



Escuela de
Ingeniería y Arquitectura
Universidad Zaragoza



Universidad de Zaragoza

Escuela de Ingeniería y Arquitectura

Departamento de Ingeniería Electrónica y Comunicaciones

Proyecto Fin de Carrera

CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGÍA OPEN SOURCE

Autor

Víctor Lapuente Solórzano

Directores

Luis Antonio Martín Nuez

Jorge Casanova Dionisio

Ponente

Bonifacio Martín del Brío

Ingeniería Técnica Industrial, Electrónica Industrial

Curso 2014 – 2015

Convocatoria Marzo 2015



Escuela de
Ingeniería y Arquitectura
Universidad Zaragoza

**PROPUESTA y ACEPTACIÓN DEL
PROYECTO FIN DE CARRERA DE INGENIERÍA TÉCNICA**

DATOS PERSONALES

APELLIDOS, Nombre
LAPUENTE SOLÓRZANO, VÍCTOR

Nº DNI 73010036D Dirección C/ BERNA, Nº 13, 7º A

C.P. 50003 Localidad ZARAGOZA

Provincia ZARAGOZA Teléfono 976444841 NIA: 589370

Firma:

DATOS DEL PROYECTO FIN DE CARRERA

INGENIERIA TECNICA INDUSTRIAL, Especialidad ELECTRÓNICA INDUSTRIAL

TITULO CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGÍA OPEN SOURCE

DEPÓSITO EN: ZAGUAN (Obligatorio) y CD-ROM (si PFC es tipo B aplicación informática)

DIRECTOR BONIFACIO MARTÍN DEL BRÍO

VERIFICACIÓN EN SECRETARÍA

El alumno reúne los requisitos académicos (1) para la adjudicación de Proyecto Fin de Carrera

SELLO DEL CENTRO EL FUNCIONARIO DE SECRETARIA

Escuela de Ingeniería y Arquitectura Universidad Zaragoza

Fdo.:

<p>SE ACEPTA LA PROPUESTA DEL PROYECTO (2)</p> <p>En Zaragoza, a <u>3</u> de <u>FEB</u> de 2.0<u>15</u></p> <p></p> <p>Fdo.: <u>BONIFACIO MARTIN DEL BRIO</u></p> <p>DIRECTOR DEL PFC</p>	<p>SE ACEPTA EL DEPÓSITO DEL PROYECTO</p> <p>En Zaragoza, a <u>16</u> de <u>FEB</u> de 2.0<u>15</u></p> <p></p> <p>Fdo.: <u>BONIFACIO MARTIN DEL BRIO</u></p> <p>DIRECTOR DEL PFC</p>
---	---

(1) Requisitos académicos: tener pendientes un máximo de 24 créditos o dos asignaturas para finalizar la titulación.

(2) Para que la propuesta sea aceptada por el Director, es imprescindible que este impreso esté sellado por la Secretaría de la EINA una vez comprobados los requisitos académicos.



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGÍA OPEN SOURCE

Resumen

Este proyecto ha sido realizado en la empresa Libelium Comunicaciones Distribuidas S.L. El presente proyecto se encuentra englobado dentro del movimiento maker o “hazlo tú mismo” iniciado por Cooking Hacks, el departamento de Open Source de Libelium, con el fin de aportar un nuevo enfoque a la tecnología de Libelium, abriéndole las puertas al usuario particular al Internet de las Cosas a un bajo coste.

Su motivación surge tras la finalización del proyecto Open Aquarium, para descubrir las posibilidades de desarrollar aún más el producto integrando otros productos de Cooking Hacks.

El objeto del proyecto es el desarrollo de un sistema electrónico basado en el proyecto Open Aquarium que integre a la plataforma Raspberry, para poder monitorizar y controlar la información necesaria, recibida por diferentes sensores, para el mantenimiento de acuarios. Open Aquarium será el encargado de gestionar los datos proporcionados por los sensores activando los actuadores necesarios cuando llegue a alguno de los límites definidos en la programación, por ejemplo desactivar la bomba de CO₂ durante el día cuando el pH haya disminuido de un valor máximo prefijado evitando la posible asfixia de los peces, permitiendo al usuario activar alguno de los actuadores si todavía no ha llegado a uno de esos límites. Y a través de Raspberry el usuario podrá interactuar con Open Aquarium, pudiendo visualizar los datos de los sensores y utilizar los actuadores, sino están restringidos por alguna tarea de la automatización, a través de una pantalla táctil, y a su vez, se utilizará para que en una aplicación web el usuario pueda visualizar los mismos datos y un streaming del acuario.

Como resultado de este proyecto se ha obtenido un sistema de bajo coste, que sustituya a los caros dispositivos utilizados en el sector de la automatización de acuarios, permitiendo al usuario tener un control total sobre su acuario de un modo sencillo y eficaz.



Índice de contenido

1. Introducción	9
1.1 Marco de trabajo.....	9
1.2 Objeto y Motivación	12
1.3 Objetivos y Alcance	13
1.4 Desarrollo del proyecto.....	15
2. Campo de aplicación	17
2.1 Posibles aplicaciones del sistema	17
2.2 Antecedentes.....	18
2.3 Estado del arte.....	19
2.3.1 Sistemas comerciales.....	19
2.3.2 Sistemas Open Source.....	20
3. Especificaciones del sistema	21
3.1 La tecnología de partida: Arduino	21
3.1.1 Introducción a Arduino.....	21
3.1.2 Ventajas de Arduino.....	22
3.1.3 Aplicaciones de Arduino	23
3.2 La tecnología de control: Open Aquarium.....	25
3.2.1 Open Aquarium Aquaponics	27
3.3 La tecnología para la monitorización: Raspberry Pi.....	28
3.3.1 Introducción a Raspberry Pi	28
3.3.2 Especificaciones técnicas de Raspberry Pi.....	29
3.3.3 Aplicaciones Raspberry Pi.....	30
3.4 Parámetros de interés a medir	32
3.4.1 La dureza total (GH)	33
3.4.2 Carbonatos (KH)	34
3.4.3 Grado de acidez (pH)	35
3.4.4 Amoníaco, Nitrito y Nitrato (NH ₃ , NO ₂ , NO ₃).....	36
3.4.5 Temperatura del agua.....	37
3.4.6 Salinidad en acuarios marinos	38
3.5 Actuadores de interés	41
3.5.1 Sistema de cambios de agua	41
3.5.2 Alimentación de peces.....	41
3.5.3 Regleta controlada por RF	42
3.5.4 Inyección de CO ₂	42
3.5.5 Iluminación del acuario.....	42
3.5.6 Sistema de control de temperatura	44
3.6 Comunicación por puerto serie.....	45



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15



3.7 Protocolos de comunicación inalámbrica.....	46
3.7.1 Modulación por desplazamiento de frecuencia	46
3.7.2 Estándar 802.11	47
3.8 Decisión final	49
4. Hardware	50
4.1 Diagrama de bloques del sistema	50
4.2 Montaje en Arduino.....	51
4.2.1 Pinout en Arduino	51
4.2.2 Sensores	52
4.2.2.1 Sensor de nivel	52
4.2.2.2 Sensor de fugas de agua	53
4.2.2.3 Sensor de temperatura.....	53
4.2.2.4 Sensor de pH	54
4.2.2.5 Sensor de electro conductividad	54
4.2.3 Actuadores	55
4.2.3.1 Bombas de agua	55
4.2.3.2 Bombas peristálticas	55
4.2.3.3 Alimentador de peces.....	56
4.2.3.4 Regleta controlada por RF	56
4.2.3.5 Bomba de CO ₂	56
4.2.3.6 Lámpara LED	57
4.2.3.7 Ventilador	58
4.2.3.8 Calentador.....	58
4.3 Montaje en Raspberry	59
4.3.1 GPIOs Raspberry Pi	59
4.3.2 Pantalla LCD touchscreen.....	60
4.3.3 Cámara para Raspberry Pi.....	61
4.3.4 Adaptador USB 802.11 b/g/n	62
4.3.5 Caja soporte pantalla LCD.....	62
5. Diseño Software	64
5.1 Introducción	64
5.2 Software de base: Arduino IDE.....	64
5.3 Lenguaje de programación: C/C++	66
5.4 Librería Open Aquarium.....	67
5.5 Raspbian	68
5.6 Python	69
5.6.1 Interfaces gráficas	70
5.6.2 wxPython	71
5.6.3 Módulos y librerías implicadas	72



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15



5.6.4 Diseño final de la GUI.....	73
5.7 Configuración Raspberry Pi.....	75
5.8 PHP y HTML.....	77
5.9 Creación servidor web.....	78
5.10 Aplicación web.....	79
5.11 Diagramas de flujo.....	82
6. Pruebas realizadas	88
6.1 Pruebas cambios de agua.....	88
6.2 Pruebas del alimentador de peces.....	89
6.3 Pruebas comunicación Arduino/Raspberry.....	89
6.4 Pruebas regulación luminosidad.....	90
7. Conclusiones	92
7.1 Conclusiones personales.....	93
7.2 Agradecimientos.....	93
8. Glosario	94
9. Bibliografía	95
Anexo A: Código aplicación web	97
Anexo A.1 Códigos streaming.....	101
Anexo B: Documentación online generada	103



Índice de figuras

Fig. 1 Logos Open Source y Open Hardware	14
Fig. 2 Controlador para bombas de bajo voltaje comercial	18
Fig. 3 Controlador AT-control de Aqua Medic.....	19
Fig. 4 Montaje Jarduino	20
Fig. 5 Montaje Prototipo CAO1	20
Fig. 6 Cara TOP Arduino UNO.....	21
Fig. 7 Ejemplo de aplicación de Arduino para regular la luminosidad.....	22
Fig. 8 Arduino con OpenAquarium shield.....	25
Fig. 9 Cara Top y Bottom de Open Aquarium shield	26
Fig. 10 Cara Top Open Aquarium Aquaponics	27
Fig. 11 Placa Raspberry Pi B	28
Fig. 12 Raspberry Pi B+.....	30
Fig. 13 Tabla dureza general agua.....	33
Fig. 14 Grafica relación entre CO2, KH y pH.....	36
Fig. 15 Tabla relación entre conductividad y salinidad.....	40
Fig. 16 Plantas creciendo con iluminación artificial.....	43
Fig. 17 Calentador tradicional	44
Fig. 18 Ventilador electrónico acuario	45
Fig. 19 Pareja de emisor y receptor de RF FSK	46
Fig. 20 Ejemplo de modulación BFSK	47
Fig. 21 Logo Wi-Fi	47
Fig. 22 Tabla Estándares físicos	48
Fig. 23 Cara Top Wifi shield	48
Fig. 24 Shield de comunicaciones con módulo wifi RovingvRN-171	48
Fig. 25 Diagrama de bloques del sistema completo	50
Fig. 26 Sensor de nivel de Open Aquarium	52
Fig. 27 Sensor de fugas de Open Aquarium	53
Fig. 28 Sensor de temperatura de Open Aquarium	54
Fig. 29 Sensor de pH de Open Aquarium	54
Fig. 30 Sensor de electro conductividad de Open Aquarium	54
Fig. 31 Bomba de agua de Open Aquarium.....	55
Fig. 32 Bomba peristáltica de Open Aquarium.....	55
Fig. 33 Alimentador de peces de Open Aquarium.....	56
Fig. 34 Regleta RF de Open Aquarium	56
Fig. 35 Extintor de CO2	57
Fig. 36 Lámpara LED.....	57
Fig. 37 Ventilador de Open Aquarium	58



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15



Fig. 38 Calentador de Open Aquarium.....	58
Fig. 39 Entrada dimmer de la lámpara.....	59
Fig. 40 Disposición GPIOs Raspberry Pi B+.....	60
Fig. 41 TFT touchscreen.....	60
Fig. 42 Cámara Raspberry Pi	61
Fig. 43 Adaptador inalámbrico USB.....	62
Fig. 44 Diseño para caja pantalla.....	63
Fig. 45 IDE de programación de Arduino.....	65
Fig. 46 Símbolo Raspbian.....	68
Fig. 47 Logo Python.....	69
Fig. 48 Panel principal GUI	74
Fig. 49 Gráfica temperatura con la GUI	74
Fig. 50 Gráficas realizadas por la página ThingSpeak.....	79
Fig. 51 Vista de la página web.....	81
Fig. 52 Tabla funciones de la librería Open Aquarium.....	83
Fig. 53 Tabla de funciones de la clase TerminalFrame	86
Fig. 54 Tabla funciones de las clases TempGraphDialog y pHgraphDialog	87
Fig. 55 Variaciones del Duty Cycle	91
Fig. 56 Ejemplo de un paso de configuración de Raspberry del artículo	103



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15



1. Introducción

1.1 Marco de trabajo

Este Proyecto de Fin de Carrera (PFC) se ha realizado en colaboración con la empresa Libelium Comunicaciones Distribuidas S.L. bajo la dirección de Luis Antonio Martín Nuez y Jorge Casanova Dionisio, ingenieros técnicos en electrónica responsables del departamento Cooking Hacks, y bajo la supervisión de D. Bonifacio Martín del Brío como ponente, profesor titular del departamento de Ingeniería Electrónica y Comunicaciones de la Universidad de Zaragoza.

El proyecto se ha llevado a cabo íntegramente en las instalaciones de Libelium, tanto en el Centro Europeo de Empresas e Innovación de Aragón (CEEI) como en su siguiente y actual ubicación en el polígono El Pilar, desde julio de 2014 hasta febrero de 2015, bajo el formato de prácticas/proyecto en empresa de UNIVERSA.

Libelium es una empresa de diseño y fabricación de hardware para la implementación de redes sensoriales inalámbricas, redes malladas y protocolos de comunicación para todo tipo de redes inalámbricas distribuidas.

Nace en 2006 como empresa spin-off de la Universidad de Zaragoza. Sus productos se han vendido en más de 40 países y sus esfuerzos se han visto reconocidos con diversos premios. En ella se utiliza en gran medida el software libre y el firmware tiene una licencia pública GPL (General Public License). Entre los productos propios de Libelium cabe destacar los más relevantes:

- **Waspnote:** dispositivo sensorial para la creación de redes sensoriales inalámbricas (actuando como nodo) de bajo consumo.
- **Meshlium:** dispositivo router multitecnología que integra tecnología Wifi Mesh (2.4GHz - 5GHz), ZigBee, GPRS, GPS y Bluetooth.

Las redes sensoriales inalámbricas, también conocidas como WSN, son redes que buscan monitorizar el entorno. Están compuestas por pequeños ordenadores (nodos o motes), equipados con distintos tipos de sensores, que se coordinan para llevar a cabo una aplicación específica

Las redes de sensores están formadas por un grupo de sensores con ciertas capacidades sensitivas y de comunicación inalámbrica los cuales permiten formar redes *ad hoc* sin infraestructura física preestablecida ni administración central. Un sensor es capaz de procesar una



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

limitada cantidad de datos. Pero cuando coordinamos la información entre un importante número de nodos, éstos tienen la habilidad de medir un medio físico dado con gran detalle.

Las redes de sensores tienen una serie de características propias y otras adaptadas de las redes Ad-Hoc:

- Topología Dinámica: En una red de sensores, la topología siempre es cambiante y éstos tienen que adaptarse para poder comunicar nuevos datos adquiridos.
- Variabilidad del canal: El canal radio es un canal muy variable en el que existen una serie de fenómenos como pueden ser la atenuación, desvanecimientos rápidos, desvanecimientos lentos e interferencias que puede producir errores en los datos.
- No se utiliza infraestructura de red: Una red sensora no tiene necesidad alguna de infraestructura para poder operar, ya que sus nodos pueden actuar de emisores, receptores o enrutadores de la información. Sin embargo, hay que destacar en el concepto de red sensora la figura del nodo recolector (también denominados sink node), que es el nodo que recolecta la información y por el cual se recoge la información generada normalmente en tiempo discreto. Esta información generalmente es adquirida por un ordenador conectado a este nodo y es sobre el ordenador que recae la posibilidad de transmitir los datos por tecnologías inalámbricas o cableadas según sea el caso.
- Tolerancia a errores: Un dispositivo sensor dentro de una red sensora tiene que ser capaz de seguir funcionando a pesar de tener errores en el sistema propio.
- Comunicaciones multisalto o broadcast: En aplicaciones sensoras siempre es característico el uso de algún protocolo que permita comunicaciones multi-hop, léase AODV, DSDV, EWMA u otras, aunque también es muy común utilizar mensajería basada en broadcast.
- Consumo energético: Es uno de los factores más sensibles debido a que tienen que conjugar autonomía con capacidad de proceso, ya que actualmente cuentan con una unidad de energía limitada. Un nodo sensor tiene que contar con un procesador de consumo ultra bajo así como de un transceptor radio con la misma característica, a esto hay que agregar un software que también conjugue esta característica haciendo el consumo aún más restrictivo
- Limitaciones hardware: Para poder conseguir un consumo ajustado, se hace indispensable que el hardware sea lo más sencillo posible, así como su transceptor radio, esto nos deja una capacidad de proceso limitada.
- Costes de producción: Dada que la naturaleza de una red de sensores tiene que ser en



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

número muy elevada, para poder obtener datos con fiabilidad, los nodos sensores una vez definida su aplicación, son económicos de hacer si son fabricados en grandes cantidades.

Las redes de sensores tienen una amplia variedad de aplicaciones:

- Monitorización de un hábitat (para determinar la población y comportamiento de animales y plantas).
- Monitorización del medio ambiente, observación del suelo o agua.
- El mantenimiento de ciertas condiciones físicas (temperatura, luz).
- Control de parámetros en la agricultura.
- Detección de incendios, terremotos o inundaciones.
- Control del hogar y automatización de edificios.
- Control de tráfico.
- Asistencia militar o civil.
- Control de inventario.
- Control médico.
- Detección acústica
- Cadenas de montaje, etc.

Entre sus ventajas destacar:

- Sin cables
- Auto-organización
- Escalabilidad
- Robustez
- Fiabilidad
- Seguridad
- Mínimo mantenimiento



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

1.2 Objeto y Motivación

El objeto del proyecto es el desarrollo de un sistema electrónico basado en el proyecto Open Aquarium que integre a la plataforma Raspberry, para poder monitorizar y controlar la información necesaria, recibida por diferentes sensores, para el mantenimiento de acuarios. Open Aquarium será el encargado de gestionar los datos proporcionados por los sensores activando los actuadores necesarios cuando llegue a alguno de los límites definidos en la programación, como apagar la bomba de CO₂ si siendo de día el pH ha disminuido demasiado, permitiendo al usuario activar alguno de los actuadores si todavía no ha llegado a uno de esos límites. Y a través de Raspberry el usuario podrá interactuar con Open Aquarium, pudiendo visualizar los datos de los sensores y utilizar los actuadores, sino están restringidos por alguna tarea de la automatización, a través de una pantalla táctil, y a su vez, se utilizará para que en una aplicación web el usuario pueda visualizar los mismos datos y un streaming del acuario.

El proyecto abarca multitud de campos y tareas a realizar, desde la investigación de entornos gráficos para Python, hasta el estudio del desarrollo de una aplicación web de monitorización.

Para ello comenzaré primero con una investigación de las diferentes alternativas de estos toolkits de Python para la creación de interfaces gráficas. Tras ello estudiaré el Shield Open Aquarium para Arduino, con el objetivo de familiarizarme con su Hardware (sensores y actuadores), y sus librerías. Se continuará con distintas pruebas de comunicación entre Arduino y Raspberry utilizando la GUI (Graphical User Interface) escogida, mediante comunicación por puerto serie. Finalizando, con el diseño final de la GUI y de una aplicación web de monitorización de los datos de los sensores y del acuario a través de una cámara.

El presente proyecto se encuentra englobado dentro del movimiento maker o “hazlo tú mismo” iniciado por Cooking Hacks, el departamento de Open Source de Libelium, con el fin de aportar un nuevo enfoque a la tecnología de Libelium, abriéndole las puertas al usuario particular al Internet de las Cosas a un bajo coste. El principal objetivo de la empresa es el de ampliar el campo de aplicación de su tecnología, pero manteniendo los niveles de rendimiento del sistema (bajo consumo, velocidad de transmisión...).

Desde hace mucho tiempo a la gente le gusta tener y cuidar su propio y diferente tipo de acuario/terrario, y ha querido tenerlo todo bien controlado para evitar problemas que puedan dañar el acuario y así no tener que estar totalmente pendiente de su cuidado, por lo tanto se ha buscado utilizar la tecnología actual para poder automatizar personalmente las labores de cuidado de los



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

acuarios.

En vista de que los dispositivos electrónicos comerciales actuales de control de acuarios tienen unos altos costes y son solo accesibles a personal cualificado, se busca la creación de un sistema que permita realizar esta tarea de manera libre, sencilla y en tiempo real, aportándole también la visión de todos los datos y del propio acuario a través de la web, todo ello sin un precio elevado, y permitiendo al usuario una total libertad de programación adaptada a sus necesidades de un modo sencillo y eficaz.

1.3 Objetivos y Alcance

El objetivo principal de este PFC es el desarrollo, construcción, programación y puesta a punto de un sistema electrónico que incorpore la posibilidad de transmisión de datos de los principales parámetros a medir en un acuario a una unidad central que almacene esta información y que tome decisiones de forma autónoma activando diversos actuadores. El usuario podrá consultar esta información en tiempo real y a través de la web, y activar o desactivar actuadores manualmente, por ejemplo la iluminación del acuario.

Podemos distinguir también una serie de objetivos secundarios relacionados con la finalización del proyecto:

- Comprender en profundidad todo el *hardware* utilizado (*datasheet*, documentación técnica...)
- Conocer el lenguaje de programación C-C++ y Processing/Wiring siendo estos la base de todos los códigos realizados dentro del entorno de desarrollo de Arduino.
- Conocer el lenguaje de programación Python siendo la base del script que se realizará en la plataforma Raspberry Pi.
- Investigar los diferentes tipos de toolkits para la creación de interfaces gráficas que hay para Python.
- Desarrollo del código de comunicación entre Arduino y Raspberry utilizando la GUI elegida.
- Estudio y creación de la Raspberry como servidor Apache en aras de crear una aplicación web de monitorización del acuario.

Gracias a que la tecnología que se va a utilizar se basa prácticamente toda en “open hardware-software”, la curva de aprendizaje es exponencial, lo que facilita notablemente la



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

investigación.

En estos últimos años, la tecnología basada en hardware y software libre, como por ejemplo Arduino, ha iniciado un movimiento muy importante entre la sociedad electrónica, propulsado por las numerosas ventajas que este tipo de dispositivos facilitan. Las personas comparten sus avances con el resto de la comunidad, lo que permite tener una referencia continua para la realización de distintos proyectos.

Entre este tipo de tecnologías destaca la desarrollada por Libelium bajo licencia Open Source, la cual está en continuo desarrollo y mejora. El término Open Source se acuñó específicamente para el desarrollo de software e implicaba que un programador diera a conocer el código para cierto programa, aplicación, etc. de manera que otros programadores pudieran utilizarlo e implementarlo en sus propios desarrollos. Esto derivó también a lo que llamamos Open Hardware que son aquellos dispositivos electrónicos cuyas especificaciones y diagramas esquemáticos son de acceso público.

Este proyecto aporta un nuevo valor añadido a dicha tecnología, puesto que es la continuación del proyecto Open Aquarium, permitiendo en cierta medida introducir el hardware y software libre al usuario particular, además de acercar las ventajas que proporciona la comunicación inalámbrica de información.

Gracias a esto, se espera que un gran número de usuarios particulares que encuentren problemas tanto de costes como de otros tipos (conocimiento especializado, mantenimiento...) para la adquisición de equipos de automatización y control de acuarios, vean en este sistema una opción económica para su realización.

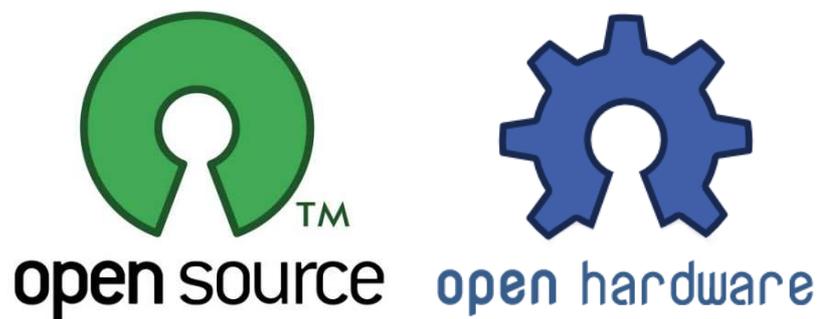


Fig. 1 Logos Open Source y Open Hardware



1.4 Desarrollo del proyecto

El desarrollo del proyecto se ha dividido en tres fases principalmente:

- En primer lugar se han estudiado las diferentes tecnologías que intervienen en el proyecto: los parámetros del acuario que puede medir el shield Open Aquarium para la plataforma Arduino, los distintos sensores y actuadores que utiliza este shield, la tecnología base sobre la que se ha trabajado (Arduino), la plataforma Raspberry y su posible comunicación con Arduino, el lenguaje de programación y el entorno de programación que se han utilizado (C-C++, IDE Arduino, Python ...), las distintas posibilidades de hardware a utilizar y todos los demás temas relacionados en menor medida con el proyecto.
- En segundo lugar se ha ido implementado un prototipo, gracias a toda la investigación previa, a la par que se han ido realizando pruebas de funcionamiento de los diversos módulos de ese prototipo (recepción correcta de datos, posibilidad de actuar desde la GUI sobre los actuadores, gráficas en tiempo real...)
- Por último, se ha revisado el prototipo para realizar una versión final, y se han llevado a cabo todas las pruebas necesarias para evaluar sus prestaciones y su correcto funcionamiento como sistema autónomo. En paralelo a estas fases se ha ido realizando una memoria de proyecto, en la que se toma nota de todos los avances e investigaciones. Esta memoria ha sido la base de la actual documentación del proyecto.

