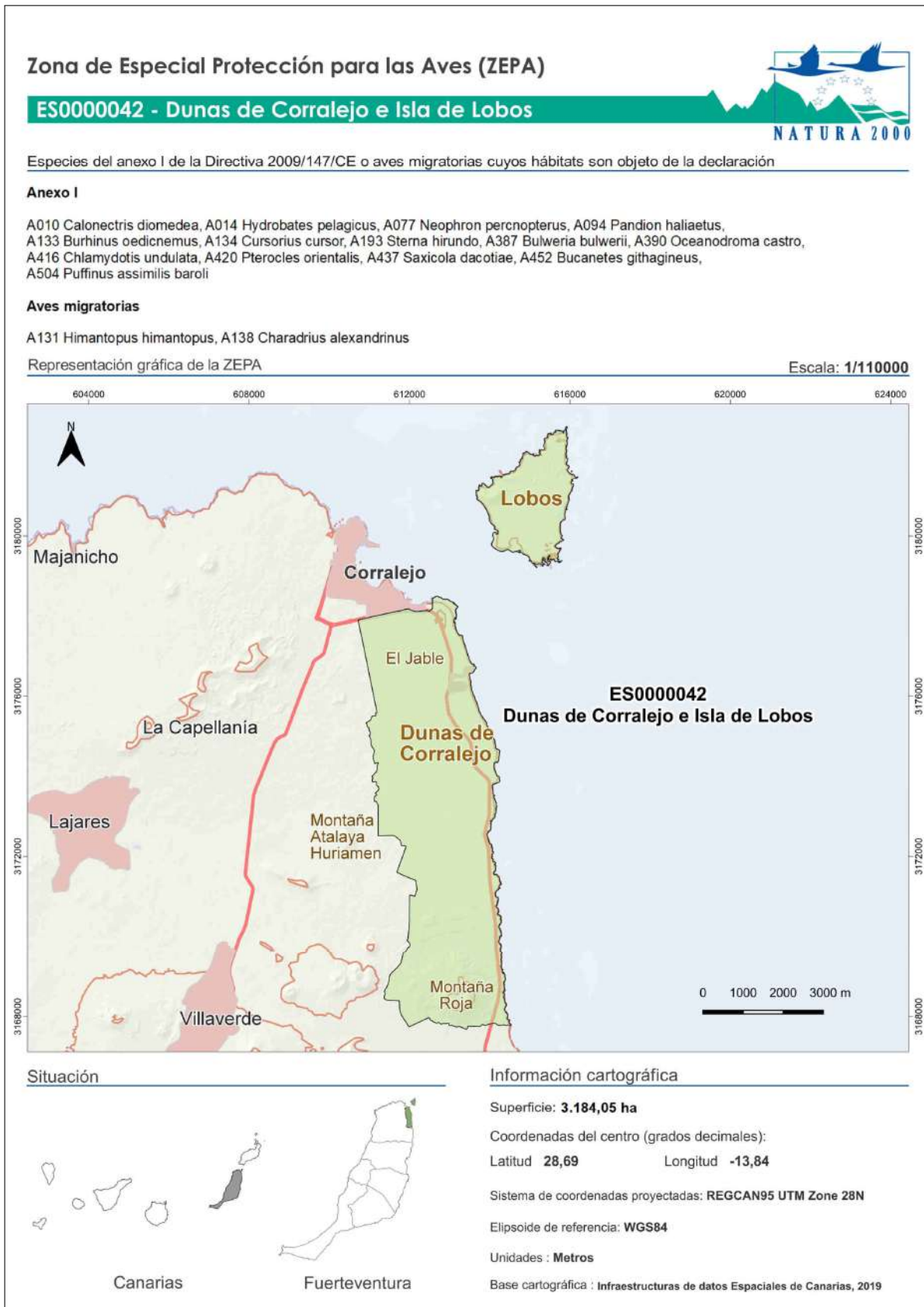






### F.1.7. Zonas de especial protección avifauna de Fuerteventura





## Zona de Especial Protección para las Aves (ZEPA)

### ES0000096 - Pozo Negro



Especies del anexo I de la Directiva 2009/147/CE o aves migratorias cuyos hábitats son objeto de la declaración

#### Anexo I

A010 *Calonectris diomedea*, A077 *Neophron percnopterus*, A094 *Pandion haliaetus*, A103 *Falco peregrinus*, A133 *Burhinus oedipnemus*, A134 *Cursorius cursor*, A193 *Sterna hirundo*, A420 *Pterocles orientalis*, A431 *Calandrella rufescens*, A437 *Saxicola dacotiae*, A452 *Bucanetes githagineus*.

#### Aves migratorias

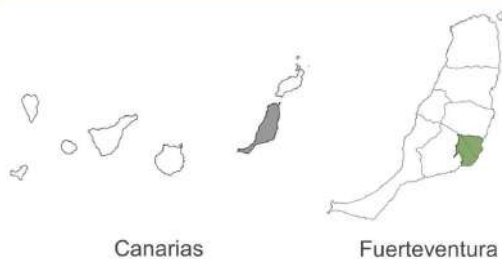
A397 *Tadorna ferruginea*

Representación gráfica de la ZEPA

Escala: 1/100000



#### Situación



#### Información cartográfica

Superficie: **9.150,85 ha**

Coordenadas del centro (grados decimales):

Latitud **28,28** Longitud **-13,95**

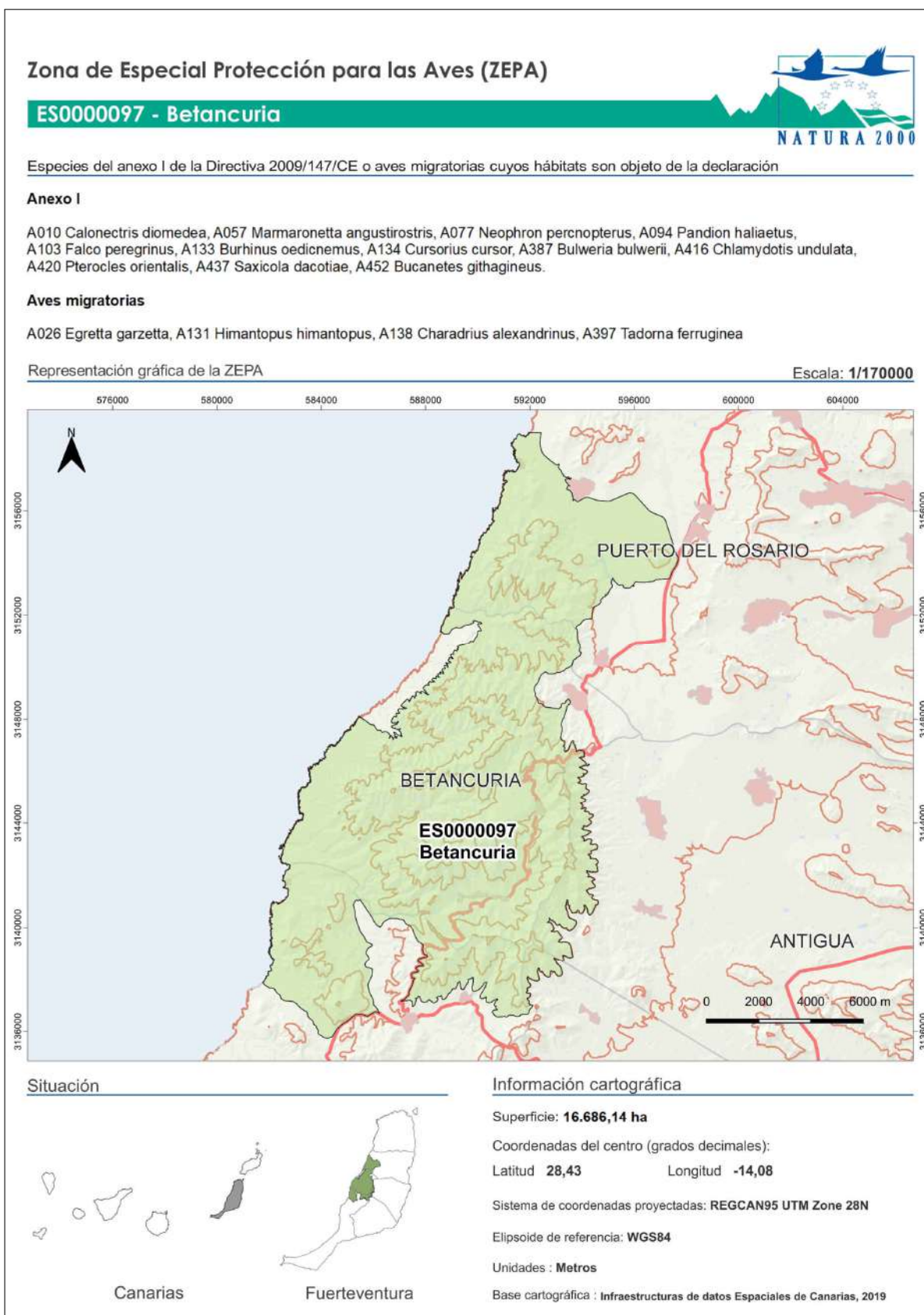
Sistema de coordenadas proyectadas: **REGCAN95 UTM Zone 28N**

