

Universidad Tecnológica Nacional

Proyecto Final

**Sistema de Aviso para Personas con
Disminución Auditiva**

Autores:

- Cipriani, Franco.
- Verde, Nicolás.

Director:

*Proyecto final presentado para cumplimentar los requisitos
académicos
para acceder al título de Ingeniero Electrónico en la*

Facultad Regional Paraná

Junio de 2018

Declaración de autoría:

Declaramos que el Proyecto Final “Sistema de Aviso para Personas con Disminución Auditiva” y el trabajo realizado son propios.

- Este trabajo fue realizado en su totalidad, o principalmente, para acceder al título de grado de Ingeniero Electrónico, en la Universidad Tecnológica Nacional, Regional Paraná.
- Establecemos claramente que el desarrollo realizado y el informe que lo acompaña no han sido previamente utilizados para acceder a otro título de grado o pre-grado.
- Siempre que utilizamos trabajo de otros autores, el mismo ha sido correctamente citado. El resto del trabajo es de autoría propia.
- Indicamos y agradecemos correctamente a todos aquellos que han colaborado con el presente trabajo.
- Cuando el trabajo forma parte de un trabajo de mayores dimensiones donde han participado otras personas, indicamos claramente el alcance del trabajo realizado.

Firmas:

-
-

Fecha:

Agradecimientos:

Agradecemos formalmente a todas aquellas personas que de una u otra manera han colaborado con el presente trabajo, haciéndolo posible. A aquellos que han contribuido a nuestra formación profesional, y aquellos que nos han brindado su apoyo durante la realización del mismo.

Franco Cipriani

Nicolás Verde

Universidad Tecnológica Nacional

Abstract

Facultad Regional Paraná

Ingeniería en Electrónica

**Sistema de Aviso para Personas con Disminución
Auditiva**

Franco Cipriani

Nicolás Verde

Abstract:

In order to obtain a device that allows people with hearing impairment to know when they are requested, we design and implement an intelligent bracelet, a server and an application for Android devices, whose main function is to notify users with hearing problems when they are requested. The notifications are captured by the user through an Oled screen placed on the front of the bracelet and by vibrations of the same.

This project is mainly based on the programming and development of a communication equipment that uses the WiFi network to achieve a wide range of coverage with a more than 24 hours autonomy. The software of the equipment is mainly based on the WIFI module ESP8266, which was reprogrammed to take full advantage of its internal microprocessor, obtaining a product with low consumption, which represents an advantage according to the desired objectives in a product of this nature. As a complement, we program an internal web server and an application compatible with Android devices.

We obtained a functional equipment, suitable for people with hearing impairment, and with a wide possibility of being modified to satisfy the extra requirements that may arise.

Keywords:

Communication, ESP8266, WIFI module, Hearing disorders

Resumen:

Con el fin de obtener un dispositivo que permita a personas con disminución auditiva saber cuándo son solicitadas, diseñamos e implementamos una pulsera inteligente, un servidor y una aplicación para dispositivos Android, cuya principal función consiste en notificar a usuarios con problemas de audición cuando éstos son solicitados. Las notificaciones son captadas por el usuario a través de una pantalla Oled colocada en el frente de la pulsera y mediante vibraciones de la misma.

El presente proyecto se basa principalmente en la programación y desarrollo de un equipo de comunicación que utiliza la red WiFi para lograr un amplio rango de cobertura con una autonomía de más de 24 horas. El software del equipo se basa principalmente en el módulo WIFI ESP8266, el cual se reprogramó para aprovechar al máximo su microprocesador interno, obteniendo un producto de bajo consumo, lo cual representa una ventaja de acuerdo a los objetivos deseados en un producto de esta índole. Como complemento, programamos un servidor web interno y una aplicación compatible con dispositivos Android.

Obtuvimos de esta manera un equipo funcional, apto para personas con deficiencia auditiva, y con una amplia posibilidad de ser modificado para satisfacer los requerimientos extras que pudieran surgir.

Palabras Clave:

Comunicación, ESP8266, Módulo WIFI, Discapacidad auditiva

Reconocimientos:

A los profesores de la Universidad Tecnológica Nacional, por brindarnos los conocimientos necesarios para aplicar en nuestro proyecto final.

A Luciano Peroni por facilitarnos las impresiones 3D del gabinete del servidor y carcasa de la pulsera.

Índice:

Capítulo 1: Introducción	1
Capítulo 2: Desarrollo	4
2.1. Investigación de componentes	6
2.1.1. SERVIDOR	6
2.1.2. PULSERA	13
2.2. Pruebas, experiencias y análisis de funcionamiento	20
2.2.1. Servidor	23
2.2.2. Pulsera	27
2.3. Sistema de Carga y consumo	34
2.4. Desarrollo del Software	44
2.4.1. Pulsera	54
2.4.2. Servidor	60
2.5. Página Web	65
2.6. Aplicación Android:	69
2.7. Diseño completo	70
2.7.1. Servidor:	70
2.7.2. Pulsera:	72
Capítulo 3: Resultados	75
Capítulo 4: Análisis de Costos	75
Capítulo 5: Discusión y Conclusión.	80
Capítulo 6: Literatura Citada.	83
Capítulo 7: Anexos.	85

Lista de Figuras:

Figura 1: Equipo Alternativo.....	3
Figura 2: Esquema Completo.....	5
Figura 3: ESP8266.....	7
Figura 4: CPU ESP8266.....	7
Figura 5: Diagrama Funcional ESP8266.....	8
Figura 6: Dimensiones motor vibrador.....	16
Figura 7: Circuito entrada principal (timbre).....	24
Figura 8: Circuito entradas extras.....	24
Figura 9: Circuito Salidas Servidor.....	25
Figura 10: Circuito Regulador de Tensión.....	26
Figura 11: Circuito entrada de retención.....	27
Figura 12: Circuito RTC.....	28
Figura 13: Bus de datos SPI-I2C.....	29
Figura 14: Esquema I2C.....	30
Figura 15: Diagrama funcional DRV2605L.....	34
Figura 16: Dimensiones Pantalla Pulsera.....	40
Figura 17: Gráficos Consumo Pulsera.....	43
Figura 18: Página Web.....	66
Figura 19: Dispositivos conectados Página Web.....	67
Figura 20: Aplicación Android.....	70
Figura 21: Tiempos de respuesta: Pulsera- Aplicación.....	69
Figura 22: Circuito Completo Servidor.....	71
Figura 23: Imágenes Servidor.....	72
Figura 24: Circuito completo pulsera.....	73
Figura 25: Imágenes Pulsera.....	74

Lista de Tablas:

Tabla 1: Especificaciones ESP8266	10
Tabla 2: Pines encapsulado SOT-223	12
Tabla 3: Especificaciones motor vibrador.....	16
Tabla 4: Especificaciones DS3231	18
Tabla 5: Anillo Johnson	27
Tabla 6: Dirección de memoria Dispositivos I2C	31
Tabla 7: Especificaciones DRV2605L	33
Tabla 8: Consumo elementos pulsera	35
Tabla 9: Tensiones elementos pulsera	36
Tabla 10: Nivel de batería ADC	41
Tabla 11: Prueba de consumo	42
Tabla 12: Horas hombre de trabajo.....	76
Tabla 13: Presupuesto estimado.....	79

Lista de Abreviaciones

ACK: acknowledgement - Acuse de recibo
ADC: Analogical Digital Converter- Convertidor analógico a digital
AP: Access Point- Punto de acceso
DHCP: Protocolo de configuración dinámica del host
DNS: Domain Name System- Sistema de denominación de dominios
E/S: Entradas/Salidas
ERM: Eccentric Rotation Mass- Masa de rotación Excéntrica
GHz: Gigahertz
GPIO: General Purpose Input/Output – Entrada/Salida de Propósito General
IDE: Integrated Development Environment- Entorno de desarrollo integrado
JSON: JavaScript ObjectNotation - Notación de Objetos de JavaScript
kB: Kilobyte
LRA: Linear Resonance Actuator- Actuador de Resonancia Lineal
mAh: mili Amper hora
MCU: Microcontrolador
MHz: Megahertz
NACK: Acuse de recibo negado
NTP: Network Time Protocol
OLED: Organic Light-Emitting Diode- Diodo emisor de luz orgánico
PC: Personal Computer- Computadora personal
RF: Radiofrecuencia
RTC: Real Time Clock – Reloj en tiempo real
RTC: Real Time Clock- Reloj en tiempo real
SCL: System Clock- Reloj del Sistema
SDA: System Data
SNMP: Protocolo simple de administración de red
SoC: System on a chip - Sistema en chip
SPI: Serial Peripheral Interface- Interfaz Periférica Serial
SPIFFS: SPI Flash File System
TCP: Transmission Control Protocol – Protocolo de control de transmisión
TFTP: Protocolo trivial de transferencia de archivos

TTL: Time To Live – Tiempo de Vida

UDP: User Datagram Protocol- Protocolo de datagrama de Usuario

WPS: WiFi Protected Setup

Dedicado a:

Franco Cipriani:

Quiero agradecer en primera instancia a mi familia, de forma especial a mis padres que con mucho esfuerzo hicieron posible que pueda estudiar una carrera de grado, con los costos que esto implica; a mis hermanas por el acompañamiento y el apoyo para no dejar la carrera en sus momentos más difíciles; a mi abuela la cual me sacaba de la rutina para disfrutar un poco el momento. Por otra parte mis amigos cumplieron un rol fundamental en cuanto a no abandonar, brindando ideas de proyectos, o simplemente compartiendo momentos para despejar la mente en épocas de evaluaciones.

Nicolás Verde:

Agradezco en gran medida a mi familia quienes me brindaron el apoyo, tanto económico como emocional para abordar la carrera que decidí realizar.

También a mi novia, quien estuvo en todo momento para acompañarme y brindarme fuerzas en las situaciones más difíciles. Siendo un pilar fundamental para atravesar el camino de la carrera.

A mis amigos, quienes fueron de gran ayuda a la hora de despejar la mente, lo cual considero muy necesario.

Capítulo 1: Introducción

Debido a la dificultad que observamos en personas con deficiencia auditiva para reconocer cuando son solicitadas (ej: para oír el timbre de su casa), surgió la iniciativa de implementar un sistema capaz de resolver esta problemática. Es así que decidimos desarrollar este proyecto que consiste en una pulsera con comunicación WiFi, capaz de notificar al individuo que está siendo llamado, mediante vibraciones y luces.

Investigamos sobre el tema y encontramos que, para esta dificultad, en el mercado solamente hay equipos que se ubican en un lugar fijo y producen destellos lumínicos para indicar a la persona que está siendo solicitada. El mayor inconveniente que presentan es que sólo son útiles cuando se encuentran dentro del campo visual del individuo.

Otro punto que analizamos fue el modo de comunicación utilizado en los equipos existentes; siendo la mayoría mediante Radiofrecuencia. Si bien este tipo de comunicación posee un alcance suficiente para su finalidad, el mismo se encuentra acotado en cuanto a la posibilidad de configuraciones personalizadas que pudiera requerir el usuario.

Teniendo en cuenta todo lo mencionado, procedimos al diseño de un dispositivo que permitiera al individuo mayor libertad y comodidad. Por ello diseñamos una pulsera inteligente que permite a la persona continuar su vida cotidiana, y al mismo tiempo ser notificado cuando es solicitado.

Otra de las ventajas que se consideramos fue la utilización de la red Wi-Fi para establecer la comunicación. De esta manera, logramos una gran virtud correspondiente al mundo del “Internet de las cosas”, posibilitando al usuario configurar su equipo de acuerdo a sus necesidades y permitiendo utilizarlo para diferentes aplicaciones.

ESTUDIO DE MERCADO:

Según los datos estadísticos del último censo realizado en la Argentina, la cantidad de personas con dificultad auditiva es de más de 1 Millón. La misma corresponde al 18% de todas las discapacidades. A nivel mundial, según la Organización Mundial de la Salud, 360 millones de personas sufren pérdida auditiva en el mundo.

Por otra parte, debido al envejecimiento de la población mundial, la OMS señaló que actualmente una de cada tres personas mayores de 65 años – un total de 165 millones de personas en el mundo – padece problemas de audición; pero la producción global de dispositivos o implantes está muy lejos de satisfacer la demanda. En los países en desarrollo, menos de una en 40 personas con discapacidad auditiva tiene acceso a un dispositivo.

Comercialmente encontramos en el mercado dispositivos que solamente, por medio de una conexión RF, emiten luces de aviso, condicionando a la persona a tener la visual dentro del rango del dispositivo. Así como también la condición de tener un equipo emisor de luz por cada ambiente del hogar.

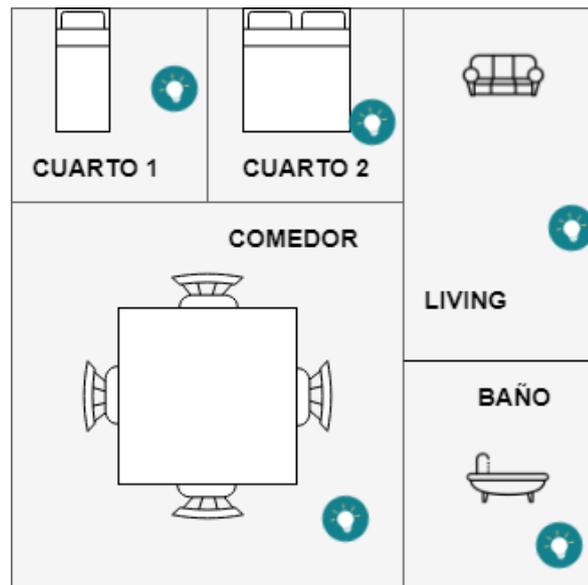


Figura 1: Equipo Alternativo

Nuestro producto amplía los sentidos que permiten a la persona con discapacidad auditiva saber que está siendo llamada, ya que posee avisos tanto por vibración como visual.

Otro aspecto importante que tuvimos en cuenta fue la opción de adición de distintos tipos de entradas por ejemplo: sensor de incendio, sensores de movimiento, puerta abierta, etc. ampliando el uso para distintas aplicaciones, incrementando la demanda del mercado.

Capítulo 2: Desarrollo

En el presente proyecto definimos, desarrollamos e implementamos una mejora en el sistema de aviso para personas con discapacidad auditiva. Para el mismo incluimos nuevas tecnologías surgidas en los últimos años, como es el caso del micro controlador ESP8266.

Este microcontrolador posee internamente un módulo que utiliza la tecnología WiFi. Esto nos simplificó a la hora de desarrollar nuestro producto ya que contábamos con las herramientas de comunicación ya integradas en un pequeño paquete.

Para explicar el funcionamiento del sistema vamos a dividirlo en dos dispositivos:

- **Servidor:** contiene las entradas y salidas que permiten la comunicación con el resto de los dispositivos. En cuanto a las entradas, el mismo posee una principal que corresponde al timbre, y dos adicionales compatibles con diferentes tipos de sensores que se hallan en el mercado. Por su parte, las salidas son independientes entre sí, siendo las mismas configuradas por el usuario para distintas acciones.
- **Pulseras:** permiten recibir la llamada de aviso enviada por el servidor y traducirla en una señal visual y de vibración para que el usuario pueda percibirlas.

Ambos elementos utilizan la red WiFi doméstica como medio de comunicación mediante protocolo UDP (UserDatagramProtocol).

También complementamos el producto creando una **aplicación** para dispositivos Android que permitiera al usuario obtener una alternativa a la pulsera. El servidor, al recibir una señal de entrada, envía el aviso a todos los dispositivos que se encuentren dentro de la misma red doméstica.

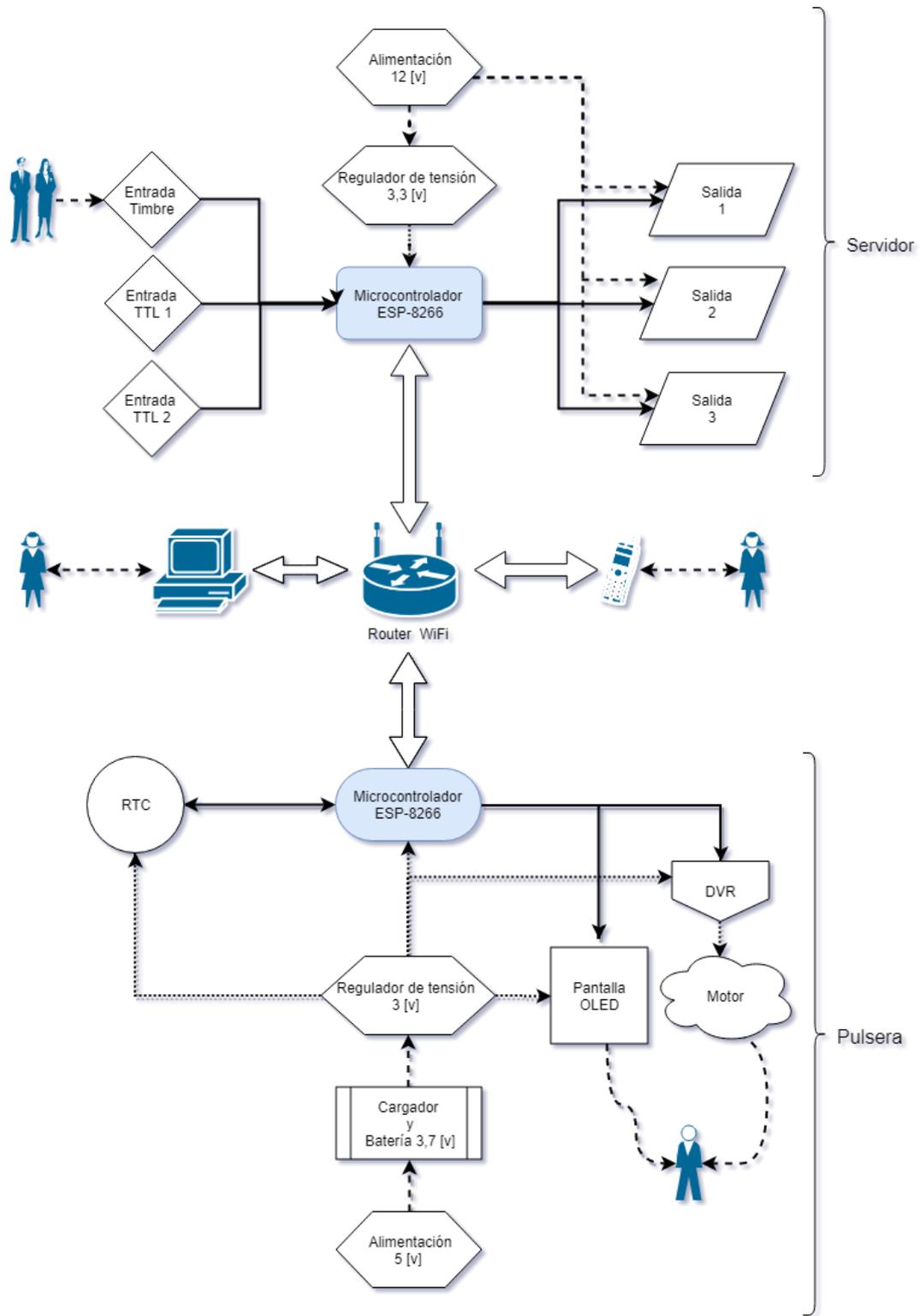


Figura 2: Esquema Completo

2.1. Investigación de componentes

Como se puede observar en el diagrama en bloque, nuestro proyecto se basa principalmente en el módulo ESP-8266, cuyas características se amoldaban a los requerimientos primordiales para el desarrollo del presente proyecto.

Puntos necesarios para llevar a cabo el proyecto:

- Requisito de espacio mínimo: la pulsera debía poseer la mayor comodidad y liviandad posibles.
- Uso eficiente de la energía: el desafío fue lograr una autonomía acorde al uso del sistema implementado.
- Desempeño confiable en la industria del internet: necesitábamos un sistema que nos proporcione una conectividad segura y estable.

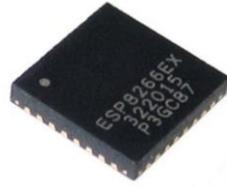
En base a la idea inicial del proyecto, investigamos cuáles eran los componentes que se conseguían en el mercado para su realización. Es así como hallamos el módulo ESP8266, el cual nos resultó conveniente por su bajo costo, sus características, sus variantes y aplicaciones.

A continuación se describe el análisis y selección de los componentes que utilizamos para el presente proyecto.

2.1.1. SERVIDOR

ESP8266:

ESP8266 es el nombre de un microcontrolador diseñado por una compañía china llamada Espressif Systems. En principio, su producción se basaba en un pequeño chip de montaje superficial de apenas 5 milímetros. Sin embargo, distintos fabricantes han potenciado su uso adecuando placas de circuito impreso integrándolo y dejándolo listo para su uso.