Se ha llevado a cabo un seguimiento de la búsqueda de información y de la ejecución del proyecto, tanto por parte de la universidad, como por parte de la empresa.

Estas son las dos vías distintas de seguimiento que se han llevado a cabo:

- Por medio de las distintas reuniones con el Dr. D. Bonifacio Martín del Brío, ponente y director del proyecto, profesor de la Universidad de Zaragoza.

En dichas reuniones se han tratado los siguientes temas:

- Inicio de PFC.
- Definición del proyecto.
- Informes de avances.



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

- Revisión y modificación de documentos.
- Dudas generales.
- En la empresa, diariamente, se han presentado los avances en el proyecto en reuniones con Jorge Casanova y Luis Martín y aproximadamente cada mes con D. David Gascón. Esto ha permitido un avance continuo, sin parones que lo ralenticen demasiado.

Los principales problemas tratados han sido:

- Problemas en la investigación.
- Problemas en algún dispositivo.
- Fallos de programación.
- Maquetación de la instalación del producto.



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

2. Campo de aplicación

2.1 Posibles aplicaciones del sistema

El proyecto se enfoca al uso de particulares que deseen monitorizar en todo momento y automatizar el cuidado de su acuario, pudiendo ser un acuario de agua dulce, marino, plantado o acuaterrario, debido a que mide los parámetros básicos de cualquier tipo de acuario y permite un cambio fácil de los rangos de estos parámetros para cada uno de estos tipos de acuario en la automatización.

Los beneficios del sistema son entre otros:

- Monitorización exhaustiva del acuario.
- Conocimiento del estado de las plantas, los peces y sus necesidades.
- Permite un análisis global gracias a la creación de gráficos con los datos recogidos.
- Automatización de los actuadores.
- Acceso instantáneo y remoto a datos gracias a la implementación de una aplicación web.
- Ahorro de agua y de tratamientos de salud de las plantas y los peces.
- Comparación de datos con meses o años anteriores.
- Mínimo mantenimiento.
- Flexibilidad y total posibilidad de personalización.
- Precio reducido.

2.2 Antecedentes

Aunque en la antigüedad ya hubo la aparición de algunos acuarios, es desde el siglo XIX cuando esta creación de acuarios se hizo poco a poco muy popular. Pero no fue hasta después de la Primera Guerra Mundial que los acuarios llegaron a las casas debido a la aparición del suministro eléctrico. La electricidad permitió la iluminación artificial, así como la aireación, la filtración y el calentamiento del agua.

El cuidado de estos acuarios se realizaba con sistemas de control basados en circuitos eléctricos con relés, interruptores y otros componentes comúnmente utilizados para el control de los sistemas de lógica combinatorial, hasta que en las últimas décadas cuando la industria buscó en las nuevas tecnologías electrónicas una solución más eficiente, se recurrió a los PLC (controlador lógico programable), que buscaban controlar procesos secuenciales en tiempo real. Los primeros controladores para acuarios nacieron uniendo estos PLC con algunas bombas que controlaban los cambios de agua.



Fig. 2 Controlador para bombas de bajo voltaje comercial

Pero los controladores de acuarios actuales suponen una gran inversión económica o tienen escasas prestaciones, es decir, dejan muy pocas posibilidades de programación al usuario, y además la mayoría de ellos solo se ocupan de unas pocas tareas de automatización. La fabricación de un sistema que no solo permitiera el control de los principales parámetros característicos de los acuarios, sino que lo aunara con la supervisión a través de la red, y todo ello a un precio asequible, supondría un dispositivo con mucho potencial y una gran oportunidad en el mercado.

2.3 Estado del arte

2.3.1 Sistemas comerciales

Existen controladores de acuarios de diversas empresas como Aqua Medic o Profilux que permiten medir y controlar pH y temperatura, regular la iluminación, y la conexión con ordenadores con el fin de guardar datos y realizar gráficas



Fig. 3 Controlador AT-control de Aqua Medic

Los programadores más complejos de la misma gama suelen permitir el conexionado con otros módulos que midan y controlen el resto de parámetros que interesen al usuario, o incluso, es el mismo controlador el que te permite la medida de todos los datos. Estos controladores son muy caros, y aun así, se suelen vender sin algunos sensores, y sin los actuadores, que suelen tener que estar conectados a una regleta en la que ya se ha definido su posición.

Por todo ello este proyecto se descarta de estos sistemas comerciales porque permite hacer lo que estos hacen además de ofrecer muchas otras ventajas a un precio menor. El hecho de apoyarse en plataformas Open Source como Arduino y Raspberry, permite la completa programación por parte del usuario, la creación de bases de datos y gráficas, la monitorización a través de una aplicación web, o el control de la iluminación del acuario, por ejemplo, entre una infinidad de posibilidades.

2.3.2 Sistemas Open Source

Existen varios proyectos Open Source disponibles en internet relacionados con la automatización de acuarios, por ejemplo:

- El proyecto Jarduino está basado en la plataforma Arduino, se centra en la medición y control de temperatura, y en vez de medir otros parámetros ha buscado una mejora estética para el acuario y unas condiciones más parecidas al entorno en el que viven los peces, en este caso el mar porque usa un generador de olas (bomba de agua), luz lunar... Este proyecto usa un Arduino Mega, debido a la necesidad de más pines ya que quiere controlar los actuadores por una pantalla táctil, propia para Arduino. Para utilizar esta se necesita del shield ITDB02 que se encarga de la interfaz gráfica.

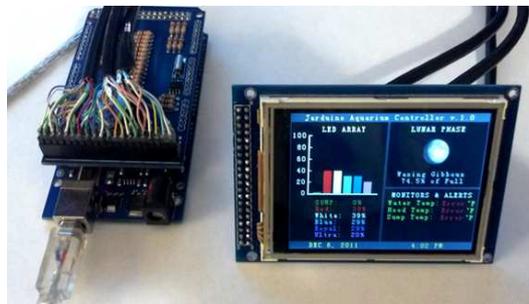


Fig. 4 Montaje Jarduino

- Otro proyecto basado en Arduino es el proyecto CAO (Control de Acuarios por Ordenador). En este caso hay sensores tanto de temperatura como de pH, un controlador de luminosidad (también fase lunar) y una alarma por fallos. En un LCD controla varios métodos de funcionamiento: chequeo, monitorización, menú de cambios y funcionamiento en alarma.

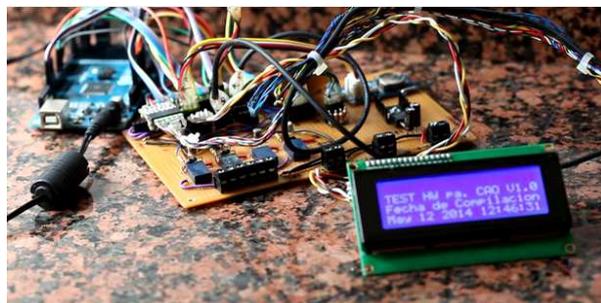


Fig. 5 Montaje Prototipo CAO1

3. Especificaciones del sistema

3.1 La tecnología de partida: Arduino

3.1.1 Introducción a Arduino

Arduino es una plataforma de hardware libre (Open Source), basada en una placa con un microcontrolador y un entorno de desarrollo, diseñada para facilitar el uso de la electrónica en proyectos multidisciplinarios. El hardware consiste en una placa con un microcontrolador Atmel AVR y puertos de entrada/salida (E/S) analógicas y digitales que incluye un entorno de desarrollo integrado (IDE) basado en Processing que incluye soporte para los lenguajes de programación C++.

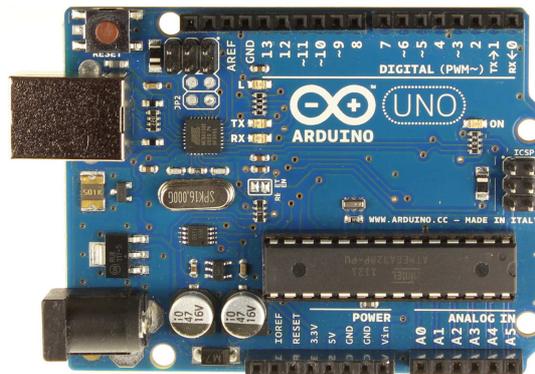


Fig. 6 Cara TOP Arduino UNO

Arduino se inició en el año 2005 como un proyecto para estudiantes en el Instituto IVREA, en Ivrea (Italia) y los líderes del proyecto trataron de proporcionar una forma económica y fácil para los aficionados, estudiantes y profesionales de crear dispositivos que interactúan con su entorno mediante sensores y actuadores. Su propósito fue es de ser una placa de desarrollo de fácil programación a través de lenguajes de muy alto nivel, en comparación con otras plataformas como PIC, que resultaba muy complicada para los iniciados y nuevos estudiantes. Pero Arduino ha trascendido más allá y hoy en día y no sólo se usa para prototipado, sino que incluso es una placa sobre la que desarrollan pequeños proyectos para empresas y entidades. Se han diseñado muchos módulos pensados para pinchar directamente sobre Arduino para aumentar la funcionalidad, como diversos módulos de comunicaciones inalámbricas que han hecho que Arduino se haya ganado un hueco muy importante en el mundo de la electrónica.

La placa más usada de Arduino, Arduino UNO R3, Cuenta con 14 pines digitales de

entrada / salida (de los cuales 6 pueden utilizarse para salidas PWM), 6 entradas analógicas con conversores AD, un oscilador de 16MHz, una conexión USB, un conector de alimentación, una cabecera ICSP, y un botón de reinicio. Además tiene salidas de alimentación de 5 y 3,3V. Contiene todo lo necesario para apoyar el microcontrolador; basta con conectarlo a un ordenador con un cable USB, un adaptador de CA a CC o una pila externa que se conecta al jack de alimentación.

Gracias a las librerías de Arduino, leer un simple sensor que requiere de configuración de muchos registros del micro-controlador, se reduce a una única llamada a una función que nos devuelve el valor de la entrada analógica y sin necesidad de tener que recurrir al datasheet del micro-controlador.

Otra ventaja es que el diseño de la placa Arduino está disponible en Internet, con que cualquier persona puede construirla en su casa de forma independiente. Todos los ficheros CAD y las librerías de programación y bootloaders están totalmente disponibles de forma gratuita.

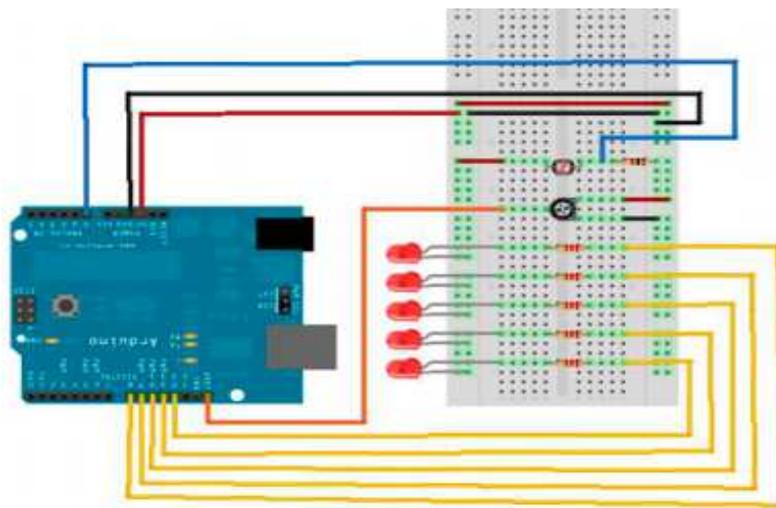


Fig. 7 Ejemplo de aplicación de Arduino para regular la luminosidad

3.1.2 Ventajas de Arduino

Aunque hay una gran variedad de micro-controladores y plataformas de desarrollo, son difíciles de usar para personas con pocos conocimientos de electrónica y programación. Arduino ha conseguido convertirse en la placa preferida para el desarrollo de prototipos y proyectos, tanto para aficionados como para profesionales del sector. Las ventajas que presenta Arduino son las siguientes.

- Open Source: Arduino es una plataforma de código y hardware abierto, es decir, puedes



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

acceder a todo aspecto del funcionamiento circuital y algorítmico de las placas, y mucho mejor que eso, te dan todos los archivos Eagle, diagramas y componentes para que tú mismo crees tú versión de Arduino.

- Fácil de programar: Arduino te ofrece un entorno de desarrollo integrado (IDE) con funciones preestablecidas que reducen la lógica a lectura de entradas, control de tiempos y salidas de una manera semántica e intuitiva. Por eso lo convierte en una herramienta perfecta para los que quieran iniciarse en el mundo de la electrónica y luego, probar cosas más difíciles.
- Documentación y tutoriales en exceso: Si algo tiene Arduino es que Internet está inundado literalmente de documentación sobre esta plataforma, así que decir que no se encuentran tutoriales para aprender no es una excusa válida para este caso. Desde la misma página de la empresa, el IDE que también viene con multitud de ejemplos y los incontables tutoriales en Youtube y la red sobre esta plataforma la hacen una de las más fáciles de desarrollar.
- Librerías: Una de las ventajas más grandes que tiene Arduino es que poseen librerías para prácticamente cualquier componente externo que se le quiera acoplar haciendo innecesario aprenderse el datasheet del componente y desarrollar el software necesario.
- Precio: El precio es uno de los elementos más atractivos de Arduino. Hay placas de Arduino por precios muy bajos, desde los 18€ que vale la placa Arduino Leonardo, hasta placas más potentes como Arduino YÚN con un precio de 52€. La placa que más se usa es Arduino UNO tiene un precio de 20€.

3.1.3 Aplicaciones de Arduino

Gracias a su versatilidad, Arduino se ha vuelto un proyecto muy popular en la red y es la base sobre la que se construyen y desarrollan un buen número de proyectos relacionados con los microcontroladores y la electrónica en general. Entre sus aplicaciones más conocidas están:

- Control y monitoreo de sensores.
- Efectos con leds.
- Transmisores/receptores.
- Educación.



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

- Monitoreo ambiental.
- Biomedicina.
- Telemedicina
- Domótica

Para realizar todos estos proyectos se ha creado alrededor de Arduino una serie de comunidades que se encargan de diseñar placas (shields) para dar a la placa muchas más funcionalidades. Algunos de los shields más importantes son:

- Shield Xbee: permite a la placa Arduino comunicarse inalámbricamente mediante el protocolo ZigBee.
- Shield GSM: conecta Arduino a Internet utilizando la red inalámbrica GPRS.
- Shield Ethernet: proporciona a Arduino capacidad para conectarse a Internet mediante un cable RJ-45.
- Shield GPS: permite la geolocalización por GPS de nuestro Arduino.

Muchas otras placas se han desarrollado por empresas privadas (como es el caso de Open Aquarium por Libelium) o por desarrolladores aficionados lo que hace que la competitividad entre ellos por dar nuevas funcionalidades a Arduino sea constante. Esto contribuye a una continua publicación de proyectos y nuevos desarrollos en la red.

3.2 La tecnología de control: Open Aquarium

Para la realización de este proyecto voy a apoyarme en la placa Open Aquarium, creada y distribuida por el departamento Cooking Hacks de Libelium Comunicaciones Distribuidas S.L.

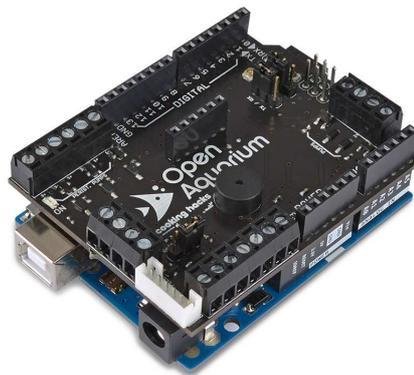


Fig. 8 Arduino con OpenAquarium shield

Esta placa es totalmente compatible con Arduino Uno, y ha sido diseñada para facilitar el uso de diversos sensores y actuadores que pueden ayudar en el mantenimiento de los acuarios y estanques. Y contiene unas librerías de fácil manejo con las que a través del IDE de Arduino llamar de un modo sencillo a las funciones que se precisen para interactuar con los sensores o con los actuadores. Características:

- Eléctricas: Open Aquarium debe ser alimentado por una fuente de alimentación externa de 12V-2A. Además varios de los dispositivos que utiliza han sido diseñados para trabajar con una tensión de 220V.
- Comunicación inalámbrica: puede comunicarse con una regleta controlada por RF gracias a la capacidad de conexión de un módulo RF siendo está comunicación de modulación FSK (modulación por desplazamiento de frecuencia) a 433MHz.
- Conexionado: esta placa tiene incorporado un buzzer que sirve de indicación sonora y una célula RTC conectada por I2C para realizar todas aquellas tareas que necesiten un tiempo específico para realizarse. Y dispone de varios conectores que facilitan la conexión entre los dispositivos y el gateway:
 - 1 conector para un alimentador de peces
 - 1 conector para un sensor de temperatura

- 1 conector para una lámpara
- 1 conector para el sensor de goteo
- 2 conectores para sensores de nivel
- 1 conector para módulo RF
- 2 conectores para bombas de agua
- 3 conectores para bombas peristálticas
- 1 conector para el módulo de Aquaponics, con el que se pueden obtener valores de pH y conductividad.

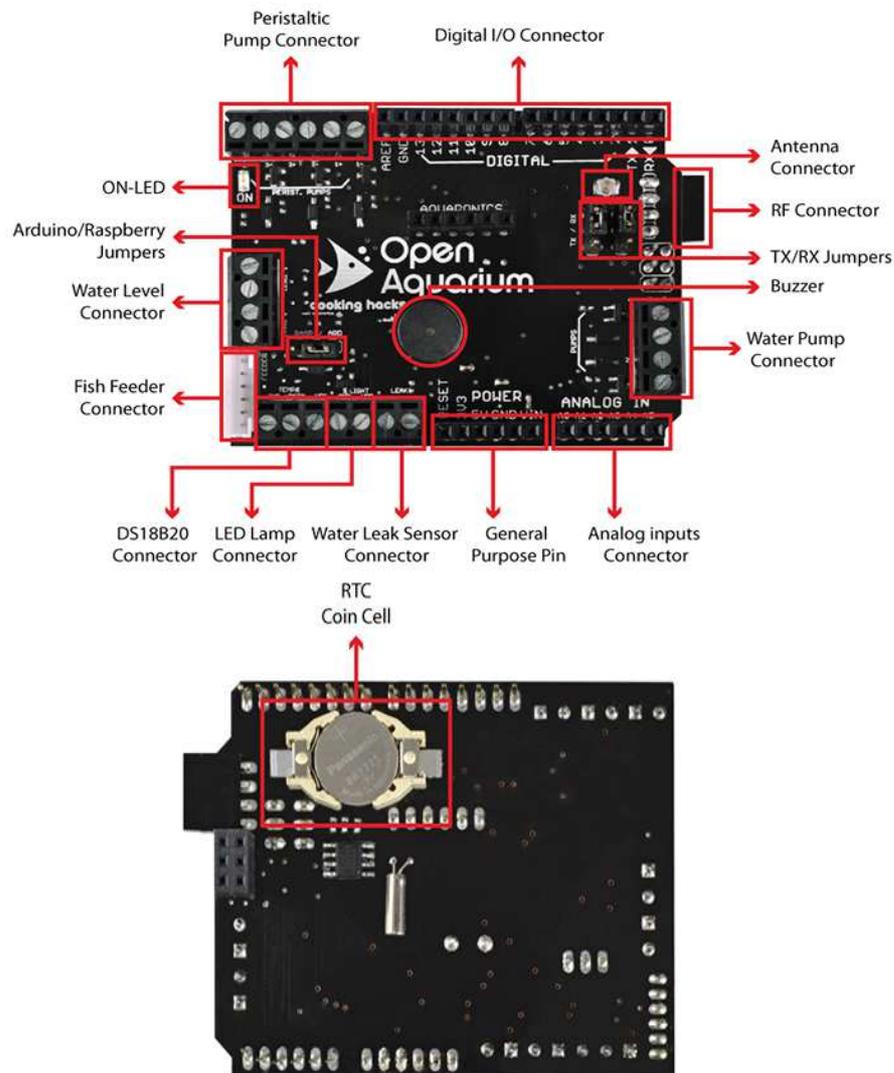


Fig. 9 Cara Top y Bottom de Open Aquarium shield

3.2.1 Open Aquarium Aquaponics

Open aquarium Aquaponics es un módulo creado y distribuido por el departamento Cooking Hacks de Libelium Comunicaciones Distribuidas S.L., para el cuidado de sistemas acuapónicos, que son aquellos sistemas que combinan los métodos de la hidroponía, cultivo de plantas en agua, con la acuicultura, cría de animales acuáticos, para producir cultivos orgánicos extraordinariamente eficientes.

Este módulo necesita obligatoriamente estar conectado del shield Open Aquarium para poder funcionar, y tiene una conexión BNC para un sensor de pH y un conector para cables para un sensor de electro-conductividad.

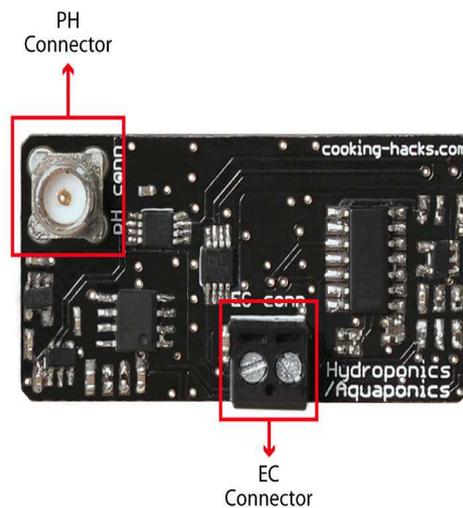


Fig. 10 Cara Top Open Aquarium Aquaponics

3.3 La tecnología para la monitorización: Raspberry Pi

3.3.1 Introducción a Raspberry Pi

Raspberry Pi es una es un ordenador de placa reducida o placa única (SBC) de bajo coste desarrollada en Reino Unido por la Fundación Raspberry Pi, con el objetivo de estimular la enseñanza de ciencias de la computación en las escuelas.



Fig. 11 Placa Raspberry Pi B

Es un proyecto "con ADN abierto", pensado para compartir y producir conocimiento, que se convierte en un acontecimiento comercial. Un triunfo de la cultura libre que reúne licencias GPL, Creative Commons, con software y hardware libre, inspirado en grandes como Linux y Arduino...

El diseño incluye un System-on-a-chip (SoC) Broadcom BCM2835, que contiene un procesador central (CPU) ARM1176JZF-S a 700 MHz (el firmware incluye unos modos Turbo para que el usuario pueda hacerle overclock de hasta 1 GHz sin perder la garantía), un procesador gráfico (GPU) VideoCore IV, y 512 MB de memoria RAM aunque originalmente al ser lanzado eran 256 MB. El diseño no incluye un disco duro o una unidad de estado sólido, ya que usa una tarjeta SD para el almacenamiento permanente; tampoco incluye fuente de alimentación o carcasa.

La fundación da soporte para las descargas de las distribuciones para arquitectura ARM, Raspbian (derivada de Debian), RISC OS 5 y Pidora (derivado de Fedora); y promueve principalmente el aprendizaje del lenguaje de programación **Python**, y otros lenguajes como Tiny BASIC, C y Perl.

Otra ventaja es que los esquemáticos de Raspberry PI están disponibles en internet por lo que cualquier persona podría fabricar la placa en su casa de forma independiente. Por otra parte el precio de las placas es bastante asequible para la gran funcionalidad que tienen, valiendo el nuevo modelo A+ unos 20 € y el modelo B+ unos 35 €.