Elipsoide de referencia: **WGS84**

Unidades: **Metros**

Base cartográfica: **Infraestructuras de datos Espaciales de Canarias, 2019**

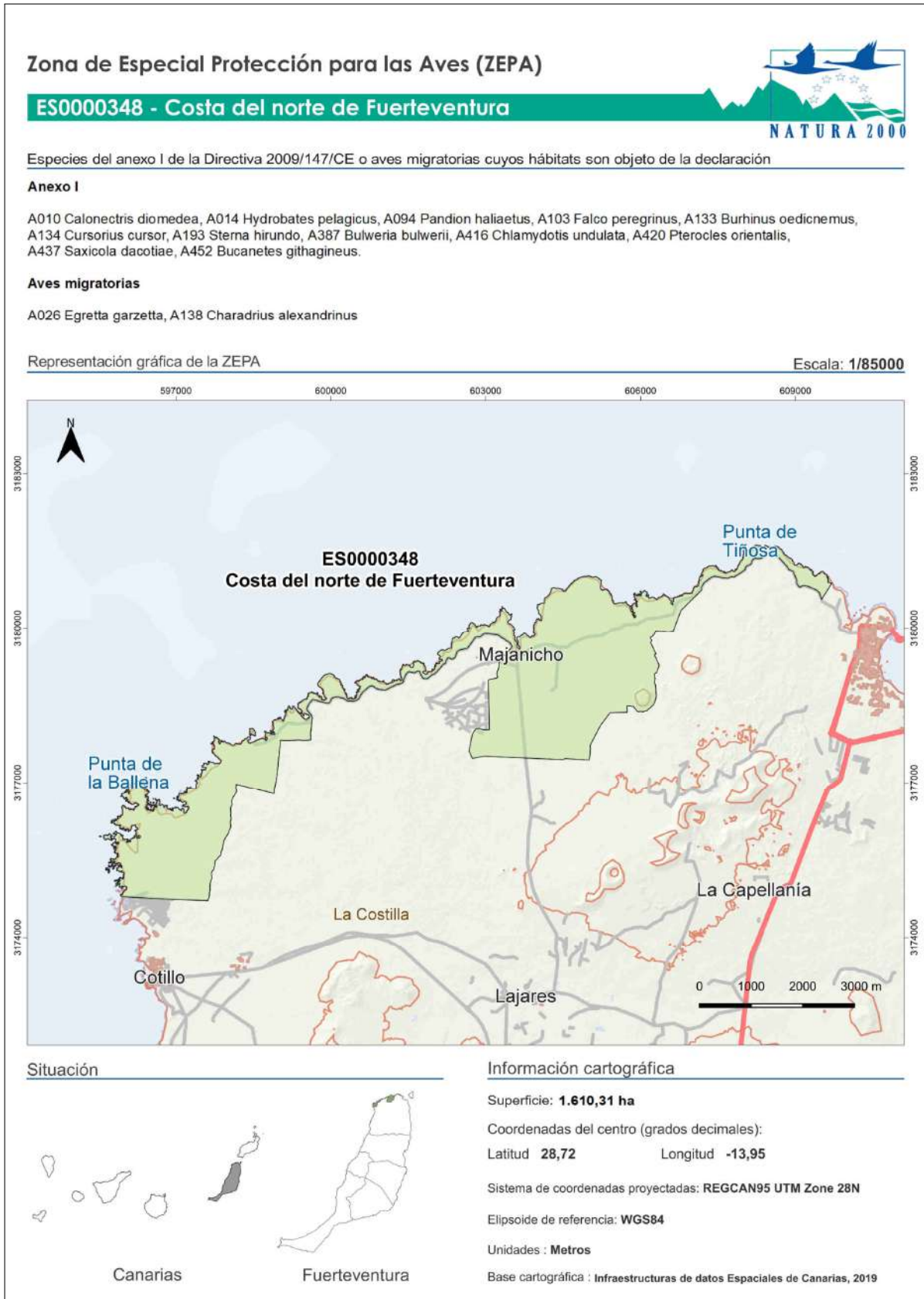






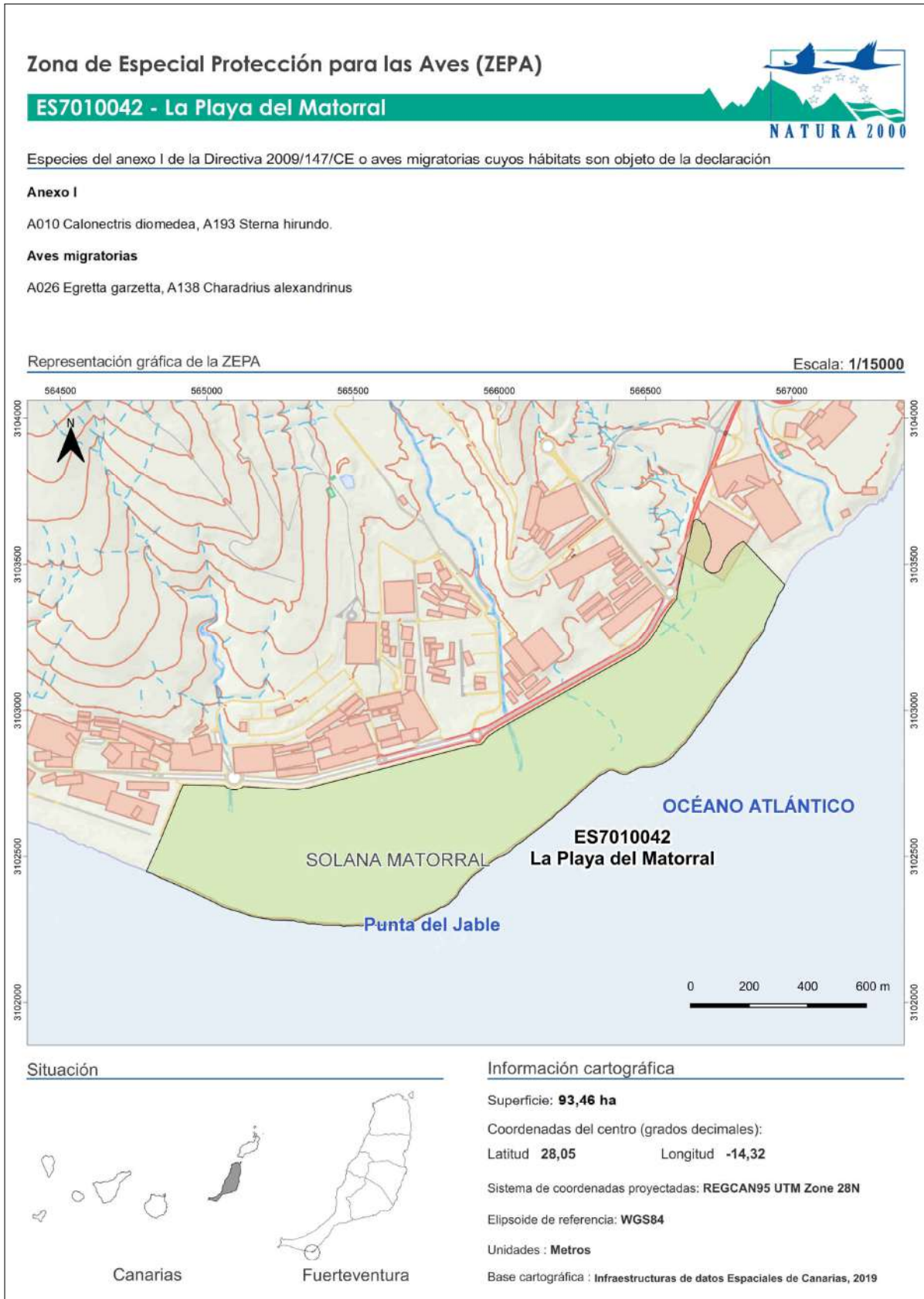








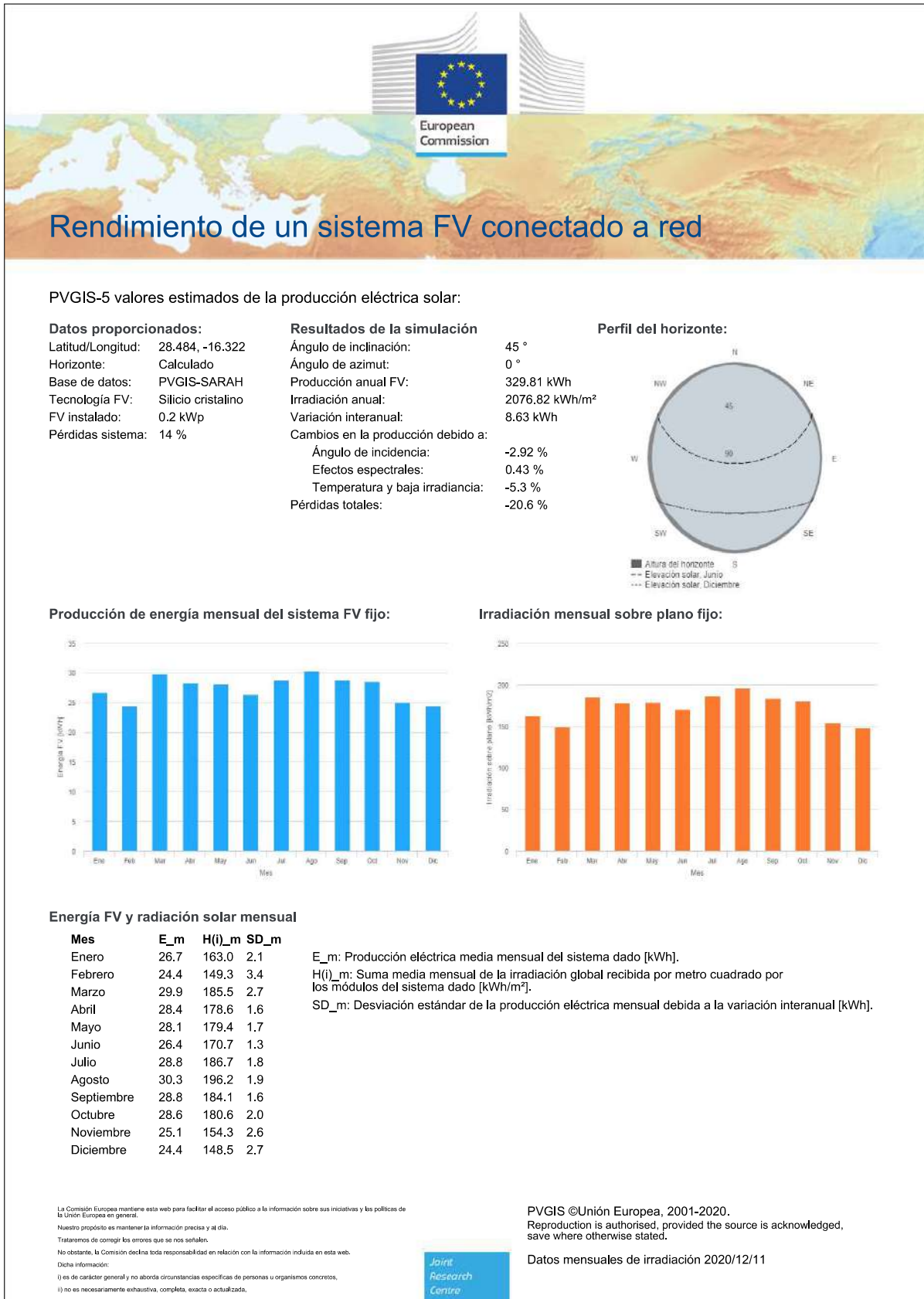




## **F.2. Datos de insolación**

En esta sección del anexo se adjuntan las diferentes hojas del rendimiento fotovoltaico referente a las ubicaciones propuestas en el desarrollo de este proyecto.

## F.2.1. Resultados de insolación a 45° en La laguna





## F.2.2. Resultados de insolación a 45<sup>0</sup> en Las Palmas de Gran Canaria



PVGIS-5 valores estimados de la producción eléctrica solar:

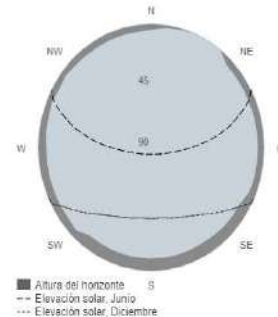
**Datos proporcionados:**

Latitud/Longitud: 28.004, -15.419  
 Horizonte: Calculado  
 Base de datos: PVGIS-SARAH  
 Tecnología FV: Silicio cristalino  
 FV instalado: 0.2 kWp  
 Pérdidas sistema: 14 %

**Resultados de la simulación**

Ángulo de inclinación: 45 °  
 Ángulo de azimut: 0 °  
 Producción anual FV: 323.97 kWh  