Debemos identificar bien las distintas estructuras en las que se puede encontrar lo que denominamos ESP8266.



Figura 3: ESP8266

Como se observa en la imagen anterior, el ESP8266 integra en sí mismo un SoC, que dentro posee un microcontrolador o MCU. Por lo tanto, el ESP8266 no es un microcontrolador, sino que dentro del mismo posee uno, Tensilica L106 de 32-bit. La MCU gestiona todas las entradas, salidas y cálculos necesarios para hacer funcionar el programa cargado. Trabaja a una velocidad de 80 MHz, con la posibilidad de alcanzar los 160 MHz.

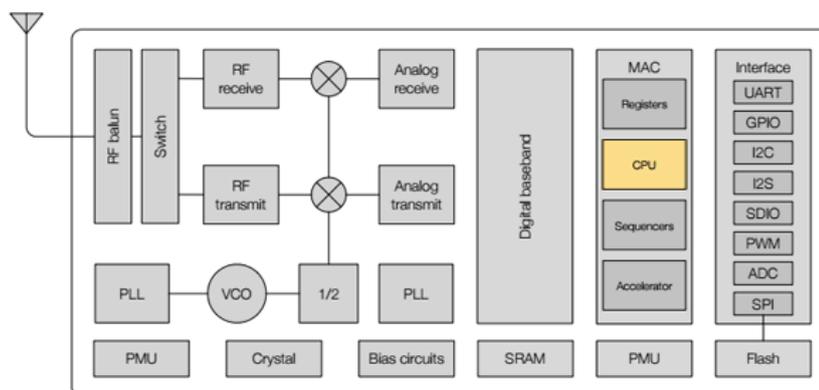


Figura 4: CPU ESP8266

SoC ESP8266:

Consiste en un chip que posee integrado todo lo necesario para funcionar de manera autónoma como si fuera una PC. En el caso del ESP8266, lo único que no posee es una memoria para almacenar el código o programa.

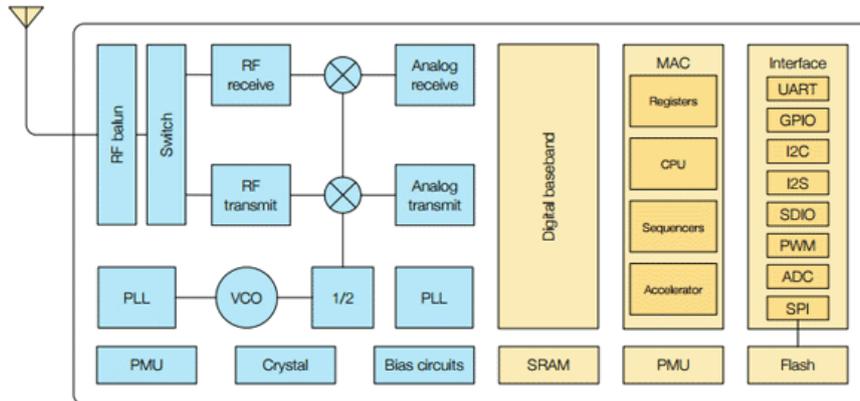


Figura 5: Diagrama Funcional ESP8266

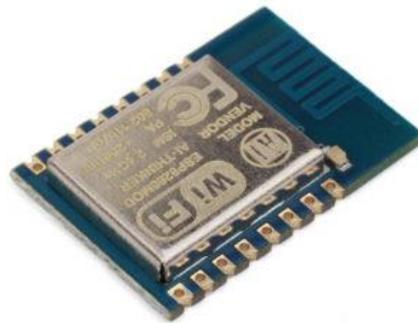
Esto supuso un inconveniente ya que parte de los pines de entrada y salida debimos utilizarlos para conectarse a una memoria Flash externa.

Las características principales son las siguientes:

- Incorpora una MCU de 32-bit de bajo consumo (Tensilica L106)
- Módulo WiFi de 2.4 GHz
- RAM de unos 50 kB
- 1 entrada analógica de 10-bit (ADC)
- 17 pines de entrada y salida GPIO (de propósito general)

Módulo ESP-12:

Dentro de los diferentes módulos del ESP8266, encontramos el ESP-12E, el cual incorpora una memoria Flash para almacenar el programa, y una antena WiFi. Además, facilita el acceso a los demás conectores de SoC y del microcontrolador. A continuación mencionamos sus características:



ESP-12-E — ESP-12-F:

- Dimensiones: 24,00 mm × 16,00 mm
- Conexiones: 22 conexiones de superficie dispuestas en tres lados (8+8+6)
- Antena impresa en la PCB
- Alimentación: 3,3 V

CATEGORIA	ÍTEMS	PARÁMETROS
WiFi	• Estándares	FCC / CE / TELEC / SRRC
	• Protocolos	802.11 b / g / n / e / i
	• Rango de frecuencias	2.4G ~ 2.5G (2400M ~ 2483.5M)
	• Potencia de Tx	802.11 b: +20 dBm
		802.11 g: +17 dBm
		802.11 n: +14 dBm
	• Sensibilidad Rx	802.11 b: -91 dbm (11 Mbps)
		802.11 g: -75 dbm (54 Mbps)
		802.11 n: -72 dbm (MCS7)
	• Antena	Sobre PCB, Externa, Conector IPEX , Chip Ceramic
• CPU	Tensilica L106 microcontrolador de 32 bits	
• Interfaz periférica	UART / SDIO / SPI / I2C / I2S / Control remoto IR	
	GPIO / ADC / PWM / LED luz y botón	
Hardware	• Tensión de funcionamiento	2.5V ~ 3.6V
	• Corriente de funcionamiento	Valor promedio: 80 mA
	• Rango de temperatura de funcionamiento	-40 ° C ~ 125 ° C
	• Tamaño del paquete	QFN32-pin (5 mm x 5 mm)
Software	• Modo Wi-Fi	Estación / SoftAP / SoftAP + Estación
	• Seguridad	WPA / WPA2
	• Cifrado	WEP / TKIP / AES
	• Actualización de firmware	Descarga de UART / OTA (a través de la red)
	• Desarrollo de software	Compatible con Cloud Server Development / Firmware y SDK para una rápida programación en chip
	• Protocolos de red	IPv4, TCP / UDP / HTTP / FTP
	• Configuración de usuario	Set de instrucciones AT, servidor de la nube, aplicación Android / iOS

Tabla 1: Especificaciones ESP8266

Lenguaje de programación seleccionado: Arduino IDE

Uno de los entornos de desarrollo integrado (IDE) más usado es Arduino, una plataforma libre que nos permite descargarla desde Internet de manera gratuita.

Además de proporcionar un editor de lenguaje C de Alto nivel, más herramientas para compilar e implementar, Arduino IDE proporciona bibliotecas pre-suministradas de rutinas C que "ocultan" complejos detalles de la implementación que de otra manera podrían ser necesarios cuando se programa.

Arduino entra en conexión con el ESP8266 gracias a una cantidad de personas que han construido un proyecto Open Source Github que proporciona un "plug-in" o "extensión" a la herramienta IDE. Dicha extensión permite escribir bocetos en el IDE de Arduino que aprovechan las interfaces de la biblioteca Arduino que, en tiempo de compilación y despliegue, generan un código que se ejecutará en el ESP8266.

El ESP8266 Arduino además incluye librerías para comunicar con WiFi utilizando TCP y UDP, crear servidores, usar un sistema de archivos en la memoria flash, trabajar con tarjetas SD, servos, y demás periféricos.

Dentro de las variantes que encontramos en el mercado, sin dudas, el ESP8266 es la más viable. Sobre todo por su bajo costo como se mencionó anteriormente y su diseño compacto.

REGULADOR DE TENSIÓN 3.3 v

Como vimos anteriormente, el ESP8266 posee una alimentación de entrada de 3.3 V, admitiendo un máximo de 3.6 V. es por esto que la misma debe permanecer estable.

Investigamos los distintos tipos de reguladores, y encontrando el LM1117MP-3.3, fabricado por la compañía Texas Instruments. El mismo es un

regulador lineal que proporciona una tensión de salida estable. Ofrece a su vez limitador de corriente y protección térmica.



Observando su hoja de datos encontramos que ofrece cinco salidas de tensiones: 1.8 – 2.5 – 3.3 – y 5 V. Siendo la configuración de 3.3 V la que se utiliza para alimentar el módulo ESP8266 de nuestro servidor. Estos rangos de tensiones los ofrece con una entrada de hasta 20 V.

Un dato relevante es que posee una tensión de caída de 1.2 V a 800 ma, es decir, que debe proporcionar 1.2 V más de los que entrega.

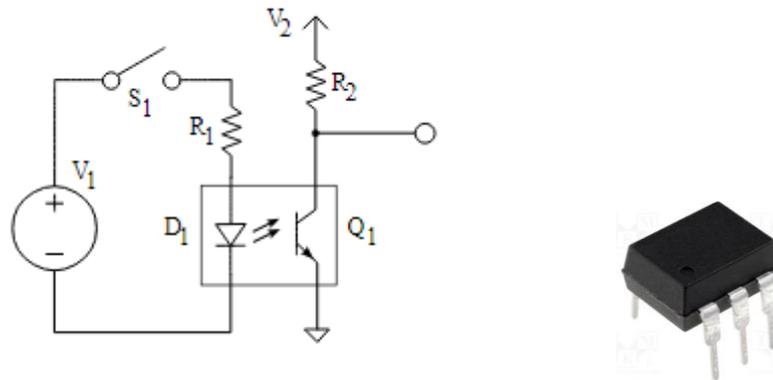
El tipo de encapsulado que compramos es el SOT-223, el cual posee la siguiente disposición de pines:

PIN		I/O	DESCRIPCIÓN
NOMBRE	SOT-223		
ADJ/GND	1	—	Pin de ajuste para opción de salida ajustable. Pin GND para opción de salida de tensión fija.
V _{IN}	3	I	Pin de tensión de entrada del regulador
V _{OUT}	2, 4	O	Pin de tensión de salida del regulador

Tabla 2: Pines encapsulado SOT-223

OPTOACOPADORES DE ENTRADA

Las entradas que posee el dispositivo Servidor se encuentran aisladas ópticamente. Para esto utilizamos el conocido 4N28. Este es un optoacoplador de un canal con salida del fototransistor con conexión a la base. Se compone de LED infrarrojo de arseniuro de galio y un fototransistor NPN de silicio.



Características:

- Tipo de salida del optoacoplador: Phototransistor
- Número de canales: 1
- Voltaje de aislamiento: 5 kV
- Corriente directa I_f máxima: 60mA
- Voltaje de colector-emisor $V_{(br)ceo}$: 80V
- Tipo encapsulado: DIP
- Número de pines: 6

De esta manera nos aseguramos de tener nuestro producto protegido frente a posibles cargas eléctricas provenientes del exterior.

2.1.2. PULSERA

Como vimos anteriormente, la pulsera también se encuentra concentrada en el módulo ESP8266. La principal característica por la cual optamos por este módulo está relacionada con la necesidad de optimizar el consumo, ya que aspirábamos a una pulsera inalámbrica alimentada por batería.

Fue por esta misma necesidad que investigamos los distintos componentes y elementos que forman la pulsera, teniendo en cuenta el consumo de cada uno, a fin de incrementar la autonomía de la misma lo más posible.

DISPLAY O-LED

Uno de los modos de aviso que posee la pulsera es el visual, por medio de una pantalla incorporada. La misma constituyó un desafío en cuanto al consumo de energía.

Investigamos las opciones disponibles, y nos encontramos con una tecnología nueva: OLED. Un diodo orgánico de emisión de luz, también conocido como OLED (Organic Light-Emitting Diode), es un diodo que se basa en una capa electroluminiscente formada por una película de componentes orgánicos que reaccionan a una determinada estimulación eléctrica, generando y emitiendo luz por sí mismos. Es decir, mediante un compuesto orgánico se consigue que cada píxel esté iluminado por separado, logrando un control mucho más preciso y exacto de brillo y contraste. Esto corresponde a una tecnología recientemente desarrollada y aplicada por varias empresas mundiales.

Es así que se obtuvo un display OLED de 0.96 pulgadas y 128x64 píxeles.



Si bien es una pantalla pequeña, posee un alto contraste que le permite ser legible incluso en ambientes muy iluminados.

Cada píxel posee un chip controlador que lo activa o desactiva; por esta misma razón, no requiere luz de fondo, lo cual reduce considerablemente el consumo.

Internamente, todo el conjunto funciona a 3.3 V. Sin embargo, posee un regulador que se encarga de mantener la tensión requerida, posibilitando una

entrada de 5 V. El consumo de potencia depende de la cantidad de píxeles encendidos, pero en promedio posee un consumo aproximado de 20 mA.

El display tiene incorporado un chip controlador que dispone de múltiples formas de comunicación como I2C, SPI, entre otras. En nuestro caso, optamos por la comunicación I2C por conveniencias que se explicarán posteriormente.

MOTOR – VIBRADOR

La otra manera de aviso al usuario es mediante un motor de vibración. Siendo éste también un desafío en cuanto a la eficiencia que deseábamos obtener, ya que los motores se caracterizan por su elevado consumo de corriente.

Encontramos en el mercado un tipo de motor comúnmente utilizado en teléfonos celulares, juguetes, etc. económico y de bajo consumo. Se trata de un motor de masa giratoria excéntrica (Eccentric Rotating Mass). El mismo está compuesto por un motor de corriente continua, giratorio, con una masa desplazada acoplada al eje. A medida que el motor gira, la fuerza centrípeta de la masa compensada es asimétrica, creando una fuerza centrífuga neta que causa un desplazamiento del motor. Al girar a altas velocidades, los desplazamientos rápidos se perciben como vibraciones.

Existe una amplia variedad de presentaciones de estos motores en el mercado, de acuerdo a los distintos tipos de aplicaciones posibles. En nuestro caso elegimos el motor tipo “moneda”; su diseño cilíndrico, con un radio de 10 [mm] y un espesor de 2 [mm], nos permitió utilizarlo apoyado sobre la placa del circuito impreso, optimizando el espacio y haciendo la pulsera más compacta.

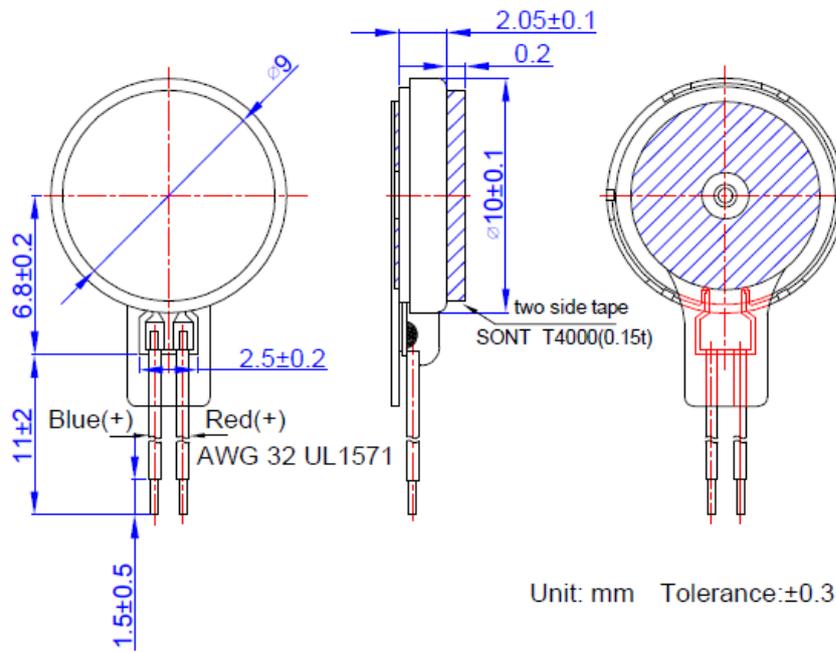


Figura 6: Dimensiones motor vibrador



PARÁMETRO	ESPECIFICACIONES
Tensión Nominal	3 V DC
Tensión de Funcionamiento	2.7 – 3.3 V DC
Rotación	Sentido agujas del reloj / Sentido contrario agujas del reloj
Corriente a 25°C	Máximo 900 mA
Velocidad nominal	Mínimo 9000 RPM
Corriente de arranque	Máximo 120 mA
Tensión de arranque	Máximo 2.3 V DC
Ruido mecánico	Máximo 50 dB
Temperatura de funcionamiento	-20 - +60 °C

Tabla 3: Especificaciones motor vibrador

DRV2605L

Una vez que elegimos el tipo de motor vibrador que usaríamos, evaluamos las distintas formas de controlarlo. Es así que, en base a las limitaciones que teníamos en cuanto a pines para lograr la comunicación de nuestro ESP8266 y el motor, decidimos utilizar un controlador diseñado específicamente para este tipo de aplicación: el DRV2605L de la compañía Texas Instruments.

Este controlador está diseñado para proporcionar control háptico de actuadores ERM y ERS, y admite la comunicación I2C, característica que nos proporcionó una solución en cuanto a capacidad de pines. De esta forma obtuvimos un control preciso del ERM mediante la integración de librerías, permitiendo infinidad de efectos y modos de arranque y frenado.



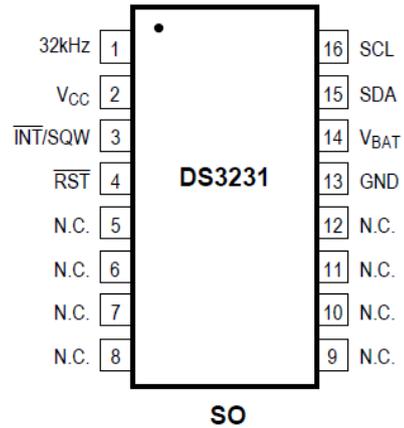
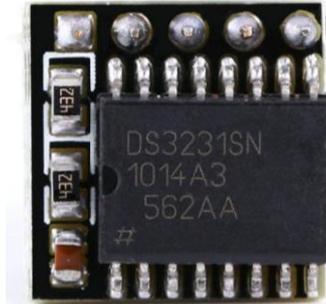
RTC

Una de las funciones que deseábamos incluir en la pulsera, era la de reloj. Para dicho propósito, necesitábamos un método preciso para obtener la hora actual, por lo cual incorporamos un integrado externo denominado DS3231 que nos permitiera liberar recursos del sistema principal (ESP8266).

El DS3231 es un reloj en tiempo real, RTC (Real Time Clock), de alta precisión que cuenta con un oscilador a cristal con compensación de temperatura (TCXO). La integración del cristal en el propio integrado asegura la exactitud a largo plazo del reloj.

El DS3231 se comunica con el microcontrolador a través del bus I2C con solamente 2 pines, lo cual constituyó una gran ventaja debido a las limitaciones de pines antes mencionadas.

El RTC posee una tensión de alimentación de 2.3 V a 5 V, la cual se adecuaba a los voltajes del resto de los dispositivos.



CARACTERÍSTICAS
Cuenta Segundos, Minutos, Horas, Días, Meses y años
Formato 12-horas o 24-horas y AM o PMz
Sensor integrado de temperatura para auto corrección del cristal
Interface de conexión I2C bidireccional de 100 kHz a 400 KHz.
Entrada exclusiva para una batería de backup
Cristal interno
Entrada externa de reseteo
Salida de interrupciones para alarmas o señal cuadrada de frecuencia fijas 1 Hz, 1.024 kHz, 4.096 kHz o 8.192 KHz
Salida de señal cuadrada de frecuencia fija de 32.768 KHz

Tabla 4: Especificaciones DS3231

BATERÍA Y SISTEMA DE CARGA:

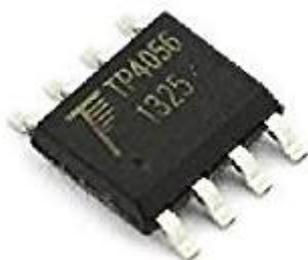
Para alimentar la pulsera, debíamos contar con un tipo de batería que se adecuara a nuestros requerimientos. Debía ser pequeña, liviana y con la mayor autonomía posible.

Optamos por un tipo de batería de Litio – Ion de 230 mAh. Su pequeño tamaño nos permitió incluirla dentro del dispositivo.

Otro aspecto importante que tuvimos en cuenta a la hora de incluir este tipo de batería a nuestro sistema es la forma en que se recarga. Para ello, se incorporó el integrado TP4056, un cargador lineal de corriente constante para baterías de Ion-Litio. El mismo, requiere un mínimo de componentes externos para su funcionamiento, y posee retroalimentación térmica que regula la corriente de carga para limitarla en caso de temperatura ambiente alta. La tensión de carga es de 4.2 V y la corriente puede programarse externamente hasta 1 A.