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

3.3.2 Especificaciones técnicas de Raspberry Pi

Hay dos modelos de Raspberry Pi, A y B, teniendo el modelo A unas características más básicas pero aun así muy completas. Las diferencias entre estos dos modelos son sus memorias RAM, ya que el modelo A tiene 256 MB y el modelo B 512 MB, el número de puertos USB, 1 para el modelo A y 2 para el B, la capacidad de conectividad que tiene el modelo B gracias a su puerto Ethernet, y el consumo energético de 500 mA del modelo A por 700 mA del modelo B.

Además de lo descrito anteriormente tienen una serie de funcionalidades comunes que hacen que Raspberry sea un dispositivo tremendamente eficiente para el desarrollo de proyectos de todo tipo:

SoC:	Broadcom BCM2835
CPU:	ARM 1176JZF-S a 700 MHz
GPU:	Videocore 4
Video:	HDMI, RCA e Interfaz DSI para panel LCD
Audio:	HDMI y 3.5 mm
Alimentación:	5V vía Micro USB o GPIO header
Entradas/salidas:	8 GPIO propósito general, 1 UART, 1 SPI, 1 bus I ² C
Almacenamiento:	Tarjeta SD

La Raspberry Pi no viene con reloj en tiempo real, por lo que debe usar un servidor de hora en red, o pedir al usuario la hora en el momento de arrancar el ordenador. Sin embargo se podría añadir un reloj en tiempo real (como el DS1307) con una batería mediante el uso de la interfaz I²C.

A finales del año pasado se hicieron unas revisiones de estos modelos renombrándolos por modelo A+ y B+. Estas revisiones modificaron el sistema de almacenamiento, sustituyéndolo por tarjetas micro SD, se mejoró el audio, se eliminó el conector RCA haciendo al conector de 3,5 mm un conector de audio/video y en su lugar añadieron más GPIOs (General Purpose Input/Output), disminuyeron el consumo en 0,5W y en el caso del modelo B+ se añadieron 2 puertos USB.



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15



Fig. 12 Raspberry Pi B+

Este mismo mes acaba de salir al mercado Raspberry Pi 2 con un procesador central Quad-core ARM Cortex-A7 a 900 MHz, 1 GB de SDRAM y es totalmente compatible con Raspberry Pi 1. Además, debido a que tiene un procesador ARMv7, puede usar la gama de distribuciones ARM GNU / Linux, incluyendo Snappy Ubuntu Core, así como Microsoft Windows 10.

Raspberry Pi usa mayoritariamente sistemas operativos basados en el núcleo Linux. Raspbian una distribución derivada de Debian que está optimizada para el hardware de Raspberry Pi es la distribución recomendada por la fundación para iniciarse. Otros sistemas que da soporte la fundación son Pidora basado en Fedora que es otra distribución en base linux, Risc OS que es un sistema operativo con núcleo o Kernel propio, distinto a los kernels más comunes como Linux y Windows, y RaspBMC y OpenELEC que son sistemas operativos orientados en hacer de la Raspberry un centro multimedia.

3.3.3 Aplicaciones Raspberry Pi

La versatilidad que ofrece este dispositivo basado en Creative Commons y en GPL ha proporcionado una plataforma perfecta para que los aficionados a la programación den rienda suelta a su creatividad proponiendo interesantes y divertidas aplicaciones de Raspberry Pi. Algunos proyectos que se pueden destacar son:

- Tener un miniordenador: conectándole una pantalla, un teclado y un ratón es posible utilizar todo el potencial de la placa: navegar, programar e incluso jugar.
- Acceder a tu ordenador: con el programa adecuado, por ejemplo usando VNC, es posible establecer conexión remota con otros ordenadores, sólo habría que instalar el servidor en el ordenador que quieras acceder y un cliente en tu Raspberry.



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

- Centro Multimedia: gracias a su salida HDMI y a que tiene la potencia suficiente para reproducir vídeo lo convierte en el complemento ideal para transformarlo en un media Center de coste reducido. La forma más común es recurrir directamente a XBMC o a través de distribuciones Linux preparadas para Raspberry Pi que ya lo integran como OpenELEC.
- Consola de juegos arcade: hay diversos simuladores para Raspberry y si se conectan un joystick y unos botones a los GPIOs, integrándolo todo en una carcasa curiosa, es posible tener una máquina recreativa.
- Proyectos robóticos: existen placas de expansión que permiten, además de domotizar la casa, construir pequeños proyectos robóticos.

Existen diversos accesorios contruidos específicamente que han ayudado a la realización de diversos proyectos, entre estos podemos destacar la Picamera que permite la toma de fotos y videos con una gran resolución a un precio muy reducido, Pi Noir que es la versión infrarroja de la cámara, y Getboard que es un módulo para ampliar el número de GPIOs para el control de LED, interruptores, señales analógicas, sensores y otros dispositivos.



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15



3.4 Parámetros de interés a medir

El fin último de este proyecto es permitir al usuario comunicarse con su acuario en las dos direcciones, es decir, el acuario nos dará información en tiempo real pudiendo actuar de forma automática o manual si se considera que hay que actuar sobre algún parámetro, pero el sistema no ha actuado al no haber llegado todavía a alguno de los límites establecidos.

Todo esto nos permite mantener siempre a los peces y/o a las plantas con un nivel óptimo de temperatura y nutrientes, o evitar posibles enfermedades de las plantas o peces.

Los sensores son dispositivos capaces de detectar magnitudes físicas del entorno y transformarlas en variables eléctricas con las que podemos trabajar. Estas magnitudes físicas pueden ser por ejemplo: temperatura, intensidad lumínica, distancia, aceleración, inclinación, desplazamiento, presión, fuerza, torsión, humedad, movimiento, pH, etc. Un sensor está siempre en contacto con la magnitud física a medir con lo que puede decirse también que es un dispositivo que aprovecha una de sus propiedades con el fin de adaptar la señal que mide para que la pueda interpretar otro dispositivo.

Respecto a los acuarios, cada tipo de acuario tiene unas preferencias en cuanto a los factores externos, pero aun así se ha de conocer las características de los peces y plantas que habitan en el acuario para el perfecto cuidado del mismo.

Respecto a las peces, cada especie tiene unas preferencias en cuanto a los factores externos. La mayoría de peces puede tolerar condiciones ambientales que difieren de las condiciones naturales en la cuales han evolucionado. Sin embargo, esto no significa que van a estar tan sanos o que van a vivir tanto como normalmente. Por ejemplo, tener un pez en una agua más fría (o caliente) que la que son su condiciones preferidas obligan a los órganos del cuerpo a trabajar más para mantenerlo vivo. Esto es, estas condiciones sitúan el pez bajo estrés.

Un estrés aumentado reduce la capacidad del pez para protegerse de enfermedades y de curarse el mismo. Además, el estrés reduce la capacidad de los peces de criar con éxito y acorta su esperanza de vida natural. Un poco de estrés por sí solo no suele ser fatal, pero a medida que el estrés aumenta su capacidad para resistirlo disminuye.

Se debería anotar que eliminar el estrés no garantiza que el acuario va a estar sano. Pero aumenta mucho las desavenencias. Reducir el estrés simplemente va a aumentar la probabilidad de que el tanque va a permanecer sano.



Identificamos ahora los siguientes parámetros medibles como los más relevantes.

3.4.1 La dureza total (GH)

La dureza total del agua viene determinada por diversas sales (carbonatos, sulfatos, nitratos, fosfatos, cloruros, etc.) en total, basándose en la concentración de los iones metálicos calcio (Ca) y magnesio (Mg). Si el porcentaje de estas sales es elevado (con un gH alto), el agua es calificada de dura. Por ejemplo, las aguas típicas de manantial o pozo suelen ser aguas duras, ricas en sales disueltas. Un agua blanda es un agua pobre en sales disueltas y, por eso, con un gH bajo.

Para subir el gH lo mejor es aportar sales al agua. Por eso, muchas veces las piedras o las gravas que se usan suben la dureza. Si la piedra es caliza, está compuesta por muchas de estas sales y, además, tiene tendencia a disolverse en el agua, por lo que tener una piedra caliza en el acuario es como ir añadiendo todos los días una pequeña cantidad de sales. También acuarios con mucha suciedad (ricos en nitratos y fosfatos derivados de la descomposición de los alimentos) verán como el gH sube (aunque no de manera tan espectacular como con una piedra caliza).

Para bajar el gH la única forma es disminuir la concentración de sales. Se puede mezclar el agua con agua con menor concentración de sales (agua destilada o agua osmotizada). Esta mezcla no hace que necesariamente varíe el pH, pero sí va a disminuir el kH.

La dureza total influye de forma decisiva en las funciones orgánicas de todos los seres vivos en el agua. La dureza total favorable que permite mantener muy bien a la mayoría de los peces ornamentales oscila entre 6 grados dH y 16 grados dH (grados dH = grado de dureza alemana).

Dureza general en dH	Dureza general en ppm	Tipo de agua
0 - 4 dH	0 - 70 ppm	Muy blanda
4 - 8 dH	70 - 140 ppm	Blanda
8 - 12 dH	140 - 210 ppm	Poco dura
12 - 18 dH	210 - 320 ppm	Bastante dura
18 - 30 dH	320 - 520 ppm	Dura

Fig. 13 Tabla dureza general agua



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15



3.4.2 Carbonatos (KH)

El kH (también llamado dureza temporal) es un medidor de la concentración de grupos carbonatos/bicarbonatos en el agua. Aparte de las ya mencionadas sales de calcio y magnesio, prácticamente todas las aguas contienen bicarbonatos, cuya presencia se refleja en el valor KH.

A los bicarbonatos les corresponde un papel importante en el acuario. Tienen el efecto de tampón de pH e impiden con ello una alteración demasiado fuerte y rápida del valor pH (como por ejemplo la caída de acidez). La capacidad tampón (buffer) se refiere a la habilidad del agua para mantener estable el pH cuando se le añaden ácidos o bases. El pH y la capacidad tampón están entrelazados uno con el otro; aunque uno podría pensar que mezclando el mismo volumen de un ácido y el de un agua neutra se obtiene un pH a mitad camino entre los dos, esto pocas veces sucede en la práctica. Si el agua tiene suficiente capacidad tampón, ésta capacidad tampón puede absorber y neutralizar el ácido añadido sin apenas modificar el pH. El concepto es que el tampón actúa como una gran esponja. A medida que se añade ácido, la "esponja" absorbe el ácido sin cambiar mucho el pH. Sin embargo, la capacidad de la "esponja" está limitada; una vez que la capacidad tampón se ha gastado, el pH cambia más deprisa a medida que se añaden ácidos.

La capacidad tampón tiene consecuencias tanto positivas como negativas. En el lado positivo, el ciclo de nitrógeno produce ácido nítrico (nitrato). Sin la capacidad tampón, el pH de su tanque bajaría a lo largo del tiempo (una cosa mala). Con suficiente capacidad tampón, el pH se mantiene estable (una cosa buena). En el lado negativo, el agua de grifo casi siempre dura tiene una gran capacidad tampón. Si el pH del agua es demasiado alto para su pez, la capacidad tampón hace difícil bajar el pH a valores más adecuados. Intentos ingenuos de modificar el pH del agua normalmente fallan porque se ignoran los efectos tampón.

En acuarios de agua dulce, gran parte de la capacidad tampón del agua es debida a los carbonatos y bicarbonatos. Por ello, los términos "dureza carbónica" (KH), "alcalinidad" y "capacidad tampón" pueden ser usadas indistintamente. Aunque técnicamente no son lo mismo, son equivalentes a nivel práctico en el contexto de la acuariofilia. Nota: el término "alcalinidad" no debería confundirse con "alcalino". Alcalinidad se refiere a capacidad tampón, mientras que "alcalino" se refiere a que es básico (esto es, pH superior a 7).

Debido a la estrecha relación entre el valor KH y el valor pH, la dureza de carbonatos tiene igualmente una influencia directa sobre el bienestar de todos los organismos en el agua del acuario.



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15



El KH no afecta a los peces directamente, así que no hay ninguna necesidad de adecuar el KH a una determinada especie de pez. Aunque se recomienda un valor KH entre 3 grados dH y 10 grados dH para la mayoría de los peces de agua dulce.

3.4.3 Grado de acidez (pH)

El valor pH resulta de todas las materias acidas y básicas disueltas en el agua, que la acidifican o la convierten en alcalina.

El agua con un valor pH de 7 es denominada neutra. En este caso, los componentes ácidos y alcalinos estarán en equilibrio. Tanto más ácido, cuanto más bajo es el valor pH. Tanto más bases, tanto más aumenta el valor pH.

Conviene verificar el valor pH por lo menos una vez a la semana. Para casi todos los peces ornamentales en acuarios de agua dulce se consideran apropiados valores pH entre 6,5 y 8,5. Los peces procedentes de ríos tropicales de agua negra prefieren agua blanda con valores pH entre 7,5 y 8,5. El nivel del PH en la mayoría de los acuarios marinos se sitúa entre 7,6 y 8,3 siendo preferible mantenerlo por encima de 7,8 para acuario con " solo peces" y por encima de 8,0 en aquellos que contengan invertebrados marinos. Los denominados Acuarios de Arrecifes o Reeftanks el PH tiende a elevarse a niveles superiores a 8,3 debido a la alta concentración de oxígeno disuelto en el agua, producto de la sobre saturación del mismo producida por las algas en ambiente de alta luminosidad. Para estos casos se puede utilizar "Dióxido de Carbono" inyectados a través de cámaras de alta presión, dosificados por controladores de ORP para estabilizarlo.

Gracias a los productos que existen en el mercado, se puede regular la dureza de carbonatos y el valor pH del agua del acuario de manera muy sencilla y duradera.

El valor pH en el acuario depende directamente de la dureza de carbonatos y del contenido en dióxido de carbono (CO₂). Si se modifica la dureza de carbonatos o el contenido en CO₂, se cambiara automáticamente el valor pH.

En la siguiente gráfica podemos observar los niveles adecuados de CO₂ (remarcados en verde) que ha de contener el acuario para un determinado pH y una determinada dureza temporal.



CO ₂ (mg/l)								
	KH 2	KH 4	KH 6	KH 8	KH 10	KH 12	KH 14	KH 16
pH 8.0	1	2	2	3	4	5	6	6
pH 7.8	1	3	4	5	6	8	9	10
pH 7.6	2	4	6	8	10	12	14	16
pH 7.4	3	6	10	13	16	19	22	25
pH 7.2	5	10	15	20	25	30	35	40
pH 7.0	8	16	24	32	40	48	56	64
pH 6.8	13	25	38	51	63	76	89	101
pH 6.6	20	40	60	80	100			
pH 6.4	32	64	95					

Fig. 14 Grafica relación entre CO₂, KH y pH

3.4.4 Amoníaco, Nitrito y Nitrato (NH₃, NO₂, NO₃)

A través de los excrementos de los peces así como restos de plantas y de comida, el agua del acuario recibe combinaciones de nitrógeno que son desintegradas en varias fases.

En primer lugar se forma el amoníaco, que es tóxico, o el amonio no tóxico en una proporción que depende del pH. Mientras que con valores pH superiores a 7,0 aumenta la formación de amoníaco, con valores pH más bajos se forma el amonio no tóxico.

Si se trata de peces sensibles, un contenido de amoníaco de 0,1 mg/l durante un tiempo prolongado puede tener ya efectos dañinos. Ya con una concentración de 0,5 mg/l, pueden darse casos de muerte de peces.

Las bacterias del género nitrosomas, que se anidan en el filtro del acuario, desintegran al amoníaco o el amonio, convirtiéndolos en nitrito. El nitrito es igualmente muy nocivo para los peces en el acuario. Cuida por ello que haya un filtraje biológicamente activo, ya que solo los acuarios con un sistema de filtraje intacto garantizan concentraciones de nitrito inferiores a 0,1 mg/l. El contenido en nitrito no debe rebasar 0,25 mg/l durante un tiempo prolongado, dado que ya un valor de 0,5 mg/l puede ser problemático para tus peces de agua dulce.

Durante la última fase de desintegración, las bacterias nitrobacter convierten el nitrito en nitrato, que es relativamente inofensivo. El nitrato sirve, entre otros, como sustancia nutritiva para las plantas; sin embargo, en concentraciones demasiado elevadas es dañino para los peces ornamentales y plantas y fomenta el molesto crecimiento de algas.



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

Con un contenido de nitrato de 25 mg/l, el agua se considera contaminada, entre 50 mg/l y 100 mg/l, se recomienda una renovación parcial y si es superior a 100 mg/l, es necesario cambiar el agua en proporciones mayores.

3.4.5 Temperatura del agua

La temperatura del agua del acuario es un factor importante en la salud de los peces. Esto es particularmente cierto en la cría de peces, en el tratamiento de enfermedades, e incluso en la selección de distintas clases de peces para mantenerlos juntos.

Los expertos discrepan sobre la necesidad de mantener una temperatura constante que nunca flaquea. Algunos piensan que los peces que no experimentan los cambios de temperatura día / noche que ocurren en la naturaleza, obtienen el desarrollo de un sistema inmunológico menos robusto, y por lo tanto son más susceptibles a enfermedades. Otros creen que todos estos cambios de temperatura producen factores de estrés que pueden conducir a la mala salud de los peces.

Sin embargo, todos los expertos coinciden en que los cambios de temperatura rápidos y significativos, así como los cambios de temperatura frecuentes durante el día, son estresantes para los peces. Estos tipos de cambios repentinos o frecuentes de temperatura del agua pueden ocurrir por una multitud de razones, incluyendo:

- Que el tanque este situado al lado de una puerta o ventana.
- Que el tanque reciba la luz del sol directamente durante una parte del día.
- Cambios de grandes cantidades de agua.
- Calor producido por iluminación.
- Equipo defectuoso.

La ubicación del tanque puede tener un efecto significativo sobre la frecuencia y el tipo de cambio de la temperatura del agua que puede ocurrir. Los acuarios que se encuentran cerca de una ventana o una puerta pueden verse afectados cada vez que la puerta o la ventana se abren y se cierran. Incluso las ventanas que no se abren plantean un problema, ya que permiten la luz del sol, lo cual puede elevar rápidamente la temperatura del agua.

Las luces del acuario también pueden afectar a la temperatura del agua. Esto es particularmente cierto en pequeños tanques, algunos de los cuales todavía utilizan bombillas incandescentes que pueden producir una gran cantidad de calor. Incluso los nuevos tipos de



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15



iluminación pueden calentar el agua en el transcurso del día. Lo mejor que se puede hacer es vigilar y registrar la temperatura en el tanque durante todo el día para que saber los efectos de la iluminación sobre la temperatura del agua. Si se trata de más de unos pocos grados, es recomendable el sustituir la fuente de iluminación.

Un simple cambio del agua puede afectar la temperatura de manera significativa, dependiendo del tamaño del tanque y el volumen de agua cambiado. Los pequeños cambios de agua son siempre preferibles a los grandes cambios, y la temperatura del agua a reemplazar debe tener una temperatura cercana a la temperatura del agua del acuario.

La mayoría de los peces requieren rangos específicos de temperatura en su crianza. A menudo ésta es una superior a la temperatura normal, pero en algunos casos de cría se desencadena por una caída en la temperatura. Por eso cuando se realiza la cría de peces, es importante conocer los requisitos de temperatura del agua de las especies que se generaran.

La temperatura depende de la especie, pero en general en los peces tropicales están más saludables en el rango de 24-27 ° C. Y los peces de agua fría usan una temperatura inferior a este rango, algunos de ellos muy por debajo de 20 ° C, lo cual no es apto para cualquier pez tropical.

La temperatura ideal en acuarios marinos es la cercana a los 25 ° C, con rangos entre 22 ° C y 28 ° C, teniendo sumo cuidado con ella al momento de ingresar nuevos peces en el acuario y en las estaciones extremas (invierno, verano). Y en acuarios de agua dulce suele oscilar entre los 24 y 25 ° C. Aunque en última instancia la mejor temperatura dependerá de las especies de peces en el acuario.

3.4.6 Salinidad en acuarios marinos

La salinidad se refiere a la cantidad total de sustancias disueltas. La medida de la salinidad mide ambos componentes GH y KH así como otras sustancias como el sodio. Conocer la salinidad del agua es importante en los acuarios de agua salada. En los tanques de agua dulce, conocer el pH, GH y el KH es suficiente.

La salinidad normalmente se expresa en términos de peso específico, la relación entre el peso de una solución y el peso del mismo volumen de agua destilada. Dado que el agua se dilata cuando se calienta (cambiando su densidad), generalmente se usa como temperatura de referencia los 20 ° C. La salinidad se mide con un densímetro el cual está calibrado para una determinada temperatura (p.e. para 25 ° C).



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

Un componente de la salinidad que no incluye ni el GH ni el KH es el sodio. Algunos peces de agua dulce toleran (o incluso prefieren) una pequeña cantidad de sal (estimula el crecimiento del moco protector). Además, los parásitos (p.e. punto blanco) no toleran la sal en absoluto. Por ello, concentraciones de sal (hasta) de una cucharilla por cada 20 litros puede prevenir y curar el punto blanco y otras enfermedades parasitarias.

Por otra parte, algunas especies de peces no toleran nada de sal. Los peces sin escamas (en general) y ciertos Coridoras son mucho más sensibles a la sal que la mayoría de peces de agua dulce.

Su valor se expresa en gramos por litro, miligramos por litro o partes por millón (ppm). El único parámetro físico que afecta a la salinidad es la temperatura ya que cualquier aumento de la temperatura significa mayor solubilidad de las sustancias disueltas a excepción del cloruro de sodio.

Observando la salinidad de los acuarios que mantenemos podemos determinar que los acuario de agua dulce pueden presentar una salinidad hasta de 3 gr/l, los acuarios salobres hasta 15 gr/l y los acuarios marinos hasta 40 gr/l.

La salinidad y la conductividad están relacionadas porque la cantidad de iones disueltos aumentan los valores de ambas. Por ello una forma de obtener la salinidad es a través de conductividad. La conductividad en el mar suele estar entre los 5 S/m.

A continuación se puede observar la relación entre estos dos parámetros en la siguiente tabla.



Cond. (mS/cm)	Salinidad (g/l)	Cond. (mS/cm)	Salinidad (g/l)	Cond. (mS/cm)	Salinidad (g/l)
1,00	0.50 - 0.91	26,00	13.28 - 24.08	52,00	26.64 - 48.37
2,00	1.01 - 1.83	27,00	13.79 - 25.01	54,00	27.67 - 50.24
3,00	1.52 - 2.74	28,00	14.31 - 25.94	56,00	28.70 - 52.11
4,00	2.02 - 3.66	29,00	14.82 - 26.8	58,00	29.73 - 53.98
5,00	2.53 - 4.59	30,00	15.33 - 27.81	60,00	30.76 - 55.86
6,00	3.04 - 5.51	31,00	15.85 - 28.74	62,00	31.79 - 57.73
7,00	3.55 - 6.43	32,00	16.36 - 29.67	64,00	32.82 - 59.61
8,00	4.06 - 7.36	33,00	16.87 - 30.61	66,00	33.85 - 61.48
9,00	4.57 - 8.28	34,00	17.39 - 31.54	68,00	34.88 - 63.36
10,00	5.08 - 9.21	35,00	17.90 - 32.47	70,00	35.91 - 65.23
11,00	5.60 - 10.13	36,00	18.41 - 33.41	72,00	36.94 - 67.11
12,00	6.11 - 11.06	37,00	18.93 - 34.34	74,00	37.97 - 68.99
13,00	6.62 - 11.99	38,00	19.44 - 35.27	76,00	39.00 - 70.86
14,00	7.13 - 12.92	39,00	19.96 - 36.21	78,00	40.03 - 72.74
15,00	7.64 - 13.84	40,00	20.47 - 37.14	80,00	41.06 - 74.62
16,00	8.15 - 14.77	41,00	20.98 - 38.08	82,00	42.10 - 76.49
17,00	8.67 - 15.70	42,00	21.50 - 39.01	84,00	43.13 - 78.37
18,00	9.18 - 16.63	43,00	22.01 - 39.95	86,00	44.16 - 80.25
19,00	9.69 - 17.56	44,00	22.53 - 40.88	88,00	45.19 - 82.13
20,00	10.20 - 18.49	45,00	23.04 - 41.82	90,00	46.22 - 84.01
21,00	10.72 - 19.42	46,00	23.56 - 42.75	92,00	47.25 - 85.89
22,00	11.23 - 20.35	47,00	24.07 - 43.69	94,00	48.28 - 87.77
23,00	11.74 - 21.28	48,00	24.58 - 44.62	96,00	49.32 - 89.65
24,00	12.25 - 22.21	49,00	25.10 - 45.56	98,00	50.35 - 91.53
25,00	12.77 - 23.15	50,00	25.61 - 46.49	100,00	51.38 - 93.41

Fig. 15 Tabla relación entre conductividad y salinidad



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

3.5 Actuadores de interés

El siguiente paso después de conocer todas las magnitudes que nos interesan de nuestro acuario es actuar sobre ellas cuando consideramos que superan o descienden de ciertos valores umbral.