 Irradiación anual: 2055.82 kWh/m<sup>2</sup>  
 Variación interanual: 7.59 kWh  
 Cambios en la producción debido a:  
 Ángulo de incidencia: -2.82 %  
 Efectos espectrales: 0.37 %  
 Temperatura y baja irradiancia: -6.07 %  
 Pérdidas totales: -21.21 %

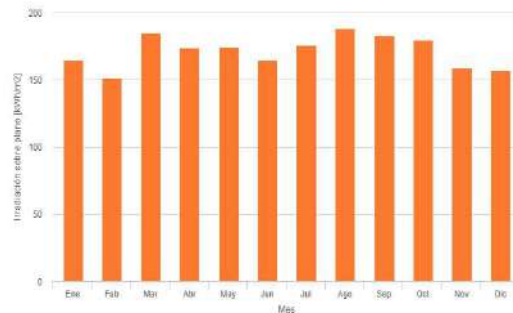
**Perfil del horizonte:**



**Producción de energía mensual del sistema FV fijo:**



**Irradiación mensual sobre plano fijo:**



**Energía FV y radiación solar mensual**

Mes	E_m	H(i)_m	SD_m
Enero	26.6	164.4	2.4
Febrero	24.4	151.3	2.7
Marzo	29.5	185.0	2.2
Abril	27.4	173.6	1.5
Mayo	27.1	174.3	1.7
Junio	25.3	164.4	1.3
Julio	27.1	175.7	1.6
Agosto	29.1	188.3	1.7
Septiembre	28.5	183.2	1.9
Octubre	28.2	179.6	2.2
Noviembre	25.5	159.1	2.3
Diciembre	25.4	157.0	2.6

E\_m: Producción eléctrica media mensual del sistema dado [kWh].  
 H(i)\_m: Suma media mensual de la irradiación global recibida por metro cuadrado por los módulos del sistema dado [kWh/m<sup>2</sup>].  
 SD\_m: Desviación estándar de la producción eléctrica mensual debida a la variación interanual [kWh].