El TP4056 incluye un monitor de corriente, bloqueo de baja tensión, recarga automática y dos pines de estado para indicar la terminación de carga y la presencia de una tensión de entrada.

Es así que se puede realizar un recarga sencilla del dispositivo mediante un puerto micro USB.



2.2. Pruebas, experiencias y análisis de funcionamiento

Antes de realizar el presente proyecto, investigamos la existencia de productos similares dentro del mercado nacional. Dado que no se encontraron similitudes, expandimos la búsqueda a mercados extranjeros, obteniendo resultados similares. En ambos casos, la problemática se encontraba sólo cubierta en ciertos aspectos; es así que vimos la posibilidad de mejorar la oferta existente. Nos concentramos en lograr un producto que además de satisfacer la necesidad principal (informar a la persona hipoacúsica cuando es llamada), sea cómodo, práctico y amigable con el usuario.

Luego de realizar el análisis y estudio de componentes que deseábamos utilizar para nuestro proyecto, comenzamos a armar los diseños prácticos de ambas partes: Servidor y Pulsera. Primero, diseñamos el servidor. Para ello, armamos un esquema del circuito mediante el simulador Proteus. Diseñamos y analizamos cada bloque por separado, para luego concluir en una única placa final de cada dispositivo.

Con respecto a los componentes, algunos se lograron conseguir únicamente en placas de prototipos como el RTC y el chip de carga. Ambos debieron ser desoldados cuidadosamente mediante la utilización de una pistola de calor ya que por la cantidad de pines que presentaban, resultaba complicada y forzada la extracción mediante un soldador de punta. La mayoría de los componentes y los PCB debieron ser importados a causa de los costos y la falta de stock en el país, esto significó una pérdida de tiempo en la puesta en funcionamiento de ambos dispositivos.

Con respecto a la programación, comenzamos con el servidor, el cual controla las entradas y salidas del mismo, y es el encargado de establecer la comunicación con los dispositivos receptores. Optamos por implementar en ambos módulos un servidor interno HTTP. En un principio, la codificación se realizaba íntegramente mediante un código de línea, implementado dentro de funciones, las cuales eran llamadas cuando el usuario solicitaba una página mediante el navegador web; sin embargo, para mejorar este punto buscamos un programa (WYSIWYG web builder) que nos permitiera realizar una mejora considerable en el sitio web y posea carga de archivos mediante FTP. A través

de dicho programa mejoramos el entorno gráfico de la página Web, añadimos imágenes, fondos y encabezados entre otros. Además implementamos las funciones Javascript para realizar el intercambio de información con el hilo principal de ejecución del servidor. Guardamos los archivos generados, junto con las imágenes, en la memoria spi del microcontrolador.

Finalmente se programa la pulsera para así establecer la comunicación con el servidor. Para ello se modifica la programación del servidor para así introducir las funciones propias de la pulsera. En este punto, se realiza la investigación de los componentes externos, que acompañan a dicho modulo. La pantalla inicialmente utilizada de tecnología OLED posee un tamaño de 0,96 pulgadas (128x32 pixeles), la misma se utiliza para presentar mensajes simples. El modulo a su vez implementa un sistema de alerta extra a la pantalla (visual), la recepción por medio de un sistema de sonido no es viable para un usuario hipocacusico, por este motivo se busca captar las alertas mediante el sentido del tacto, usando vibraciones. Se buscó un conjunto compatible de controlador y motor para dicho fin, el cual se programó y se verificó su correcto funcionamiento una vez obtenida la placa final de la pulsera, ya que dado su formato SMD, no podíamos utilizarlo en una protoboard. . Con el objetivo de hacer más funcional la pulsera, optamos por integrar un sistema de reloj (RTC) externo al microcontrolador, para poder informar al usuario mediante la pantalla la hora y fecha.(correcto)

Para verificar el funcionamiento de todos los componentes estábamos obligados a diseñar y probar en un PCB. Buscamos en este punto reducir al menor tamaño posible la placa, ya que ésta sería utilizada por el usuario en su muñeca. Luego de varios diseños, optamos por utilizar una pantalla más grande de 1,3 pulgadas (128x 64 pixeles) dado que no podíamos reducir más el PCB del microcontrolador. De esta forma obtuvimos un conjunto de dos placas interconectadas, mediante pines de alimentación y bus de datos I2C, distanciadas mediante separadores, con el fin de no provocar un cortocircuito entre ellas.

Con el fin de ahorrar tiempo y dinero, diseñamos el PCB del servidor para realizar un solo pedido. Para proteger el microcontrolador optamos por

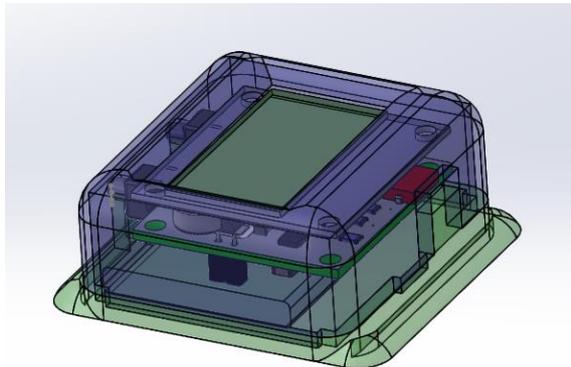
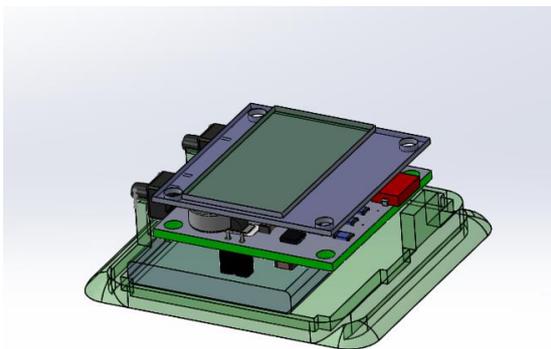
entradas opto acopladas y salidas con relays permitiendo al usuario conectar dispositivos de tensiones diferentes (12 [v] – 48 [v] – 220 [v]).

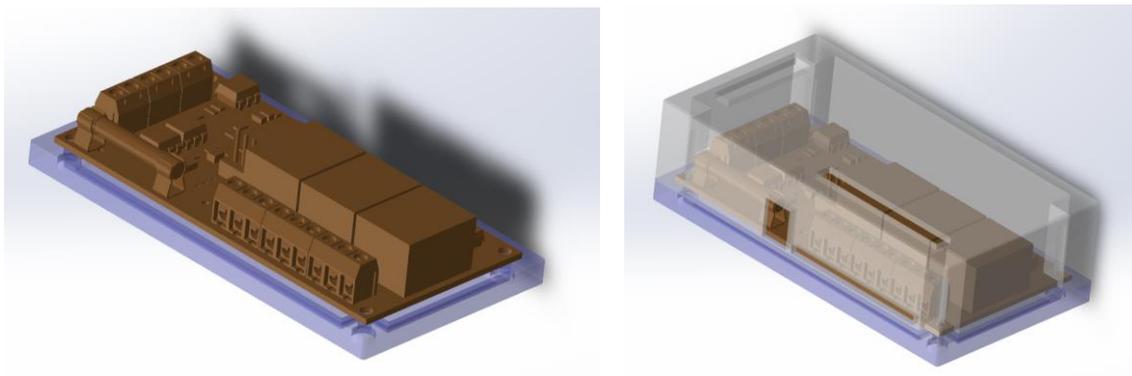
Una vez obtenido los componentes y las placas procedimos a armarlos para luego verificarlos. Finalmente con los sistemas funcionando, implementamos la comunicación mediante el protocolo TCP. La dificultad que tuvimos en este punto fue que este protocolo provocaba desconexiones y mal funcionamiento, por lo cual decidimos reemplazarlo por el protocolo UDP, con un sistema de transferencia de datos mediante JSON.

Realizamos el testeo de consumo de la pulsera con baterías, lo cual nos permitió elegir correctamente su capacidad y forma.

Para complementar el uso de la pulsera, desarrollamos una aplicación celular mediante Android Studio, la cual permite realizar ciertas configuraciones sobre el servidor y recibir las alertas generadas por este, mediante notificaciones y vibraciones.

Para finalizar, diseñamos la carcasa del servidor utilizando el programa SolidWorks. Le colocamos orificios en la parte superior para ajustar los cables de las entradas y salidas y en los laterales establecimos las entradas y salidas, junto a la alimentación. En cuanto a la pulsera, armamos su diseño teniendo como eje la colocación de la pantalla en la parte superior; en los laterales, añadimos el puerto de carga mini usb y dos botones de usos múltiples. En resumen, logramos un diseño útil, acorde a los requerimientos del usuario





La problemática más difícil de superar fue la autonomía que presentaba la pulsera, ya que una duración de pocas horas ponía en peligro la continuidad del proyecto. Para solucionar esto, debimos buscar alternativas como aumentar el tamaño de la batería, apagar los dispositivos complementarios que no sean necesarios (pantalla y dvr) y realizar un sistema de suspensión y reactivación del ESP8266.

2.2.1. Servidor

Entradas

En principio incluimos la entrada principal, la del timbre propiamente dicho. Luego se propuso agregar dos entradas más para incluir distintos tipos de dispositivos o sensores.

Todas las entradas se encuentran optocopladas por el integrado 2N28; el mismo proporciona una aislación máxima de 5000 [Vrms], permitiendo que las entradas del microprocesador se encuentren seguras ante algún problema externo. Su corriente máxima de entrada es de 60 [mA] con un valor típico de 10 [mA]; debido a esto, sus entradas están limitadas mediante una resistencia acorde a la tensión.

Entrada del timbre

Constituye la entrada principal del dispositivo. Trabaja con una tensión continua de 12 [V], y posee un fusible que previene posibles daños a las personas que lo utilizarán, en caso que existiera alguna falla en el circuito.

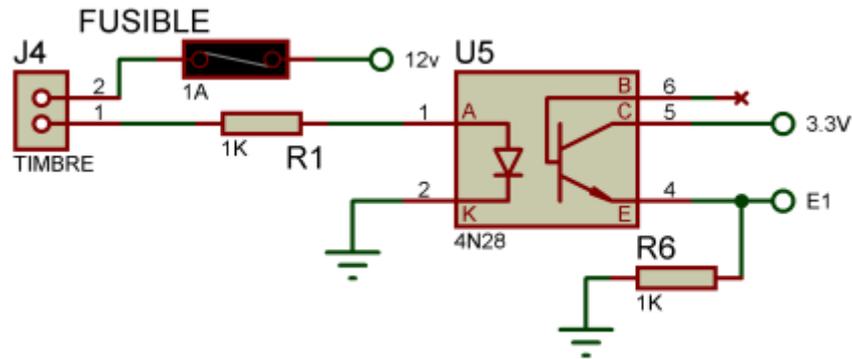


Figura 7: Circuito entrada principal (timbre)

Entradas extras

Para darle más funcionalidad al dispositivo incorporamos dos entradas, las cuales pueden utilizarse para cualquier fin. Para que sea compatible con diferentes tipos de sensores que se encuentran en el mercado, adaptamos las entradas para funcionar con tensiones TTL, es decir que cuando la entrada sea de 5 a 2 [V] (High) tomará un uno lógico, y cuando sea de 0,8 a 0 [V] (Low) interpretará un cero lógico.

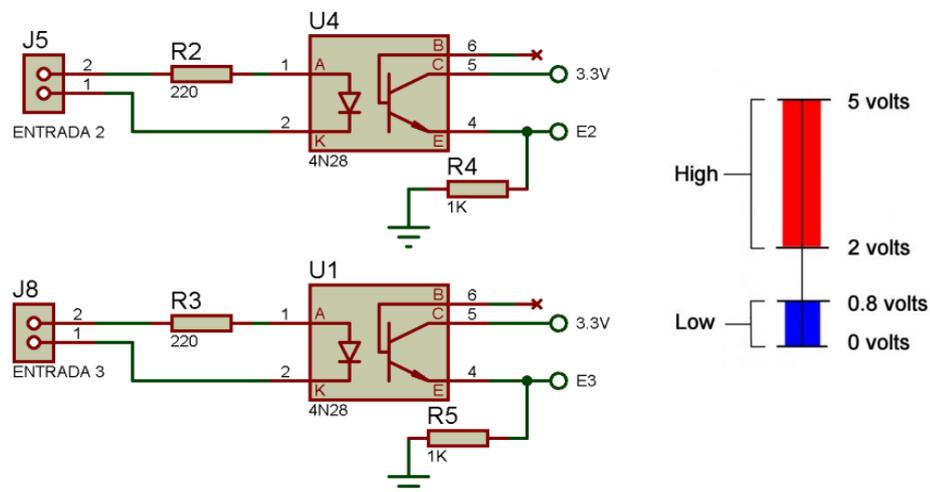


Figura 8: Circuito entradas extras

Para comprobar el funcionamiento de los circuitos realizamos la prueba en Protoboard. Ambos tipos de entradas obtuvieron resultados satisfactorios trabajando con los niveles de tensión que necesita la entrada del microcontrolador para reconocerlas.

Salidas

Luego nos ocupamos de implementar las salidas que debía poseer nuestro servidor para brindar más funcionalidad al dispositivo. Logramos incorporar tres salidas gracias a la cantidad de pines disponibles que posee el ESP8266.

Establecimos que el usuario pueda configurarlas para utilizarlas de forma independiente o en conjunto, según su necesidad, además de elegir una conexión normal abierta o normal cerrada.

Sin embargo, cuando realizamos la prueba del circuito nos encontramos con la dificultad de que el ESP8266 no posee la capacidad de activar las bobinas de los relés que trabajan a 12 [V]. Por ello, incorporamos un Array de Darlington, el ULN2803, que posee características muy convenientes para nuestro propósito: una salida de 500 [mA] con una tensión admitida de hasta 50 [V]. De esta manera logramos obtener la corriente necesaria para activar el relé y proporcionar las salidas deseadas.

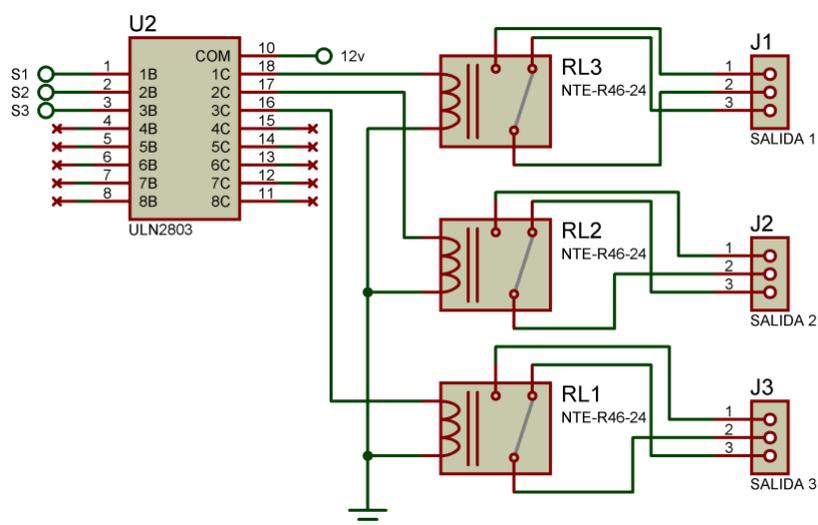


Figura 9: Circuito Salidas Servidor

Regulador de tensión

Para adecuar la tensión necesaria para alimentar el ESP8266, partiendo de la alimentación del servidor (12 [V]) decidimos utilizar dos reguladores de tensión. En un primer paso, mediante el LM7005 obtuvimos una tensión estable de 5 [V] para luego por medio del LM1117MP-3.3 obtener los 3.3 [V] necesarios para alimentar el ESP8266.

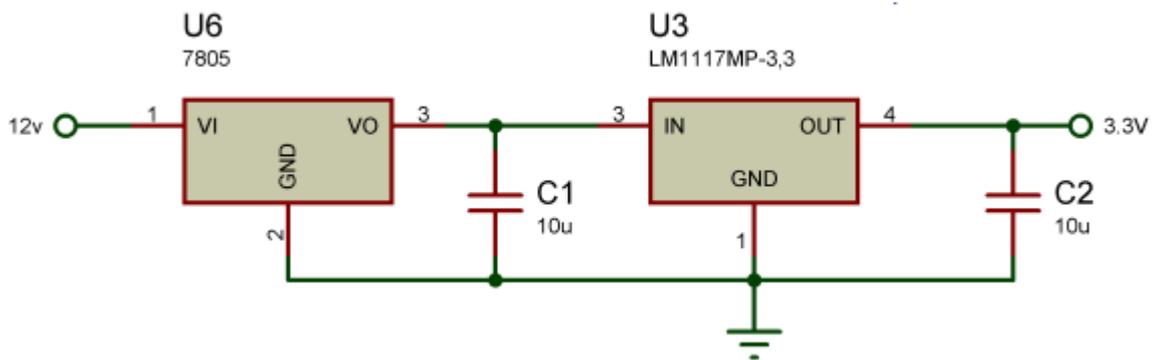


Figura 10: Circuito Regulador de Tensión

2.2.2. Pulsera

Entrada de retención

Implementamos un método que permitiera indicar el estado de la pantalla: encendida o apagada. Esto es debido a que la hora se visualiza cuando el usuario presiona uno de los botones que incluye la pulsera.

Con el fin de establecer si la pantalla se va a encontrar encendida o no, realizamos un circuito de anillo Johnson mediante un flip flop tipo D, el cual consiste únicamente en dos estados como vemos en la tabla de verdad. Añadimos además un sistema anti-rebote para no causar muchos pulsos en la entrada del clock del flipflop.

Tabla de verdad			
Botón	D	Q	!Q
0	1	0	1
1	0	1	0
2	1	0	1

Tabla 5: Anillo Johnson

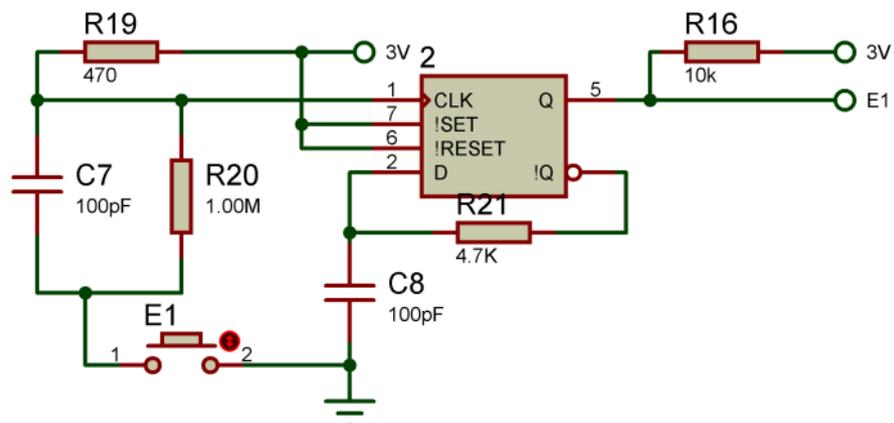


Figura 11: Circuito entrada de retención

RTC

Contábamos con varias opciones para obtener la hora actual en la pulsera. Si bien mediante programación hubiéramos podido simular un RTC (Real Time Clock- Reloj en tiempo real) de manera sencilla, utilizando los

contadores internos; el mismo posee como defecto un desfase de unos 300 segundos al año, lo cual generaría errores en la lectura horaria en tiempo real.

En nuestro caso, sólo simulamos un RTC para generar demoras (delay), en las cuales no se programaron secuencias de código útiles, sino ciclos basura para generar el tiempo requerido en ciertas situaciones, como la espera de un mensaje desde otro dispositivo.

En base a las opciones disponibles, decidimos utilizar el integrado DS3231, que nos brindó la mayor precisión y el menor margen de desfase en el tiempo.

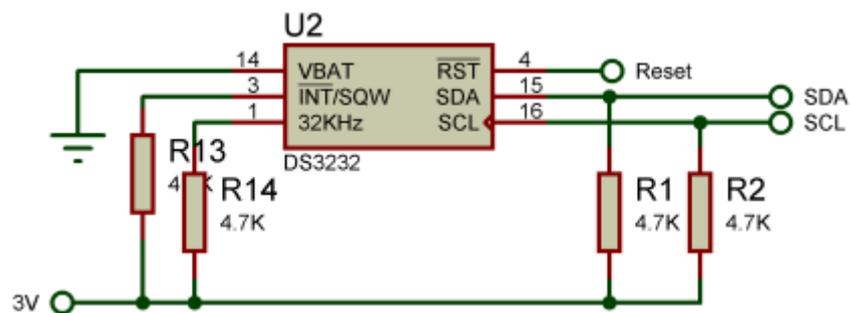


Figura 12: Circuito RTC

BUS DE DATOS

Como se mencionó anteriormente, debíamos contar con un sistema de bus de datos, entre los componentes que forman la pulsera, efectivo y con la menor cantidad de pines posibles. Esto es porque contábamos con varios periféricos externos al microcontrolador y poseíamos una cantidad limitada de E/S disponibles.