3.5.1 Sistema de cambios de agua

En todos los acuarios se producen sustancias contaminantes que surgen de los excrementos de los peces así como de restos de plantas y de comida, la solución a la contaminación es la dilución, los cambios de agua sustituyen una parte del agua "sucia" por una parte igual de agua limpia, diluyendo así las concentraciones de las sustancias indeseables en el acuario. El nitrato es el contaminante que predominantemente se acumula en tanques ya establecidos. Los cambios de agua periódicos son la forma más barata, segura y efectiva de mantener la concentración de nitrato a niveles razonables.

Debido a que los cambios de agua son la primera línea de defensa en la lucha contra problemas tales como enfermedades, sería deseable estar en condiciones de efectuar cambios parciales de agua en gran cantidad y con frecuencia, y para realizar estos cambios se utilizan las bombas de agua ya que son fácilmente manejables y sustituibles.

Por otro lado es común la inyección de medicinas, que están en estado líquido, para los peces o de otros fluidos para el tratamiento del agua. Por ello se recurre a las bombas peristálticas ya que son usadas para bombear fluidos limpios o estériles porque la bomba no puede contaminar el líquido, o para bombear fluidos agresivos porque el fluido puede dañar la bomba.

3.5.2 Alimentación de peces

Este es un aspecto que no debe ser improvisado, puesto que de una buena alimentación depende que los peces estén sanos, sean resistentes a las enfermedades y tengan una mayor longevidad. Por otro lado, la alimentación tiene efecto directo sobre la apariencia, la vitalidad, lucimiento de los colores, conformación del cuerpo, crecimiento apropiado y una buena reproducción.

La alimentación tiene que producirse una o dos veces al día y siempre por el mismo lado, para que los peces sepan adonde acudir y de este modo se coman toda la comida antes de que llegue al fondo, lo que minimizar el riesgo de que ésta se descomponga ya que es fuente de



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

compuestos nitrogenados, altera el pH y consume grandes cantidades de oxígeno.

3.5.3 Regleta controlada por RF

Es interesante permitir al usuario permitir el control de diferentes dispositivos alimentados a 220V de tensión alterna, por ejemplo para encender el calefactor cuando se detecta que la temperatura del acuario ha descendido del nivel que se haya prefijado.

3.5.4 Inyección de CO₂

El CO₂ ayuda a realizar la fotosíntesis a las plantas, de ahí que se considere, junto a la luz, el principal abono. Los niveles de CO₂ que se utilizan en los acuarios plantados son mucho mayores que los que hay en la naturaleza (como norma general) pero también la luz y resto de minerales que se aportan a las plantas.

Si se tiene un acuario con una iluminación media y sin plantas complicadas el CO₂ no será necesario. Si bien su uso no es necesario tampoco es perjudicial. Facilita la fotosíntesis (menos acelerada por lógica) y, por lo general a muchas algas el CO₂ no las sienta muy bien.

Si por el contrario se tiene un cañón de luz y se quiere tener un verdadero jardín acuático con plantas de muchas variedades y complicadas su uso será imprescindible. Lo que si hay que tener muy en cuenta es que luz, CO₂ y abonado van ligados y en escala, si el primero de ellos falla no será necesario el resto.

3.5.5 Iluminación del acuario

No existe una regla general para la iluminación de los acuarios, ya que cada acuario es un mundo, y hay muchas variables que influyen en la iluminación adecuada, como volumen del acuario, altura de la columna de agua, dureza del agua, cobertura del acuario, tipo de plantas,....

La iluminación juega un papel muy importante en los procesos químicos y, fundamentalmente, en el desarrollo de las plantas naturales. Por eso es necesario que se utilice el alumbrado del acuario correctamente. La iluminación es importante para cubrir parte de las necesidades de todos los seres vivos y organismos que viven en los acuarios.

La luz fluorescente permite simular la luz natural. Dependiendo de las dimensiones del acuario hay que calcular la luz que va a necesita. Hoy en día, la mejor forma de calcular la luz de nuestro acuario es en lúmenes (lm). Para acuarios de hasta 60 cm. de altura, incluido el sustrato,



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

se recomienda el uso de tubos fluorescentes y ampollitas compactas. Sobre esa altura se necesitará otro tipo de iluminación, como las ampollitas de vapor de mercurio (HQL) o las de ioduro metálico (HQI). El fluorescente (FL) y el haluro metálico (HQI) también son muy utilizadas, aunque últimamente se utiliza la iluminación LED.

Dependiendo del tipo de iluminación que se utilice, el crecimiento y desarrollo de las plantas del acuario variará. Se modifica el tamaño, las hojas y la forma de las mismas. Es por esto, que se utiliza iluminación con una temperatura de color sobre los 6.500° K. Aunque las plantas responden bien entre los rangos 5.000 y 6.500° K.

Una temperatura de color que está situada entre 2.700° K y 4.500° K, será ideal si se quieren que las plantas crezcan en altura. Una iluminación sobre la base de tubos fluorescente o ampollitas compactas de color frío da una tonalidad muy blanca en general, cuya temperatura de color está situada entre 7.000° K y 10.000° K. Genera plantas de crecimiento hacia los costados, es decir, plantas más anchas. Además de esto, el aspecto general del acuario no es muy atractivo, debido a que estas plantas generalmente son demasiado blancas en color, muy anchas y con dificultad en el crecimiento.

Se han hecho estudios y se ha determinado que una intensidad de luz adecuada para el promedio de las plantas y los peces de un acuario de agua dulce es de 30 a 35 lúmenes por litro de agua.

Para un acuario biotipo de los lagos africanos, casi sin plantas, se necesitan sólo 20 lúmenes por litro. Estos datos sirven para calcular la cantidad de luz en acuarios de no más de 60 cm. de altura, ya que, en una altura superior la intensidad luminosa que llega al sustrato disminuye considerablemente.



Fig. 16 Plantas creciendo con iluminación artificial

3.5.6 Sistema de control de temperatura

A lo largo del día se producen variaciones en la temperatura del acuario, por lo que necesitamos de mecanismos para mantenerlo en una temperatura cuasi constante.

Los calentadores aseguran que el acuario no se enfríe demasiado, y que la temperatura se mantenga constante a lo largo del día, incluso durante la noche cuando la temperatura de la habitación baja.

Hay dos tipos principales de calentadores. Los calentadores sumergibles están completamente bajo el agua. Un segundo tipo, tipo tradicional, tiene un tubo de cristal (el cual contiene las resistencias) que permanece parcialmente sumergido, pero los controles quedan fuera del agua. Los calentadores sumergibles son preferibles, ya que pueden ser colocados horizontalmente en el fondo del acuario. Esto ayuda a mantener las temperatura constante (el calor sube), y evita que el calentador quede fuera del agua durante los cambios parciales de agua. Con el tipo tradicional, uno debe acordarse de desconectarlo antes de cambiar el agua; si el calentador permanece fuera del agua mientras está encendido, el tubo se calienta y podría romperse cuando se vuelva a poner el agua en el acuario (por el cambio brusco de temperatura).



Fig. 17 Calentador tradicional

Es probable que la temperatura del acuario suba demasiado, ya sea porque sea verano o por el calor producido por las lámparas, para ello se utilizan los ventiladores. El reducido tamaño de estos ventiladores permite utilizarlo en acuarios de cualquier tamaño y en sitios estratégicos para inducir la evaporación. Los ventiladores electrónicos operan normalmente con corriente directa de 12V lo cual lo hace más seguro para las personas, pero tienen el riesgo de producir electrólisis si llegan a entrar en contacto con el agua salada, así que hay que usarlo con mucha precaución en ese sentido. Todos los terminales deberán de estar protegidos y las bases de sujeción tienen que estar firmemente colocados y ajustados. Los ventiladores electrónicos se colocan normalmente en los borde del acuario para que provoquen evaporación en la superficie del agua, y esto puede bajar la temperatura en 2°C ~ 3°C si las condiciones son favorables para la



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

evaporación.



Fig. 18 Ventilador electrónico acuario

3.6 Comunicación por puerto serie

Los puertos serie son la forma principal de comunicar una placa Arduino con un ordenador. Existen un sin fin de posibilidades en las que se requiere el empleo del puerto serie. Por tanto el puerto serie es un componente fundamental de una gran cantidad de proyectos de Arduino, y es uno de los elementos básicos que se deben de aprender para poder sacar todo el potencial de Arduino.

Un puerto serie envía la información mediante una secuencia de bits. Para ello se necesitan al menos dos conectores para realizar la comunicación de datos, RX (recepción) y TX (transmisión). No obstante, pueden existir otros conductores para referencia de tensión, sincronismo de reloj, etc. En ocasiones se refiere a los puertos serie como UART. La UART (universally asynchronous receiver/transmitter) es una unidad que incorporan ciertos procesadores, encargada de realiza la conversión de los datos a una secuencia de bits y transmitirlos o recibirlos a una velocidad determinada.

Prácticamente todas las placas Arduino disponen al menos de una unidad UART. Las placas Arduino UNO y Mini Pro disponen de una unidad UART que operan a nivel TTL 0V / 5V, por lo que son directamente compatibles con la conexión USB. Por su parte, Arduino Mega y Arduino Due disponen de 4 unidades UART TTL 0V / 5V.

Los puertos serie están físicamente unidos a distintos pines de la placa Arduino. Lógicamente, mientras usamos los puertos serie no podemos usar como entradas o salidas digitales los pines asociados con el puerto de serie en uso. En Arduino UNO y Mini Pro los pines empleados son 0 (RX) y 1 (TX). En el caso de Arduino Mega y Arduino Due el puerto de serie 1 está conectado a los pines 0 (RX) y 1 (TX), el puerto de serie 1 a los pines 19 (RX) y 18 (TX) el puerto de serie 2 a los pines 17 (RX) y 16 (TX), y el puerto serie 3 a los pines 15 (RX) y 14 (TX).

3.7 Protocolos de comunicación inalámbrica

Debido a que algunos actuadores son de alta tensión me interesa emplear comunicación inalámbrica para manejar una regleta de alta tensión 220V AC, posibilitando al usuario controlar desde el código o desde la GUI, el apagado y/o encendido de cualquier de estos dispositivos.

También me interesa utilizar comunicación inalámbrica para la recolección de datos con el fin de realizar de una aplicación web, donde el usuario podrá controlar en todo momento tanto los parámetros medidos por los sensores.

3.7.1 Modulación por desplazamiento de frecuencia

Este tipo de modulación es útil porque se emplea en regletas comerciales de enchufes de 220V AC por lo que se puede manejar alta tensión desde mi dispositivo a través de radiofrecuencia sin ningún peligro de forma eficaz y sin cables. Se puede realizar con un par de dispositivos de muy bajo coste en el que uno es el emisor y otro es el receptor en una comunicación unilateral uno a uno.

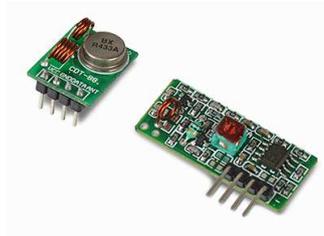


Fig. 19 Pareja de emisor y receptor de RF FSK

Se comercializan en dos frecuencias diferentes: 315MHz y 433MHz y existen librerías adaptadas para trabajar con ellos en Arduino. Open Aquarium utiliza este tipo de modulación a 433MHz.

La modulación por desplazamiento de frecuencia o FSK (Frequency Shift Keying), es una técnica de transmisión digital de información binaria (ceros y unos) utilizando dos o más frecuencias diferentes. La señal moduladora solo varía entre dos valores de tensión discretos formando un tren de pulsos donde uno representa un "1" o "marca" y el otro representa el "0" o "espacio". Esta modulación es más robusta frente al ruido que la modulación por desplazamiento de amplitud o ASK (Amplitude Shift Keying).

En esta forma de modulación la portadora sinusoidal toma dos valores de frecuencia, determinados directamente por la señal de datos binaria. El modulador puede realizarse en varios

modos; entre los más difundidos podemos mencionar:

- Un oscilador controlado por tensión (VCO).
- Un sistema que transmite una de las dos frecuencias, en función de la señal de datos.
- Un divisor gobernado por la señal de datos.

En el caso de que existan dos posibles frecuencias para transmitir la señal, hablamos de BFSK (*Binary Frequency Shift Keying*), como se muestra en la siguiente figura.

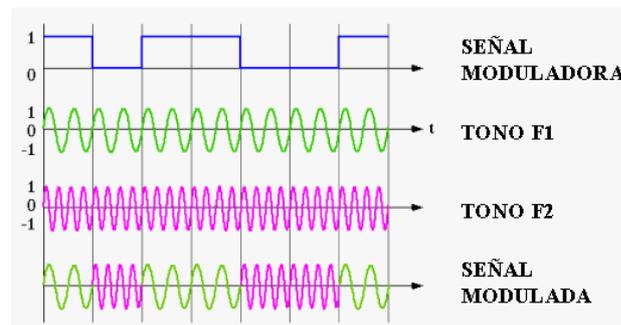


Fig. 20 Ejemplo de modulación BFSK

3.7.2 Estándar 802.11

La especificación IEEE 802.11 (ISO/IEC 8802-11) es un estándar internacional que define las características de una red de área local inalámbrica (WLAN). **Wi-Fi** es el nombre de la certificación otorgada por la Wi-Fi Alliance, anteriormente WECA (Wireless Ethernet Compatibility Alliance), grupo que garantiza la compatibilidad entre dispositivos que utilizan el estándar 802.11. Por el uso indebido de los términos el nombre del estándar se confunde con el nombre de la certificación. Una red Wi-Fi es en realidad una red que cumple con el estándar 802.11. A los dispositivos certificados por la Wi-Fi Alliance se les permite usar este logotipo:



Fig. 21 Logo Wi-Fi

Con Wi-Fi se pueden crear redes de área local inalámbricas de alta velocidad siempre y cuando el equipo que se vaya a conectar no esté muy alejado del punto de acceso. En la práctica, Wi-Fi admite ordenadores portátiles, equipos de escritorio, PDA o cualquier otro tipo de dispositivo de alta velocidad con propiedades de conexión también de alta velocidad (11 Mbps o superior) dentro de un radio de varias docenas de metros en ambientes cerrados (de 20 a 50 metros en general) o dentro de un radio de cientos de metros al aire libre.

El estándar 802.11 es el primer estándar y permite un ancho de banda de 1 a 2 Mbps. El estándar original se ha modificado para optimizar el ancho de banda (incluidos los estándares 802.11a, 802.11b y 802.11g, denominados estándares físicos 802.11) o para especificar componentes de mejor manera con el fin de garantizar mayor seguridad o compatibilidad.

Estándar	Frecuencia	Velocidad	Rango
802.11a	5 GHz	54 Mbit/s	10 m
802.11b	2,4 GHz	11 Mbit/s	100 m
802.11g	2,4 GHz	54 Mbit/s	100 m

Fig. 22 Tabla Estándares físicos

Es posible conectar Arduino a internet de forma inalámbrica usando el WiFi shield, este shield se conecta a redes inalámbricas con encriptación WEP o WPA2 a través de los estándares 802.11b y 802.11g. Su conexión con Arduino es a través del bus SPI utilizando los conectores ICSP. En Arduino UNO se conecta a través de los pines 11,12 y 13, y el Arduino MEGA a través de los pines 50, 51 y 52.

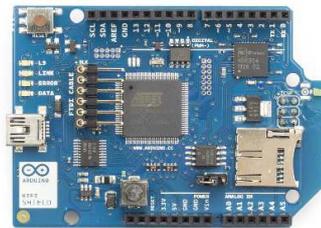


Fig. 23 Cara Top Wifi shield

Otra manera de conectar Arduino a internet es utilizando el shield XBee conectando el módulo wifi Roving RN-171. Este módulo incorpora una radio 802.11 b/g y permite la conexión con redes con seguridad WEP, WPA-TKIP y WPA2-AES. Su conexión a Arduino es a través de TTL UART.

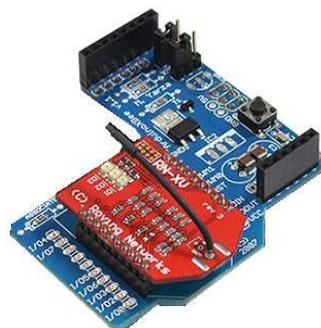


Fig. 24 Shield de comunicaciones con módulo wifi RovingRN-171



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

3.8 Decisión final

Tras el estudio de los parámetros que se necesitan medir para el buen cuidado de un acuario, se procedió a la búsqueda de sensores de los parámetros que no mide Open Aquarium para intentar integrarlos, pero debido a la falta de pines libres y sobre todo el costoso precio de estos sensores, se desechó esta opción y se procedió a utilizar solo los que utiliza Open Aquarium.

Por lo tanto se va a usar un sensor de temperatura, 2 de nivel de agua, 1 de fugas de agua, 1 de pH, y 1 de electro conductividad. Este último sensor, como nuestro acuario es plantado y no marino, no haría falta, pero se escoge con el fin de usar todos los sensores y así sacarle el máximo provecho a la placa Open Aquarium.

Aunque en principio el no medir el resto de parámetros pueda parecer peligroso, no es tan preocupante puesto que la dureza del agua de una zona no cambia, con lo que utilizando sensores químicos (que son más baratos) una vez y con los cambios de agua constantes, ya no lo necesitaríamos hacer más. En cuanto al amoníaco, los nitritos y los nitratos, gracias a estos cambios de agua, el filtro y la grava, son fácilmente reducibles.

Respecto a la elección de los actuadores, se van a usar las bombas de agua, las bombas peristálticas, el alimentador de peces, el calentador y el ventilador que se venden para Open Aquarium, pero se va a elegir otra lámpara que produzca más potencia ya que las plantas necesitan más potencia para crecer que la que produce la que se vende en Cooking Hacks, y se añade una bomba de CO₂.

La comunicación entre Arduino y Raspberry se realiza por puerto serie, esta comunicación se realizará a través de los puertos USB debido a la facilidad de conexión y a la cercanía que tienen que tener Arduino y Raspberry puesto que tienen que estar cerca del acuario, bien por la conexión con los sensores y actuadores, o bien por la conexión con la pantalla.

Y en cuanto al envío de los datos y el streaming para la aplicación web se realiza a través de WiFi con la conexión de un adaptador inalámbrico USB 802.11 b/g/n a uno de los puertos USB de Raspberry Pi. Se elige esta opción debido a que la cámara está conectada a Raspberry, por lo que como ésta es la encargada de mandar el streaming a la web se aprovecha para mandar los datos y así ahorrar la conexión de un módulo en Arduino, además de que hay una gran posibilidad de que el acuario no esté cerca de un router para poder conectar Raspberry con un cable Ethernet.

4. Hardware

4.1 Diagrama de bloques del sistema

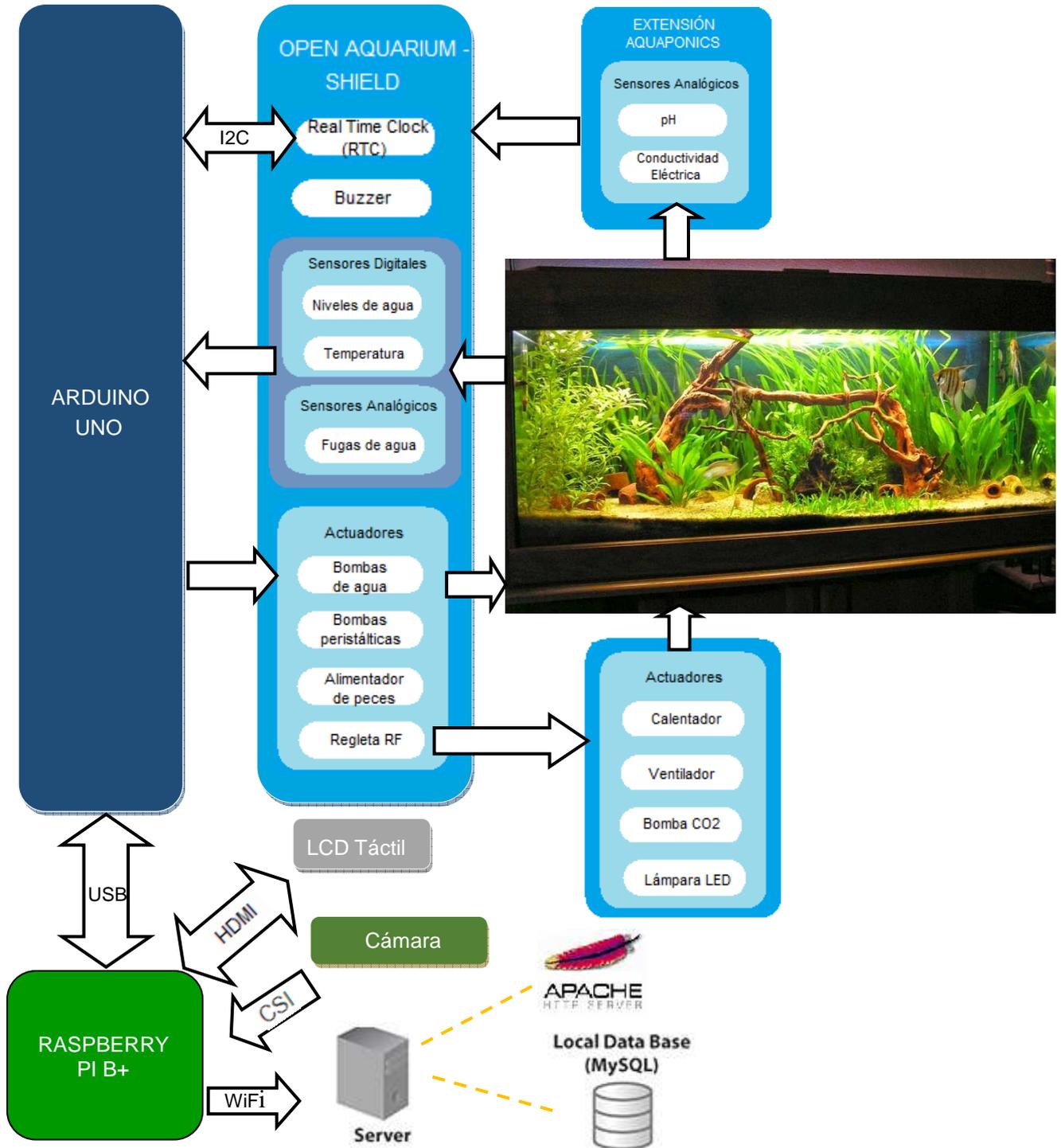


Fig. 25 Diagrama de bloques del sistema completo



4.2 Montaje en Arduino

4.2.1 Pinout en Arduino

La disposición de los pines en la placa Open Aquarium es:

Pines digitales

pin	Open Aquarium shield	Aquaponics
0	RX Conexión USB	RX Conexión USB
1	TX Conexión USB	TX Conexión USB
2	RF Regleta	RF Regleta
3	Bomba de agua 2	Bomba de agua 2
4	Bomba de agua 1	Bomba de agua 1
5	Buzzer	Buzzer
6	Lámpara	Lámpara
7	Sensor de nivel de agua 1	Sensor de nivel de agua 1
8	Sensor de nivel de agua 2	Alimentación sensor pH
9	Bomba peristáltica 3	Alimentación sensor conductividad eléctrica
10	Bomba peristáltica 2	Bomba peristáltica 2
11	Bomba peristáltica 1	Bomba peristáltica 1
12	Sensor de Temperatura	Sensor de Temperatura
13	Alimentador de peces	Alimentador de peces

Entradas analógicas

pin	Open Aquarium shield	Aquaponics
0	-	Sensor de conductividad eléctrica
1	-	Sensor de pH
2	Posición alimentador de peces	Posición alimentador de peces
3	Sensor de fugas de agua	Sensor de fugas de agua
4	RTC (I2C-SDA)	RTC (I2C-SDA)
5	RTC (I2C-SCK)	RTC (I2C-SCK)

La entrada analógica 3 se utiliza como entrada digital para saber si el alimentador de peces ha dado una vuelta completa.



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

Tras la última revisión del módulo Aquaponics, el nuevo diseño necesitaba otros pines además de los que se habían habilitado en la placa Open Aquarium, en este caso los pines digitales 8 y 9, correspondientes al sensor de nivel de agua 2 y a la bomba peristáltica 3 respectivamente. Tras diversas pruebas, vi que el sensor de nivel aún podía utilizarse, cosa que viene bien por su utilidad en los cambios de agua, pero que la bomba era inutilizable, ya que la función encargada de medir el pH manda una señal 0-5 V constantemente, lo que hace que la bomba esté todo momento encendiéndose y apagándose, por lo que podría dañarse la bomba además de que no haría la función que se le programara.