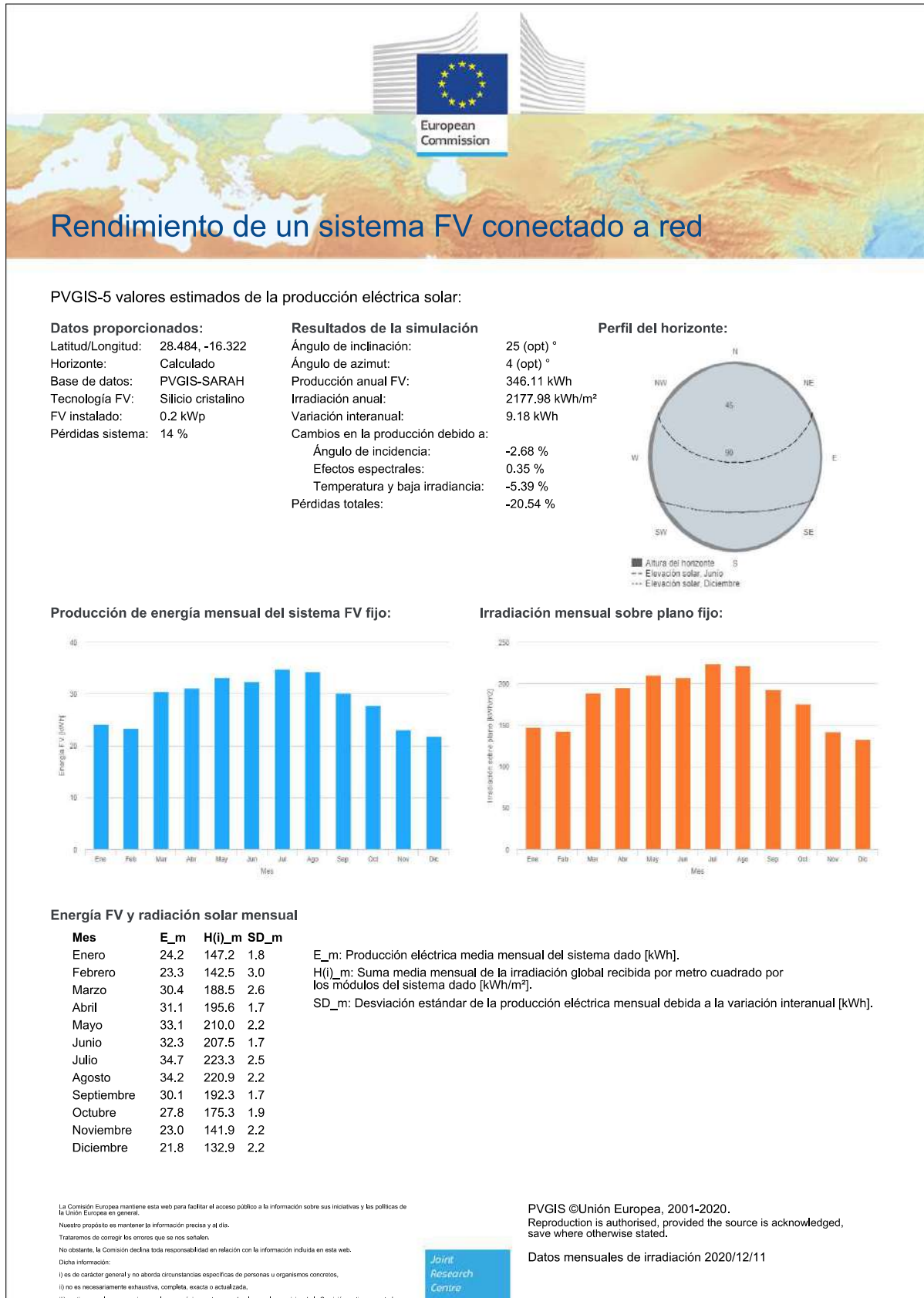
La Comisión Europea mantiene esta web para facilitar el acceso público a la información sobre sus iniciativas y las políticas de la Unión Europea en general.  
 Nuestro propósito es mantener la información precisa y al día.  
 Trataremos de corregir los errores que se nos señalen.  
 No obstante, la Comisión declina toda responsabilidad en relación con la información incluida en esta web.  
 Dicha información:  
 i) es de carácter general y no aborda circunstancias específicas de personas u organismos concretos,  
 ii) no es necesariamente exhaustiva, completa, exacta o actualizada,  
 iii) contiene en algunas ocasiones enlaces a páginas externas sobre las que los servicios de la Comisión no tienen control



PVGIS ©Unión Europea, 2001-2020.  
 Reproduction is authorised, provided the source is acknowledged, save where otherwise stated.

Datos mensuales de irradiación 2020/12/11

### F.2.3. Resultados de insolación optimizados en La laguna



### F.2.4. Resultados de insolación optimizados en Las Palmas de Gran Canaria



PVGIS-5 valores estimados de la producción eléctrica solar:

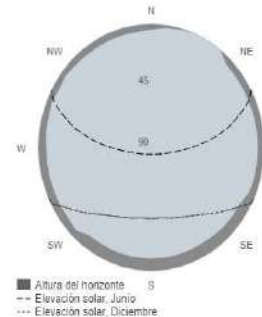
**Datos proporcionados:**

Latitud/Longitud: 28.004, -15.419  
 Horizonte: Calculado  
 Base de datos: PVGIS-SARAH  
 Tecnología FV: Silicio cristalino  
 FV instalado: 0.2 kWp  
 Pérdidas sistema: 14 %

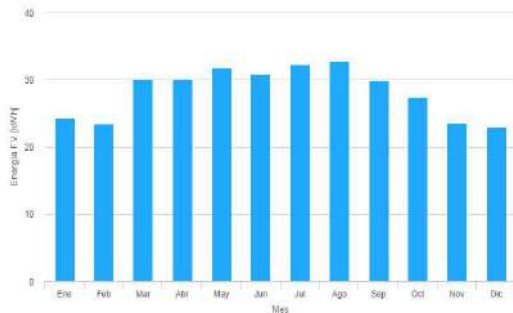
**Resultados de la simulación**

Ángulo de inclinación: 25 (opt) °  
 Ángulo de azimut: 8 (opt) °  
 Producción anual FV: 339.21 kWh  
 Irradiación anual: 2150.36 kWh/m<sup>2</sup>  
 Variación interanual: 8.28 kWh  
 Cambios en la producción debido a:  
 Ángulo de incidencia: -2.61 %  
 Efectos espectrales: 0.29 %  
 Temperatura y baja irradiancia: -6.1 %  
 Pérdidas totales: -21.13 %

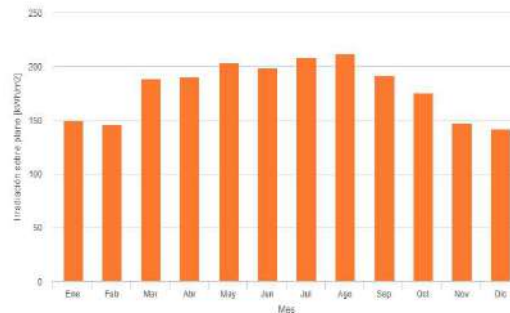
**Perfil del horizonte:**



**Producción de energía mensual del sistema FV fijo:**



**Irradiación mensual sobre plano fijo:**



**Energía FV y radiación solar mensual**

Mes	E_m	H(i)_m	SD_m
Enero	24.3	149.7	2.0
Febrero	23.5	145.3	2.4
Marzo	30.0	188.6	2.1
Abril	30.0	190.2	1.6
Mayo	31.8	203.0	2.1
Junio	30.8	198.4	1.8
Julio	32.3	207.9	2.2
Agosto	32.8	211.5	2.1
Septiembre	29.9	191.9	2.0
Octubre	27.4	175.1	2.0
Noviembre	23.6	147.4	1.9
Diciembre	22.9	141.4	2.1

E\_m: Producción eléctrica media mensual del sistema dado [kWh].  
 H(i)\_m: Suma media mensual de la irradiación global recibida por metro cuadrado por los módulos del sistema dado [kWh/m<sup>2</sup>].  
 SD\_m: Desviación estándar de la producción eléctrica mensual debida a la variación interanual [kWh].

La Comisión Europea mantiene esta web para facilitar el acceso público a la información sobre sus iniciativas y las políticas de la Unión Europea en general.  
 Nuestro propósito es mantener la información precisa y al día.  
 Trataremos de corregir los errores que se nos señalen.  
 No obstante, la Comisión declina toda responsabilidad en relación con la información incluida en esta web.  
 Dicha información:  
 i) es de carácter general y no aborda circunstancias específicas de personas u organismos concretos,  
 ii) no es necesariamente exhaustiva, completa, exacta o actualizada,  
 iii) contiene en algunas ocasiones enlaces a páginas externas sobre las que los servicios de la Comisión no tienen control



PVGIS ©Unión Europea, 2001-2020.  
 Reproduction is authorised, provided the source is acknowledged, save where otherwise stated.

Datos mensuales de irradiación 2020/12/11





# Anexo G

## *Scripts* utilizados

En este anexo se adjuntan los *scripts* utilizados en las diferentes partes del desarrollo del proyecto y en cada sección se detalla para que han sido usados.