Contábamos con dos posibilidades para utilizar. Primero, el bus SPI (Serial Peripheral Interface), el cual posee una comunicación full-dúplex, a una velocidad de 50 MHz, no posee límite de tamaño en las palabras de datos, y utiliza direccionamiento lineal por cada dispositivo (chip select). El problema que hallamos al querer utilizarlo en nuestro proyecto fue la cantidad de salidas que necesitábamos para realizar el bus: 4 líneas para el primer dispositivo y

una línea más por cada dispositivo que agregábamos, como podemos observar en la imagen del conexionado.

El bus I2C (Inter-IntegratedCircuit) como segunda posibilidad, nos presentaba una ventaja clara al utilizar solo 2 líneas entre todos los dispositivos. Aunque presentaba una conexión más lenta entre 400 kbps a 5 Kbps, no utilizaba una línea dedicada para el direccionamiento, sino que el mismo se realizaba con los primeros 7 bits de la transmisión. El tamaño de la palabra de datos estaba limitado a 8 bits. Para su correcto funcionamiento, el bus I2C necesitaba de una resistencia pull-up por cada línea.

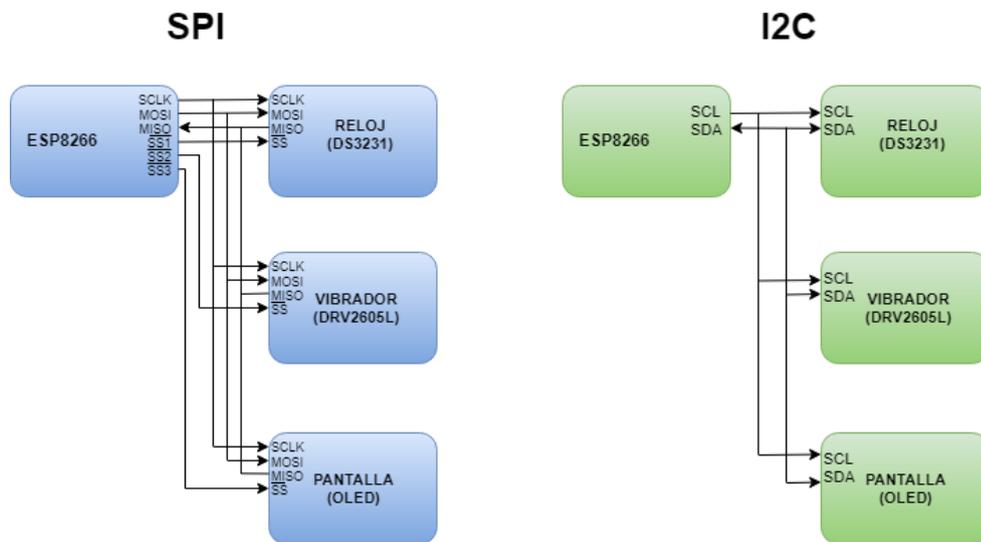


Figura 13: Bus de datos SPI-I2C

Finalmente nos inclinamos por bus de comunicación de datos I2C, obteniendo la máxima velocidad acorde al funcionamiento de los 4 dispositivos, 400 kbps

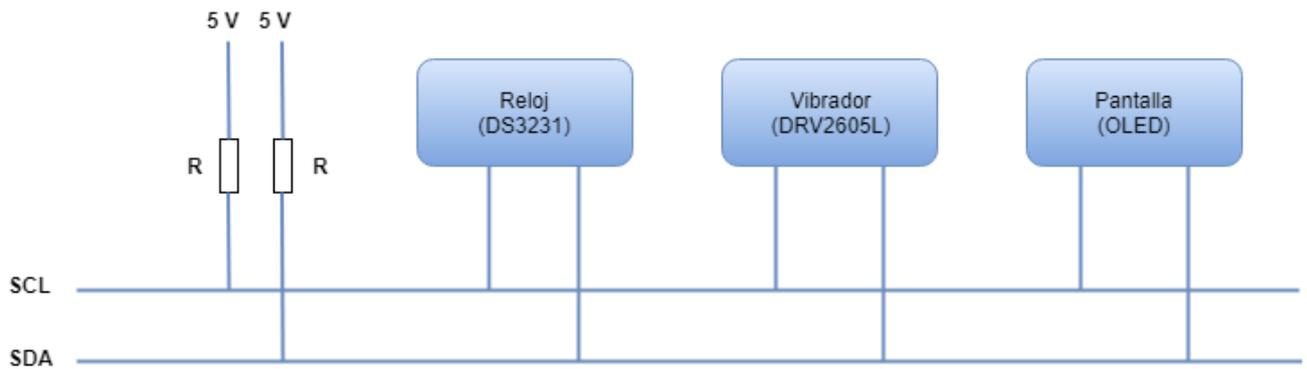


Figura 14: Esquema I2C

La metodología de comunicación de datos del bus I2C es en serie y sincrónica. Una de las señales del bus marca el tiempo (pulsos de reloj) y la otra se utiliza para intercambiar datos.

- **SCL** (System Clock) es la línea de los pulsos de reloj que sincronizan el sistema.
- **SDA** (System Data) es la línea por la que se mueven los datos entre los dispositivos.

PROTOCOLO DE COMUNICACIÓN I2C

Detallaremos ahora el protocolo de comunicación que utilizamos entre los tres dispositivos conectados sobre el bus. En primer lugar, existe un dispositivo maestro (ESP8266) y varios dispositivos esclavos (reloj, vibrador, pantalla). Sólo el maestro puede iniciar una comunicación.

El protocolo de transferencia comienza indicado por la señal de inicio del maestro, seguido de la dirección. Ésta es confirmada por el ACK-Bit del esclavo correspondiente. En función del R/W-Bit (el bit 8 del dato de direccionamiento) se escriben bytes de datos (datos al esclavo) o se leen (datos al maestro). El ACK (acuse de recibo) es enviado desde el esclavo al

escribir, y desde el maestro al leer. El último byte leído es reconocido por el maestro como un NACK (acuse de recibo negado), para indicar el final de una transmisión. Una transmisión es finalizada por la señal de parada.

Cada dispositivo cuenta con una dirección distinta, para no tener un conflicto en la interpretación de los datos y el uso del bus. En nuestro caso, las direcciones de memoria son:

DISPOSITIVO	DIRECCIÓN
DS3231	0x68
DRV2605L	0x5A
OLED	0x3C

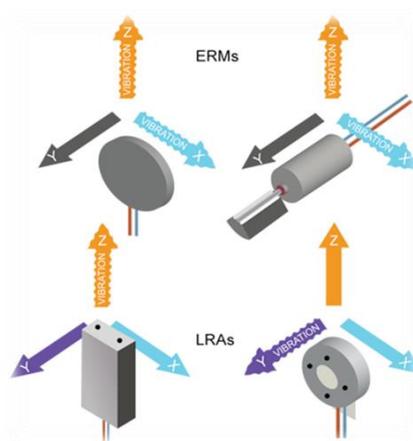
Tabla 6: Dirección de memoria Dispositivos I2C

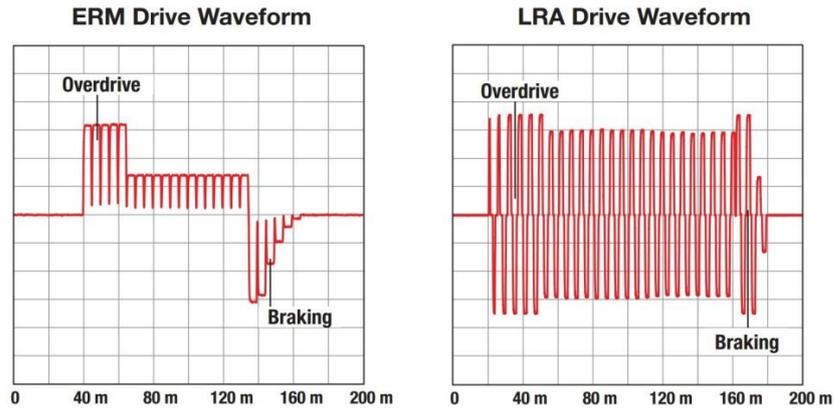
MOTOR VIBRADOR

Como se mencionó anteriormente, una de las formas de alerta al usuario es mediante un motor vibrador. Utilizamos un motor de tipo ERM, con características propicias para nuestra aplicación. La dificultad que se nos presentó en este caso fue el modo de accionar dicho motor para que genere vibraciones.

Dentro de las opciones que analizamos para solucionar la dificultad antes mencionada, la primera fue utilizar una salida de PWM del ESP8266. Esta opción no fue viable, dado que la corriente de salida máxima de los puertos GPIO es de 12 [mA], lo cual eliminaba la posibilidad de usar transistores que necesitaban una corriente de base alta para su conmutación. Otra opción era utilizar un Mosfet, cuya activación se base en la tensión gate-surtidor; sin embargo, no hallamos ninguno que se adapte a las tensiones utilizadas en nuestro dispositivo.

Finalmente optamos por el controlador DRV2605L, el cual permite un manejo sencillo y completo de la actuación del motor vibrador mediante comunicación I2C, característica que fue determinante para los requerimientos de nuestro proyecto. Dicho controlador puede manejar distintos tipos de motores vibradores como LRA (Linear Resonance Actuator) o ERM, ambos sistemas poseen distintas señales para su funcionamiento; a continuación se puede apreciar la diferencia de forma más gráfica, en donde se especifica el arranque y el frenado.





CARACTERÍSTICAS
Comandos de control mediante bus I2C a una velocidad de 400 [mbps]
Admite control mediante entrada PWM con un rango 0% a 100% del ciclo de duty, con una frecuencia de 10 a 250 [kHz]
Voltaje de operación 2 a 5,2 [V]
Posee control de arranque y freno
Integración de librerías con distintos efectos
La frecuencia de salida de la señal PWM posee un rango entre 19,5 a 21,5 [kHz]
Posee 123 efectos preestablecidos, durante la alerta se puede realizar una combinación de los mismos

Tabla 7: Especificaciones DRV2605L

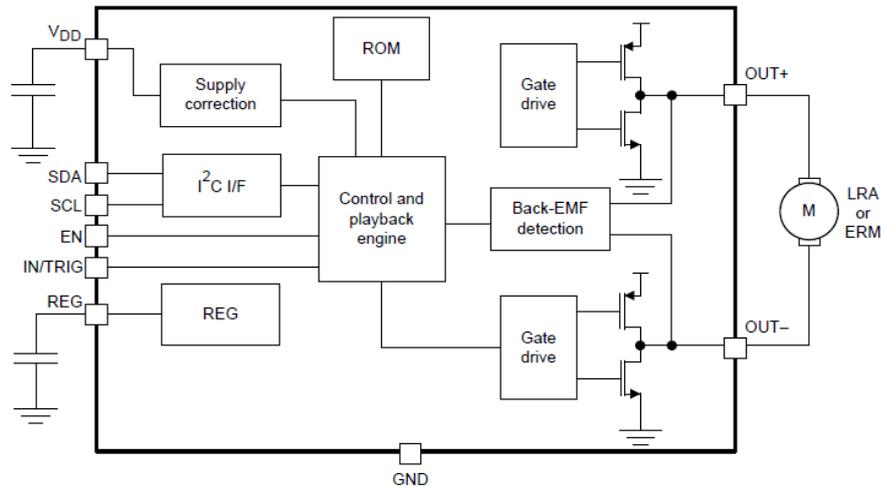


Figura 15: Diagrama funcional DRV2605L

2.3. Sistema de Carga y consumo

El chip TP4056 es el encargado de cargar la batería de litio-ion del dispositivo, a una tensión y corriente constantes. Presenta una corriente de carga máxima de 1000 mA con una tensión promedio de 4,2 [V]. Su tensión de funcionamiento oscila entre 4 y 8 [V], permitiendo utilizar cargadores universales USB de 5 [V].

La corriente de carga es regulada a través de una resistencia de programación (R_{prog}). Como el reloj queda encendido cuando está en modo carga, debemos suministrar una corriente más alta para abastecerlo ($I_{consumo}$) sumada a la corriente que carga la batería (I_{bat}), entonces:

$$I_{carga} = I_{bat} + I_{consumo}$$

Para calcular la R_{prog} obtenemos la fórmula de la hoja de datos:

$$I_{carga} = \frac{V_{prog}}{R_{prog}} * 1200 \text{ si } V_{prog} = 1[v]$$

Despejando y reagrupando obtenemos:

$$R_{prog} = \frac{1200}{I_{bat} + I_{consumo}}$$

La *I*consumo cuando el dispositivo se encuentra en modo de carga es de aproximadamente 100 [mA].

Elegimos para este proyecto una batería de 3,7 [V] – 250 [mAh] Li-PO, por su capacidad y sus dimensiones, las cuales nos resultaron adecuadas. El tiempo de carga total de la misma se estipuló en 2,5 horas. Para ello calculamos la corriente necesaria $I_{bat} = 100 \text{ [mA]}$.

Entonces:

$$R_{prog} = \frac{1200}{0,100 + 0,100} = 6 \text{ [kOHM]}$$

	CONSUMO MÁXIMO [mA]	CONSUMO PROMEDIO [mA]
ESP8266	170	60
DS3231	0.2	0.2
DRV2605L	0.0035	0.0035
OLED	40	20
MOTOR VIBRADOR	120	90
EXTRAS	-	20

Tabla 8: Consumo elementos pulsera

Tensión de alimentación

La tensión de alimentación entre los distintos chips del módulo del reloj debía ser menor o igual a la tensión de la batería, es por esto que la elección de cada componente debía estar acorde una tensión en común. Confeccionamos una tabla para comparar las tensiones:

	TENSIÓN MINIMA [V]	TENSION TIPICA [V]	TENSION MAXIMA [V]
ESP8266	2.5	3.3	3.6
DS3231	2.3	3.3	5.5
DRV2605L	2.0	3.3	5.2
OLED	2.8	3.0	3.3
MOTOR VIBRADOR	2.7	3.0	3.3

Tabla 9: Tensiones elementos pulsera

Como se puede observar, el límite superior es de 3,3 [V] y el límite inferior 2,0 [V]. Por esta razón, buscamos un punto medio de 3,0 [V] de los dispositivos más críticos (motor y OLED) en cuanto a la cercanía entre sus límites de funcionamiento, y que a su vez permita el correcto funcionamiento de todos los dispositivos.

Seleccionamos el chip MCP1700 para regular la tensión, que presenta una salida regulable de 1,2 a 5 [V]; para esta ocasión, seleccionamos una tensión fija de 3 [V]. Su tensión de entrada puede variar de 2,3 a 6 [V] por lo que se encuentra dentro del rango de funcionamiento de la batería (3,2 a 4,2 [V]). Además, incorporamos capacitores de 1 [uF] en la entrada y en la salida para otorgar mayor estabilidad.

La corriente continua suministrada por el regulador es de 250 [mA] para una tensión de salida mayor o igual a 2,5 [V], aunque puede soportar picos de hasta 550 [mA]. Una vez alcanzado este límite, entra en protección por sobre corriente. En nuestro proyecto medimos un consumo pico de 300 [mA] como se

puede observar en la tabla N; por ello, administramos los dispositivos encendidos para evitar llegar a un pico de consumo que ocasionaría daños en el regulador.

Seleccionamos el encapsulado según las formulas obtenidas del fabricante, optando por utilizar una temperatura ambiente de $T_{A(MAX)} = 45^{\circ}\text{C}$.

La potencia de disipación interna $PLDO_{(MAX)}$ debía ser menor a la que puede soportar el encapsulado, entonces:

$$PLDO_{(MAX)} = (V_{IN(MAX)} - V_{OUT(MIN)}) \times I_{B2t(MAX)}$$

$$PLDO_{(MAX)} = (4,2 \text{ [V]} - 3 \text{ [V]}) \times 250 \text{ [mA]} = \boxed{300 \text{ [mW]}}$$

La máxima disipación hacia el ambiente con los encapsulados SMD se calcula:

$$\text{Encapsulado SOT-23: } \left(R\theta_{JA} = 230.0 \left[\frac{\text{e C}}{\text{W}} \right] \right)$$

$$PD_{(MAX)} = \frac{(125 \text{ [}^{\circ}\text{C]} - 45 \text{ [}^{\circ}\text{C]})}{\square} 230 \left[\frac{\text{e C}}{\text{W}} \right]$$

$$PD_{(MAX)} = \boxed{347.8 \text{ [mW]}}$$

$$\text{Encapsulado SOT-89: } \left(R\theta_{JA} = 52 \left[\frac{\text{e C}}{\text{W}} \right] \right)$$

$$PD_{(MAX)} = \frac{(125 \text{ [}^{\circ}\text{C]} - 45 \text{ [}^{\circ}\text{C]})}{\square} 52 \left[\frac{\text{e C}}{\text{W}} \right]$$

$$PD_{(MAX)} = \boxed{1.538 \text{ [W]}}$$

Con los datos obtenidos de los cálculos podemos observar que la **PD(MAX)** del encapsulado SOT-23 se encuentra muy cercana a sus límites operativos, por eso optamos por seleccionar el SOT-89 que nos permitía una disipación de temperatura 5 veces más alta a nuestro consumo máximo. De esta forma podría soportar cualquier pico de corriente que pudiera ocurrir.

Luego estimamos la temperatura de la junta con el encapsulado SOT-89:

$$T_{J(RISE)} = P_{TOTAL} \times R\theta_{JA}$$

$$T_{J(RISE)} = 300 [mW] \times 52 \left[\frac{^{\circ}C}{W} \right]$$

$$T_{J(RISE)} = 15.6 [^{\circ}C]$$

REFERENCIAS

P_{LDO} = LDO Pass device internal power dissipation

$V_{IN(MAX)}$ = Maximum input voltage

$V_{OUT(MIN)}$ = LDO minimum output voltage

$T_{J(RISE)}$ = Rise in device junction temperature over the ambient temperature.

P_{TOTAL} = Maximum device power dissipation.

$R\theta_{JA}$ = Thermal resistance from junction to ambient.

T_J = Junction Temperature.

$T_{J(RISE)}$ = Rise in device junction temperature over the ambient temperature.

T_A = Ambient temperature.

Pantalla OLED

Para poder mostrarle la hora y las alertas enviadas por el servidor al usuario, utilizamos una pantalla gráfica de 128 x 64 píxeles. Su construcción está basada en el diodo orgánico de emisión de luz OLED (Organic Light-Emitting Diode), que genera y emite luz por sí mismo. Esto nos permitió tener una pantalla mucho más fina que las pantallas TFT.

Una característica importante es su bajo consumo respecto a los demás paneles, ya que sólo los píxeles encendidos consumen energía, además del controlador; esta característica permite tener un contraste más alto porque el color negro no es simulado, si no que el píxel se encuentra apagado. La duración del panel ronda las 50.000 horas según el color de los píxeles.

El controlador SH1106 es el encargado de administrar el panel; el mismo presenta varios protocolos de comunicación como 3-SPI, 4-SPI, 8 bit 64XX Parallel, 8 bit 64XX Parallel y I2C de 100 a 400 [kbps], éste último fue el que seleccionamos. El SH1106 posee un generador interno que se encarga de alimentar cada oled con su tensión correspondiente.

Las dimensiones de la pantalla nos resultaron adecuadas al tipo de dispositivo que queríamos implementar. Sobre todo el espesor, el cual nos permitió adaptar el pcb del resto de componentes a la misma.