Por otro lado el pin digital 6, utilizado para el control de una lámpara alimentada de 12 V, no se utiliza debido a que la lámpara a utilizar necesita mayor alimentación.

4.2.2 Sensores

4.2.2.1 Sensor de nivel

Los sensores de nivel son una manera de conocer el nivel del agua del acuario, que es un parámetro necesario para saber cuándo reponer el agua del acuario o para los cambios de agua a la hora de automatizar el acuario. El motivo de que use 2 sensores de nivel es por este último motivo, uno indicará cuando la caja de cambios esté llena para volver a rellenar el acuario, y el otro indicará cuando ya esté lleno el acuario para parar de llenar.

El sensor que se va a utilizar tiene una salida resistiva que varía con el nivel del fluido y es fácilmente utilizable para sistemas electrónicos de control. La envoltura del sensor es comprimida por la presión del líquido en el que está inmerso, lo que origina un cambio en la resistencia correspondiente a la distancia desde la parte superior del sensor a la superficie del líquido. La salida resistiva del sensor es inversamente proporcional a la altura del líquido, cuanto menor sea el nivel del líquido, mayor es la resistencia de salida; cuanto mayor sea el nivel del líquido, menor es la resistencia de salida.

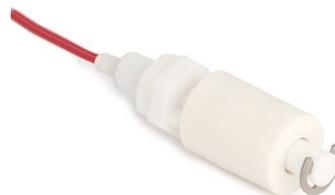


Fig. 26 Sensor de nivel de Open Aquarium

Este sensor soporta una potencia de 10 W, una tensión de conmutación de 100 V, una corriente de conmutación de 0,5 A, una tensión de ruptura de 220 V, una corriente de carga de 1 A y una temperatura entre -10 y 85 grados Celsius.

4.2.2.2 Sensor de fugas de agua

Este sensor es muy útil para el ahorro de agua, ya que cuando el sensor detecte el agua parará las bombas si están rellenando el acuario por la pérdida del agua y a la vez activará una alarma que avisará al usuario para salvaguardar la integridad del acuario.

Con el sensor que se va a utilizar se puede detectar la presencia de agua u otros líquidos conductores en una línea continua sin puntos ciegos. Gracias a la forma del cordón / cadena del sensor se puede asegurar que cualquier gota de agua será detectado tan pronto como entra en contacto con el sensor. Es capaz de detectar fugas de agua en tiempo real en un rango de hasta 100m.



Fig. 27 Sensor de fugas de Open Aquarium

Este sensor está hecho de polietileno (PE) con una aleación de plomo y tiene un núcleo resistivo de 3 ohm por cada 100 m. Soporta temperaturas hasta un máximo de 75° y fuerzas de tracción de hasta 60 Kg. Y los contactos no tienen polaridad.

4.2.2.3 Sensor de temperatura

La medición de la temperatura es uno de los aspectos básicos en el control de acuarios, para medir la temperatura del agua se utiliza un sensor impermeabilizado a prueba de agua.

El sensor que se utiliza es el DS18B20, es un sensor digital de temperatura de Maxim Integrated que se comunica con el microprocesador a través de la librería One- Wire, la cual solo necesita una línea digital de datos. Proporciona lecturas de temperatura de 9-12 bit (configurable). Es rápido, robusto, fiable y tiene un bajo coste.

Tiene una tensión de alimentación de 3,3 a 5 V, mide en un rango de temperatura de -55°

C a +125° C y una precisión de $\pm 0,5^{\circ}$ C en el rango de -10° C a +85° C.



Fig. 28 Sensor de temperatura de Open Aquarium

4.2.2.4 Sensor de pH

Para medir el pH del se va a utilizar el sensor 201DH PH que se está usando actualmente en diversos productos de Libelium. Este sensor es de tipo de combinación de electrodo. Tiene una sensibilidad de 56mV/grado de pH. El rango de medida es de 0 a 14 grados de pH. Puede trabajar entre 0 y 60°C. Y su conector es de tipo BNC.



Fig. 29 Sensor de pH de Open Aquarium

4.2.2.5 Sensor de electro conductividad

Para medir la electro-conductividad del agua se utiliza el sensor 208DH que está usando actualmente Libelium, da una respuesta analógica directamente proporcional con la cantidad de sales que contiene el agua. Por eso se utiliza en los acuarios marinos para conocer la salinidad del acuario. Puede trabajar entre 0 y 60°C. Es un sensor con 2 electrodos de platino. Y la constante de la celda de conductividad es de $1 \pm 0.2 \text{ cm}^{-1}$.



Fig. 30 Sensor de electro conductividad de Open Aquarium

4.2.3 Actuadores

4.2.3.1 Bombas de agua

Las bombas de agua son uno de los actuadores imprescindibles en el cuidado de los acuarios ya que se usan para los cambios de agua que son la primera defensa en este cuidado.

Se utiliza una bomba sumergible para reducir espacio. Su alimentación es de 12 a 24 V. Y el agujero de salida del agua es de 4mm por el que se pueden mover de 3 a 4 litros de agua por minuto.



Fig. 31 Bomba de agua de Open Aquarium

4.2.3.2 Bombas peristálticas

El uso de bombas peristálticas es con el fin de suministrar medicamentos ya que no contaminan el líquido que pasa por sus gomas, y estas son fácilmente sustituibles cuando se degradan.

Con las que se trabajará se alimentan a 12 V, su temperatura de trabajo es de 0 a 40° C y mueven fluidos de 20 a 60 mililitros por minuto.



Fig. 32 Bomba peristáltica de Open Aquarium

4.2.3.3 Alimentador de peces

Para tener automatizado la alimentación de los peces se utiliza un alimentador programable, alimentado a 3,3 V.



Fig. 33 Alimentador de peces de Open Aquarium

4.2.3.4 Regleta controlada por RF

Como ya se comentó, es interesante usar una regleta controlada por RF para usar dispositivos alimentados a 220 V. La regleta que se va a usar tiene una tensión de trabajo de 240 V a 50 Hz, una potencia máxima de carga de 2400 W, un consumo de potencia en standby de 0,3 W y una frecuencia de transmisión de 433,92 MHz.



Fig. 34 Regleta RF de Open Aquarium

4.2.3.5 Bomba de CO₂

El CO₂ ayuda a realizar la fotosíntesis a las plantas, puesto que el acuario va a ser mayoritariamente plantado el utilizar una bomba inyectora de CO₂ es muy beneficioso. En este proyecto se va a utilizar un extintor de CO₂. Es una alternativa muy válida y más económica a la bomba de CO₂. Se utiliza solamente acoplándole el manorreductor con el solenoide (que es una válvula de corte) y dejando la maneta apretada con una brida. El manorreductor es el aparato encargado de hacer que el CO₂ de la bombona salga de manera controlada y a la presión que se



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

desea, mostrando la cantidad de gas que queda y la presión con la que se trabaja. La bomba está conectada a la regleta puesto que trabaja a 220 V.



Fig. 35 Extintor de CO₂

4.2.3.6 Lámpara LED

Para el buen crecimiento de las plantas además del CO₂ se necesita una fuente de luz suficiente para que hagan la fotosíntesis. Por ello en vez de utilizar la lámpara de Cooking Hacks que solo vale para iluminar, se utiliza una lámpara con una potencia mayor que la mitad de litros del acuario que es lo que se recomienda, en este caso mayor de 40 W debido a que el acuario es de 80 litros.

La lámpara elegida es de 48 W con LEDs que proporcionan luz blanca y azul para simular la luz diurna y de luna. Esta luminosidad puede ser controlada a través de una entrada dimmer con dos canales para poder elegir entre los dos tipos de luminosidad.



Fig. 36 Lámpara LED

Su tensión de trabajo es de 24 V, pero viene con un adaptador para poder conectarse a redes de 100 a 240 V, por lo que se conecta también a la regleta para controlar el encendido y apagado.

4.2.3.7 Ventilador

El ventilador es necesario para reducir la temperatura por el aumento provocado por el calor de la luminosidad de la lámpara o el calor del verano. Se va a utilizar el que se vende en Cooking Hacks.



Fig. 37 Ventilador de Open Aquarium

El ventilador tiene una corriente nominal de 0,43 A y una tensión nominal de 24 V, aunque se conecta a la regleta puesto que tiene integrado un transformador de 220/24 V. Es capaz de lanzar un flujo de aire máximo de 13,6 CFM (cubic feet per minute) y su rango de operación es de -10 a 60° C.

4.2.3.8 Calentador

El calentador es necesario para que el acuario no se enfríe demasiado, y que la temperatura se mantenga constante a lo largo del día, incluso durante la noche cuando la temperatura de la habitación baja. Se va a utilizar el que se vende en Cooking Hacks.



Fig. 38 Calentador de Open Aquarium

Se alimenta a una tensión de 220/240 V por lo que es el último de los actuadores que se conectan a la regleta. Tiene una potencia de 100 W. El rango de temperatura es entre 17 y 35° C. Es recomendado para trabajar en acuario de 75 a 125 litros. Y es totalmente sumergible aunque por seguridad se recomienda dejar la parte eléctrica por fuera del agua.

4.3 Montaje en Raspberry

4.3.1 GPIOs Raspberry Pi

Cuando se abrió la lámpara se vio que para controlar la luminosidad de ésta solo hacía falta meterle una señal PWM a través de su entrada dimmer.

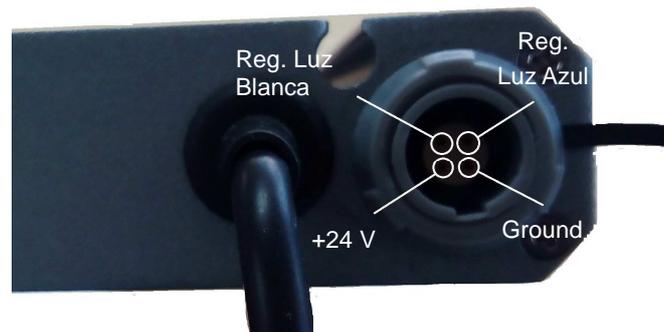


Fig. 39 Entrada dimmer de la lámpara

Debido a que esta lámpara es distinta a la que se vende para Open Aquarium, y con su conexión con la regleta queda un pin libre, en un principio se pensó en controlar la luminosidad a través de Arduino. Pero como son dos canales, o solo se controlaba un canal o se eliminaba un elemento del sistema para que quedara libre su pin y así poder controlar los dos canales.

Con el fin de no reducir ningún elemento del sistema y sacarle el máximo partido a Raspberry Pi, se van a utilizar los GPIOs de ésta, que mediante un módulo para Python pueden producir señales PWM especificando tanto su duty cycle como su frecuencia.

Los GPIOs pueden ser configurados como pin de entrada o de salida de propósito general, y algunos pueden ser utilizados para comunicarse con otros dispositivos ya sea por UART, SPI o I2C.

Para el control de luminosidad se realizan 3 conexiones: la masa de la entrada dimmer con cualquiera de los GPIO que son masa en la Raspberry Pi, la entrada de regulación lumínica de luz blanca del dimmer con el GPIO 18 y la entrada de regulación lumínica de luz azul con el GPIO 17.

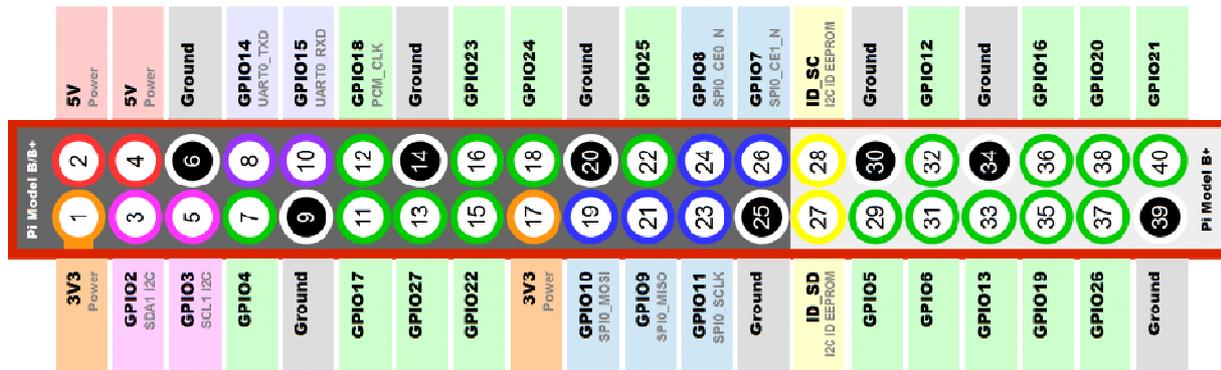


Fig. 40 Disposición GPIOs Raspberry Pi B+

4.3.2 Pantalla LCD touchscreen

Para la monitorización de los datos se va a utilizar una pantalla táctil como panel de control. Se va a utilizar una pantalla TFT de 7 pulgadas eGalax touchscreen de SaintSmart.



Fig. 41 TFT touchscreen

La pantalla se conecta a través de su falda a una placa controladora VS-TY2662-V1, que tiene posibilidad de conexión a HDMI (que es la conexión a utilizar), VGA y AV2. Esta placa también tiene un conector para una botonera con la que poder manejar la configuración de la pantalla (contraste, salida de video...) y se alimenta a 12 V. Para usar la función táctil hay que unir la falda correspondiente con otra placa controladora que va conectada a un USB que se introducirá en uno de los puertos de Raspberry Pi.

4.3.3 Cámara para Raspberry Pi

Con el objetivo de que los elementos del sistema sean siempre que se pueda, productos que se venden en Libelium, se va a utilizar para realizar el streaming la cámara propia para Raspberry Pi.



Fig. 42 Cámara Raspberry Pi

La cámara para Raspberry Pi se puede utilizar tanto para tomar video como fotos de alta definición. Es fácil de utilizar para los principiantes, y tiene mucho que ofrecer a los usuarios avanzados para ampliar su conocimiento. Es muy popular en las aplicaciones de seguridad para el hogar.

La cámara cuenta con un sensor Omnivisión de 5 megapíxeles que además de vídeo en alta definición y a 30 fotogramas por segundo, es capaz de hacer fotos con una resolución de 2.592 X 1.944 píxeles. Soporta los modos 1080p30, 720p60 y vídeo VGA90. Se une a través de un cable de cinta al puerto CSI en la Raspberry Pi. Se puede acceder a través de las API (Application Programming Interface) MMAL y V4L, y hay numerosas bibliotecas de terceros creadas para ello, incluyendo la biblioteca Picamera Python.

La cámara se conecta mediante su cable de 15 pines directamente con Raspberry a través de su conector CSI (Camera Serial Interface). Tiene un tamaño de 25x20x9 mm y pesa unos 3 gramos.

El soporte software está disponible en la distribución Raspbian, que hay que tener estar actualizado a la última versión para asegurar su buen funcionamiento, y hay que ejecutar la utilidad raspi-config y habilitar la opción de la cámara para dar soporte a este módulo. Eso permitirá utilizar la captura de fotos y vídeos con las herramientas **raspistill** y **raspivid**.



4.3.4 Adaptador USB 802.11 b/g/n

Debido a que con gran probabilidad el acuario del usuario no va a estar cerca de un router, se opta por enviar los datos y el streaming a través de WiFi. Para ello se va a utilizar el adaptador USB Edimax EW-7811UN que es totalmente compatible con Raspberry Pi, y no necesita de hub alimentado como otros adaptadores ya que es suficiente con la corriente que le suministra Raspberry Pi.



Fig. 43 Adaptador inalámbrico USB

Este adaptador cumple con los estándares inalámbricos IEEE802.11b/g/n. Además como está construido con la última tecnología inalámbrica, el EW-7811Un puede aumentar su cobertura inalámbrica hasta 3 veces más. La tasa de transmisión de datos puede ir hasta 150 Mbps cuando se conecta a un dispositivo 802.11n, con un viaje de la señal más lejos y conexión mucho más estable en comparación con su red 802.11g anterior. Es capaz de reducir el consumo de un 20-50% en los momentos de inactividad. Soporta encriptación 64/128 bit WEP, WPA, WPA2 y es compatible con WPS. Y funciona en entornos de 0 - 40° C y con un 10 - 90 % de humedad sin condensación.

4.3.5 Caja soporte pantalla LCD

Con el fin de realizar un encapsulado para la pantalla se ha optado por la tecnología de impresión 3D ya que en la empresa Libelium disponemos de impresoras 3D con las que hemos podido realizar pruebas de cara al diseño del encapsulado final.

Para realizar el modelo 3D se ha utilizado el software online Tinkercad que permite de forma fácil e intuitiva el diseño de modelos en 3D y posteriormente la exportación de los archivos en diferentes formatos estándar para su impresión 3D.



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

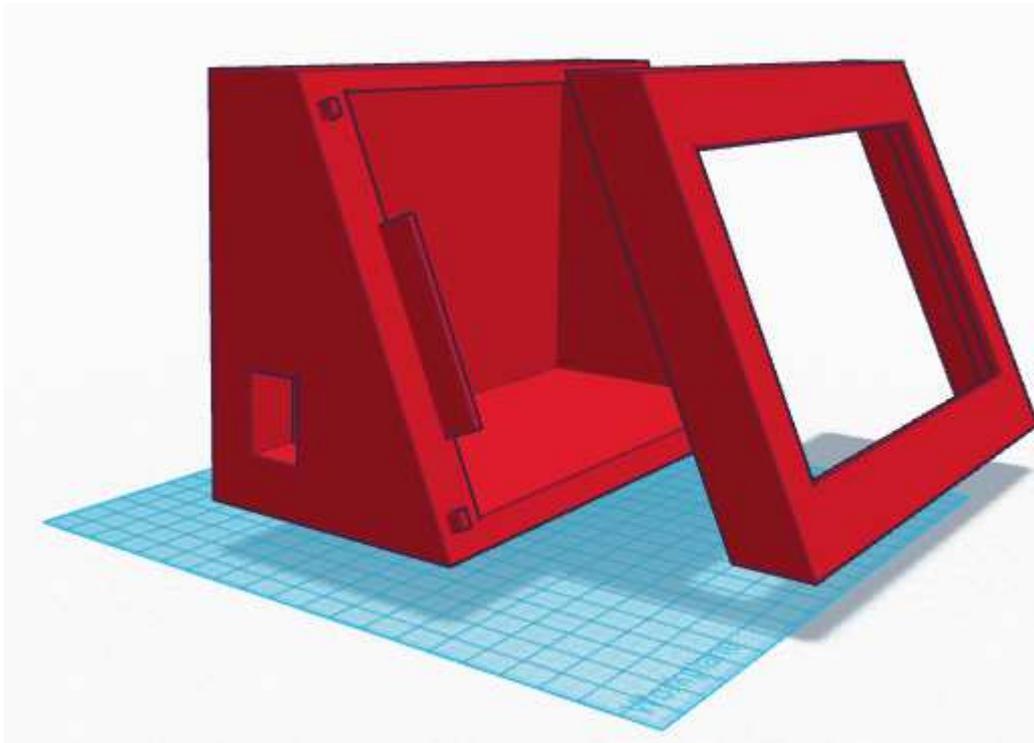


Fig. 44 *Diseño para caja pantalla*

Una vez que estuvo claro el diseño final se encargó su impresión definitiva a la empresa Shapeways dedicada profesionalmente a la impresión 3D ofreciendo un gran acabado final y permitiendo elegir entre una gran cantidad de materiales y colores.



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

5. Diseño Software

5.1 Introducción

En este apartado se explica el funcionamiento de las distintas partes que componen los códigos de programación mediante diagramas de flujo o explicando las funciones utilizadas más importantes. También se cuenta las distintas librerías usadas en los códigos, así como los entornos de desarrollo de programación y lenguajes que estas usan. Igualmente se habla de las pequeñas modificaciones que han sido necesarias realizar en alguna de ellas para el correcto funcionamiento de todos los módulos. Aunque se hayan utilizado librerías ya existentes ha sido necesario mucha programación en el código para dotarlo de inteligencia y que sea robusto.

Por otro lado también se cuenta los diversos procedimientos a hacer para tener todo bien configurado, así como del desarrollo de una aplicación web de monitorización.

Todo el código generado tiene licencia GPL por lo que cualquier persona tiene derecho a acceder al código, estudiarlo y modificarlo.

5.2 Software de base: Arduino IDE

Para realizar cualquier tipo de programación con la tecnología de Arduino, es necesario descargarse de la página web de Arduino el entorno de desarrollo (IDE). Se dispone de versiones para Windows y para MAC, así como las fuentes para compilarlas en LINUX. En el caso de disponer de una placa USB es necesario instalar los drivers FTDI. Estos drivers vienen incluidos en el paquete de Arduino mencionado anteriormente. Existen en la web, versiones para distintos sistemas operativos. Es importante usar la última versión que se encuentra en la web.

Lo primero que tenemos que hacer para comenzar a trabajar con el entorno de desarrollo de arduino es configurar las comunicaciones entre la placa Arduino y el PC. Para ello deberemos abrir en el menú "Herramientas" la opción "Puerto Serie". En esta opción deberemos seleccionar el puerto serie al que está conectada la placa.

La estructura de códigos se divide en dos partes fundamentales: setup y loop. El setup es la primera parte del código que se ejecuta, haciéndolo solo una vez al iniciar el código. El loop es un bucle que se ejecuta continuamente, formando un bucle infinito.

En el entorno existen una serie de botones que sirven para un manejo rápido del código:



Verificar: Comprueba el código en busca de errores.



Cargar: Compila el código y lo vuelca en la placa E/S de Arduino.



Nuevo: Crea un nuevo sketch.



Abrir: Presenta un menú de todos los programas sketch del "sketchbook" (librería de sketch). Un clic sobre uno de ellos lo abrirá en la ventana actual.



Guardar: Salva el programa sketch.



Monitor Serial: Inicia la monitorización serie.

```
aquarium
#include "OpenAquarium.h"
#include "Wire.h"

//variables
DateTime now;
int status1;
int status2;
long medianvalue = 0;
float temperature;
int flag = 0;

//calibration points
#define point_1_cond 40000 // Write here your EC calibration val
#define point_1_cal 40 // Write here your EC value measured
#define point_2_cond 10500 // Write here your EC calibration val
#define point_2_cal 120 // Write here your EC value measured

#define calibration_point_4 2221 //Write here your measured value
#define calibration_point_7 2065 //Write here your measured value
```

Fig. 45 IDE de programación de Arduino



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

5.3 Lenguaje de programación: C/C++

C es un lenguaje de programación orientado a la implementación de Sistemas Operativos, concretamente Unix. Es un lenguaje de medio nivel. Trata con objetos básicos como caracteres, números. . . Posee una gran portabilidad. Los compiladores suelen ofrecer extensiones al lenguaje que posibilitan mezclar código en ensamblador con código C o acceder directamente a memoria o dispositivos periféricos.

C++ es un lenguaje de programación orientado a objetos que toma la base del lenguaje C. Es un lenguaje versátil, potente y general. Mantiene las ventajas del C en cuanto a riqueza de operadores y expresiones, flexibilidad, concisión y eficiencia y ha eliminado algunas de las dificultades y limitaciones del C original. El C++ es a la vez un lenguaje procedural (orientado a algoritmos) y orientado a objetos. Como lenguaje procedural se asemeja al C y es compatible con él, aunque ya se ha dicho que presenta ciertas ventajas. Como lenguaje orientado a objetos se basa en una filosofía completamente diferente, que exige del programador un completo cambio de mentalidad. Las características propias de la Programación Orientada a Objetos de C++ son modificaciones mayores que sí que cambian radicalmente su naturaleza.

Las principales características de este tipo de lenguajes son:

- Tiene un conjunto completo de instrucciones de control.
- Permite la agrupación de instrucciones.
- Incluye el concepto de puntero (variable que contiene la dirección de otra variable).
- Los argumentos de las funciones se transfieren por su valor.
- E/S no forma parte del lenguaje, sino que se proporciona a través de una biblioteca de funciones.
- Permite la separación de un programa en módulos que admiten compilación independiente.
- Programación de bajo nivel (nivel bit).

El poder manejarse con este lenguaje ha sido fundamental tanto para comprender y usar las librerías de Arduino, como para poder utilizar el toolkit wxPython para el desarrollo de interfaces gráficas, ya que tiene un lenguaje más parecido a C++ que a Python, como se explicará posteriormente.



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

5.4 Librería Open Aquarium

La librería Open Aquarium está desarrollada buscando resultar de fácil utilización al usuario final, sin necesidad de conocer los componentes con los que ha sido diseñada.

Está implementada en lenguaje C-C++ íntegramente, y en una arquitectura orientada a objetos, siguiendo el modelo de las demás librerías de Arduino.

La librería engloba el conjunto de subprogramas que sirven para el funcionamiento correcto del software. Está compuesta por dos archivos: OpenAquarium.h, la cabecera de la librería, en donde se definen todas las variables y funciones a utilizar, y sus propiedades (públicas, privadas...), y OpenAquarium.cpp, que es el objeto de la librería, en la que se programan cada una de las funciones definidas en el otro archivo.