### G.1. Generación de señal

En esta sección del anexo se adjunta el primer código utilizado para la generación de señal en alta frecuencia, dicho código es optimizado y utilizado con posterioridad en las diferentes secciones que se muestran.

```
1
2 void clockgenerator (unsigned long time_delay){
3
4   static bool flag = false;
5   static bool flag2 = false;
6   static bool flag3 = false;
7   static bool flag4 = false;
8
9   unsigned long currentMillis = millis();
10  unsigned long previousMillis;
11
12  if (flag == false){
13    analogWrite(3,127);
14    previousMillis = millis();
15    flag = true;
16    Serial.println("Salida PWM3 activada");
17  }
18  Serial.print("Current milis: ");
19  Serial.println(currentMillis);
20  Serial.print("Prev milis: ");
21  Serial.println(previousMillis);
22
```

```

23
24 if ((currentMillis <= time_delay * 2) && (currentMillis >=
    time_delay + previousMillis ) && (flag == true)) {
25 analogWrite(9,127);
26 previousMillis = millis();
27 flag2 = true;
28 Serial.println("Salida PWM9 activada");
29 }
30
31
32 if ((currentMillis <= time_delay * 3) && (currentMillis >=
    time_delay + previousMillis ) && (flag2 == true)) {
33 analogWrite(10,127);
34 previousMillis = currentMillis ;
35 flag3 = true;
36 Serial.println("Salida PWM10 activada");
37 }
38
39 if ((currentMillis <= time_delay * 4) && (currentMillis >=
    time_delay + previousMillis) && (flag3 == true)) {
40 analogWrite(11,127);
41 flag4 = true;
42 Serial.println("Salida PWM11 activada");
43 }
44
45 }
46
47
48
49 void setPWMPrescaler(uint8_t pin , uint16_t prescale) {
50
51     byte mode;
52
53     if(pin == 9 || pin == 10) {
54         switch(prescale) {
55             case 1: mode = 0b001; break;
56             case 8: mode = 0b010; break;
57             case 64: mode = 0b011; break;
58             case 256: mode = 0b100; break;
59             case 1024: mode = 0b101; break;
60             default: return;
61         }
62
63     } else if(pin == 3 || pin == 11) {
64         switch(prescale) {
65             case 1: mode = 0b001; break;
66             case 8: mode = 0b010; break;
67             case 32: mode = 0b011; break;
68             case 64: mode = 0b100; break;

```

```
69     case 128: mode = 0b101; break;
70     case 256: mode = 0b110; break;
71     case 1024: mode = 0b111; break;
72     default: return;
73   }
74 }
75
76 if(pin==5 || pin==6) {
77   TCCR0B = TCCR0B & 0b11111000 | mode;
78 } else if (pin==9 || pin==10) {
79   TCCR1B = TCCR1B & 0b11111000 | mode;
80 } else if (pin==3 || pin==11) {
81   TCCR2B = TCCR2B & 0b11111000 | mode;
82 }
83 }
84
85
86
87
88 void setup() {
89   Serial.begin(9600);
90   setPWMPrescaler(9,64);
91   setPWMPrescaler(10,64);
92   setPWMPrescaler(3,64);
93   setPWMPrescaler(11,64);
94 }
95
96 void loop() {
97   clockgenerator(5000);
98 }
99 }
```

## G.2. Adquisición de los datos del Osciloscopio

En esta sección del anexo, se adjunta el código utilizado para obtener los datos desde el osciloscopio en los diferentes ensayos detallados en el anexo B. Este *script* no fue realizado durante el desarrollo del proyecto, se trata de un código realizado por el co-tutor de este proyecto para una de sus investigaciones previas y que fue cedido por el mismo, para el uso necesario en el desarrollo de este proyecto.

```
1
2 #!/usr/bin/python3
3 # -*- coding: utf-8 -*-
4 '''
5 Tektronix TDS 210 RS232
6
7 *** Script para la comunicación RS232 con el osciloscopio
8     Tektronix TDS 210 ***
9
10 This program is free software: you can redistribute it and/or
11 modify
12 it under the terms of the GNU General Public License as
13 published by
14 the Free Software Foundation, either version 3 of the License,
15 or
16 (at your option) any later version.
17
18 This program is distributed in the hope that it will be useful,
19 but WITHOUT ANY WARRANTY; without even the implied warranty of
20 MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
21 GNU General Public License for more details.
22
23 You should have received a copy of the GNU General Public
24 License
25 along with this program. If not, see <https://www.gnu.org/licenses/>.
26
27 Iván Rodríguez (irodrigu@ull.es) – 2021
28 '''
29
30 from matplotlib import pyplot as plt
31 from datetime import datetime
32 import serial, time, csv, sys
33 import numpy as np
34
35 def log_msg(msg):
36     '''Función para crear mensajes de log del sistema'''
37     script_name = "tds210-rs232.py"
```

```

33
34     now = datetime.now()
35     dt_string = now.strftime("[%d/%m/%Y %H:%M%S] ") # Creamos
36     el mensaje que queremos guardar
37     f_msg = dt_string+script_name+": "+msg+"\n"
38     print(f_msg[:-1])
39
40 def print_serial_info(serial):
41     '''Función para imprimir la información serial recibida'''
42     str = ser.readline()[:-1].decode("utf-8")
43     log_msg("Respuesta: "+str) # Imprimimos el resultado
44     return str
45
46 def get_data(serial, voltage_scale, time_scale):
47     ''' Función para obtener la forma de onda por medio del
48     RS232 '''
49     curva = ser.readline()[:-1].decode("utf-8")
50     curva_lst = curva.split(",")
51
52     # Adaptación de escala de datos
53     data_int = [int(integer) for integer in curva_lst]
54     data = [(((integer+32768)*voltage_scale*8)/65536.0)-
55     voltage_scale*4)*1.25 for integer in data_int]
56
57     # Datos temporales
58     time = []
59     for i in range(0,2500):
60         time.append(((i*10*time_scale)/2500))
61
62     return data, time
63
64 def write_csv(name, field1, field2, data, time, data2, time2):
65     '''Función para escribir un fichero .csv con los datos
66     obtenidos'''
67     # Nombres de campos
68     campos = ["Tiempo", field1, "Tiempo", field2]
69
70     filename = name+"-ttds210.csv"
71
72     log_msg("Abriendo el fichero "+filename+".")
73     with open(filename, "w") as file:
74         write = csv.writer(file)
75
76         # Escribimos los campos del fichero
77         write.writerow(campos)
78
79         # Escribimos los datos
80         for i in range(len(data)):
81             write.writerow([time[i], data[i], time2[i], data2[i]])

```

```

78     log_msg("Datos escritos en el fichero "+filename+".")
79
80
81 if __name__ == "__main__":
82     log_msg("Iniciando el programa.")
83
84     #Comprobamos el número de argumentos de entrada
85     if len(sys.argv) < 4:
86         print("Número de argumentos de entrada insuficiente")
87         print("Uso: tds210-rs232.py <nombre-fichero-destino> <
etiqueta-medida-1> <etiqueta-medida-2>")
88         sys.exit()
89
90     with serial.Serial('/dev/ttyUSB0', 9600, timeout=1, parity=
serial.PARITY_EVEN) as ser:
91         log_msg("Conexión RS232 establecida.")
92
93         ser.write(b"\r") # Limpiamos la posible basura que
tengamos en el buffer
94         ser.readline()
95
96         log_msg("Obteniendo información del equipo.")
97         ser.write(b"*IDN?\r")
98         print_serial_info(ser)
99
100        log_msg("Obteniendo el canal de medida.")
101        ser.write(b"MEASUREMENT:IMMED:SOURCE?\r")
102        print_serial_info(ser)
103
104        # Intentamos leer una señal
105        ser.write(b>Data:Source CHI\r")
106        ser.write(b>Data:Encdg ASCII\r")
107        ser.write(b>Data:Width 2\r")
108        ser.write(b>Data:Start 1\r")
109        ser.write(b>Data:Stop 2500\r")
110
111
112        log_msg("Obteniendo la escala horizontal.")
113        ser.write(b"HORizontal:MAIn:SECdiv?\r")
114        # Guardamos la escala de tiempo
115        time_scale = float(ser.readline()[:-1].decode("utf-8"))
116        log_msg("%f Segundos por división"%time_scale)
117
118        log_msg("Obteniendo la escala vertical.")
119        ser.write(b"CHI:SCALE?\r")
120        # Guardamos la escala de votaje
121        vol_scale = float(ser.readline()[:-1].decode("utf-8"))
122        log_msg("%f Voltios por división"%vol_scale)
123

```



```

124     log_msg("Obteniendo datos de la curva.")
125     ser.write(b"Curve?\r")
126     data1, time1 = get_data(ser, vol_scale, time_scale)
127     log_msg("Datos obtenidos.")
128     ser.close()
129
130     with serial.Serial('/dev/ttyUSB0', 9600, timeout=1, parity=
serial.PARITY_EVEN) as ser:
131         log_msg("Cambiamos al canal 2.")
132
133         ser.write(b"DATA:SOURce CH2\r")
134         ser.write(b"Data:Encdg ASCII\r")
135         ser.write(b"Data:Width 2\r")
136         ser.write(b"Data:Start 1\r")
137         ser.write(b"Data:Stop 2500\r")
138         time.sleep(2)
139
140         log_msg("Obteniendo el canal de medida.")
141         ser.write(b"DATA:SOURce?\r")
142         print_serial_info(ser)
143
144         log_msg("Obteniendo la escala vertical.")
145         ser.write(b"CH2:SCAlE?\r")
146         # Guardamos la escala de votaje
147         vol_scale = float(ser.readline()[:-1].decode("utf-8"))
148         log_msg("%f Voltios por división"%vol_scale)
149
150         log_msg("Obteniendo datos de la curva.")
151         ser.write(b"Curve?\r")
152         data2, time2 = get_data(ser, vol_scale, time_scale)
153         log_msg("Datos obtenidos.")
154
155         log_msg("Cerrando el puerto serie.")
156         ser.close()
157
158     log_msg("Realizando la representación gráfica")
159
160     # Representación de datos
161     plt.plot(time1, data1, label="CH1".format(sys.argv[2]))
162     plt.plot(time2, data2, label="CH2".format(sys.argv[3]))
163     plt.ylim([-vol_scale*4, vol_scale*4])
164     plt.title('Señal "cuadrada" de entrada y salida {}'.format(
sys.argv[1]))
165     plt.xlabel("Tiempo (s)")
166     plt.ylabel("Voltaje (V)")
167     plt.grid()
168     plt.savefig("{} .png".format(sys.argv[1]))
169