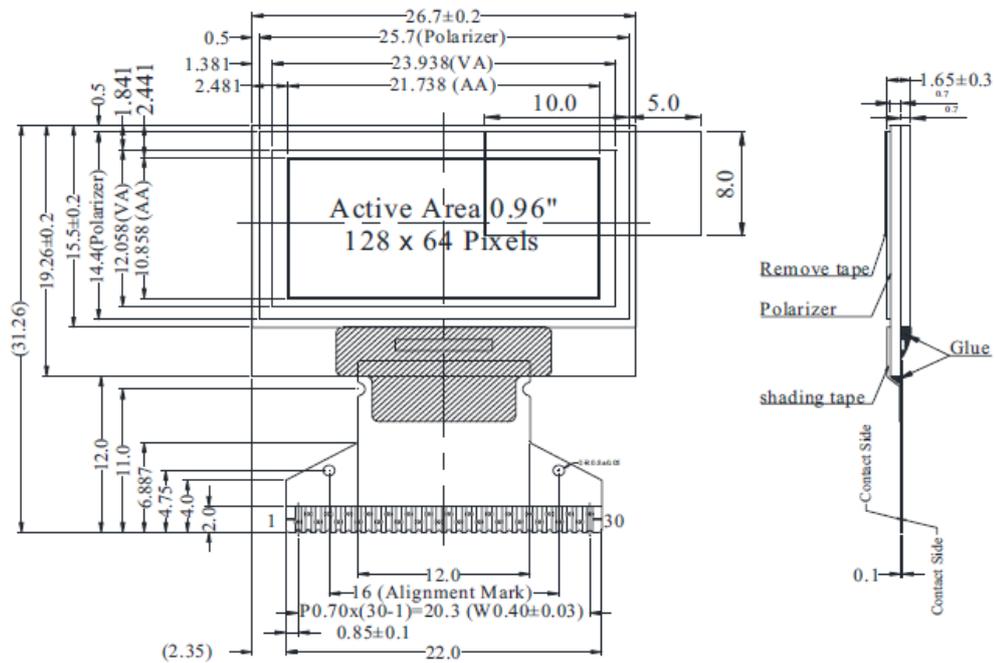
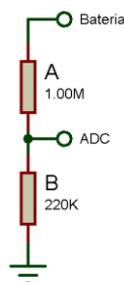


Figura 16: Dimensiones Pantalla Pulsera

Nivel batería

El ESP8266 dispone de un único ADC (convertidor analógico a digital) de 10 bits, es decir que posee 1024 valores de conversión; se puede utilizar para medir tanto la tensión del dispositivo o una señal externa, como la tensión de la batería.

La entrada se caracteriza por ser acotada de 0 a 1 volt, por lo cual se debe realizar una adaptación para el rango de tensión de la batería (3,2 a 4,2 [V]). Colocamos entonces un divisor resistivo de valores elevado, dado que siempre va a consumir corriente.



$$ADC = \frac{B * Bateria}{B + A}$$

NIVEL DE BATERIA	ADC	Bit
3.2 V.	0.577 V.	591
4.2 V.	0757 V.	775

Tabla 10: Nivel de batería ADC

En el cuadro podemos observar la tensión correspondiente a los niveles de la batería que obtenemos a través del divisor, junto con su correspondiente conversión al valor digital. Los valores digitales nos establecen el límite inferior y superior con el cual determinamos la carga.

Consumo

Se realizaron distintas pruebas para determinar la autonomía que poseía la pulsera. Las baterías utilizadas fueron de polímero de iones de litio de dos capacidades: 230 mAh y 420 mAh.

A continuación se incluye una tabla con las características de cada prueba y su resultado neto de horas de funcionamiento.

Pruebas de consumo pulsera					
Prueba N°	Hora	Nivel de carga	Condición de prueba	Tiempo neto de funcionamiento	Capacidad de la batería [mA]
1	16:00	100	<ul style="list-style-type: none"> ■ Se programó para uso continuo tanto de la pantalla como de los tiempos de interrogación al servidor 	02:00	
	16:30	82			
	17:00	55			
	17:30	21			
	18:00	0			
2	13:00	100	<ul style="list-style-type: none"> ■ Interrogación cada 30 seg. ■ Sin alertas. ■ Pantalla apagada. 	06:10	
	14:00	84			
	15:00	66			
	16:00	44			
	17:00	21			
	18:10	0			
3	22:00	100	<ul style="list-style-type: none"> ■ Interrogación cada 30 seg. ■ Sin alertas. ■ Pantalla apagada. ■ Se optimizó la potencia (5 db) 	23:42	230
	01:00	93			
	04:00	88			
	08:00	73			
	10:00	68			
	13:00	51			
	15:30	45			
	18:30	38			
	21:00	16			
21:42	0				
4	19:42	100	<ul style="list-style-type: none"> ■ Interrogación cada 30 seg. ■ Pantalla apagada ■ 10 alertas de timbre 	24:03:00	
	19:45 *	100			
	21:00 *	97			
	22:30 *	94			
	02:00 *	90			
	04:00 *	84			
	13:07 *	46			
	15:03 *	41			
	16:35 *	35			
	18:23 *	24			
	18:55 *	15			
	19:45 *	0			
5	06:00	100	<ul style="list-style-type: none"> ■ Interrogación cada 30 seg. ■ Pantalla apagada ■ 10 alertas de timbre 	37:15:00	420
	07:30 *	100			
	09:30 *	92			
	11:30 *	85			
	14:00	78			
	17:30 *	71			
	20:00 *	63			
	00:15 *	51			
	06:00 *	38			
	08:00	33			
	11:30 *	27			
	13:00 *	18			
	15:30 *	12			
	17:00 *	8			
	19:13	0			

Tabla 11: Prueba de consumo

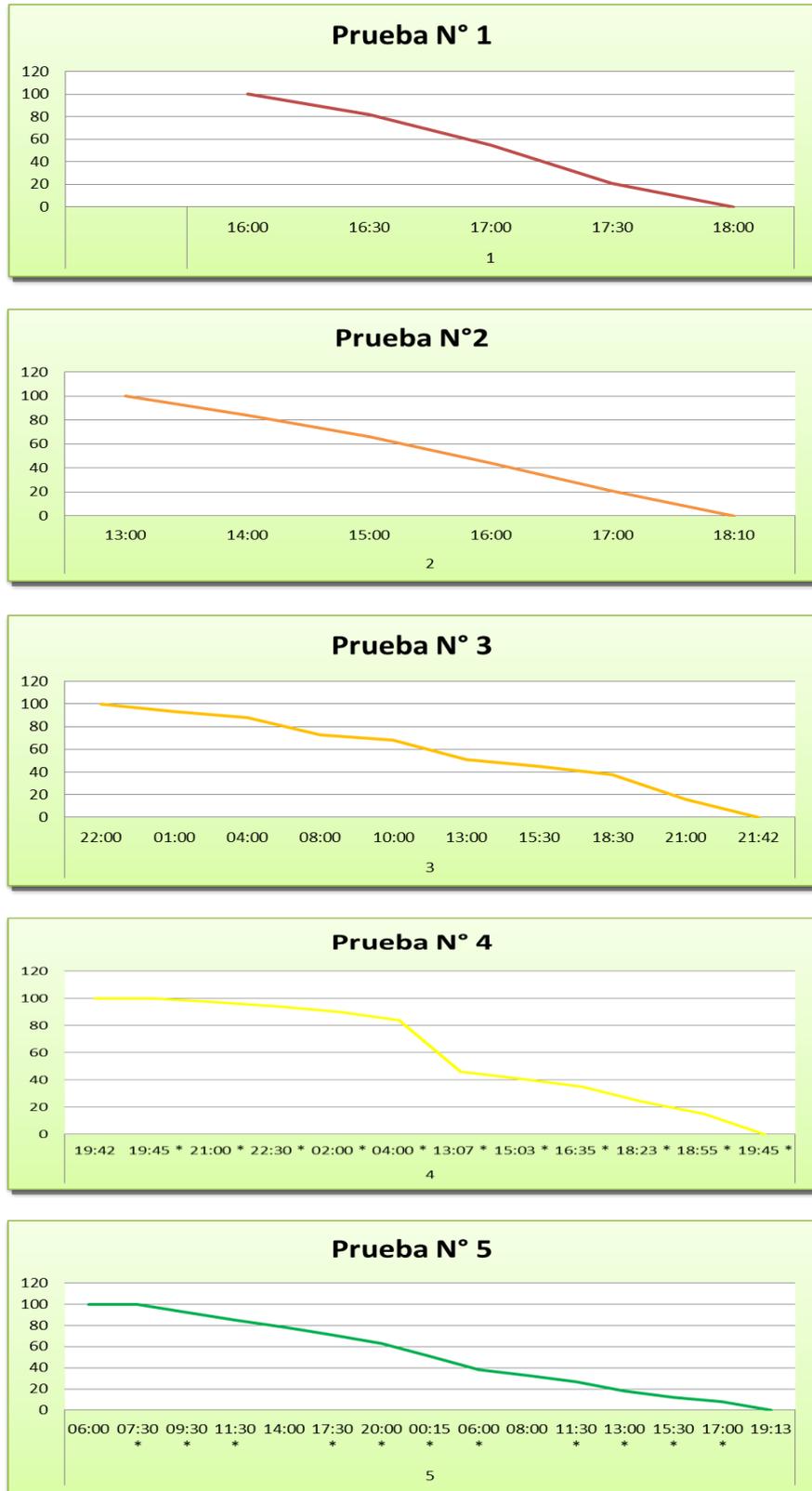


Figura 17: Gráficos Consumo Pulsera

2.4. Desarrollo del Software

Protocolos de Comunicación

La intercomunicación entre el servidor, los relojes y los dispositivos con Android se basó en la implementación del protocolo UDP (User Datagram Protocol) ubicado en la capa 4 del modelo OSI. El puerto de origen y destino que utilizamos fue el 9999.

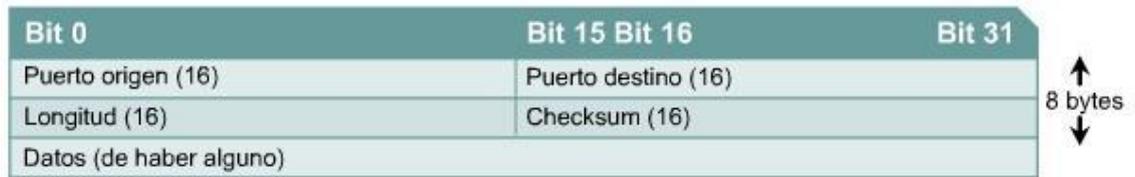
El UDP es un protocolo basado en el envío de datagramas a través de la red sin haber establecido una conexión previamente, por este motivo no posee confirmación ni control de flujo. El datagrama incorpora la información suficiente en su cabecera para el direccionamiento.

Los protocolos que usan UDP son:

- TFTP (Protocolo trivial de transferencia de archivos)
- SNMP (Protocolo simple de administración de red)
- DHCP (Protocolo de configuración dinámica del host)
- DNS (Sistema de denominación de dominios)

Las siguientes son las definiciones de los campos de un segmento UDP:

- *Puerto origen*: Número del puerto que realiza la llamada
- *Puerto destino*: Número del puerto al que se realiza la llamada
- *Longitud*: Número de bytes que se incluyen en el encabezado y los datos
- *Checksum* (suma de comprobación): Suma de comprobación calculada a partir de los campos del encabezado y de los datos.
- *Datos*: Datos de protocolo de capa superior



Servidor FTP

FTP es un servicio confiable orientado a conexión que utiliza TCP para transferir archivos entre sistemas que admiten FTP. El propósito principal es transferir archivos, copiándolos y moviéndolos desde los servidores hacia los clientes, y viceversa. Cuando los archivos se copian de un servidor, FTP primero establece una conexión de control entre el cliente y el servidor. Luego, se establece una segunda conexión, que es un enlace entre los computadores a través del cual se transfieren los datos, en modo ASCII o en modo binario. Estos modos determinan la codificación que se usa para el archivo de datos que, en el modelo OSI, es una tarea de la capa de presentación. Cuando termina la transferencia de archivos, la conexión de datos se termina automáticamente. Una vez que se ha completado toda la sesión para copiar y trasladar archivos, el vínculo de comandos se cierra cuando el usuario se desconecta y finaliza la sesión.

Utilizamos la conexión FTP en el presente proyecto específicamente para la carga de los archivos directamente desde la aplicación en donde fue desarrollada la página web. Esto posibilitó agilizar las pruebas, pasando de unos 5 a 10 minutos de transferencia mediante el puerto serie, a transferir o actualizar todos los archivos en menos de 30 segundos mediante el uso de la red. Sin embargo, para que la comunicación ocurriera en forma fluida se debían presentar dos condiciones: que el módulo estuviera conectado a la misma red, y que el microcontrolador no tuviera delays (tiempos muertos de ejecución).

DNS

Internet está basado en un esquema de direccionamiento jerárquico. Este esquema permite que el enrutamiento se base en clases de direcciones en lugar de basarse en direcciones individuales. El problema que esto crea para el usuario es la asociación de la dirección correcta con el sitio de Internet. Es muy fácil olvidarse cuál es la dirección IP de un sitio o dispositivo en particular dado que no hay ningún elemento que permita asociar el contenido del sitio con su dirección.

Imaginamos lo difícil que sería recordar direcciones IP del servidor o de los relojes. Por eso desarrollamos un sistema de denominación de dominio para poder asociar el contenido del sitio de configuración con su dirección. El Sistema de denominación de dominios (DNS: Domain Name System) es un sistema utilizado en Internet para convertir los nombres de los dominios y de sus nodos de red publicados abiertamente en direcciones IP.

Cabe destacar que la implementación del DNS solo se puede utilizar mediante una PC que ejecute el sistema operativo Windows, ya que estamos realizando una redirección local.

Para poder facilitar al usuario final la configuración de los módulos optamos por el uso de distintos DNS ya que la mayoría de las personas no usan en sus routers direcciones fijas, si no que utilizan el protocolo DHCP, el cual asigna una dirección IP a cada dispositivo conectado a la red. Cuando los módulos estén conectados a la red, tendremos dificultad de obtener la dirección IP en la cual está asignado el dispositivo. La solución que implementamos entonces fue que el módulo servidor pueda ser accedido mediante la siguiente dirección `http://servidor/` en cualquier tipo de navegador web.

Debido a que la misma red admite una o varias pulseras, se buscó la forma de identificar a cada una de ellas mediante el número único que posee el chip ESP8266, de esta forma podemos tener direcciones de muchas pulseras conectadas en la misma red, y acceder fácilmente a cada una de ellas. Entonces la dirección web queda conformada de la siguiente manera `http://pulsera_XXXXXX/`, donde XXXXXX es un conjunto alfa numérico de 6

caracteres que corresponderá a cada una, el mismo deberá ser proporcionado individualmente para cada dispositivo, ya sea en su caja, manual o en el mismo dispositivo.

Ejemplos:

`http://pulsera_9B5778/`

`http://pulsera_0F1EA1/`

JSON

Con el fin de hacer más ordenada la transferencia de datos entre los distintos dispositivos, y desde el dispositivo hacia el navegador web, optamos por la utilización del formato ligero de archivos JSON (JavaScript ObjectNotation - Notación de Objetos de JavaScript).

JSON es un formato de texto completamente independiente del lenguaje, pero utiliza convenciones que son ampliamente conocidas por los programadores de la familia de lenguajes C, incluyendo C, C++, C#, Java, JavaScript, Perl, Python, y muchos otros. Estas propiedades hacen que JSON sea un lenguaje ideal para el intercambio de datos.

Tipos de datos transferibles:

- Números: Se permiten números positivos y negativos, opcionalmente pueden contener parte fraccional separada por puntos.

Ejemplo: 123.456

- Cadenas: Representan secuencias de cero o más caracteres. Se ponen entre doble comilla y se permiten cadenas de escape.

Ejemplo: "Hola"

- Booleanos: Representan valores booleanos y pueden tener dos valores: true y false

- Null: Representan el valor nulo.

- Array: Representa una lista ordenada de cero o más valores los cuales pueden ser de cualquier tipo. Los valores se separan por comas y el vector se coloca entre llaves.

Ejemplo: {"juan","pedro"}

- Objetos: Son colecciones no ordenadas de pares de la forma <nombre>:<valor> separados por comas y puestas entre llaves. El nombre tiene que ser una cadena, y entre ellas el valor puede ser de cualquier tipo.

Ejemplo de una empresa:

```
{  
  "departamento":8,  
  "nombredepto":"Ventas",  
  "director":"juanRodriguez",  
  "empleados":[{"nombre":"Pedro","apellido":"Fernandez"}, {"nombre":"Jacinto",  
"apellido":"Benavente"}]  
}
```

Dentro de nuestra codificación lo utilizamos específicamente en estos casos:

- Archivo donde se guarda la configuración realizada, el mismo es almacenado en la memoria integrada del ESP8266. A continuación visualizamos como se guarda config.json donde se utilizan principalmente objetos de diferentes tipos:

```
{  
  "Modo":"WiFi",  
  "Firmware":" Firmware: Reloj: v5.5 + v18.5 ",  
  "E1_nombre":"Timbre",  
  "E2_nombre":"Entrada 2",
```

```
"E3_nombre":"Entrada 3",
"S1_nombre":"Salida 1",
"S2_nombre":"Salida 2",
"S3_nombre":"Salida 3",
"E1_Estado":"1",
"E2_Estado":"0",
"E3_Estado":"1",
"S1_Estado":"0",
"S2_Estado":"1",
"S3_Estado":"0",
"E1_Relacion":"1",
"E2_Relacion":"5",
"E3_Relacion":"7",
"ssid":"",
"password":"",
"rssi":"",
"ip":"",
"canal":"",
"softAP_ssid":"admin",
"softAP_password":"admin",
"www_username":"admin",
"www_password":"admin",
"update_username":"admin",
"update_password":"admin",
"OTA_password":"admin",
"FTP_username":"admin",
"FTP_password":"admin"
}
```

- Comunicación entre los dispositivos: acciones a realizar, así como la identificación de quien envió el paquete.

Ejemplo: {"Registro":["Pulsera_9B5778","Reloj",0,Pulsera]}

- Comunicación entre los dispositivos (el servidor o un reloj) y la página web cuando se escanean las redes wifi disponibles.

Ejemplo: {"Disponibles":["SSID: Cipriani - RSSI: -51 [dBm] - Canal: 6","SSID: Luna - RSSI: -88 [dBm] - Canal: 11"]}

DEEP SLEEP

El modo deepsleep del ESP8266 es responsable del reinicio normal por software/hardware para el cual se establece un tiempo mínimo y un tiempo de reinicio programado, el cual utilizamos para ahorrar batería en el reloj. Este modo deja inoperativa la placa dado que apaga el wifi, el clock del sistema y el CPU, dejando únicamente funcionando el RTC interno, el cual enviará la señal para que se reinicie el microprocesador cuando finalice la cuenta regresiva de tiempo de suspenso que hayamos establecido.

El consumo en este modo es de aproximadamente 20 [uA], lo cual constituye un considerable ahorro en relación al consumo normal (100 [mA]), permitiendo así extender el tiempo de uso de la batería del módulo reloj. Mientras no ocurra ningún evento que deba ser transmitido al usuario, el sistema quedará inoperativo durante el tiempo asignado, en modo deepsleep.

La función tiene dos variables:

ESP.deepSleep(tiempo_en_us, modo_reinicio), donde:

- ◆ tiempo_en_us (uint32_t): tiempo en microsegundos que debe estar el microprocesador en suspenso. El tiempo máximo que se puede mantener el microprocesador en suspenso es de aproximadamente 71 minutos y 30 segundos. Esto se debe a que los microprocesadores ESP8266 trabajan a 32 bits, por lo tanto el número más alto que admiten es $2^{32} = 4.294.967.295$. Dado que el tiempo de suspenso se da en

microsegundos (μs) tenemos que: $4.294.967.295 \mu\text{s} = 4.294,967295 \text{ s} \approx 71' 30''$

- ◆ `modo_reinicio`: indica cómo se debe proceder cuando se reinicia el microprocesador con la calibración de la señal de radio para la WiFi. Existen cuatro posibilidades:

1 = `WAKE_RF_DEFAULT`: cuando se reinicia el microprocesador, únicamente se calibra la señal de radio si el chequeo da error (`init data byte 108 > 0`).

2 = `SLEEP_TIME, WAKE_RFCAL`: cuando se reinicia el microprocesador siempre se calibra la señal de radio. Esto incrementa el consumo.

3 = `SLEEP_TIME, WAKE_NO_RFCAL`: cuando se reinicia el microprocesador no se calibra la señal de radio. Esto reduce el consumo.

4 = `WAKE_RF_DISABLED`: cuando se reinicia el microprocesador se deshabilita la señal de radio (como en el modo `modem sleep`). Este es el modo con menor consumo, pero no permite ni enviar ni recibir datos vía WiFi.

Cuando haya finalizado el tiempo establecido para que el microprocesador quede en suspenso, el Real Time Clock enviará una señal para que el microprocesador se reinicie (Reset). En nuestro proyecto, conectamos el pin GPIO16 con el pin de Reset para que el módulo ESP se reinicie junto con el microprocesador.