Estas librerías podemos encontrarlas en la página de Cooking Hacks en el tutorial de Open Aquarium.

Para la realización del sketch de Arduino, es indispensable el utilizar estas librerías, y se van a utilizar en gran medida las funciones que implementan estas librerías. Sin embargo, debido a que la comunicación entre Arduino y Raspberry se realiza por puerto serie, hay que modificar alguna de las funciones a utilizar ya que en su programación incluyen comentarios para salir por ventana y esto ocasiona problemas de intercomunicación con la GUI.

Las funciones que pueden ocasionar problemas son **feedfish()**, que es una función que hace girar una vuelta completa al alimentador de peces, **empty()**, que es una función que vacía el acuario y cuando se llena el recipiente de cambios de agua, manda la orden para empezar a rellenar, y **fill()**, que comprueba si se ha llenado el acuario, y si es así ordena que se pare la bomba de agua.

A la hora de modificar estas opciones se ha optado por comentar estas salidas por pantalla para no modificar la esencia de la librería. Por otro lado, en la función **empty()** se usa la bomba peristáltica 3, y como se ha comentado en el apartado de Hardware, esta bomba no se usa por estar en conflicto con el sensor de electro-conductividad. Por lo tanto, también se comenta la zona en la que esta bomba actúa.



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

5.5 Raspbian

Raspbian es una distribución del sistema operativo GNU/Linux y por lo tanto libre basado en Debian Wheezy (Debian 7.0) para Raspberry Pi, orientado a la enseñanza de informática. El lanzamiento inicial fue en junio de 2012.

Técnicamente el sistema operativo es un port no oficial de Debian wheezy armhf para el procesador (CPU) de Raspberry Pi, con soporte optimizado para cálculos en coma flotante por hardware, lo que permite dar más rendimiento en según qué casos. El port fue necesario al no haber versión Debian wheezy armhf para la CPU ARMv6 que usa Raspberry PI.

Debido a que Raspbian se basa estrechamente en Debian, hay una enorme cantidad de documentación disponible para Raspbian.

La distribución usa LXDE como escritorio y Epiphany como navegador web. Además contiene herramientas de desarrollo como IDLE para el lenguaje de programación Python o Scratch, y diferentes ejemplos de juegos usando los módulos Pygame.

Destaca también el menú "raspi-config" que permite configurar el sistema operativo sin tener que modificar archivos de configuración manualmente. Entre sus funciones, permite expandir la partición root para que ocupe toda la tarjeta de memoria, configurar el teclado, aplicar overclock, etc.

El manejo de este sistema operativo ha sido fundamental para la instalación y seguida configuración de los paquetes necesarios para el correcto funcionamiento del sistema.

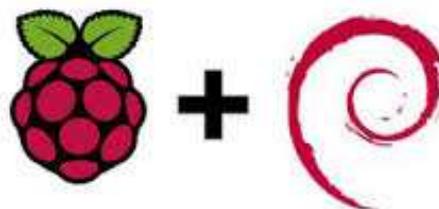


Fig. 46 Símbolo Raspbian



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

5.6 Python

Python es un lenguaje de programación interpretado que a diferencia de muchos de los lenguajes de programación provee de reglas de estilos, a fin de poder escribir código fuente más legible y de manera estandarizada.

Se trata de un lenguaje de programación multiparadigma, ya que soporta orientación a objetos, programación imperativa y, en menor medida, programación funcional. Es un lenguaje interpretado, usa tipado dinámico y conteo de referencias para la administración de memoria, y es multiplataforma. Una característica importante de Python es la resolución dinámica de nombres; es decir, lo que enlaza un método y un nombre de variable durante la ejecución del programa (también llamado enlace dinámico de métodos). Otro objetivo del diseño del lenguaje es la facilidad de extensión. Se pueden escribir nuevos módulos fácilmente en C o C++. Python puede incluirse en aplicaciones que necesitan una interfaz programable.

El intérprete de Python estándar incluye un modo interactivo en el cual se escriben las instrucciones en una especie de intérprete de comandos: las expresiones pueden ser introducidas una a una, pudiendo verse el resultado de su evaluación inmediatamente, lo que da la posibilidad de probar porciones de código en el modo interactivo antes de integrarlo como parte de un programa. Esto resulta útil tanto para las personas que se están familiarizando con el lenguaje como para los programadores más avanzados.

Es administrado por la Python Software Foundation. Posee una licencia de código abierto, denominada Python Software Foundation License, que es compatible con la Licencia pública general de GNU a partir de la versión 2.1.1, e incompatible en ciertas versiones anteriores.



Fig. 47 Logo Python

Python es el lenguaje de programación que es recomendado por los fundadores de la Raspberry Pi, pues piensan que es un lenguaje de sintaxis sencilla y clara que puede venir bien en los temas de educación. Y es el lenguaje que va a ser la base para la realización del script que permitirá al usuario conocer el estado del acuario y actuar sobre él.



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15



5.6.1 Interfaces gráficas

Para que la comunicación entre el sistema y el usuario sea más sencilla y de una forma estéticamente más bonita, se valoró la creación de una GUI. Python tiene disponible un gran número de toolkits para la creación de interfaces gráficas, desde TkInter (tradicionalmente incluido con Python, usando Tk) a una serie de otras soluciones multiplataforma, así como bindings a específicas plataformas (también conocido como "tecnologías nativas "). A continuación se va a realizar una pequeña descripción de algunos de los toolkits que se han probado.

- **TKInter:** es un módulo que permite construir interfaces gráficas de usuario multiplataforma en Python utilizando el conocido toolkit Tk. Python incluye este módulo por defecto, lo que hace que sea un toolkit muy popular. TkInter, además, es robusto, maduro y muy sencillo de aprender y de utilizar, contando con una amplia documentación.

TkInter se distribuye bajo la PSFL (Python Software Foundation License) una licencia compatible con la GPL creada para la distribución de software relacionado con el proyecto Python. La PSFL carece de la naturaleza viral de la GPL, por lo que permite crear trabajos derivados sin que estos se conviertan necesariamente en software libre.

- **PyGTK:** posiblemente sea la segunda opción más utilizada para la creación de interfaces gráficas con Python, solo por detrás de wxPython, con la que compite de tú a tú. PyGTK es un binding de GTK, la biblioteca utilizada para desarrollar GNOME. Cuenta con una API muy clara, limpia y elegante y es, además, muy sencillo de aprender, solo superado en ese aspecto por Tkinter. Las aplicaciones PyGTK son multiplataforma y son capaces de ejecutarse, sin modificar, en Linux, Windows, MacOS X y otras plataformas. PyGTK también cuenta con grandes herramientas para construir la interfaz de forma gráfica, como Glade o Gazpacho. PyGTK se distribuye bajo licencia LGPL.
- **PyQt:** es un binding de Qt para Python, sencillo de utilizar y con muchas posibilidades. Es especialmente interesante para el desarrollo en KDE dado que Qt es la librería utilizada para crear este entorno. No obstante el interés en Qt no se limita a KDE, sino que es una biblioteca multiplataforma que utiliza widgets nativos para las distintas plataformas. PyQt utiliza un modelo de licencias similar al de Qt, con una licencia dual GPL/PyQt Commercial.



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15



5.6.2 wxPython

De entre todos los toolkits estudiados se eligió **wxPython** para la realización del script por la comodidad y facilidad de uso de este lenguaje de programación.

wxPython es un wrapper Open Source para el toolkit anteriormente conocido como wxWindows: wxWidgets. wxPython es posiblemente el toolkit para desarrollo de interfaces gráficas en Python más popular, superando incluso a TKinter. Ya que cuenta con más y mejores widgets que TKinter, y ofrece un muy buen aspecto en todas las plataformas, utilizando MFC en Windows y GTK en Linux. Fue creada para proveer una manera barata y flexible de maximizar la inversión realizada en el desarrollo de GUIs y combina eficiencia, sencillez de uso e independencia de la plataforma. La biblioteca wxWidgets se caracteriza por ser multiplataforma, por lo que su uso junto a Python permite el desarrollo rápido de aplicaciones gráficas multiplataforma. Funciona como un módulo más de Python, debiendo ser importado al igual que cualquier otro.

Combina eficiencia, sencillez de uso e independencia de la plataforma, y entre sus características se puede destacar que está disponible en todas las plataformas más populares, es una API orientada a objetos y da soporte para la comunicación en red mediante sockets y librerías de multi-threading.

wxPython cuenta además con herramientas muy interesantes como wxGlade, una aplicación RAD para diseñar las interfaces gráficas de forma visual.

Sin embargo, la API (Application Programming Interface) adolece de una cierta falta de consistencia y un estilo alejado de Python y más cercano a C++, ya que, de hecho, uno de sus objetivos es no distanciarse demasiado del estilo de wxWidgets.

Tanto wxPython como wxWidgets se distribuyen bajo una licencia "wxWindows Licence", que consiste esencialmente en una LGPL con la excepción de que las obras derivadas en formato binario se pueden distribuir como el usuario crea conveniente.

Para la instalación de este toolkit en Raspberry Pi, al utilizar Raspbian que es un sistema basado en Debian, solo hace falta bajarse el paquete python-wxgtk2.8 de los repositorios a través del comando: **sudo apt-get install python-wxgtk2.8**



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15



5.6.3 Módulos y librerías implicadas

Para la resolución del script se han necesitado la utilización de diversos módulos para la realización de diferentes tareas.

Para la comunicación entre Arduino y Raspberry Pi, se ha utilizado el módulo **pyserial**. Este módulo ya viene instalado por defecto en Python, y es de los más importantes puesto que es la comunicación entre estos dos elementos en lo que se basa el script.

El módulo **RPI.GPIO** es necesario para el control de los GPIOs usando Python. Este módulo dará soporte para el control de la luminosidad de la lámpara, ya que su control depende de señales PWM, y con esta librería se es capaz de mandar señales PWM a través de los GPIOs especificando tanto la frecuencia de trabajo como el duty cycle. Este módulo también viene instalado por defecto con Python.

Mientras se desarrollaba el script se vio interesante el poder ver gráficas de los datos que se reciben, en la misma pantalla. Por ello se recurrió a la librería **matplotlib** que es una librería para impresión 2D en python que realiza figuras de calidad en una variedad de formatos impresos y entornos interactivos a través de plataformas. Además se recurrió a la librería **numpy** puesto que es un paquete fundamental para la computación científica con Python que ayuda a la biblioteca matplotlib. Estas dos librerías tuvieron que ser instaladas de los repositorios a través de los comandos: **sudo apt-get install python-matplotlib** y **sudo apt-get install python-numpy python-scipy**.

Se utilizan el módulo **threading**, para producir hilos que permiten que en las aplicaciones se ejecutan varias operaciones simultáneamente en el mismo espacio de proceso, y el módulo **time** que es fundamental para todas aquellas tareas que necesitan realizarse con un tiempo concreto. Estos módulos ya vienen instalados por defecto con Python.

Y finalmente los módulos **urllib** y **httplib** que son los encargados de enviar los datos a la aplicación web. El módulo **urllib** proporciona una interfaz de alto nivel para ir a buscar los datos a través de la World Wide Web. En particular, la función `urlopen()` que es similar a la función interna `open()`, pero acepta localizadores universales de recursos (URL) en lugar de nombres de archivo. Y el módulo **httplib** define clases que implementan el lado del cliente de los protocolos HTTP y HTTPS. Estos dos módulos también vienen instalados por defecto con Python.



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

5.6.4 Diseño final de la GUI

La realización de la GUI ha sido realizada de manera que el usuario puede ver todos los datos en una zona e interactuar con los actuadores en otra. Estas dos zonas están bien delimitadas definiendo que es cada cosa.

La zona de los sensores a su vez está dividida en otros departamentos que son los diferentes sensores identificados por su nombre. Estos departamentos están compuestos por una ventana de control, que es donde se ven los datos, y una etiqueta indicando las unidades de las magnitudes en los casos que son necesarios.

La zona de los actuadores, igual que la de los sensores, está dividida en otros departamentos que son los diferentes tipos de actuadores identificados por su nombre. Estos departamentos están formados por botones para activar los actuadores y por un spin control para cambiar entre los diferentes rangos de luminosidad. Los botones de las bombas además tienen dos posiciones: encendido y apagado, mientras que los otros solo entran en encendido y se deja el apagado a la automatización. En cuanto al spin control tiene 4 posiciones para la luminosidad: 0 – Apagadas, 1 – Luz diurna, 2 – Luz de luna, y 3 – Automático, en el que la luminosidad va variando simulando las horas del día y por la noche simulando el ciclo de la luna.

Por otro lado, para ver las gráficas se creó un menú para llamar a una ventana emergente donde poder verlas. Desde este menú se puede llamar a la gráfica de temperatura o la de pH. En esta ventana emergente se puede ver la gráfica de la magnitud elegida a tiempo real, y debajo un selector para elegir un rango de tiempo entre tiempo real, última hora y últimas 24 horas, y un botón para cerrar esta ventana más fácilmente.

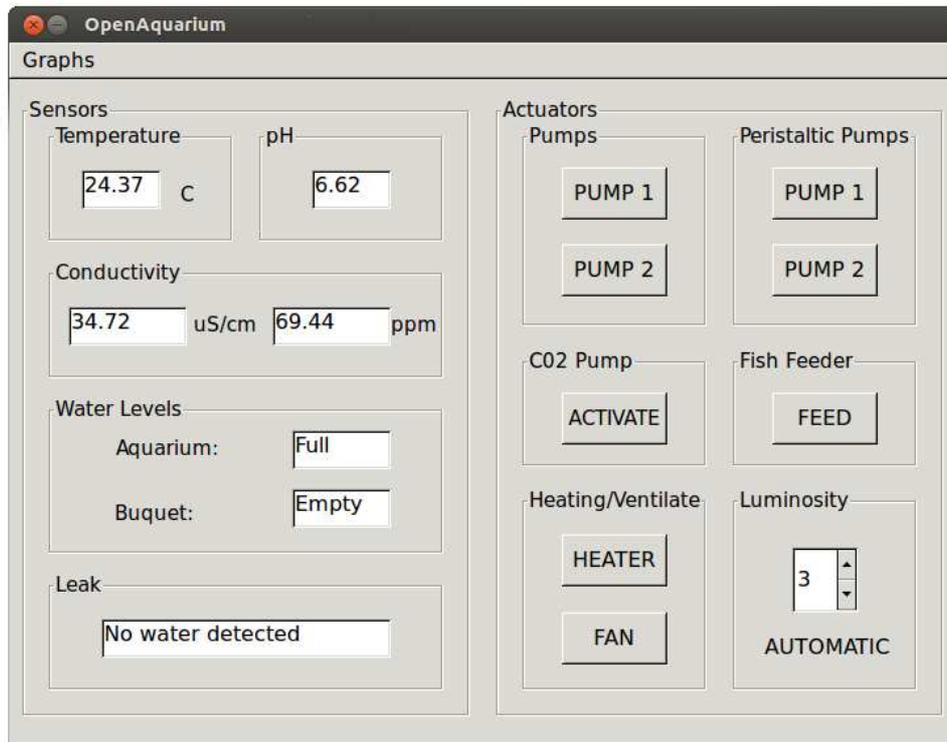


Fig. 48 Panel principal GUI

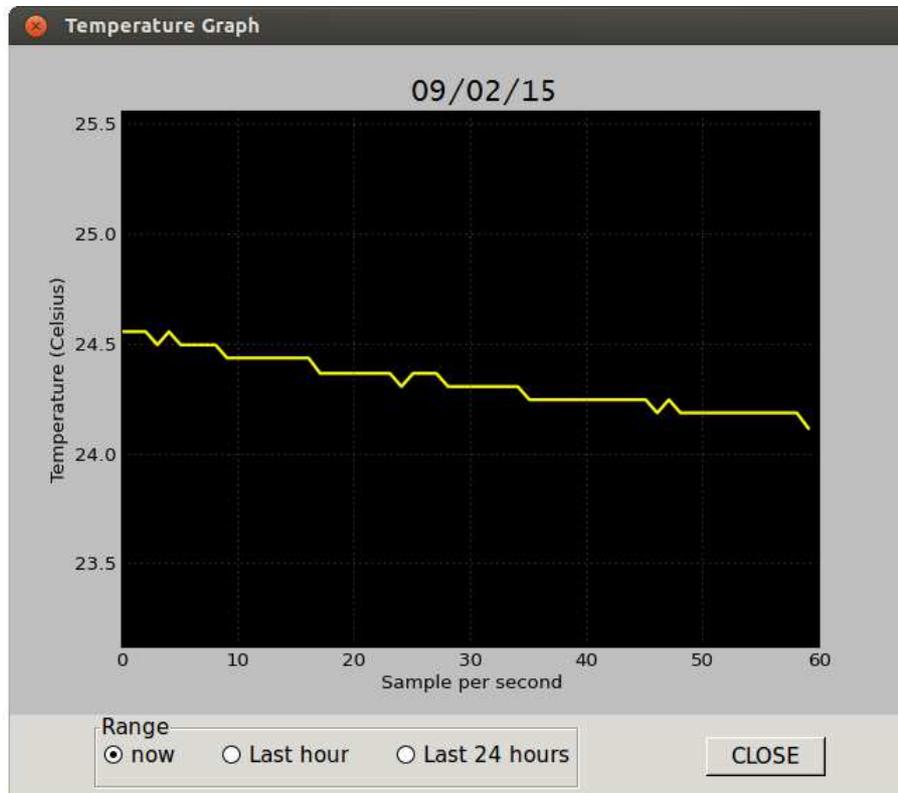


Fig. 49 Gráfica temperatura con la GUI



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

5.7 Configuración Raspberry Pi

Lo primero que hay que hacer para configurar la Raspberry Pi ya que la aplicación es una GUI, es habilitar el entorno gráfico, por ello utilizando el comando **sudo raspi-config**, se entra en **Enable Boot to Desktop/Scratch** y se selecciona **Desktop Log in as user 'pi' at the graphical desktop**.

Para adecuar el tamaño y calidad de la imagen a la pantalla, y que así al usuario le sea más fácil interactuar con la GUI, hay que acudir a **/boot/config.txt** y modificarle el apartado de la conexión HDMI configurando **HDMI_group=1** que es la configuración para las pantallas que no son monitores de ordenador, y **HDMI_mode=18** (576p a 50Hz) para adecuar el tamaño más apropiado para la pantalla. Además para que se parezca más a un panel de control, se instala un programa para que desaparezca el cursor cuando no se use: **sudo apt-get install unclutter**. También para resaltar más la GUI pero sin perder la funcionalidad de la barra de tareas de Raspbian, se va hacer de ésta lo más reducida posible. Para ello, haciendo doble click en ella, y en **configuración del panel**, se abre una ventana donde poder cambiar el tamaño. Se encontró que la altura y el tamaño de los iconos tenían que ser de 23 para poder seguir sacándole partido.

Para poder utilizar la función táctil de la pantalla es necesario reconstruir el kernel. En la web de Raspberry se explica cómo configurar el kernel a través de la herramienta menuconfig con la que añadir el driver que se necesita. Para utilizar esta herramienta hay que instalar los header de desarrollo ncurses para que pueda compilar correctamente: **sudo apt-get install libncurses5-dev**. Para modificar el kernel hay que preparar sus fuente e instalar algunas dependencias: **git clone --depth=1 https://github.com/raspberrypi/linux** y **sudo apt-get install bc**. Tras la instalación dentro de los archivos descargados, se usa el comando **make bcmrpi_defconfig** para configurar y se inicia la herramienta: **make menuconfig**. La ruta del driver dentro de esta herramienta es: Device Drivers --> Input device support --> Touchscreen --> USB Touchscreen Driver --> eGalax, eTurboTouch CT-410/510/700 device support. Cuando se guarda este driver toca compilar, que es un proceso muy lento: **make, make modules**, y **make install modules**. Por último se reemplaza este nuevo kernel por el antiguo: **sudo cp arch/arm/boot/Image /boot/kernel.img**.

Tras hacer estas modificaciones hay que reiniciar la Raspberry para que se guarden los cambios, e instalar un programa para poder calibrar el táctil, por ejemplo **xinput_calibrator**. Para que esta calibración sea permanente, hay que guardar los datos que te da este programa al finalizar de calibrar, en el directorio **/usr/share/X11/xorg.conf.d/01-input.conf**.



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15



Debido a que se va a utilizar la cámara para Raspberry Pi lo primero es habilitar su uso, por lo que se usa el comando **sudo raspi-config** y se entra en **enable camera**.

Hay unos cuantos protocolos de streaming para navegadores web. Por ejemplo, HLS es la elección de Apple, por lo que tiene un gran apoyo en iDevices pero no mucho en otros lugares. Otro, es el llamado MP4 fragmentado al que da soporte Adobe y Microsoft, pero requiere plugins de estas empresas en el equipo del usuario, por lo que los ordenadores Windows y Mac puede hacerlo, pero Linux y los móviles no pueden. Además, para todos los protocolos de streaming mencionados anteriormente hay una necesidad de tener un servidor de streaming que prepara el video para la transmisión mediante la segmentación y empaquetarlo.

Tras investigar cómo lo hacen cámaras IP vi que muchas de ellos usan un protocolo de flujo más antiguo llamado Motion JPEG o MJPEG. Esto es sólo un stream de imágenes JPEG individuales, uno tras otro. Siguiendo con mi investigación encontré MJPG-streamer, un pequeño código abierto para realizar streaming MJPEG mediante un código muy sencillo. Para poder usarlo lo primero es instalar la librería **libjpeg** utilizada para el manejo del formato de datos de imagen JPEG (**sudo apt-get install libjpeg62-dev**), y luego descárgarlo (**git clone https://github.com/Jacksonliam/mjpg-streamer.git ~/mjpg-streamer**), compilarlo e insertarlo en el directorio **/opt/**. Por último es decirle la ruta que va a usar cuando se usa:

```
LD_LIBRARY_PATH=/opt/mjpg-streamer/ /opt/mjpg-streamer/mjpg_streamer -i

```

Además, con el objetivo de tenerlo todo bajo control, se procede a que tanto al script de python como al código para realizar el streaming se ejecuten al reiniciarse Raspberry. De este modo se podrá interactuar siempre con el acuario y ver lo que ocurre en él desde la aplicación sin tener que acudir a Raspberry y acceder a su terminal para ejecutar los códigos que reinician los scripts, lo que es algo problemático. Para ejecutar el código del streaming, al ser de formato sh, sólo hay que acudir al archivo **/etc/rc.local** y escribir en él la ruta del archivo. Para ejecutar el script, lo primero es hacer un archivo que haga ejecutarse al script (con el comando **sudo python**) especificando la ruta completa de éste, luego se hace ejecutable con el comando **sudo chmod 755**, y por último se escribe la ruta de este archivo en el directorio **/etc/xdg/lxsession/LXDE/autostart** ya que este se ejecuta cuando se inicia el entorno gráfico de Raspbian.

Par crear un servidor web hay que dejar la IP de la Raspberry estática ya que muchos router son dinámicos y la IP cambia durante un tiempo, por ello hay que acudir al archivo



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

`/etc/modules/interfaces` y añadir la IP que se quiere que tenga.

5.8 PHP y HTML

Para el desarrollo de la aplicación web y la gestión de datos enviados a la base de datos, se estudiaron los lenguajes de programación PHP y HTML.

PHP es un lenguaje de programación de uso general de código del lado del servidor originalmente diseñado para el desarrollo web de contenido dinámico. Fue uno de los primeros lenguajes de programación del lado del servidor que se podían incorporar directamente en el documento HTML en lugar de llamar a un archivo externo que procese los datos. El código es interpretado por un servidor web con un módulo de procesador de PHP que genera la página Web resultante. PHP ha evolucionado por lo que ahora incluye también una interfaz de línea de comandos que puede ser usada en aplicaciones gráficas independientes. Puede ser usado en la mayoría de los servidores web al igual que en casi todos los sistemas operativos y plataformas sin ningún costo. Se considera uno de los lenguajes más flexibles, potentes y de alto rendimiento conocidos hasta el día de hoy.

HTML, siglas de HyperText Markup Language («lenguaje de marcas de hipertexto»), hace referencia al lenguaje de marcado para la elaboración de páginas web. Es un estándar que sirve de referencia para la elaboración de páginas web en sus diferentes versiones, define una estructura básica y un código (denominado código HTML) para la definición de contenido de una página web, como texto, imágenes, entre otros. Es un estándar a cargo de la W3C, organización dedicada a la estandarización de casi todas las tecnologías ligadas a la web, sobre todo en lo referente a su escritura e interpretación.