```

```
170 write_csv(sys.argv[1], sys.argv[2], sys.argv[3], data1, time1,
      data2, time2)
```

### G.3. Tratamiento de datos

En esta sección del anexo se adjunta el código utilizado para el tratamiento de los datos obtenidos en la realización del ensayo de caracterización del piezoeléctrico que se detalla en el anexo **B**.

```
1
2 #!/usr/bin/python3
3 # -*- coding: utf-8 -*-
4
5 '''
6 Script para el tratamiento de datos obtenidos en el ensayo
7
8 *** Script para la obtención de los datos necesarios para la
9     caracterización del piezoeléctrico desde los datos obtenidos
10    mediante el ensayo de laboratorio***
11
12 This program is free software: you can redistribute it and/or
13    modify
14    it under the terms of the GNU General Public License as
15    published by
16    the Free Software Foundation, either version 3 of the License,
17    or
18    (at your option) any later version.
19
20 This program is distributed in the hope that it will be useful,
21    but WITHOUT ANY WARRANTY; without even the implied warranty of
22    MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
23    GNU General Public License for more details.
24
25 You should have received a copy of the GNU General Public
26    License
27    along with this program. If not, see <https://www.gnu.org/licenses/>.
28
29 Rubén Morales (alu0100949582@ull.edu.es) – 2022
30 '''
31
32 import numpy as np
33 import math
34 import os
35 import shutil
36 import time, csv, sys
```

```

31 from tqdm import tqdm
32 from array import array
33 from glob import glob
34 from matplotlib import pyplot as plt
35
36 def buscafreq(filename):
37     pos=filename.find('hz')
38     frecuencia = int(filename[:pos].split('/')[ -1])
39     return (frecuencia)
40
41 def leer_fichero(filename):
42     '''
43     Uso: data=lee_file(filename)
44
45     filename: Es el nombre del cvs con los datos del tektronik.
46     data: devuelve un array de 2500x4 con los datos t1,v1,t2,v2,
47           tiempos
48           y voltajes de los dos canales del osciloscopio.
49     '''
49     data = np.array([[0,0,0,0]], dtype='float32')
50     with open(filename, "r") as fname:
51         contenido = fname.readlines()
52         for linea in contenido[1:]:
53             #print(linea)
54             aux = np.array([linea.strip("\n").split(',')]).
55             astype('float32')
56             aux[0,0]*=10. # corrección de la adquisición del
57             tiempo
58             aux[0,2]*=10. # corrección de la adquisición del
59             tiempo
60             data = np.append(data,aux, axis=0)
61             data = data[1:,:]
62             media1 = np.sum(data[:,1])/data.shape[0]
63             data[:,1] = data[:,1]-media1
64             media3 = np.sum(data[:,3])/data.shape[0]
65             data[:,3] = data[:,3]-media3
66             return(data)
67
68 def medir_amplitudes(datos, frecuencia):
69     dt = datos[1,0]
70     T = 1/frecuencia
71     N = len(datos[:,0])
72     x = datos[:,1]-np.mean(datos[:,1])
73     y = datos[:,3]-np.mean(datos[:,3])
74     A1 = np.sqrt(np.sum(x**2)*2/N)
75     A2 = np.sqrt(np.sum(y**2)*2/N)
76     x/=A1
77     y/=A2
78     fase = np.arccos(np.sum(x*y)*2/N)

```

```

76     return(A1,A2,A1/A2, fase)
77
78
79 def main():
80
81     files = glob(os.getcwd()+"/*ttds210.csv")
82     print("El numero de medidas tomadas es:",len(files))
83
84
85     orden_correcto = sorted(files , key=buscafreq)
86     print("El orden de los ficheros de datos es el siguiente;")
87
88
89
90
91     myData = np.zeros((len(orden_correcto),7))
92
93     for file in tqdm(orden_correcto):
94         #print(file)
95         data = leer_fichero(file)
96
97         frecuencia = buscafreq(file)
98
99         A1,A2,g,fase=medir_amplitudes(data , frecuencia)
100
101         #cálculo de la ganancia en db
102         gananciaDB = 10*math.log(A1/A2)
103
104         ## Calculo de la impedancia
105
106         vx = np.abs(A2-A1)
107         r1 = 10000
108         i =A1/r1
109         zt= vx/i
110         # zreal=g*np.cos(fase)
111         # zimag=g*np.sin(fase)
112
113         myData[orden_correcto.index(file),0]=frecuencia
114         myData[orden_correcto.index(file),1]=A2
115         myData[orden_correcto.index(file),2]=A1
116         myData[orden_correcto.index(file),3]=A1/A2
117         myData[orden_correcto.index(file),4]=gananciaDB
118         myData[orden_correcto.index(file),5]=fase
119         myData[orden_correcto.index(file),6]=zt
120
121
122
123     with open('Datos_curvas.csv', 'w',newline='') as file:
124

```

```

125     campos = ["Frecuencia", "Amplitud entrada", "Amplitud
126     salida", "Ganancia", "Ganancia (db)", "fase", "impedancia"]
127     writer = csv.writer(file, delimiter=',', quotechar='"',
128     quoting=csv.QUOTE_MINIMAL)
129     writer.writerow(campos)
130     for i in range(myData.shape[0]):
131         data = []
132         for j in range(myData.shape[1]):
133             data.append(myData[i, j])
134         writer.writerow(data)
135
136 if __name__ == "__main__":
137     main()

```

## G.4. Generación de gráficas

En esta sección del anexo se adjunta el código utilizado para la generación de las gráficas de los datos obtenidos tras tratar los datos de la realización del ensayo de caracterización del piezoeléctrico que se detalla en el anexo B.

```

1
2 #!/usr/bin/python3
3 # -*- coding: utf-8 -*-
4
5 '''
6 Script para la obtención de gráficas de un fichero de datos
7
8 *** Script para la obtención de las gráficas que caracterizan el
9     piezoeléctrico usado, desde el fichero de datos obtenido al
10    tratar los datos del ensayo***
11
12 This program is free software: you can redistribute it and/or
13    modify
14    it under the terms of the GNU General Public License as
15    published by
16    the Free Software Foundation, either version 3 of the License,
17    or
18    (at your option) any later version.
19
20 This program is distributed in the hope that it will be useful,
21    but WITHOUT ANY WARRANTY; without even the implied warranty of
22    MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
23    GNU General Public License for more details.
24
25 You should have received a copy of the GNU General Public
26    License

```

```

21 along with this program. If not, see <https://www.gnu.org/licenses/>.
22
23 Rubén Morales (alu0100949582@ull.edu.es) – 2022
24 '''
25
26 import csv
27 import numpy as np
28 from matplotlib import pyplot as plt
29
30 file = "Datos_curvas.csv"
31 data = np.array([[0,0,0,0,0,0,0]], dtype='float32')
32
33 with open(file , "r") as fname:
34     contenido = fname.readlines()
35     for i in range(1,len(contenido)):
36         linea=contenido[i]
37         aux = np.array([linea.strip("\n").split(',')]).
38         astype('float32')
39         data = np.append(data ,aux , axis=0)
40
41
42 ## Representación de los resultados obtenidos Ganancia vs
43 Frecuencia
44
45 plt.plot(data[1:,0], data[1:,4])
46 plt.title("Ganancia (db) vs Frecuencia (Hz) ")
47 plt.xscale("log")
48 plt.xlabel("Frecuencia (Hz)")
49 plt.ylabel("Ganancia (db)")
50 plt.grid()
51 #plt.show()
52 plt.savefig("Ganancia_VS_Frecuencia.png")
53
54 ## Representación de los resultados obtenidos Impoedancia vs
55 Frecuencia
56
57 plt.figure()
58 plt.plot(data[1:,0], data[1:,6])
59 plt.title("Impedancia ( $\Omega$ ) vs Frecuencia(Hz)")
60 plt.xscale("log")
61 plt.xlabel("Frecuencia (Hz)")
62 plt.ylabel("Impedancia ( $\Omega$ )")
63 plt.grid()
64 #plt.show()
65 plt.savefig("Impedancia_VS_Frecuencia.png")

```

```
66 ## Representación de los resultados obtenidos Desfase vs
    Frecuencia
67 plt.figure()
68 plt.plot(data[1:,0],data[1:,5])
69 plt.title("Desfase ( $\hat{\phi}$ ) vs Frecuencia(Hz)")
70 plt.xscale("log")
71 plt.xlabel("Frecuencia (Hz)")
72 plt.ylabel("Desfase ( $\hat{\phi}$ )")
73 plt.grid()
74 #plt.show()
75 plt.savefig("Desfase_VS_Frecuencia.png")
```



## G.5. Simulaciones acústicas

En esta sección del anexo se adjuntan los códigos que han permitido obtener las simulaciones gráficas usadas en el capítulo 3.