Memoria SPIFFS

El sistema de archivos de registro que utilizamos se basa en SPIFFS (SPI Flash File System), el cual fue diseñado para sistemas con memoria SPI NOR flash en dispositivos embebidos. El sistema de archivos tiene como objetivo principal usar la mínima cantidad posible memoria RAM; en nuestro caso, también lo utilizamos para guardar configuraciones como si fuera el eeprom del microcontrolador.

El sistema reserva un espacio en la memoria flash para la creación de archivos, que pueden variar según el modelo del módulo ESP. Esta asignación puede ir desde 64 [kB] hasta 3 [MB]. En el presente proyecto utilizamos 3 [MB] para guardar los archivos de la página web y los de configuración.

WPS

WiFi Protected Setup o WPS, es una estándar de seguridad de red presente en algunos routers que nos permite conectarnos de forma inalámbrica con sólo pulsar un botón. De esta manera, no es necesario introducir la contraseña del WiFi y el emparejamiento con dispositivos compatibles se lleva a cabo de forma mucho más sencilla.

Existen distintas formas de conectarse a una red utilizando WPS. Por ejemplo, si pulsamos el botón WPS de nuestro router podremos ver al instante qué dispositivos compatibles se encuentran a su alcance. Así, de vuelta a nuestro ordenador, tableta, Smartphone u otro aparato con conexión inalámbrica sólo tendremos que seleccionar la red en cuestión y listo: el emparejamiento habrá finalizado sin tener que teclear nada.



Existen también ciertos dispositivos con su propio botón WPS, siendo aún más fácil la configuración. Primero pulsamos el botón del router y luego el de dichos aparatos y ya estará todo preparado. Este proceso solo lo tendremos que llevar a cabo una vez, y no en cada ocasión que pretendamos

conectarnos, ya que la configuración quedará convenientemente almacenada. En nuestro caso, utilizamos esta facilidad y almacenamos los datos en el archivo de configuración, facilitándole considerablemente la conexión a nuestro potencial cliente.

En el caso de que el router del potencial cliente tuviese desactivado el acceso vía WPS, basta con entrar en su interfaz desde un navegador web y habilitarlo.



NTP

Network Time Protocol (NTP) es un protocolo de Internet que utilizamos para sincronizar el reloj de nuestro RTC a través del enrutamiento de paquetes en redes con latencia variable. El Protocolo NTP utiliza UDP como capa de transporte, mediante el puerto 123. Por ejemplo, el sistema operativo Windows utiliza este método para mantener actualizada la hora en las computadoras.

2.4.1. Pulsera

Para establecer la codificación de la pulsera seguimos los siguientes pasos:

- I. Para empezar, realizamos la configuración serial para poder hacer un debug mientras observábamos cómo se ejecutaba el código. Esto nos permitió saber si se estaba corriendo el programa o se encontraba tildado, para lo cual realizamos impresiones que nos permitieran identificar la línea en la que se encontraba ejecutando el microprocesador. Cabe destacar que esta opción en la versión final será deshabilitada llevando a una mejora del rendimiento del sistema.
- II. Inicializamos la configuración del Bus de datos I2C donde se definen los pines (GPIO5 para SDA y GPIO4 para SCL) y su velocidad de ejecución (400 [kHz]).
- III. Con el Bus ya configurado, procedemos a comunicarlo con la pantalla a fin de reconocer su conexión y establecer sus parámetros inicio. También invertimos verticalmente el formato de muestra, para que se adapte a la forma de la pulsera.
- IV. Establecemos los pines GPIO como entradas o salidas:

GPIO13-	Entrada Cargando
GPIO12-	Entrada Cargado
GPIO14-	Botón Pantalla
GPIO2-	Botón Default
A0-	Entrada Analógica para censar la batería
- V. Configuramos las interrupciones para los botones. La interrupción para volver al formato de fábrica sólo podrá utilizarse si la pulsera está en modo carga. Además configuramos el botón para mantener la pantalla encendida con flanco ascendente.

- VI.** Leemos el nombre del chip para poder configurar el nombre de la página web y su identificación.
- VII.** Inicializamos el sistema de archivos SPIFFS y leemos todos los archivos alojados en la memoria, con el fin de obtener un registro para su acceso inmediato.
- VIII.** Inicializamos el servidor FTP siempre y cuando el sistema de archivos se inicializó correctamente. Establecemos como nombre de usuario y contraseña “admin”.
- IX.** Leemos el archivo de configuración para obtener las configuraciones predeterminadas, o las que configuró el usuario. Dentro de este archivo encontraremos los siguientes datos: Modo de operación (AP o WiFi), SSID, contraseña, nombres asignados a las entradas y a las salidas, relaciones asignadas entre las salidas y entradas, versión del Firmware, alias, ultima dirección IP, contraseñas para la actualización del firmware mediante OTA y el tiempo de Deep Sleep.
- X.** Modo de ejecución del módulo inalámbrico:
 - **AP:**

El modo Access Point, solo se ejecutará cuando el sistema se encuentre en el primer inicio de fábrica o cuando el usuario lo restablezca a ese punto.

En este modo, se crea un punto de acceso WiFi hacia el módulo, como si se tratase de un router; pero su red se identifica con el nombre de la pulsera (“Pulsera_XXXXXX”). Su contraseña de acceso es PulseraDefault.

Una vez que el usuario se conecte a dicha red, deberá acceder por un navegador web a su página de configuración http://Pulsera_XXXXXX/ o mediante la aplicación del sistema Android a fin de establecer los datos correspondientes para que el dispositivo se conecte a la red WiFi que dispone en su local o residencia.

Si la prueba falla al ingresar los datos del SSID y su contraseña, el sistema volverá iniciar en modo AP hasta que la conexión sea exitosa. En este momento, se establecerá el modo WiFi y se reiniciará el dispositivo.

○ **WiFi:**

El modo WiFi o Station, obtiene las credenciales previamente cargadas del archivo de configuración y procede a la conexión de la red WiFi.

La conexión se intentará 3 veces durante 15 segundos máximos; en caso de conexión o desconexión, el sistema seguirá la interpretación del código.

Si la conexión es exitosa, obtendremos algunos datos de la misma para que puedan ser informados al usuario mediante la página web o la APP; Obtenemos la potencia de conexión (RSSI), el IP asignado a la pulsera y el canal (frecuencia) en la que se encuentra la red.

XI. Configuramos el servidor interno, establecemos las páginas a las cuales se podrá acceder mediante el navegador Web. Las primeras cinco que mencionaremos se encuentran ocultas para el usuario, la única forma de acceder para su control es ingresándola en el navegador tal cual se encuentra tipada a continuación:

- http://Pulsera_XXXXXX/jsonOUT; es la página de transferencia de datos entre el modulo y el navegador WEB, sus datos están expresados mediante el formato JSON y su modo de operación es GET
- http://Pulsera_XXXXXX/jsonIN; se utiliza para la recepción de los datos enviados desde el navegador web hacia el modulo, el formato de recepción es JSON y el modo de operación es POST
- http://Pulsera_XXXXXX/jsonSCAN; presenta las mismas características que la primera pero únicamente se utiliza para enviar los datos obtenidos cuando se realiza un escaneo de las redes cercanas a el modulo.

- http://Pulsera_XXXXXX/edit; permite ver los archivos cargados en la memoria SPIFFS, borrarlos, editarlos o cargar uno de forma manual. Esta extensión se utilizara únicamente para actualizar el código que corre el microprocesador, mediante un archivo ya compilado. Hay que destacar que si el archivo de actualización es erróneo o es corrupto, al cargarlo podría dejar fuera de servicio el módulo. La única forma de reparar esto es mediante una conexión física, que el usuario no podrá hacer.
- http://Pulsera_XXXXXX/index.htm; permite el ingreso a la página principal de configuración del módulo, las subpáginas no se configuran dado que el servidor resuelve automáticamente cuando se pide una de ellas a partir de la principal.

XII. Configuramos el sistema de interpretación DNS para poder acceder a las páginas en cualquiera de los modos de ejecución. Establecemos el nombre de acceso y añadimos el servicio HTTP para TCP, puerto 80.

XIII. Iniciamos el protocolo de comunicación UDP, estableciendo el puerto 9999 de entrada y salida de datos.

XIV. Verificamos si la pulsera está en modo “carga de batería”. En caso afirmativo, mandamos un registro hacia el servidor por medio del bus de datos UDP. Para el registro, armamos una secuencia de caracteres en formato Json, en donde incluimos los siguientes datos: Alias, nombre de identificación y porcentaje de la batería.

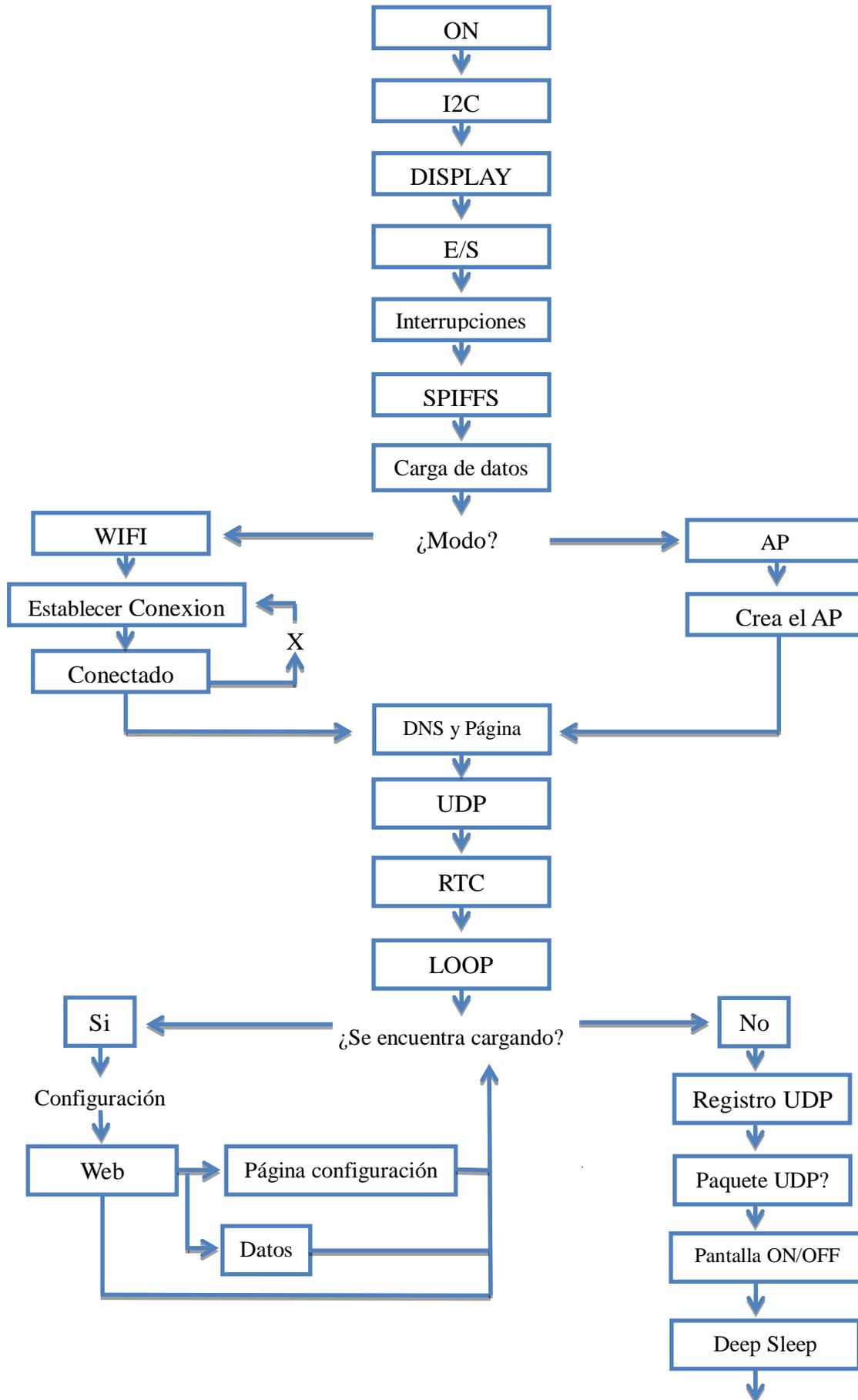
En la pantalla se visualizará el logo de la facultad y el logo de WiFi.

XV. Iniciamos la comunicación con el chip del RTC y realizamos las siguientes acciones:

- Apagamos la salida de frecuencia fija por el pin de 32 [kHz] y la salida de señal variable.
- Verificamos que el proceso interno del RTC se encuentre corriendo; en caso contrario, se lo inicializa. Esto podría ocurrir en caso que la pulsera no tenga batería para mantener el RTC funcionando.

- Ahora confirmamos si estamos conectados a una red WiFi y si ésta posee acceso a internet, ya que la actualización de la hora en la pulsera se realiza de forma automática únicamente en este paso. La hora sólo se actualiza al comparar el horario que posee el RTC y el obtenido del servidor NTC.
 - Si no se puede establecer una conexión con el servidor NTC la hora queda establecida por la del RTC.
- XVI.** Si el sistema no está cargando la batería, procederá a realizar los siguientes pasos:
- Si está conectado a la red WiFi, mandará una petición al módulo servidor, esperando como máximo 2,5 segundos para recibir una respuesta.
 - Si la respuesta llega, es analizada para verificar si hay una alerta para el usuario, ya sea una entrada externa o el timbre. En caso de alerta, prenderá el motor vibrador y mostrará en la pantalla la alerta durante un tiempo determinado.
 - Si el punto anterior terminó su ejecución o no hay respuesta por parte del servidor, continuará la ejecución del código.
 - Verificamos la entrada que determina si la pantalla debe estar encendida o apagada, según la configuración elegida por el usuario mediante el botón dedicado para tal fin.
 - Luego la pulsera entra en estado Deep Sleep, por un tiempo determinado por el usuario que va de 5 a 60 segundos. Pasado ese tiempo, la pulsera se reinicia y comienza el ciclo nuevamente.
- XVII.** Si el sistema está en modo de carga, realizará ciclos continuos hasta que el sistema deje de cargar y se reinicie. El estado de carga de la batería se mostrará en la pantalla.

En este punto la pulsera no realizará ninguna alerta ya que se considera que no está siendo portada por el usuario.



2.4.2. Servidor

Algunos pasos para la configuración del módulo servidor son idénticos a los mencionados anteriormente para el módulo pulsera. Mencionaremos a continuación las diferencias existentes:

- I. Ídem paso 1
- II. Establecemos los pines GPIO como entradas o salidas:

GPIO5-	Salida 1
GPIO4-	Salida 2
GPIO15-	Salida 3
GPIO14-	Entrada 1 (Timbre)
GPIO13-	Entrada 2
GPIO12-	Entrada 3
GPIO2-	Botón Default
- III. Configuramos las interrupciones para las 3 entradas y para los botones, con flanco descendente; al ejecutar una interrupción, se ejecuta una función específica a cada una, que se detallará más adelante.
- IV. Ídem paso 6
- V. Ídem paso 7
- VI. Ídem paso 8
- VII. Ídem paso 9
- VIII. Ídem paso 10, cambiando la contraseña del modo AP, por "ServidorDefault".
- IX. Ídem paso 11. Utilizamos una página adicional <http://servidor/jsonDispositivo>; y pasamos los datos de los dispositivos registrados en el servidor.
- X. Ídem paso 12
- XI. Iniciamos funciones Ticker, las cuales se repiten cada cierto tiempo; el RTC interno del servidor es quien marca la repetición. Las funciones son

para borrar la bandera de alerta a los 60 segundos de producirse una de ellas y el restablecimiento del perro guardián.

XII. LOOP: el ciclo indefinido de repetición posee 2 pasos internos posibles:

- a. Primera posibilidad, responder a un llamado de solicitud para el servidor web integrado. El servidor buscará en la lista de archivos cargados; una vez encontrado el archivo, transmitirá el contenido a quien lo solicitó.

También podemos solicitar por medio de una página específica el contenido para completar los datos de la página solicitada por medio del formato JSON.

- b. La segunda posibilidad, se basa en la comunicación entre los dispositivos mediante el protocolo UDP. Cuando llega un paquete de una pulsera o de un dispositivo Android, se procede a buscarlo en una lista dinámica de registro de dispositivos.
 - Si el dispositivo no se encuentra en dicha lista, se procede a generar un nuevo nodo en la lista, donde se almacena su nombre (Pulsera_XXXXXX o Android_XXXXXX) y su alias (Ejemplo: Franco, Nico, etc.), también se almacena su porcentaje de batería.
 - Cuando ya se encuentra registrado dentro de la lista, se procederá a averiguar si hay alguna alerta en curso.

Si se produjo una alerta, se verificarán las banderas para comprobar que ya no se ha mandado la señal; en caso negativo, se cambiará el estado a “Verdadero” de la bandera y se enviará la señal de alerta que corresponda.

En caso de que la bandera ya esté establecida o que no exista una alerta en curso, se responderá con un mensaje específico para que no realice ninguna acción.

Interrupciones:

Cuando una entrada lógica cambia, se genera un salto del código principal hacia la función específica que corresponde a la interrupción de la entrada que presentó el cambio de estado (bajo a alto).

Tenemos 4 entradas; una de ellas es un botón específico que provoca que se carguen los datos de fábrica, para luego proceder a un reinicio. Las restantes entradas, es decir el timbre y las dos entradas TTL, provocarán el cambio de la bandera de alerta; esto quiere decir que cualquier dispositivo que compruebe esta bandera dentro de los próximos 60 segundos, comprobará que se ejecutó un cambio en dicha entrada provocando la alerta en el dispositivo. La bandera de alerta cambiará de estado luego del tiempo establecido con la finalidad de que si se provoca la misma alerta pueda ser enviada nuevamente, este procedimiento es llevado a cabo por medio de las funciones tickets.

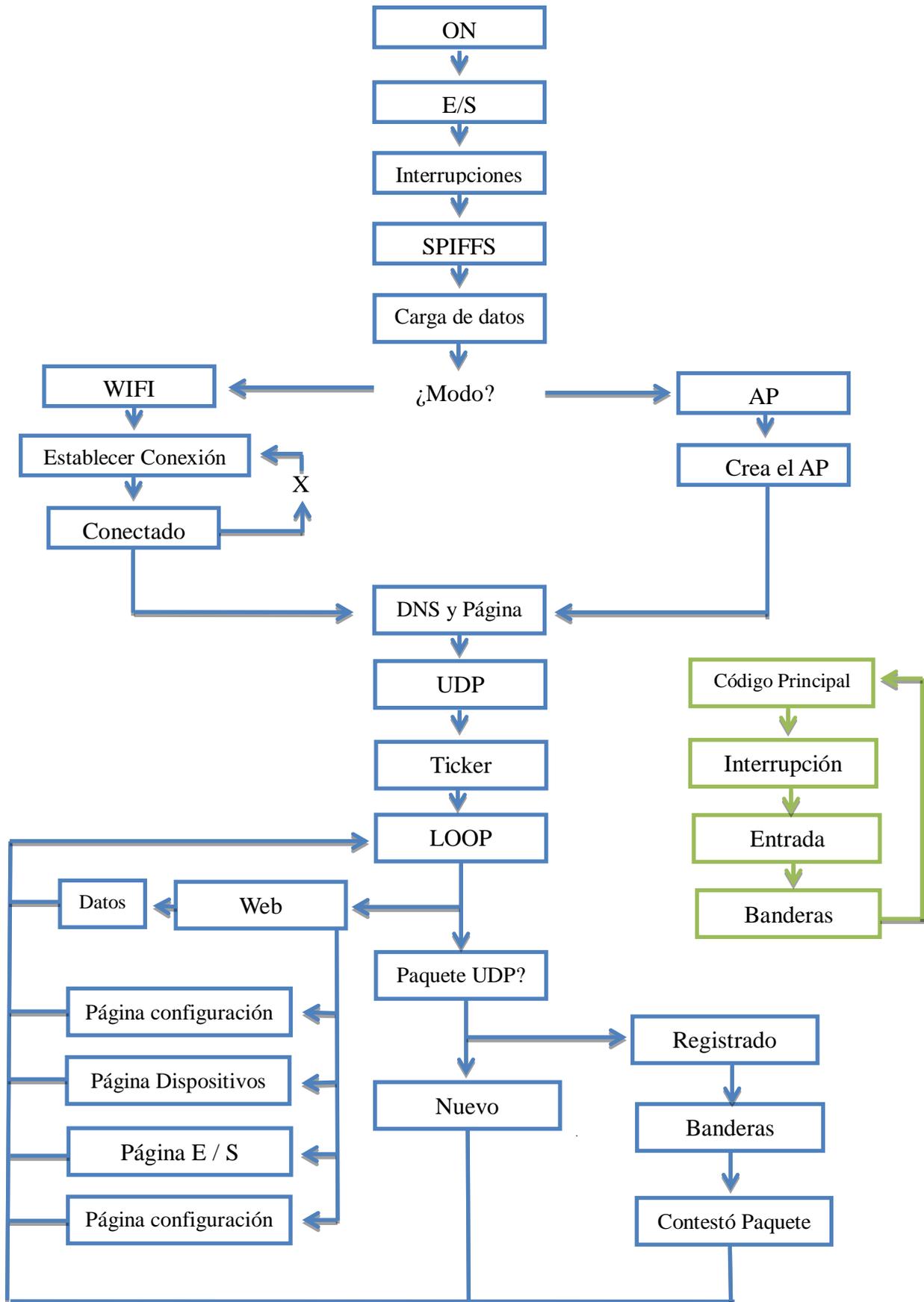
La lista dinámica

Trabajamos con una estructura que incluye el nombre de identificación, el alias, el porcentaje de batería, las banderas y el puntero al siguiente nodo.