El lenguaje HTML basa su filosofía de desarrollo en la referenciación. Para añadir un elemento externo a la página (imagen, vídeo, script, entre otros.), este no se incrusta directamente en el código de la página, sino que se hace una referencia a la ubicación de dicho elemento mediante texto. De este modo, la página web contiene sólo texto mientras que recae en el navegador web (interpretador del código) la tarea de unir todos los elementos y visualizar la página final. Al ser un estándar, HTML busca ser un lenguaje que permita que cualquier página web escrita en una determinada versión, pueda ser interpretada de la misma forma (estándar) por cualquier navegador web actualizado.



5.9 Creación servidor web

Como ya se ha comentado gracias a la gran versatilidad de Raspberry Pi, se han hecho multitud de proyectos, y entre ellos está el hacer de ésta un servidor web. Esto es de gran utilidad para no tener que usar un ordenador como servidor y así ahorrar en elementos del sistema y en energía, puesto que Raspberry Pi consume bastante menos que un ordenador normal.

Antes de empezar a instalar el servidor lo primero que hay que hacer es crear y dar permisos al grupo que usa apache por defecto: **sudo addgroup www-data** y **sudo usermod -a -G www-data www-data**.

Para instalar el servidor y que no se produzca ningún error hay que tener Raspberry actualizada (**sudo apt-get update**). Tras la actualización se instala el servidor Apache y PHP y se hace que se inicie Apache: **sudo apt-get install apache2 php5 libapache2-mod-php5** y **sudo /etc/init.d/apache2 restart**.

Tras realizar esto se procede a instalar MySQL, que es un sistema de gestión de bases de datos relacional, multihilo y multiusuario, y phpMyAdmin, que es una herramienta escrita en PHP con la intención de manejar la administración de MySQL a través de páginas web, utilizando Internet. Actualmente puede crear y eliminar bases de datos, crear, eliminar y alterar tablas, borrar, editar y añadir campos, ejecutar cualquier sentencia SQL, administrar claves en campos, administrar privilegios y exportar datos en varios formatos. Aunque antes hay que activar la interfaz loopback ya que si no se producirá un error al instalar MySQL: **sudo ifup lo**. Y luego: **sudo apt-get install mysql-server mysql-client php5-mysql phpmyadmin**. Para finalizar de configurar hay que añadir **extension=mysql.so** al archivo **/etc/php5/apache2/php.ini**.

Para ver la página web en cualquier sitio, ésta tiene que tener siempre la misma IP. Para ello hay que registrarse en una de las páginas que crean dominios gratuitos para conseguir uno. Al crear un dominio este sitio web dice si está disponible y le asignará una IP al dominio.

Respecto a Raspberry hay que instalar el programa ddclient para que gestione la IP del dominio credo para que el sitio web esté siempre visible: **sudo apt-get install ddclient**.

5.10 Aplicación web

Al inicio del proyecto cuando se consideró hacer una aplicación web, se encontró una página a la que mandar los valores de los sensores para verlos en forma de gráficas. Esta página es ThingSpeak, un sitio web donde se pueden cargar los datos en su servidor y luego recuperarlos en forma de tablas y gráficos. Estos gráficos se pueden ver en la misma página o pueden ser colocados en un sitio web personal. Tienen una extensa API para configurar los gráficos y los datos, aunque la restricción principal es que solo se pueden recuperar 8.000 datos de una vez. Por estas razones, el uso de este sitio web es una forma muy útil, sencilla, rápida y barata para conocer el estado de los principales parámetros del acuario.

Como se pueden llevar las gráficas a un sitio web personal, esta opción no se desechó, sino que se unió. Se envían los datos de temperatura, pH y conductividad a esta página y luego se retocan las gráficas al gusto para poder exportarlas a la aplicación real. El envío de estos datos se realiza desde el script de Python, dándole la dirección, y colocando los valores de un modo concreto seguidos de la clave del canal.

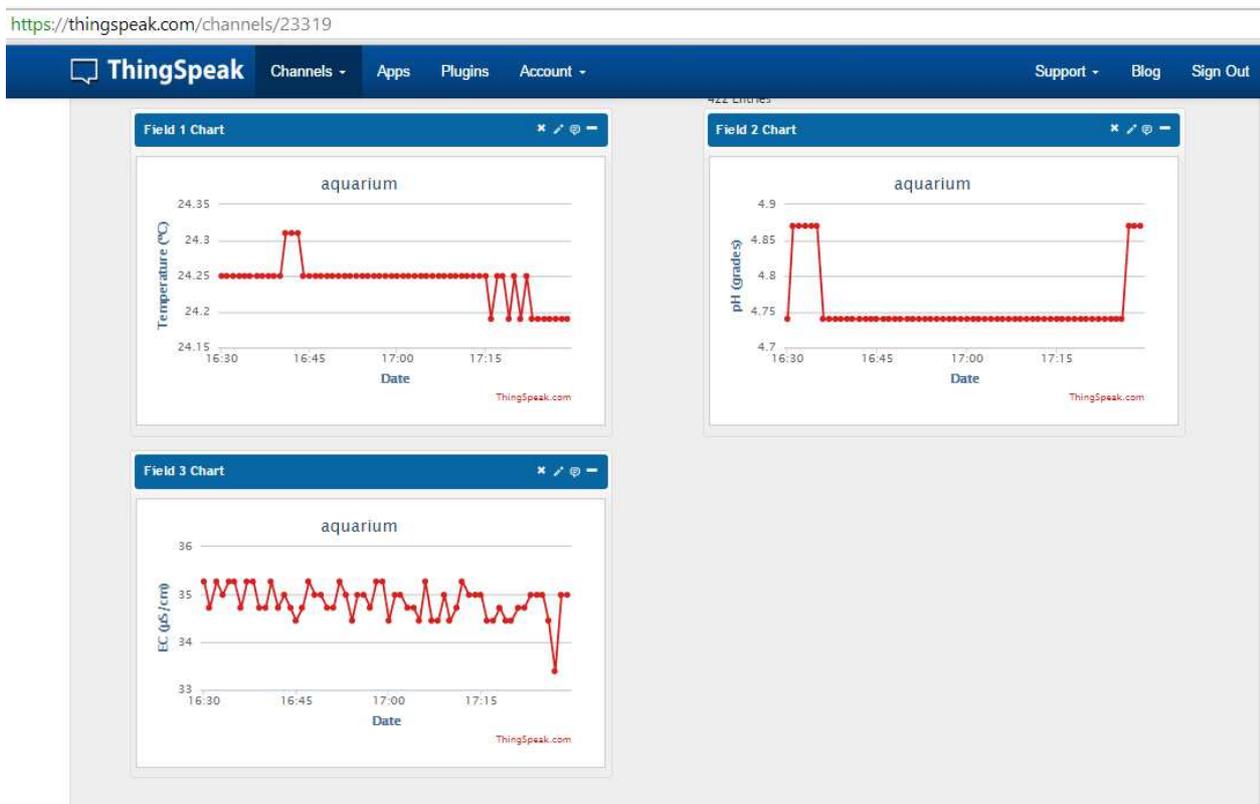


Fig. 50 Gráficas realizadas por la página ThingSpeak



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15



Ya que la filosofía de este proyecto se basa en la tecnología Open Source lo que conlleva un continuo desarrollo y mejora de los sistemas, se procede a utilizar el sistema de gestión de base de datos que se ha desarrollado para la aplicación de Open Aquarium. Este sistema se basa en varios códigos php que realizan la gestión de la base de datos, el guardar los valores de los sensores, el encendido/apagado de los actuadores, la autenticación, tanto para el usuario como para la base de datos y hay un archivo central que centraliza todas estas operaciones. De estos archivos en principio no se va a usar los códigos que utilizan los actuadores por considerarse que cualquier operación implicada con algún actuador que necesite el acuario se realizará en la automatización, pero se dejan ya que en el archivo central **openaquarium.php** se hace una llamada a estos códigos y podría producirse un error. Además se dejan también los archivos para que se puede ver los datos de los sensores, si se desea, con un formato de app Android.

Debido a que se utilizan estos códigos también se han de crear la base de datos y las tablas en MySQL como se realiza en el sistema de Open Aquarium. Para crear esta base de datos se puede hacer desde la terminal de Raspberry con el comando **mysql -u root -p**, o con la herramienta phpmyadmin, a la que se puede acceder introduciendo en un navegador web la dirección IP de Raspberry seguida por /phpmyadmin. Los archivos y el código para crear la base de datos se encuentran en el tutorial de Open Aquarium.

Estos archivos han de añadirse al directorio **/var/www/** de Raspberry Pi, que se creó cuando se hizo de ésta un servidor web. Se modificó el archivo config.php que es el de autenticación para adecuarlo a este sistema, añadiendo el nombre de la base de datos, el usuario y contraseña de MySQL, y el host que en este caso es local. Los datos son enviados desde el script de Python al archivo **set_sensors.php** que es el archivo que guarda el valor de los datos de los sensores, en el orden en que se mandan en el sistema de Open Aquarium. Además a este directorio se añade otro archivo php que hay que crear, para poder ver el streaming en la aplicación. Este archivo se llama **stream.php** y en él se configura el puerto que utiliza la cámara y la dirección IP donde enviar el streaming para no tener que configurarlo luego en la aplicación.

En el código HTML lo primero se define la fuente de todos los objetos, y luego se hace una llamada al archivo stream.php para mostrar el streaming, una llamada a las gráficas de ThingSpeak y se crea una tabla en la que se muestran los datos de los sensores con una referencia debajo expresando la última actualización. Para mostrar los datos se hace una llamada al archivo openaquarium.php cada minuto indicando el usuario y la contraseña definidas en el archivo config.php, que a su vez hace una llamada a dos funciones, una que recoge el valor de los sensores para mostrarlos en la tabla y la otra que actualiza la hora que indica la última

actualización de esos datos.

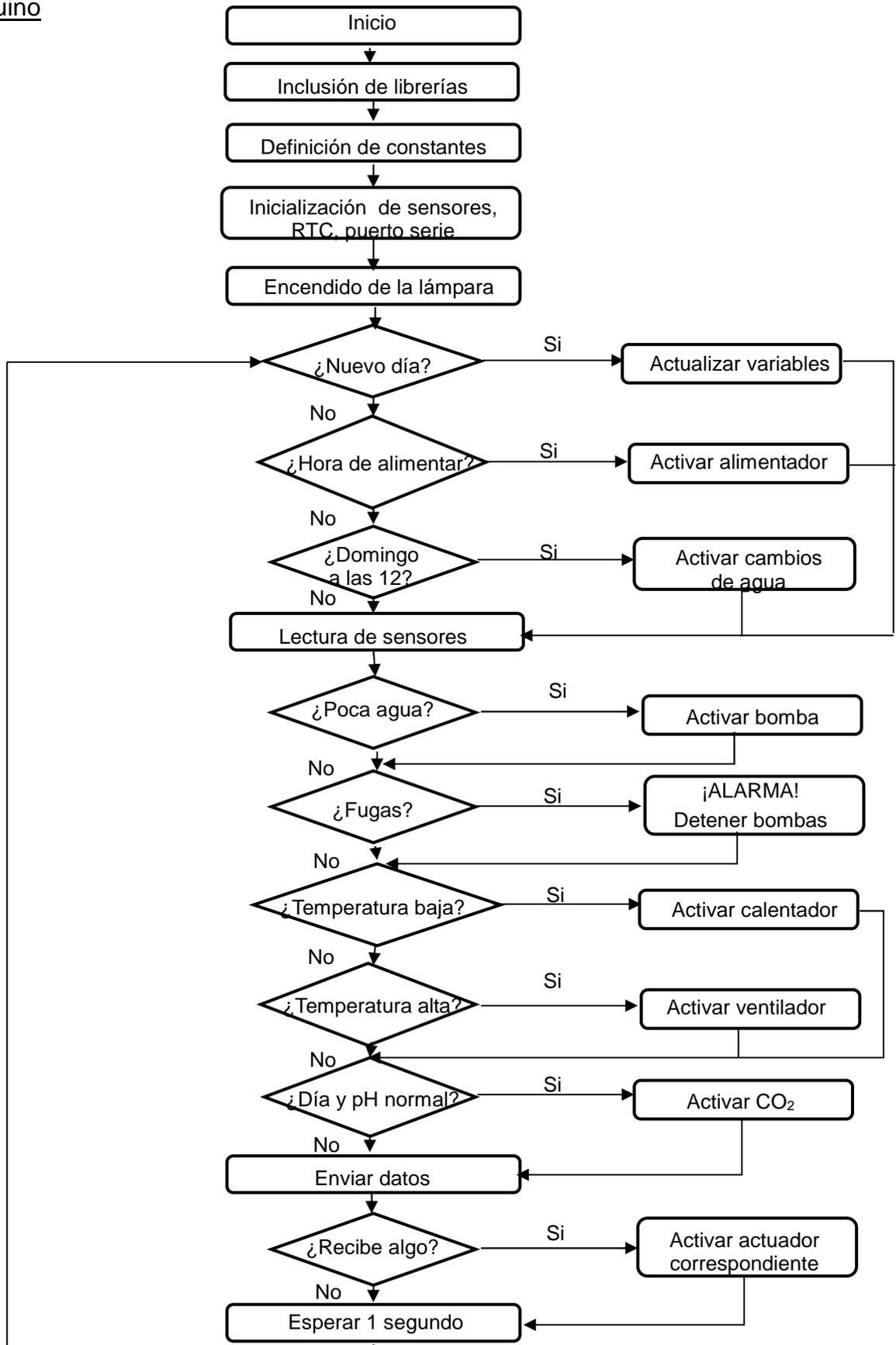
El aspecto de la página es muy sencillo. En la parte superior se encuentra el streaming del acuario junto a la tabla de los valores de los sensores, y debajo se encuentran las tres gráficas de ThingSpeak. Se ha hecho que la gráfica de temperatura sea el doble que las de pH y electroconductividad porque se considera un factor más determinante, y por colocarlos de una forma que sea visualmente más llamativa.





5.11 Diagramas de flujo

Sketch Arduino





El programa de Arduino se dedica a la automatización del acuario, envío de los datos de los sensores y repuesta, a una orden dada por parte del usuario, mediante activación del actuador correspondiente. Respecto de la automatización se encarga en un principio de encender la lámpara debido a la importancia de la luz para las plantas, aunque el usuario puede decidir después si la quiere apagar o encender la luz al máximo en vez de regulada por horas. Luego se encarga de alimentar dos veces al día a los peces, cambiar el agua una vez a la semana, inyectar agua cuando el nivel haya bajado, controlar posibles fugas, mantener la temperatura en un rango constante e inyectar CO₂ durante el día si el pH no ha disminuido demasiado.

Para la realización del sketch de Arduino, como ya se ha comentado, se han utilizado en gran medida las funciones de la librería Open Aquarium aunque alguna haya tenido que ser modificada. A continuación se hace una pequeña descripción de las funciones utilizadas.

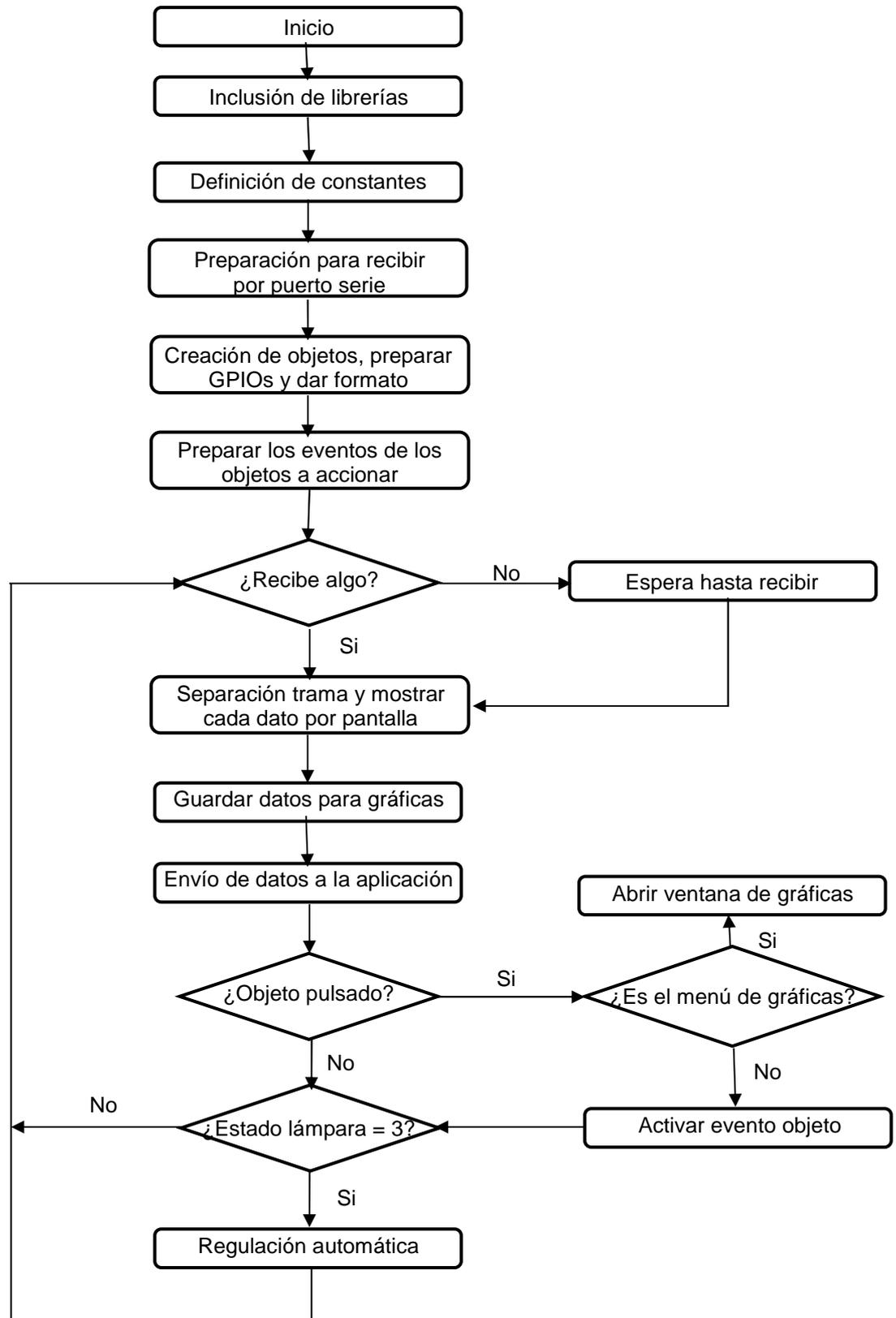
Función	Descripción
OpenAquarium.init()	Inicializa los sensores, la RTC y las variables
OpenAquarium.newday()	Reinicia las variables
OpenAquarium.getTime()	Recoge la hora de la RTC
OpenAquarium.readwaterlevel()	Lee el valor del sensor de nivel que se indique
OpenAquarium.readtemperature()	Lee el valor del sensor de temperatura
OpenAquarium.calibrateEC()	Calibra el sensor de electro-conductividad
OpenAquarium.calibratepH()	Calibra el sensor de pH
OpenAquarium.readpH()	Lee el valor del sensor de pH en mV
OpenAquarium.pHConversion()	Convierte el valor de pH de mV a grados
OpenAquarium.readResistanceEC()	Lee el valor del sensor de electro-conductividad en resistencia
OpenAquarium.ECConversion()	Convierte el valor de electro-conductividad de resistencia a $\mu\text{S}/\text{cm}$
OpenAquarium.empty()	Activa la bomba que vacía el acuario, activa las bombas peristálticas para tratar el agua, y rellena el acuario
OpenAquarium.fill()	Para la bomba activada con la función empty() cuando se detecta que el acuario está lleno
OpenAquarium.feedfish()	Activa el alimentador de peces
OpenAquarium.readyforfeeding()	Prepara el alimentador para que funcione otra vez
OpenAquarium.pumpON()	Activa la bomba de agua que se indica
OpenAquarium.perpumpON()	Activa la bomba peristáltica que se indica
OpenAquarium.pumpOFF()	Desactiva la bomba de agua que se indica
OpenAquarium.perpumpOFF()	Desactiva la bomba peristáltica que se indica
OpenAquarium.sendPowerStrip()	Envía el código por RF que activa/desactiva el actuador que se indique

Fig. 52 Tabla funciones de la librería Open Aquarium

En el CD se encuentra tanto el sketch como las librerías modificadas.

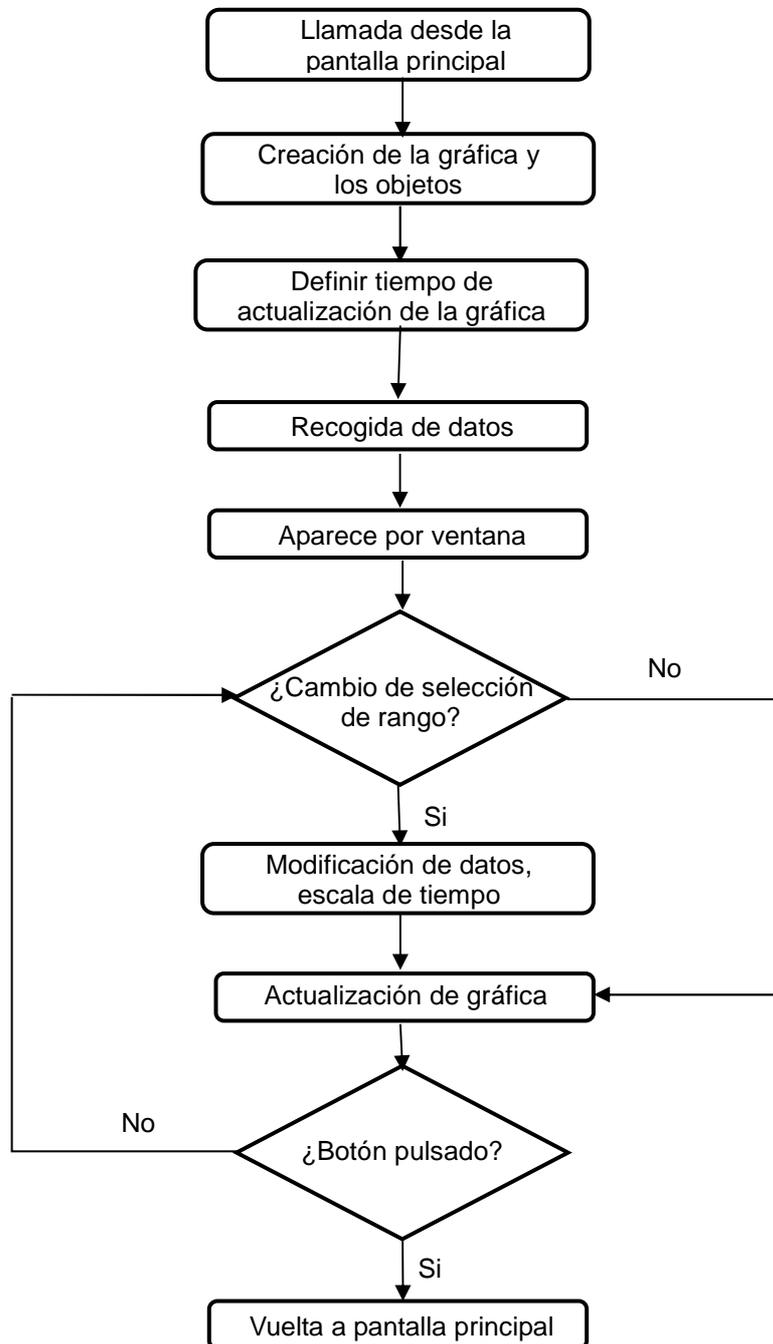


Script Python





Script Python Ventana gráficas





Para la realización del script de la GUI se han creado diversas funciones en las clases creadas: **TerminalFrame**, que es la clase en la que se desarrolla el programa principal, y **TempGraphDialog** y **pHgraphDialog**, que son las clases para crear las ventanas de dialogo donde visualizar las gráficas.