### G.5.1. Simulaciones de propagación de ondas

```
1
2 #!/usr/bin/python3
3 # -*- coding: utf-8 -*-
4
5 '''
6 Script para la obtención de la propagación de ondas atendiendo a
7     retrasos entre emisiones
8
9 *** Script para la obtención de la propagación de ondas
10 desde un determinado número de emisores a un receptor , que
11 recibe las señales en el mismo instante***
12
13 This program is free software: you can redistribute it and/or
14 modify
15 it under the terms of the GNU General Public License as
16 published by
17 the Free Software Foundation, either version 3 of the License,
18 or
19 (at your option) any later version.
20
21 This program is distributed in the hope that it will be useful,
22 but WITHOUT ANY WARRANTY; without even the implied warranty of
23 MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
24 GNU General Public License for more details.
25
26 You should have received a copy of the GNU General Public
27 License
28 along with this program. If not, see <https://www.gnu.org/licenses/>.
29
30 Rubén Morales (alu0100949582@ull.edu.es) – 2022
31 '''
32
33 import numpy as np
34 import matplotlib.pyplot as plt
35 import os
```

```

35 def circulo(centro, punto, dt=0.1, prop=1.):
36     centro=np.array(centro)
37     punto=np.array(punto)
38     relativo = punto-centro
39     relcomp = relativo[0]+1j*relativo[1]
40     radio = np.abs(relcomp)
41     angulo = np.angle(relcomp)
42     theta = np.arange(angulo-np.pi/4, angulo+np.pi/4, np.pi/100.)
43     x = centro[0]+(radio*prop-dt)*np.cos(theta)
44     x2 = centro[0]+(radio*prop+dt)*np.cos(theta)
45     y = centro[1]+(radio*prop-dt)*np.sin(theta)
46     y2 = centro[1]+(radio*prop+dt)*np.sin(theta)
47     plt.plot(x,y, 'k')
48     plt.plot(x2,y2, 'k')
49
50     return
51
52 for n,prop in enumerate(np.arange(0.1,1.4,0.01)):
53     paj=[0,5]
54     plt.figure(1)
55     for pos in [-1.5,-0.5,0.5,1.5]:
56         circulo([pos,0], paj, dt=0.1, prop=prop)
57     plt.plot(-1.5,0, 'ko')
58     plt.plot(-0.5,0, 'ko')
59     plt.plot(0.5,0, 'ko')
60     plt.plot(1.5,0, 'ko')
61     plt.plot(paj[0], paj[1], 'r*')
62     plt.axis([-10,10,-1,10])
63     plt.title('Propagación de la señal en el plano')
64     plt.xlabel('Posición de los emisores')
65     plt.ylabel('Distancia')
66     plt.savefig('filename_%03d.png'%n)
67     plt.close()

```

## G.5.2. Simulación de propagación del frente de onda

```

1 #!/usr/bin/python3
2 # -*- coding: utf-8 -*-
3
4 '''
5 Script para la obtención de la propagación del frente de onda
6
7 *** Script para la obtención de la propagación del frente de
8 onda
9 desde un determinado número de emisores a un receptor, que
10 recibe las señales en el mismo instante***

```

```

11 This program is free software: you can redistribute it and/or
    modify
12 it under the terms of the GNU General Public License as
    published by
13 the Free Software Foundation, either version 3 of the License,
    or
14 (at your option) any later version.
15
16 This program is distributed in the hope that it will be useful,
17 but WITHOUT ANY WARRANTY; without even the implied warranty of
18 MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
19 GNU General Public License for more details.
20
21 You should have received a copy of the GNU General Public
    License
22 along with this program. If not, see <https://www.gnu.org/
    licenses/>.
23
24 Rubén Morales (alu0100949582@ull.edu.es) – 2022
25 '''
26
27
28
29 import numpy as np
30 import matplotlib.pyplot as plt
31 import os
32
33
34 pemi = [-9+0j, -3+0j, 3+0j, 9+0j]
35 ppaj = [1+5j]
36
37 def dominio(xini=-10., xfin=10., xinc=0.1, yini=-1., yfin=19.,
    yinc=0.1):
38     x = np.arange(xini, xfin, xinc)
39     y = np.arange(yini, yfin, yinc)
40     xx, yy = np.meshgrid(x, y)
41     c=340
42     return xx, yy, c
43
44 def energia(pemi, ppaj, t, xx, yy, c):
45     """
46     Entradas
47     =====
48     pemi: Posición del emisor en x + y j.
49     ppaj: Posición del pájaro en x + y j.
50     t    : Tiempo de propagación.
51     """
52
53     drefer = np.abs(ppaj)

```

```

54     rel = ppaj - pemi
55     demipaj = np.abs(rel)
56     dretraso = demipaj - drefer
57     xxjyy = xx + 1j * yy
58     rr = np.abs(xxjyy - pemi)
59     zz = np.exp(-(rr - t * c - dretraso) ** 2)
60
61     return zz
62
63 #zz = energia(pos1, paj1, 0)
64 #plt.imshow(zz, origin='xy', extent=[-10.,10,-1,19])
65
66 xini = -10.
67 xfin = 10.
68 xinc = 0.01
69 yini = -1.0
70 yfin = 19.
71 yinc = 0.01
72 xx, yy, c = dominio(xini=xini, xfin=xfin, xinc=xinc, yini=yini,
73                     yfin=yfin, yinc=yinc)
74
75 for n, ti in enumerate(np.arange(-0.05, 0.05, 0.001)):
76     plt.figure(1)
77     zz1 = energia(pemi[0], ppaj[0], ti, xx, yy, c)
78     plt.plot(pemi[0].real, pemi[0].imag, 'ko')
79     zz2 = energia(pemi[1], ppaj[0], ti, xx, yy, c)
80     plt.plot(pemi[1].real, pemi[1].imag, 'ko')
81     zz3 = energia(pemi[2], ppaj[0], ti, xx, yy, c)
82     plt.plot(pemi[2].real, pemi[2].imag, 'ko')
83     zz4 = energia(pemi[3], ppaj[0], ti, xx, yy, c)
84     plt.plot(pemi[3].real, pemi[3].imag, 'ko')
85     plt.plot(ppaj[0].real, ppaj[0].imag, 'r*')
86     plt.imshow(zz1 + zz2 + zz3 + zz4, origin='xy', extent=[xini, xfin,
87                 yini, yfin])
88     plt.axis([xini, xfin, yini, yfin])
89     plt.suptitle('Propagación del frente de onda en el plano de
90                 emisión')
91     plt.title('t=%f'%ti)
92     plt.xlabel('Posición de los emisores')
93     plt.ylabel('Distancia')
94     plt.savefig('filename_%03d.png'%n)
95     plt.close()