Las funciones asociadas a la lista son:

- *Insertar elemento:* guarda una copia del primer puntero para poder asignarlo después al nuevo nodo. Procede a la asignación de un espacio de memoria con el tamaño de la estructura, utilizando la función calloc. Luego inicia los valores de forma predeterminada y asigna el valor guardado al puntero que marca ahora el segundo nodo de la lista.
- *Liberar lista:* Recorre la lista completa liberando toda la memoria asignada
- *Contar elementos:* Recorre la lista para establecer la cantidad de nodos que se encuentran en ella.
- *Buscar elemento:* busca un elemento dentro de la lista y devuelve verdadero o falso según el caso

- *Obtener bandera elementos:* busca un nodo y devuelve los valores de sus banderas.
- *Establecer bandera de elementos:* establece una bandera de un nodo específico cuando el servidor ya le envió el alerta que recibió en una de sus entradas.
- *Cambio de nombre de elementos:* Cuando un usuario cambia el alias de un módulo que ya estaba registrado en el servidor, se procede a la búsqueda de este nodo y al cambio de su alias.
- *Borrar bandera elemento:* borra una bandera para todos los nodos por igual, esto se debe a que ya se cumplió el tiempo preestablecido.
- *Mostrar elementos:* muestra la información principal de cada nodo



2.5. Página Web

Desarrollamos la programación de las páginas web en Java-script mediante un programa que posee la capacidad de transferir hacia los módulos las páginas y archivos necesarios para su funcionamiento por medio del protocolo FTP.

Diseñamos las páginas utilizando como base a una paleta de colores oscuros, a fin de evitar el cansancio en la vista a la hora de ser programada.

La interfaz de la página web es simple, permitiendo cargarla rápidamente. Esto se hizo teniendo en cuenta que la memoria que integra la ESP8266 es limitada, y que 1 [MB] es destinado al código de ejecución y los restantes 3 [MB] son destinados a archivos varios, como los de la página web y el archivo de configuración.

Ambos dispositivos comparten dos páginas: la primera, donde se informa el modo de operación actual (WiFi o AP) y un subconjunto de datos sobre la red actual.



La segunda página es la de configuración. La misma permite al usuario identificar su pulsera por medio de un alias, además de mostrar las redes WiFi disponibles. Esta página permite al usuario conectarse como prefiera, ya sea utilizando su Nombre y Apellido, o simplemente presionando el botón WPS del router.

Si la configuración se está realizando por primera vez o se ha restablecido la configuración a default (puede ser por medio de la página o por su botón dedicado), se deberá acceder por medio de un navegador web a la página ya antes mencionada http://pulsera_XXXXXX/ para configurarla.



Figura 18: Página Web

El servidor posee otras dos pestañas. La primera es la de “Dispositivos” que muestra los dispositivos que se encuentran en la red WiFi y son compatibles con el sistema de aviso. Como se puede observar, cada dispositivo aparece representado por su alias; al seleccionarlo, se accede a su configuración. En el caso de la pulsera, la misma deberá estar enchufada por medio de la entrada mini usb a un cargador o puerto Usb de una Pc o Tv para poder configurarla.

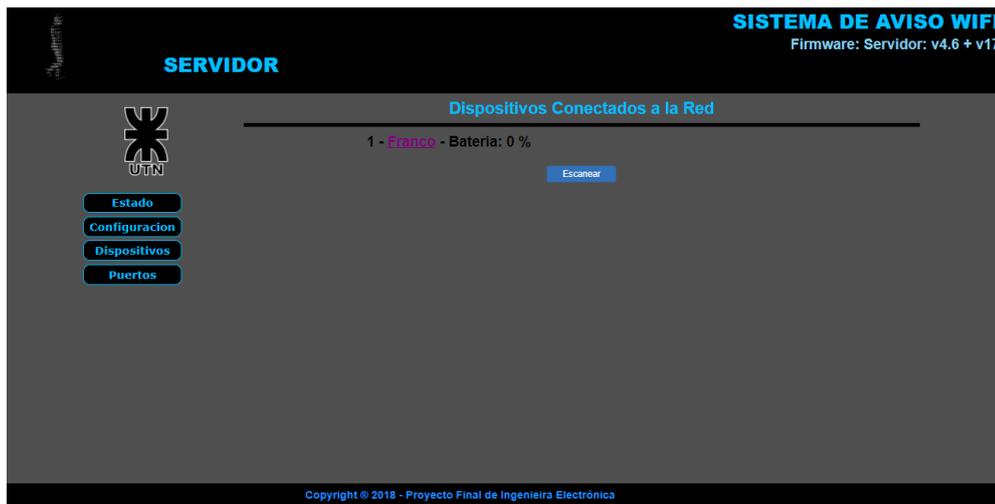


Figura 19: Dispositivos conectados Página Web

La pestaña de Puertos, se refiere a las entradas y salidas que maneja el servidor. A través de ella podemos cambiar los nombres de las entradas y salidas, y realizar las asociaciones entre las mismas. Si una entrada es activada, las salidas asociadas se activarán hasta que la entrada se vuelva a desactivar. Por ejemplo, si el botón del timbre es pulsado y rápidamente soltado; sus salidas asociadas serán activadas en ese lapso de tiempo únicamente. En cambio, si se conecta en la entrada 2 una alarma, la cual presenta una alerta que envía una señal de retención hacia el módulo del servidor y está asociada a la salida 3 donde hay luces conectadas, éstas se mantendrán encendidas mientras dure la señal en la entrada 2.

Cabe destacar que cualquier señal que active la entrada del timbre o cualquiera de las dos extras, mandará una señal de alerta hacia los dispositivos registrados, aunque estos no posean ninguna salida asociada.

SERVIDOR**SISTEMA DE AVISO WIFI**
Firmware: Servidor: v4.6 + v17


Estado
Configuración
Dispositivos
Puertos

Entradas

Nombre	Estado
Temp	<input type="checkbox"/>
E2_nombre	<input type="checkbox"/>
E3_nombre	<input type="checkbox"/>

Salidas

Nombre	Estado
S1_nombre	<input type="checkbox"/>
S2_nombre	<input type="checkbox"/>
S3_nombre	<input type="checkbox"/>

Relaciones

Entrada/Salidas	S1_nombre	S2_nombre	S3_nombre
Temp	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
E2_nombre	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
E3_nombre	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Modo Forzado de Salida

Salidas	S1_nombre	S2_nombre	S3_nombre
Forzar Salida	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Nota:
1) El botón Guardar: Guarda los cambios de las relaciones y los nombres de las entradas y las salidas.
2) El botón Forzar: Cambia la salida independientemente de la asignación de las entradas, o en caso de que una salida no tenga una entrada asociada.

Copyright © 2018 - Proyecto Final de Ingeniería Electrónica

2.6. Aplicación Android:

Aprovechando el sistema de comunicación WiFi de la residencia donde se utilizará nuestro dispositivo, decidimos integrar un sistema de aviso a través de una aplicación (APP) para celulares con sistema operativo Android.

Para obtener una alta compatibilidad con los dispositivos actuales y los más antiguos, utilizamos la API 19 correspondiente a la versión 4.4 kitKat de Android, la cual nos garantiza una compatibilidad con el 90% de los dispositivos en uso actualmente a nivel mundial. La aplicación fue compilada mediante Android Studio y el lenguaje de programación es Java. La programación que realizamos en el servidor y la pulsera se ejecuta de manera secuencial, permitiendo realizar una sola acción por vez; por el contrario, en el caso de la APP, podemos ejecutar códigos que permiten realizar varias acciones al mismo tiempo.

La APP genera una actividad (vista principal) la cual inicia un servicio. En el mismo se crea un hilo de ejecución responsable de capturar los mensajes que el servidor envía por medio del protocolo UDP. El puerto utilizado es el mismo que en el caso de la pulsera y el servidor es 9999. El servicio tiene la particularidad de que si el usuario cierra accidentalmente la APP, la misma se sigue corriendo en segundo plano para continuar recibiendo las alertas que el servidor mande. Es por esto que el hilo de ejecución se crea dentro del servicio.

La actividad principal posee un sistema de aviso donde muestra las entradas y salidas que se encuentran activas mediante una tilde. Para esto, creamos un hilo que se auto destruye en caso de cambiar de actividad, minimizar o finalizar la aplicación.

Para poder mandar información al servidor, como el nombre del dispositivo o los nombres de las salidas y entradas, debemos parar el servicio y pasar a una nueva actividad donde se podrán cambiar los valores antes mencionados. Una vez realizados los cambios se procede a enviarlos, para lo cual se crea un hilo de ejecución que obtiene los datos guardados y los envía al servidor al

puerto 9999. Cuando finaliza el envío, automáticamente regresa a la actividad principal reactivando el servicio de escucha.

Si un usuario quiere parar definitivamente el servicio para no tener alertas, deberá utilizar el botón de parar, el cual detiene el servicio, y la aplicación no se ejecutará en segundo plano.

Cuando se produce una alerta en el servidor, inmediatamente se recibe una alerta en la APP provocando que el celular vibre y genere una notificación en la barra de tareas del sistema, con la leyenda correspondiente a la entrada que se activó.



Figura 20: Aplicación Android

Utilizando la aplicación obtuvimos un tiempo de respuesta menor en la comunicación. Esto es debido al método que se implementó en cada dispositivo (pulsera - aplicación). Mientras la comunicación en la aplicación es constante, la pulsera posee un tiempo preestablecido de “preguntas” al servidor para incrementar su autonomía. Sin embargo, la mayor virtud de la pulsera no está en el tiempo de respuesta sino en su comodidad y facilidad de uso.

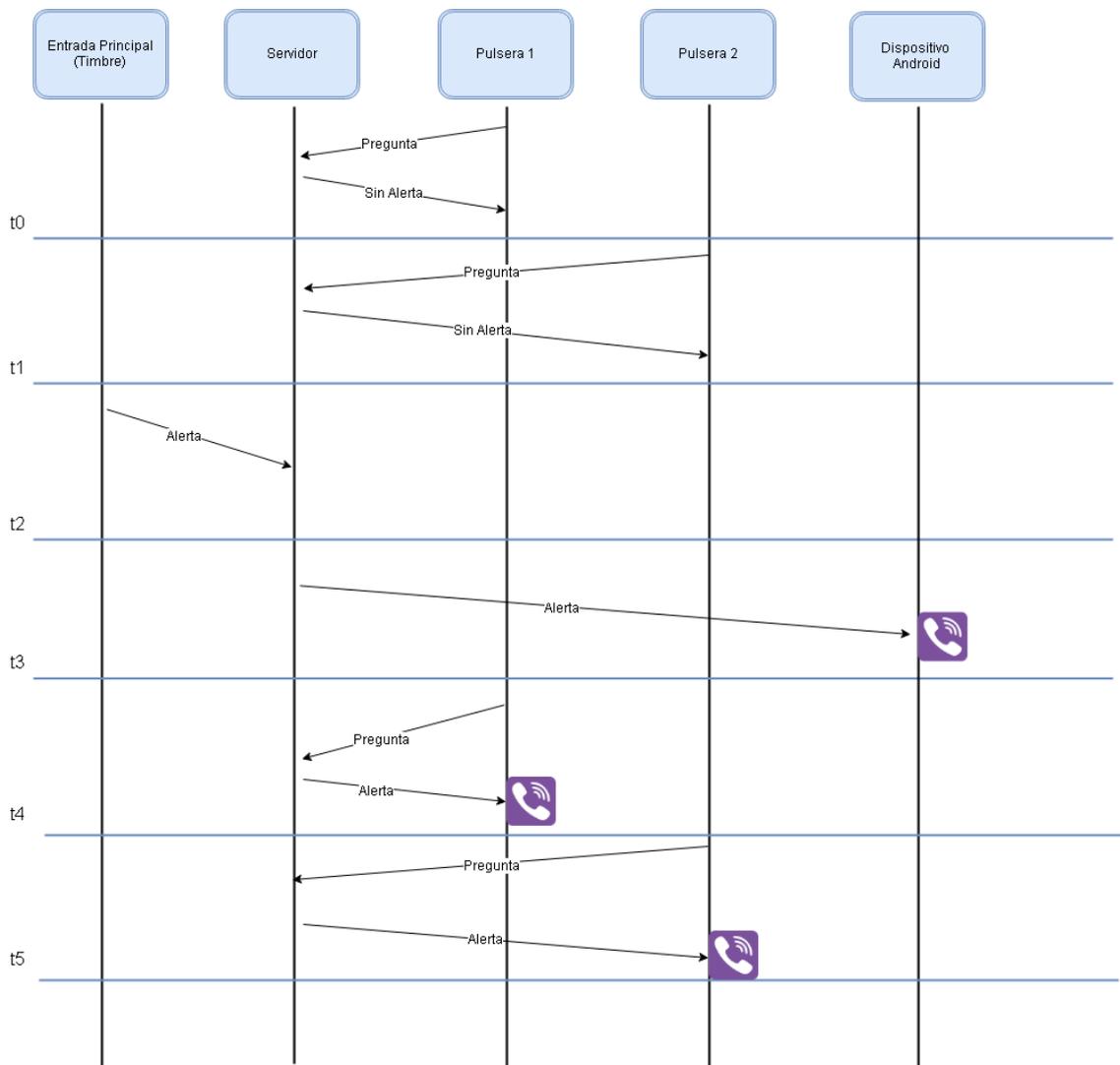


Figura 21: Tiempos de respuesta: Pulsera- Aplicación

2.7. Diseño completo

Finalmente obtuvimos un dispositivo compacto, acorde a las necesidades del usuario y de fácil manipulación. Diseñamos y enviamos a fabricar las placas tanto del servidor como de la pulsera. De esta manera logramos un acabado profesional, evitando errores de fabricación al tratarse de placas de dimensiones acotadas.

Decidimos diseñar el gabinete del servidor y la carcasa de la pulsera para luego materializarlos mediante impresión 3D. Esto significó una gran herramienta para darle a nuestro producto un acabado agradable para el usuario y competitivo en el mercado, cuyas dimensiones debían ser muy específicas.

A continuación se observa el diseño final de ambos dispositivos:

2.7.1. Servidor:

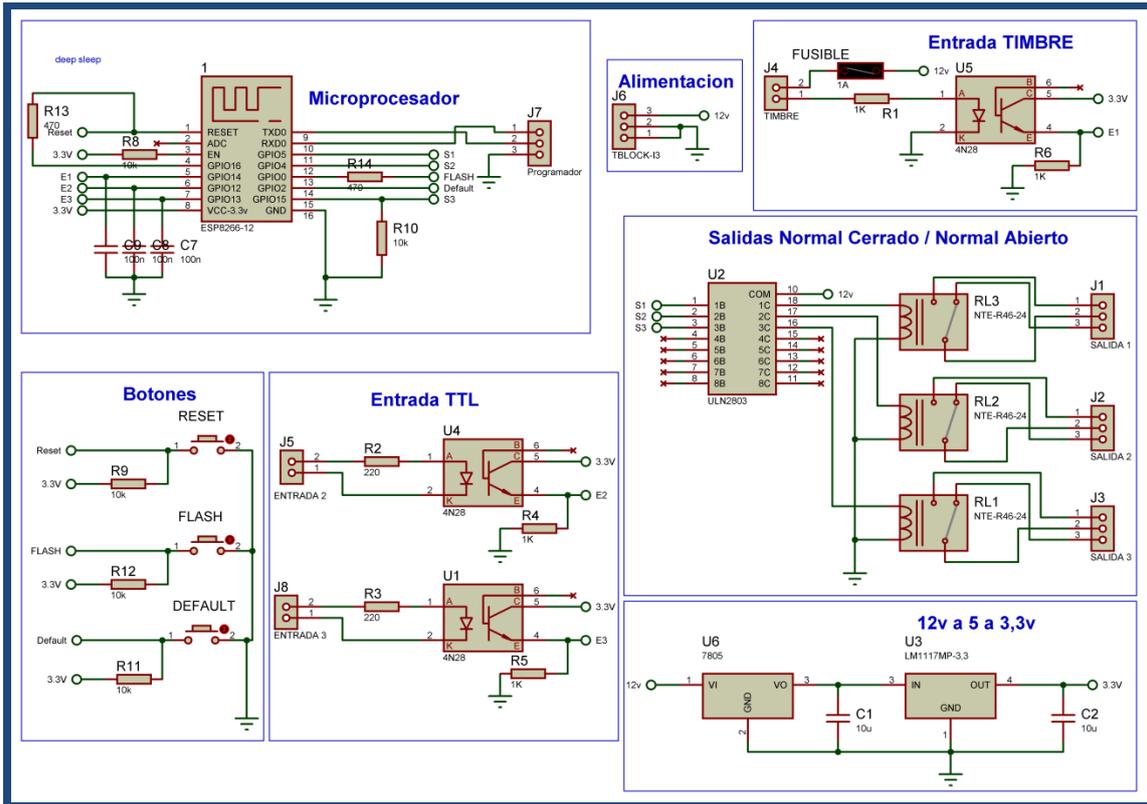
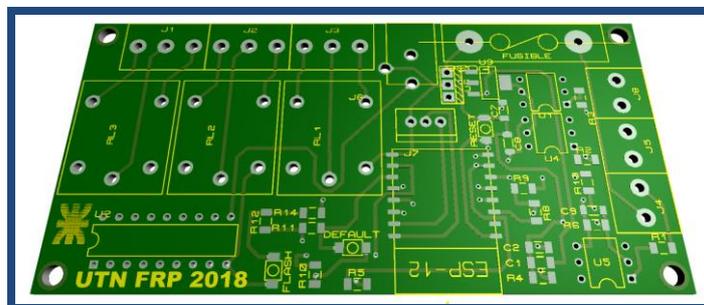


Figura 22: Circuito Completo Servidor



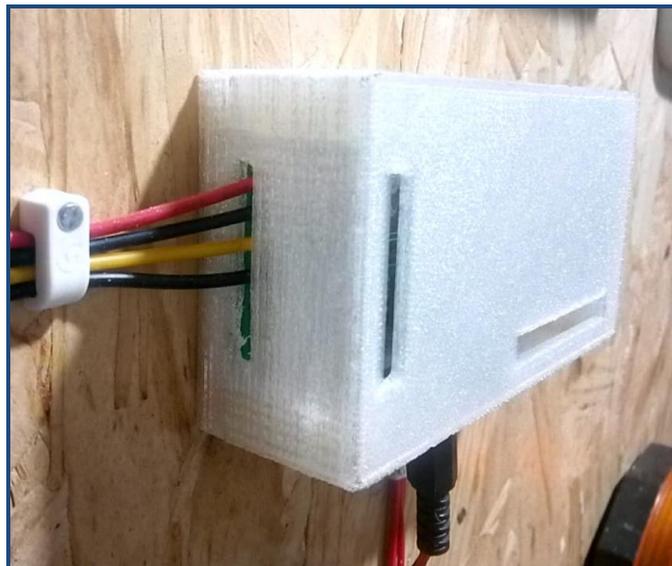
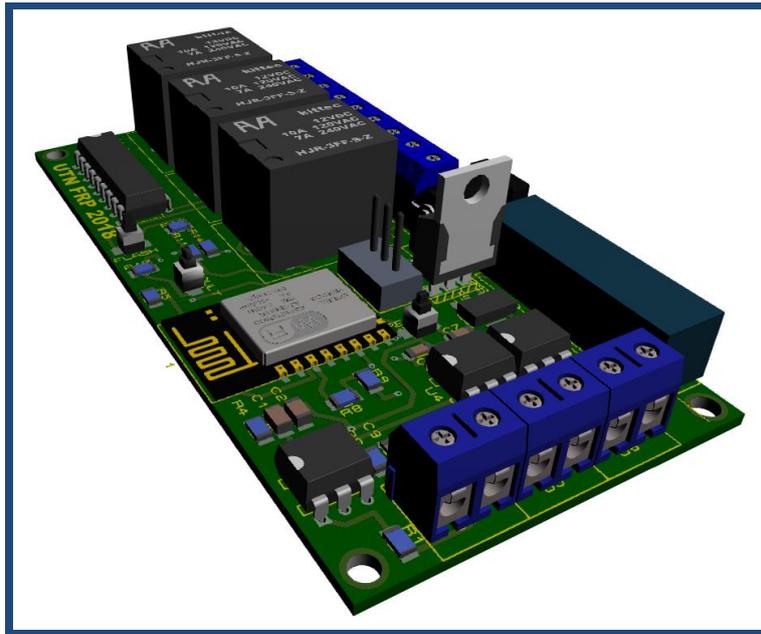