TerminalFrame:

Función	Descripción
_init()	Selecciona el puerto por el que llegan los datos, crea los objetos con los que interactuar con Arduino, el menú donde llamar a las gráficas, da la orden de empezar a mandar señales PWM a los GPIOs, y llama al resto de funciones para que empiece a funcionar
__set_properties	Define título y tamaño de la GUI
__do_layout	Da formato a la GUI
__attach_events	Crea los eventos que han de realizar los objetos cuando se les utiliza
StartThread()	Comienza el hilo llamando a la función que comprobará si se recibe algo
StopThread()	Para el hilo cuando no se recibe nada
feed(event), TurnOnfan(event), TurnOnheater(event), TurnOnC02pump(event)	Esta función se activa cuando se aprieta el botón correspondiente y manda el comando correspondiente a Arduino para activar el actuador que sea
TurnOnpump1(event), TurnOnperpump1(event), TurnOnpump2(event), TurnOnperpump2(event)	Se activa cuando se aprieta el botón correspondiente. Al hacerlo manda la orden para encender esa bomba, y cambia la etiqueta del botón hasta que se vuelve a apretar ese botón que manda el comando para apagar la bomba
Luminosity(event)	Se activa cuando se modifica el spin control, cambiando el rango de luminosidad
OnClose ()	Función que 'limpia' todo las funciones usadas cuando se cierra la GUI
TempGraph(), pHGraph()	Funciones utilizadas para llamar a las clases que crean las gráficas, que son activadas cuando se llaman desde el menú
ComPortThread()	Función usada para recoger la trama que se recibe
OnSerialRead()	Función que separa la trama y envía cada parámetro a su ventana de control para mostrarlo por pantalla y que guarda los datos para la realización de las gráficas
dimmer()	Función para saber el día del ciclo lunar, y para hacer la regulación automática horaria cuando el spin control está en estado 3
database()	Función para enviar los datos a la página ThingSpeak y a la base de datos

Fig. 53 Tabla de funciones de la clase *TerminalFrame*



TempGraphDialog y pHgraphDialog

Función	Descripción
_init()	Función para crear la gráfica, y los objetos para cambiar los rangos de la gráfica y para cerrar la ventana, da formato para dejarla más estética y especifica el tiempo de actualizar
SetVal(event)	Función a la que se accede cuando se utiliza uno de los radio button, y que modifica los datos que muestra la gráfica y el rango de tiempo
draw_plot()	Función para dibujar la gráfica, especificando sus rangos, y que la muestra por pantalla
on_redraw_timer	Función que llama a la función draw_plot, en los tiempos especificados en _init, para actualizar la gráfica
OnClose (event)	Función para cerrar la ventana y volver a la pantalla principal cuando se aprieta el botón creado para facilitar esta tarea

Fig. 54 Tabla funciones de las clases *TempGraphDialog* y *pHgraphDialog*

En el CD se encuentra el script, los códigos para el streaming y para la aplicación web.



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

6. Pruebas realizadas

En este apartado se hará una pequeña descripción de las pruebas más relevantes que se han hecho durante la realización del proyecto para el funcionamiento del sistema.

6.1 Pruebas cambios de agua

Como ya se ha comentado, los cambios de agua son fundamentales, ya que son la primera línea de defensa para evitar que los peces contraigan enfermedades.

En principio en el programa solo se contemplaba la realización de estos cambios una vez por semana. Se pudo contemplar que la función **empty()** de Open Aquarium hacía que a su hora y correctamente, las bombas procedieran primero a un vaciado del acuario, luego a un tratado del agua a usar, y por último a un rellenado del acuario.

Tras la recomendación de profesionales, se añadió una función para rellenar el acuario cuando éste se vaciara por evaporación del agua. Esta función, que es solo encender la bomba correspondiente cuando el sensor de nivel detecta que ha disminuido el agua, tras diversas pruebas se pudo observar que producía un error cuando se hacían los cambios de agua con la función **empty()**. Cuando se estaba vaciando el acuario, sin llegar a terminar esta acción, el sensor de nivel del acuario detectaba que el agua había disminuido y empezaba a la vez a rellenar agua que todavía no había sido tratada, algo que no debería hacer. Por algún motivo salía de la función y al leer el sensor activaba la bomba. Tras varias deliberaciones, se optó por añadir una condición de que el sensor solo activara la bomba todos los días salvo los que hubiera programado que se realizaran cambios de agua. Se consideró que debido a que solo un día a la semana se producían los cambios, este día no podría bajar demasiado el nivel, y si así fuera, la función **empty()** junto con la función **fill()** aseguran que cuando se rellena el acuario, este sea hasta que el sensor de nivel detecta que está lleno, por lo que cuando se produjera este cambio el acuario se quedaría en el nivel óptimo de agua.

Además, al añadir esta función hubo que añadir otra para parar la bomba cuando el sensor detectará que el acuario estaba lleno, para evitar que se produjeran desbordamientos. Pero se pudo observar que esto producía problemas cuando se querían hacer cambios de agua manuales, que no fueran suficientes para que el sensor detectara que el agua hubiera bajado el límite mínimo establecido, o inyectar alguna sustancia para mejorar la salud del acuario, desde la GUI. Para arreglarlo se añadió una variable booleana que tiene que cumplirse para que se active esta



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15



última función, y que es modificada para desactivar la función cuando se activa la bomba desde la GUI y que vuelve a su estado original para activarla cuando se apaga desde la GUI.

6.2 Pruebas del alimentador de peces

La alimentación automática de los peces es también una de las funciones fundamentales del sistema. Con la función **feedfish()**, programándola a una hora concreta, el alimentador da una vuelta completa. Tras diversas pruebas, y con la inclusión de la función **readyforfeeding()**, de obligada inclusión para poder utilizar la función **feedfish()** más de una vez, se pudo comprobar que funcionaba correctamente a la hora a la hora que estaba programada.

Aunque está función se vio que funcionaba correctamente, tras las pruebas para poder hacer alimentaciones extra a través de la GUI, se pudo observar que al mandar el comando el alimentador se encendía pero no hacía una vuelta, sino que no paraba de girar, no se desconectaba. Para arreglar esto, se hizo un programa que encendía y apagaba el alimentador y contaba cuantas acciones hacía hasta completar una vuelta completa. Tras averiguar el número exacto se añadió al programa principal esta acción con el número hallado y a continuación la función **readyforfeeding()** para que se asegurará de que se produjera la siguiente alimentación a la hora correcta. Después de probarlo todo junto se observó que funcionaba correctamente.

6.3 Pruebas comunicación Arduino/Raspberry

La comunicación entre Arduino y Raspberry es una parte muy importante del proyecto, puesto que es necesaria tanto para la monitorización como para el control manual por parte del usuario.

Esta comunicación es mediante puerto serie; los datos de los sensores son enviados en una línea separados por distintos caracteres para que cuando lleguen a Raspberry y se separe la trama sea más fácil separar e identificar cada parámetro. Por otro lado, para activar o desactivar un actuador desde Raspberry se realiza mandando un carácter distintivo para identificar la acción a realizar del actuador correspondiente.

Se pudo observar tras diversas pruebas que cuando se daba una orden de activar un actuador desde la GUI, el actuador correspondiente realizaba la acción que tenía que hacer, siempre que no hubiera una tarea dentro de la automatización que lo impidiera, como el que no se pueda encender el ventilador con temperaturas bajas, o en los casos ya comentados del alimentador de peces y de la bomba de rellenado que funcionaron bien tras las modificaciones



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

realizadas.

Por otro lado, en la recepción de los datos de los sensores en Raspberry, al utilizar las funciones para ver lo que se recibe por puerto serie, se observó que se producía una fragmentación de la trama lo que dificultaba la separación e identificación correcta de los parámetros. Después de observar las librerías de Open Aquarium, me di cuenta de que las funciones utilizadas para medir la temperatura, el pH y la electro-conductividad, al ser más complejas necesitaban de retrasos para poder medir bien estas unidades. Y era por estos retrasos por los que se producía la fragmentación de la trama.

Esta fragmentación producía que se recibieran casi todos los datos por separado. Para poder identificar mejor que parámetro era el que se recibía se introdujeron más caracteres, que el programa de Python tenía que reconocer para poder mostrarlos en la GUI. Con esta solución se logró que se vieran estos datos de forma ordenada aunque en algunos casos se produjera algún pequeño desfase cuando se modificará alguno de estos parámetros.

6.4 Pruebas regulación luminosidad

Después de la compra de la lámpara debido a la necesidad de mayor potencia para el desarrollo de las plantas, como ya se ha comentado, se procedió a realizar pruebas para saber cómo variar la luminosidad de la lámpara.

La lámpara tiene dos canales independientes para regular la luminosidad de los LEDs de luz blanca y los LEDs de luz azul. Tras abrirla se vio que había una indicación de que el control de la luminosidad se realizaba con señales PWM y se comprobó que metiéndole una señal de 5 voltios por estos canales se apagaban los LEDs del canal correspondiente.

Para la creación de estas señales PWM se recurrió a los GPIOs de Raspberry, y a la utilización del módulo **RPi.GPIO** que tiene la función para realizar señales PWM. Al utilizar el módulo se define el GPIO a utilizar a la frecuencia que se desea y luego el Duty Cycle (ciclo de trabajo), aunque es posible cambiar estos dos factores gracias a las funciones **ChangeDutyCycle()** y **ChangeFrequency()**.

Debido a que la luz se apaga con una señal de 3.3V (voltaje que suministra Raspberry por sus GPIOs), el Duty Cycle será de 1 (100 en la función) cuando se desea apagar uno de los dos tipos de luminosidad y 0 cuando se desea que esta luminosidad sea máxima.

Cuando en la GUI se selecciona que la iluminación sea automática (algo que ya ocurre al inicio) el script irá modificando el Duty Cycle. Este Duty Cycle, en el canal de luz blanca, va disminuyendo desde un amanecer ficticio y declarado en la programación, hasta el mediodía, y volverá a aumentar desde este punto hasta el anochecer. En el caso de la luz azul, el Duty Cycle será máximo durante el día, y por la noche, cada día, irá cambiando para aumentar o disminuir la luminosidad simulando las fases de la luna.

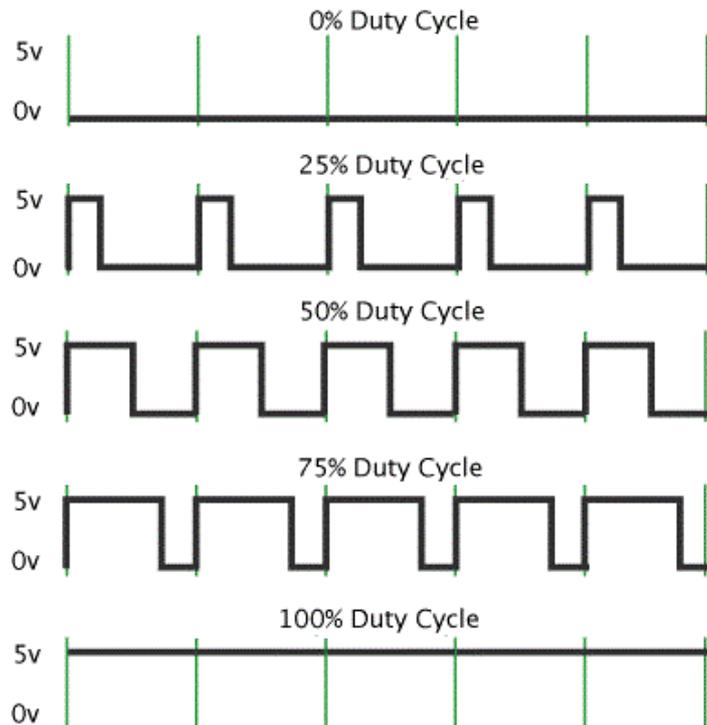


Fig. 55 Variaciones del Duty Cycle

Se comprobó que la luminosidad iba cambiando a lo largo del día debido al cambio del Duty Cycle, pero que era fluctuante, lo que era perjudicial ya que podría marear a los peces. Por ello se fueron realizando diversas pruebas cambiando la frecuencia hasta que se encontró una luminosidad lo más regular posible. Esto se alcanzó con una frecuencia de 100 Hz.



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

7. Conclusiones

Se han conseguido alcanzar los objetivos que se habían planteado, consiguiendo ampliar el proyecto previo realizado por Marcos Martínez, Open Aquarium, integrando otra plataforma totalmente distinta a Arduino, como es Raspberry, para ser capaz de monitorizar en todo momento en tiempo real y automatizar el cuidado de un acuario pudiendo ser de cualquier tipo porque cubre las necesidades básicas de todos ellos.

Se ha dedicado tiempo a realizar pruebas y ensayos del sistema, para descubrir fallos en la programación, logrando así prevenir futuros problemas que puedan aparecer. Para ello se ha puesto al sistema a trabajar en escenarios límites, como por ejemplo colocando el acuario en una habitación con poca ventilación en agosto y con las luces del acuario encendidas, o realizando pruebas durante periodos de funcionamiento largos. Los objetivos parciales establecidos han sido conseguidos en el orden ideado:

- Se completó un exhaustivo proceso de investigación inicial, relacionado con los tipos de acuario, sus necesidades y cuidados, reuniendo información general del tema.
- Se consultaron las necesidades de los usuarios en los criterios de automatización.
- Se realizó un estudio básico de la tecnología con la que se iba a trabajar: Arduino, Open Aquarium y Raspberry Pi.
- Para el diseño final del sistema, se realizó en una serie de módulos, lectura de los sensores por parte de Arduino y envió de datos a Raspberry, muestra de esos datos en la GUI, activación de los actuadores desde la GUI, y automatización de los procesos necesarios. Tras diversas pruebas de funcionamiento, se diseñó un prototipo final funcional, con la unión de todos estos módulos.
- Se realizó un estudio del uso de la cámara de Raspberry y su incorporación al sistema para crear un streaming en una aplicación web.
- Se realizó una aplicación web en la que consultar el estado del acuario a través de un streaming, de una tabla con los valores de los sensores y las gráficas de los parámetros más característicos.

Se ha conseguido que el conjunto de la placa Open Aquarium trabajando con Arduino en colaboración con Raspberry Pi formen un sistema de altas prestaciones, muy completo, fiable y flexible, lo que junto al precio competitivo y facilidad de uso permitirán llegar a un gran número de usuarios. Como el acceso al software final es sencillo y está disponible, cualquier persona con algún conocimiento de programación puede usar o personalizar su sistema más allá del original.



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

7.1 Conclusiones personales

Ha sido una experiencia muy gratificante tener la oportunidad de trabajar como ingeniero de investigación y desarrollo de Libelium, y más concretamente en el departamento Cooking Hacks, donde he podido aprender y donde se me ha provisto de todo el hardware y el conocimiento que he necesitado. La aplicación real de los conocimientos adquiridos durante la carrera y la adquisición de otros muchos durante estos meses de trabajo, ha hecho que tantos años de estudio hayan merecido la pena.

He podido ver de primera mano las dificultades por las que pasa un proyecto real en una empresa y con las que el ingeniero debe lidiar para sacarlo adelante como pueden ser la búsqueda de material entre los distintos distribuidores, tiempos de entrega, tiempos estimados para realizar las tareas.

7.2 Agradecimientos

En primer lugar quiero dar las gracias a toda la gente de Libelium, y en especial al departamento de Cooking Hacks, Luis Martín y Jorge Casanova por aceptarme para realizar este proyecto con ellos y haberme guiado en el proceso de diseño de un producto real. También quiero dar las gracias a toda la gente del departamento de investigación y desarrollo, como David Gascón, Marcos Yarza, Ahmad Saad, Alejandro Gallego, Javier Siscart o a Javier Solobera entre otros, por haberme ayudado con todas mis dudas.

En especial, quiero agradecer a Marcos Martínez por haberme guiado en la correcta utilización del shield Open Aquarium diseñado por él, que ha sido la base de este proyecto.

También quiero dar las gracias a Bonifacio Martín del Brío, tanto por haber sido el tutor de mi proyecto, como profesor de la carrera, y al resto de profesores de la escuela de ingeniería que me han dado clases en algún momento.

Por último quiero dar las gracias a mi familia por todo su apoyo durante todo este tiempo, a mis amigos y compañeros de trabajo, como Víctor Boría, Luis Miguel Martí o Andrés Faló. Todos han aportado algo necesario en algún momento en particular del proyecto.

A todos muchísimas gracias.



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

8. Glosario

WSN - Wireless Sensor Network (Red sensorial inalámbrica)

GPL - General Public License (Licencia Pública General)

GUI - Graphical User Interface (Interfaz gráfica de usuario)

PLC - Programmable Logic Controller (Controladores lógicos programables)

IDE - Integrated Development Environment (Entorno de desarrollo integrado)

SPI - Serial Peripheral Interface (Interfaz de comunicación serie)

FSK - Frequency Shift Keying (Modulación por desplazamiento de frecuencia)

SBC - Single Board Computer (Ordenador de placa reducida)

CPU - Central Processing Unit (Unidad central de procesamiento)

GPU - Graphics Processing Unit (Unidad de procesamiento gráfico)

GPIO - General Purpose Input/Output (Entrada/Salida de propósito general)

UART - Universal Asynchronous Receiver-Transmitter (Transmisor-Receptor Asíncrono Universal)



9. Bibliografía

www.libelium.com

www.cooking-hacks.com

www.arduino.com

www.raspberrypi.org

www.raspbian.org

http://dplinux.net/quia-raspberry-pi/#_RefHeading_1706_924516217

<https://www.python.org/>

<http://librosweb.es/libro/python/>

<http://mundogeek.net/archivos/2008/11/24/interfaces-graficas-de-usuario-en-python/>

<http://zetcode.com/wxpython/widgets/>

<http://www.wxpython.org/>

<http://geekytheory.com/arduino-raspberry-pi-lectura-de-datos/>

<http://batchloaf.wordpress.com/2012/05/11/complotter-a-graphical-serial-port-data-logger/>

<http://www.blendedtechnologies.com/realtime-plot-of-arduino-serial-data-using-python/231>

<http://www.raspberrypi.com/pi/pi-operating-systems/raspbian/scripts>

<http://www.raspberrypi.org/forums/viewtopic.php?f=31&t=43509>

<http://raspi.tv/2013/rpi-gpio-0-5-2a-now-has-software-pwm-how-to-use-it>

<https://thingspeak.com/>

<http://www.australianrobotics.com.au/news/how-to-talk-to-thingspeak-with-python-a-memory-cpu-monitor>

<https://docs.python.org/2/library/urllib.html>

<https://docs.python.org/2/library/httplib.html>

<http://geekytheory.com/tutorial-raspberry-pi-crear-servidor-web/>



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

<http://www.raspberrypi.org/help/camera-module-setup/>

<http://blog.miguelgrinberg.com/post/stream-video-from-the-raspberry-pi-camera-to-web-browsers-even-on-ios-and-android>

<https://miguelmota.com/blog/raspberry-pi-camera-board-video-streaming/>

<https://www.tinkercad.com>



Anexo A: Código aplicación web

```
<html>
  <head>
    <script src="openaquarium-app/scripts/jquery.min.js"></script>
    <link href='http://fonts.googleapis.com/css?family=Bree+Serif'
rel='stylesheet' type='text/css'>
    <style type="text/css">
      table{
        font-family: 'Bree Serif', serif;
        border: none;
        width:100%;
      }
      h1, h2{
        font-family: 'Bree Serif', serif;
        color:#173c58;
        margin: 0 0 15px 0;
      }
      h2{
        font-size: 22px;
        text-align:center;
      }

      #cam{
        width:720px;
        float: left;
      }
      #sidebar {
        width: 250px;
        float: left;
        margin: 0 0 0 15px;
      }
      #data .param{
        color:#173c58;
        width: 60%;
        background-color: #ddd;
        border: none;
        font-weight: bold;
        padding: 7px;
        font-size: 18px;
      }
    </style>
  </head>
</html>
```



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

```
#data .val{background-color: #ddd;
border: none;
padding: 7px;
color: #444;
text-align: right;
font-weight: bold;
font-family: Arial,Helvetica,sans-serif;
font-size: 15px;
}
#updated{
background-color:#173c58;
font-size: 1vw;
font-family: Arial,Helvetica,sans-serif;
font-weight: bold;
color: #fff;
margin:10px;
padding:3px;
text-align:center;
font-size: 14px;
}
.clear{
clear:both;
}
.board{
float: left;
width: 493px;
height: 375px;
border: 1px solid #cccccc;
}
#temperature{
width:997px;
clear:both;
margin: 10px 0;
}
#ph{
margin: 0 10px 0 0;
clear:both;
}
#conductivity{
}
</style>
</head>
```



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15



```
<body>
  <h1>Aquarium</h1>
  <div id="board">
    <div id="live">
      <div id="cam"></div>
      <div id="sidebar">
        <h2>Aquarium real-time data</h2>
        <table class="tabla" id="data" border="1"></table>
        <p id="updated">Updated: <span id="time"></span></p>
      </div>
    </div>
    <iframe id="temperature" class="board"
src="http://api.thingspeak.com/channels/23319/charts/1?width=997&height=375&res
ults=60&dynamic=true&yaxis=Temperature%20(%C2%BAC)" ></iframe>
    <iframe id="ph" class="board"
src="http://api.thingspeak.com/channels/23319/charts/2?width=493&height=375&res
ults=60&dynamic=true&yaxis=pH%20(grades)" ></iframe>
    <iframe id="conductivity" class="board"
src="http://api.thingspeak.com/channels/23319/charts/3?width=493&height=375&res
ults=60&dynamic=true&yaxis=EC%20(%C2%B5S%2Fcm)" ></iframe>
  </div>

  <script type="text/javascript">
    var base_url = "openaquarium.php";
    var updateInterval = 60000; //60sec.
    var updateIntervalID;

    $(document).ready(function(){
      clearTimeout(updateIntervalID);

      //Login
      $.ajax({
        type: 'GET',
        url: base_url+"?login=admin&pass=12345",
        data: { },
        success: function(response){
          getData();
          updateIntervalID = setInterval(getData,
updateInterval);
        }
      });
    });
  </script>
</body>
```



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15



```
});
```

```
function getData(){
    var code = "";
    $.ajax({
        type: 'GET',
        url: base_url +'?node=1',
        data: { },
        success: function(response){
            sensors = JSON.parse(response);
            $.each(sensors, function(i, item) {
                var unit = "";
                if(item.unit!=""){
                    unit = " " + item.unit;
                }
                code += '<tr><td
class="param">'+item.sensor_type+'</td><td class="val">'+item.value+
unit+'</td></tr>';
            });

            if(code==""){
                code = '<tr><td class="param">No data
from database.</td></tr>';
            }

            $('#data').html(code);
            $('#time').html(getCurrentTime());
        }
    });
}
```

```
function getCurrentTime(){
    var d = new Date();
    var _date = (d.getDate()<10) ? "0"+d.getDate() :
d.getDate();
    var _month = (d.getMonth()<10) ? "0"+(d.getMonth()+1) :
(d.getMonth()+1);
    var _year = d.getFullYear();
    var _hour = (d.getHours()<10) ? "0"+d.getHours() :
d.getHours();
    var _minutes = (d.getMinutes()<10) ? "0"+d.getMinutes()
: d.getMinutes();
```



```
        return (_date + "-" + _month + "-" + _year + " " + _hour
+ ":" + _minutes);
    }
</script>
</body>
</html>
```

Anexo A.1 Códigos streaming

Para la realización de la aplicación se ha necesitado la realización de un código php que recoge la acción del streaming.

Código streaming

```
#!/bin/bash

if pgrep mjpg_streamer > /dev/null
then
    echo "mjpg_streamer already running"
else
    LD_LIBRARY_PATH=/opt/mjpg-streamer/ /opt/mjpg-streamer/mjpg_streamer -i
"input_raspicam.so -fps 15 -q 50 -x 640 -y 480" -o "output_http.so -p 9000 -w
/opt/mjpg-streamer/www" > /dev/null 2>&1&
    echo "mjpg_streamer started"
fi
```

Código php

```
<?

/*
usage on webpage:

*/
$server = "localhost"; // camera server address
$port = 9000; // camera server port
$url = "?action=stream"; // image url on server
set_time_limit(0);
$fp = fsockopen($server, $port, $errno, $errstr, 30);
if (!$fp) {
    echo "$errstr ($errno)<br>\n"; // error handling
} else {
```



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

```
$urlstring = "GET ".$url." HTTP/1.0\r\n\r\n";  
fputs ($fp, $urlstring);  
while ($str = trim(fgets($fp, 4096)))  
header($str);  
fpassthru($fp);  
fclose($fp);  
}  
?>
```



CONTROL Y MONITORIZACIÓN DE UN ACUARIO EN TIEMPO REAL MEDIANTE TECNOLOGIA OPEN SOURCE	MEMORIA
Revisión nº: 2	Fecha: 17/02/15

Anexo B: Documentación online generada

Se está realizando una documentación online del proyecto, como un método para compartir los avances realizados con todos los usuarios del mundo, fomentando el hardware y software libre, y a su vez obtener la ayuda de las personas interesadas por la información, intercambiando emails y compartiendo experiencias, obteniendo conocimientos de manera recíproca.

Esta documentación se está realizando en formato artículo/tutorial para que cualquier usuario con unos mínimos conocimientos en las dos plataformas usadas (Arduino y Raspberry Pi) sea capaz de reproducir el proyecto o fácilmente modificarlo para adecuarlo a las necesidades de su acuario. Esto es posible gracias a las muy concretas y concienzudas explicaciones sobre cómo hay que instalar el hardware y a una muy detallada explicación del software, indicando en cada caso lo que haya que instalar y lo que hace cada apartado y función de todos los códigos.

Este artículo podrá ser visto en la página de Cooking Hacks en la sección de proyectos.



Fig. 56 Ejemplo de un paso de configuración de Raspberry del artículo