```

### G.5.3. Simulación del cambio de presión

```

1
2 #!/usr/bin/python3
3 # -*- coding: utf-8 -*-

```

```
4 '''
5
6 Script para la obtención del cambio de presión en el frente de
7 onda
8 *** Script para la obtención del cambio de presión en el frente
9 de onda
10 desde un determinado número de emisores a un receptor ,el
11 cual es rodeado
12 por el dominio de la imagen generada***
13
14 This program is free software: you can redistribute it and/or
15 modify
16 it under the terms of the GNU General Public License as
17 published by
18 the Free Software Foundation , either version 3 of the License ,
19 or
20 (at your option) any later version .
21
22 This program is distributed in the hope that it will be useful ,
23 but WITHOUT ANY WARRANTY; without even the implied warranty of
24 MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
25 GNU General Public License for more details .
26
27 You should have received a copy of the GNU General Public
28 License
29 along with this program. If not , see <https://www.gnu.org/
30 licenses/>.
31
32 Rubén Morales (alu0100949582@ull.edu.es) – 2022
33 '''
34
35 import numpy as np
36 import matplotlib.pyplot as plt
37 import os
38
39 def pulso_unitario(t, t0=0.0, tau=1.0):
40     """Función para generar la señal pulso centrado en el instante
41     de tiempo
42     t0 y de ancho tau. Por defecto , genera el pulso unitario.
43     Entradas:
44     t : Array con los instantes de tiempo en los que se quiere
45     calcular la función.
46     t0 [opcional]: Instante temporal del centro del pulso. Vale
47     0.0 por defecto.
48     tau [opcional]: Ancho temporal del pulso. Vale 1.0 por
49     defecto.
50     Salidas:
```

```

40     pu : Array con los valores de la función en los instantes de
41     entrada.
42     """
43     # Calculamos el instante inicial del pulso y el instante final
44     .
45     tini = t0-tau/2
46     tfin = t0+tau/2
47     # Uso de la función np.logical_and() que realiza la operación
48     lógica and entre
49     # dos arrays de la misma dimensión, punto a punto.
50     pu=np.logical_and(t>=tini , t <tfin).astype('float')
51     return pu
52
53 def dominio(xini=-5., xfin=5., xinc=0.1, yini=-1., yfin=9., yinc
54 =0.01):
55     x = np.arange(xini, xfin, xinc)
56     y = np.arange(yini, yfin, yinc)
57     xx,yy = np.meshgrid(x,y)
58     c=340.
59     return xx,yy,c
60
61 def presion(pemi, ppaj, t, xx,yy,c):
62     """
63     Entradas
64     =====
65     pemi: Posición del emisor en x + y j.
66     ppaj: Posición del pájaro en x + y j.
67     t    : Tiempo de propagación.
68     xx   : Imagen coordenada x
69     yy   : Imagen coordenada y
70     c    : Velocidad del sonido en m/s
71     """
72
73     drefer = np.abs(ppaj)
74     rel = ppaj-pemi
75     demipaj = np.abs(rel)
76     dretraso = demipaj-drefer
77     xxjyy=xx+1j*yy
78     rr = np.abs(xxjyy-pemi)
79     lamb = c/31250
80     zz = np.sin(2*np.pi*(rr-t*c-dretraso)/lamb) * pulso_unitario(
81     rr , t0=0.554/2+t*c+dretraso , tau=0.544)
82
83     return zz
84
85 def pinta(ti=0.02, anim=False, n=0):
86     plt.figure(1)
87     plt.clf()

```

```

84  zz1 = presion(pemi[0], ppaj[0], ti, xx, yy, c)
85  plt.plot(pemi[0].real, pemi[0].imag, 'ko')
86  zz2 = presion(pemi[1], ppaj[0], ti, xx, yy, c)
87  plt.plot(pemi[1].real, pemi[1].imag, 'ko')
88  zz3 = presion(pemi[2], ppaj[0], ti, xx, yy, c)
89  plt.plot(pemi[2].real, pemi[2].imag, 'ko')
90  zz4 = presion(pemi[3], ppaj[0], ti, xx, yy, c)
91  plt.plot(pemi[3].real, pemi[3].imag, 'ko')
92  plt.plot(ppaj[0].real, ppaj[0].imag, 'r*')
93  plt.imshow(zz1+zz2+zz3+zz4, origin='lower', extent=[xini, xfin,
94             yini, yfin])
95  plt.axis([xini, xfin, yini, yfin])
96  plt.title('Propagación de la onda de presión en el plano \n t
97           =%f' %ti)
98  plt.xlabel('Posición de los emisores')
99  plt.ylabel('Distancia')
100 if(anim==True):
101     plt.savefig('presion_%03d.png' %n)
102     plt.close()
103 else:
104     plt.show()
105 return
106 # Posiciones de los emisores codificadas en dos dimensiones como
107     complejos.
108 pemi = [-0.45+0j, -0.15+0j, 0.15+0j, 0.45+0j]
109 # Posición del ave a ahuyentar como complejo
110 ppaj = [0.2+8j]
111
112 # Establecemos el dominio
113 xini=ppaj[0].real-0.2
114 xfin=ppaj[0].real+0.2
115 xinc=0.0001
116 yini=ppaj[0].imag-0.2
117 yfin=ppaj[0].imag+0.2
118 yinc=0.0001
119
120 xx,yy,c = dominio(xini=xini, xfin=xfin, xinc=xinc, yini=yini,
121                  yfin=yfin, yinc=yinc)
122 for n, ti in enumerate(np.arange(0.0213, 0.0242, 0.00005)):
123     pinta(ti, anim=True, n=n)

```

#### G.5.4. Detección de aves



```
2 #!/usr/bin/python3
3 # -*- coding: utf-8 -*-
4 '''
5 Sistema del procesamiento para el prototipo de prueba de
6     conceptos de una protección avifauna
7
8 *** Script para el procesamiento de los datos de detección de
9     aves***
10
11 This program is free software: you can redistribute it and/or
12 modify
13 it under the terms of the GNU General Public License as
14 published by
15 the Free Software Foundation, either version 3 of the License,
16 or
17 (at your option) any later version.
18
19 This program is distributed in the hope that it will be useful,
20 but WITHOUT ANY WARRANTY; without even the implied warranty of
21 MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
22 GNU General Public License for more details.
23
24 You should have received a copy of the GNU General Public
25 License
26 along with this program. If not, see <https://www.gnu.org/licenses/>.
27
28 Rubén Morales (alu0100949582@ull.edu.es) - 2022
29 '''
30
31 import detector as dt
32 import aviso as av
33 DEBUG = False
34 dt.abrir_camara()
35
36 while True:
37     imagen = dt.leer_imagen()
38     lx, ly, eti = dt.detec_ave(imagen)
39     if DEBUG == True:
40         if len(lx) == 0:
41             print("No hay nada")
42         else:
43             print("Hay cosas")
44     for n in range(len(lx)):
45         if eti[n] == "bird":
46             mx, my, mz = dt.ima2wrl(lx[n], ly[n])
47             if DEBUG == False:
48                 print(eti[n], ": ", lx[n], ly[n])
```

```

44         retrasos = av.calcula_retrasos(mx,my,mz)
45         status = av.envia_retrasos(retrasos)
46         if status!=0:
47             print("procesar: Error")
48     else:
49         if DEBUG == True:
50             print(eti[n], ": ", lx[n],ly[n])
51 dt.cerrar_camara()

```

## G.6. Creación del servicio del sistema

En esta sección se adjunta aquella información relevante a la creación del servicio del sistema. Este servicio del sistema se crea con el fin de que el programa de procesamiento del instrumento se ejecute desde que este es encendido. Para ello se ha creado un servicio o *Daemon* del sistema, el cual consta de un fichero de texto plano, con la extensión “.service” que debe ser habilitado de una cierta manera para que al ser encendido el instrumento ejecute el fichero “.py” correspondiente.

### G.6.1. Texto plano del servicio del sistema

```

[Unit]
Description=Detector
After=network.target
StartLimitIntervalSec=0

[Service]
Type=simple
Restart=always
RestartSec=1
ExecStart= ## Poner la ruta que apunta al fichero
            que ejecuta el sistema de control

[Install]
WantedBy=default.target

```

### G.6.2. Instrucciones para el servicio del sistema

Para la puesta en marcha del fichero ".service" para la activación del sistema de control es necesario realizar la siguiente lista de acciones:

- cambiar la ruta para que apunte al fichero correcto.

- copiar el fichero `*.service` al directorio `/lib/systemd/system`
- comprobar que se hace correctamente mediante una búsqueda.
- Posteriormente se deben ejecutar los comandos que lo activan:
  - `sudo systemctl list-unit-files | grep fichero.service` (Comprobar si está o no habilitado el servicio).
  - `sudo systemctl start detector.service`.
  - `sudo systemctl enable detector.service`.
  - `journalctl -u detector.service` (Permite ver las acciones realizadas).
- si existe algún error se deben ejecutar los siguientes comandos:
  - `sudo systemctl daemon-reload` (Recargar el servicio).
  - `sudo systemctl stop detector.service`.
  - `sudo systemctl disable detector.service`.
  - `sudo systemctl enable detector.service`.
  - `sudo systemctl start detector.service`.
- Al realizar todos los pasos se debe comprobar mediante el `systemctl` que todo funciona de manera correcta.