Figura 23: Imágenes Servidor

2.7.2. Pulsera:

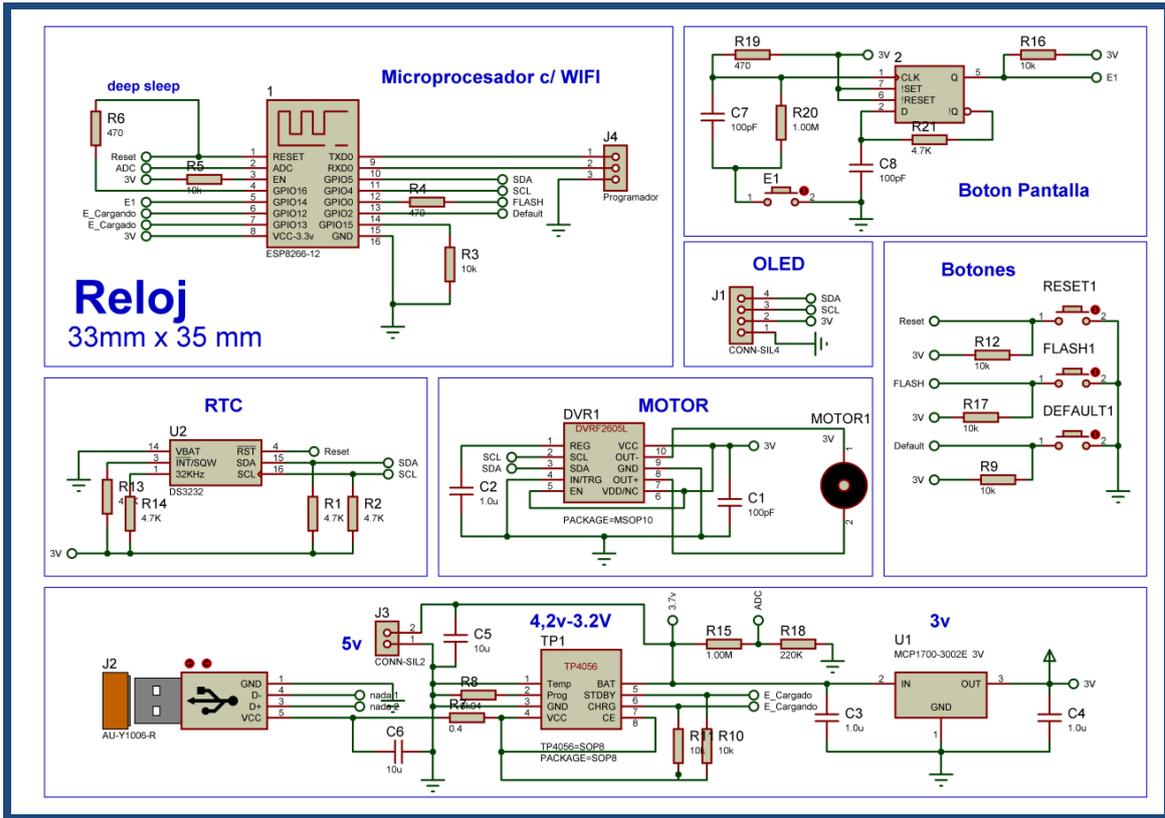
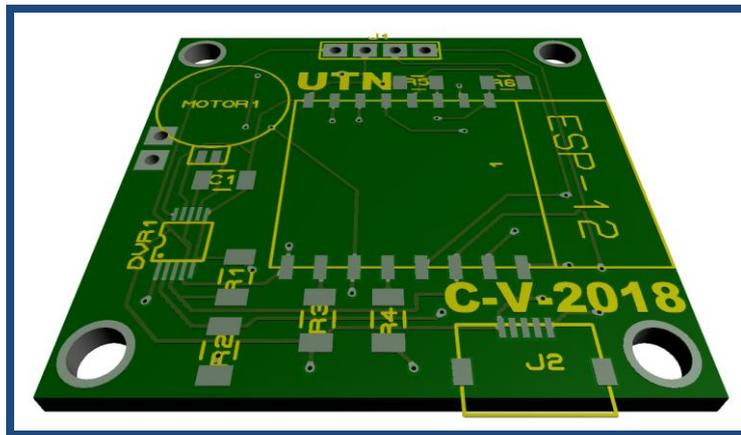


Figura 24: Circuito completo pulsera



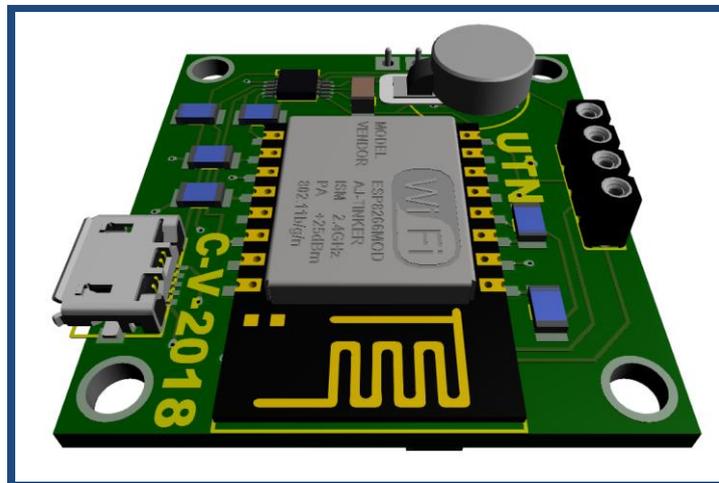
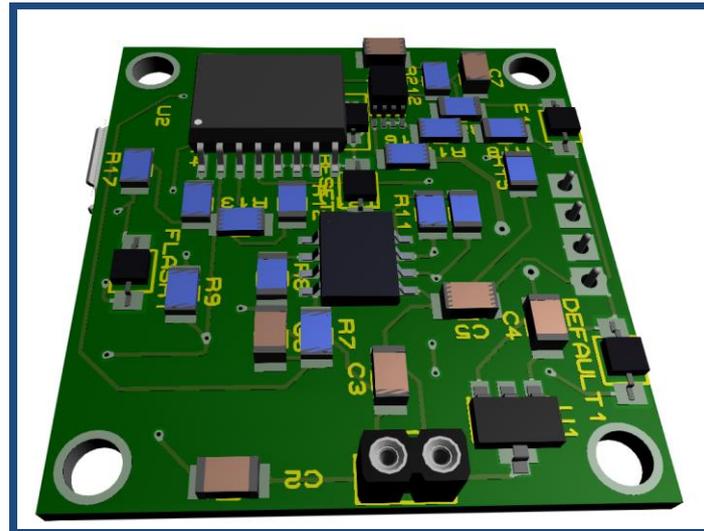


Figura 25: Imágenes Pulsera

Capítulo 3: Resultados

Finalmente obtuvimos un equipo innovador y funcional, capaz de solucionar una problemática poco conocida pero considerable para mucha gente.

La mayor ventaja que presenta nuestro producto en relación a otros de similares características que ya se encontraban en el mercado es la comodidad que posee, ya que no requiere que el usuario esté pendiente del equipo, sino que éste vibrará cuando reciba una señal de alerta.

También incorpora la posibilidad de incluir distintas entradas de alertas diferenciadas además de su entrada principal (timbre), mediante sensores colocados como entradas auxiliares. Así como también la opción de preestablecer distintos tipos de salidas anexadas o no a las señales de entrada.

Mediante la aplicación Android se amplía la incorporación de usuarios al producto. También brinda una alternativa cuando la pulsera deba ser cargada, disponiendo del sistema de alertas en todo momento.

Por último, utilizar la red WiFi del usuario permite mayor confiabilidad, a diferencia de aquellos equipos que se encuentran en el mercado, los cuales se comunican por radio frecuencia.

Capítulo 4: Análisis de Costos

Como vimos anteriormente, ambos dispositivos utilizan como “cerebro” el módulo ESP8266. El mismo se consigue aproximadamente por unos U\$S 7, precio que consideramos bastante accesible.

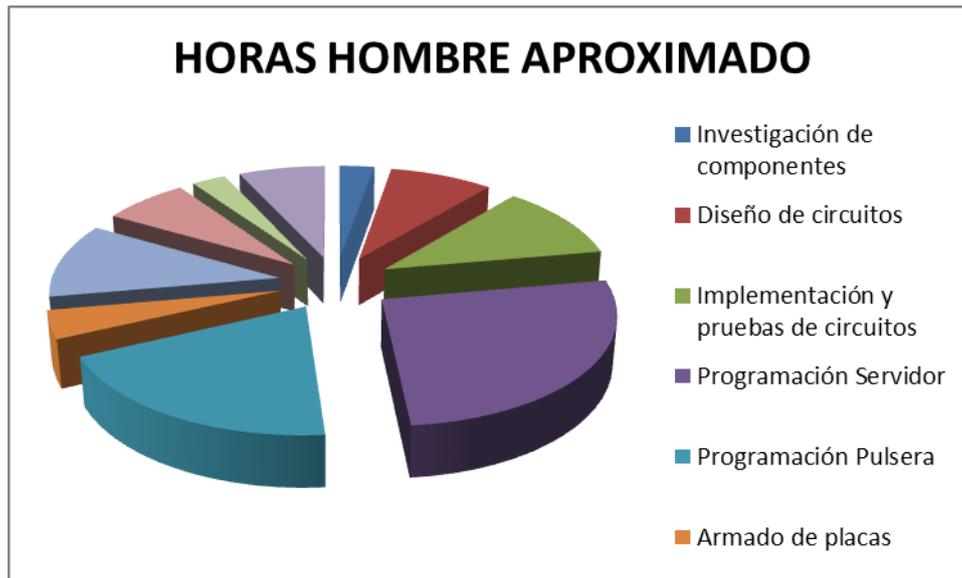
Para el presente trabajo realizamos una compra minorista que cubrió el armado de 2 servidores y 3 pulseras. Sin embargo, se estima que para una fabricación en serie de dicho producto se reduciría hasta un 50 % los costos, lo cual conllevaría a un precio final aún más competitivo.

En cuanto a las horas hombre se estima un trabajo neto de 4 meses, en jornadas de 6 horas. Aproximadamente las horas se distribuyen de la siguiente manera:

ACTIVIDAD	HORAS HOMBRE
Investigación de componentes	20
Diseño de circuitos	60
Implementación y pruebas de circuitos	80
Programación Servidor	190
Programación Pulsera	140
Armado de placas	30
Diseño página web	80
Diseño y programación aplicación Android	50
Diseño de gabinetes y carcasas	20
Otros	50

Tabla 12: Horas hombre de trabajo

Sumando la cantidad de horas hombre, obtenemos un total de 720 horas. Utilizando un promedio de USD 11 la hora, nos da una suma total de USD 7200. Este valor se considera como el precio máximo de horas hombres estipulado.



El costo del kit básico compuesto de un servidor y una pulsera rondaría los 70 dólares, considerando los componentes, las placas y las carcasas. La Aplicación para los sistemas Android no posee costo ni publicidad, por lo que se podría descargar desde todos los dispositivos compatibles que dispongan los usuarios. En el mercado, la competencia directa comercializa un transmisor y dos receptores por 40 a 50 dólares, aunque hay que destacar que este sistema no es ampliable.

Presupuesto estimado:**Bill Of Materials for Proyecto Final Ingenieria Electronica**

Design Title Proyecto Final Ingenieria Electronica
Author Cipriani - Verde
Document Number
Revision 5
Design Created miércoles, 25 de abril de 2018
Design Last Modified miércoles, 16 de mayo de 2018
Total Parts In Design 40

5 Capacitors

<u>Quantity</u>	<u>References</u>	<u>Value</u>	<u>Stock Code</u>	<u>Unit Cost</u>
2	C1-C2	10u	C1206C106M8PACTU	\$0,15
3	C7-C9	100n	CL31B104MBCNNNC	\$0,04
Sub-totals:				\$0,42

13 Resistors

<u>Quantity</u>	<u>References</u>	<u>Value</u>	<u>Stock Code</u>	<u>Unit Cost</u>
4	R1,R4-R6	1K	RC1206JR-071KL	\$0,01
2	R2-R3	220	RC1206JR-07220RL	\$0,01
5	R8-R12	10k	RC1206JR-0710KL	\$0,01
2	R13-R14	470	ERJ-8GEYJ471V	\$0,02
Sub-totals:				\$0,15

6 Integrated Circuits

<u>Quantity</u>	<u>References</u>	<u>Value</u>	<u>Stock Code</u>	<u>Unit Cost</u>
3	U1,U4-U5	4N28	4n28	\$0,54
1	U2	ULN2803	ULN2803A	\$1,08
1	U3	LM1117MP-3,3	LM1117MPX-3.3/NOPB	\$1,36
1	U6	7805	MC7805CTG	\$0,33
Sub-totals:				\$4,40

16 Miscellaneous

<u>Quantity</u>	<u>References</u>	<u>Value</u>	<u>Stock Code</u>	<u>Unit Cost</u>
1	1	ESP8266-12	firstpcb.com	\$1,00
2	DEFAULT,FLASH	Boton	B3U-1000P-B	\$0,95
1	FUSIBLE	1A		\$0,05
1	J1	SALIDA 1		\$0,25
1	J2	SALIDA 2		\$0,25
1	J3	SALIDA 3		\$0,25
1	J4	TIMBRE		\$0,20
1	J5	ENTRADA 2		\$0,20
1	J6	TBLOCK-I3		\$1,50
1	J7	Programador		\$0,00
1	J8	ENTRADA 3		\$0,20
1	RESET	Boton		
3	RL1-RL3	NTE-R46-24	G5LE1A4DC12	\$1,27
Sub-totals:				\$9,62

Totals: **\$14,58**

Bill Of Materials for Proyecto Final Ingenieria Electronica

Design Title Proyecto Final Ingenieria Electronica
Author Cipriani - Verde
Document Number
Revision 6
Design Created martes, 3 de abril de 2018
Design Last Modified miércoles, 16 de mayo de 2018
Total Parts In Design 44

8 Capacitors				
Quantity	References	Value	Stock Code	Unit Cost
3	C1,C7-C8	100pF	CL31C101JBCNNND	\$0,03
3	C2-C4	1.0u	CL31B105KOFNNNE	\$0,06
2	C5-C6	10u	C1206C106M8PACTU	\$0,15
Sub-totals:				\$0,56
21 Resistors				
Quantity	References	Value	Stock Code	Unit Cost
5	R1-R2,R13-R14,R21	4.7K	RC1206JR-074K7L	\$0,01
8	R3,R5,R9-R12,R16-R17	10k	RC1206JR-0710KL	\$0,01
3	R4,R6,R19	470	ERJ-8GEYJ471V	\$0,02
1	R7	0.4	RL1206FR-070R4L	\$0,27
1	R8	6k04	RC1206FR-136K04L	\$0,17
2	R15,R20	1.00M	RC1206JR-071ML	\$0,01
1	R18	220K	RC1206JR-07220KL	\$0,01
Sub-totals:				\$0,66
2 Integrated Circuits				
Quantity	References	Value	Stock Code	Unit Cost
1	U1	MCP1700-3002E 3V	MCP1700T-3002E/MB	\$0,34
1	U2	DS3232		\$3,00
Sub-totals:				\$3,34
13 Miscellaneous				
Quantity	References	Value	Stock Code	Unit Cost
1	1	ESP8266-12	firstpcb.com	\$1,00
1	2	Flip Flop D	74LVC2G74DP,125	\$0,29
2	DEFAULT1,E1	Boton		\$0,17
1	DVR1		DRV2605LDGST	\$3,63
2	FLASH1,RESET1	Boton	B3U-1000P-B	\$0,95
1	J1	Pantalla	OLED 128x64	\$12,00
1	J2	AU-Y1006-R	ZX62R-B-5P(30)	\$0,95
1	J3	CONN-SIL2	Bateria	\$0,01
1	J4	Programador		\$0,00
1	MOTOR1	3V	BMV1003H02	\$3,57
1	TP1	TP4056		\$2,50
Sub-totals:				\$26,18
Totals:				\$30,74

miércoles, 16 de mayo de 2018 11:15:32

Tabla 13: Presupuesto estimado

Estimamos la vida útil del producto en relación a la batería de la pulsera, a la cual consideramos como el componente que sufre el mayor desgaste. Este tipo de baterías posee un tiempo de vida estimado de 4 años, a partir del cual comienzan a retener menos capacidad de carga. De igual manera, consideramos que el usuario podría reemplazarla, prolongando la vida útil del dispositivo.

Capítulo 5: Discusión y Conclusión.

Realizamos el presente proyecto con el fin de mejorar la calidad de vida de las personas hipoacúsicas. El objetivo del mismo fue resuelto mediante la implementación de 3 dispositivos interconectados: un servidor, cuya función principal es recibir alertas y difundirlas a los dispositivos receptores; una pulsera, que le otorga la máxima autonomía al usuario, y un dispositivo Android (teléfono celular) que amplía las posibles configuraciones. La ventaja que posee tener dos dispositivos receptores es que permite seguir obteniendo alertas a través de la pulsera mientras el celular está cargándose, y viceversa. Sin embargo, previmos que la autonomía de la pulsera (más de 18 hs) posibilite realizar la carga cuando el usuario se dispone a descansar.

Para otorgarle mayor practicidad a nuestro dispositivo, elegimos que en cualquier situación el usuario sea alertado mediante vibraciones, dado que son fácilmente perceptibles para las personas hipoacúsicas, aunque se encuentren descansando o realizando alguna actividad que demande atención. Esto constituye una gran ventaja frente a los dispositivos que se encuentran actualmente en el mercado, cuyas alertas son lumínicas o sonoras, éstas últimas además solo son captadas por personas que sufren hipoacusia leve o moderada.

Finalmente se obtuvo un producto que presenta notables mejoras en cuanto a similares en el mercado. Sobre todo por su funcionalidad, permitiendo al usuario adaptar el equipo de acuerdo a lo requerido.

Creemos que en cuanto a su función principal, logramos una comunicación estable y segura mediante la red doméstica WiFi, la cual se encuentra en la gran mayoría de los hogares hoy en día. De igual manera, en el caso de que no posea internet, se consigue por un muy bajo costo y fácil instalación cualquier router WiFi.

También logramos incorporar nuevas tecnologías, como el caso de la pantalla OLED, que ofrece elevados rendimientos y bajo consumo.

De esta manera obtenemos un producto altamente superior en a la competencia que se encuentra en el mercado actualmente. Éstos sólo poseen comunicación mediante radiofrecuencia, la cual presenta una gran probabilidad de interferencias electromagnéticas.

Las mejoras que se podrían implementar en un futuro son:

- Reducir el peso y las dimensiones de la pulsera, buscando un diseño más actual y más amigable al usuario; para lo cual sería imprescindible acoplar la pantalla a la placa principal.
- Incorporar una conexión bluetooth entre la pulsera y los celulares de los usuarios con el fin de aumentar la durabilidad de la batería.
- Tener 2 modelos de pulseras con diferentes tamaños de pantallas.
- Incorporar más salidas y entradas al servidor, para darle más utilidades al sistema.
- Realizar un registro de las alertas que se realizaron.
- Diseñar la pulsera con certificación IP67, que permita al usuario lavarse las manos o bañarse con ella, sin que se produzca una falla en el funcionamiento.
- Integrar un sistema de aviso sonoro en caso de que la pérdida auditiva sea leve

El presente proyecto podría ser modificado para implementarse en otras áreas. Por ejemplo, en aquellas industrias donde los operarios usen protectores auditivos debido a los ruidos de la maquinaria que utilizan. En este caso, se

podría mejorar el tiempo de respuesta ante alguna emergencia que ocurra en el establecimiento.

Capítulo 6: Literatura Citada.

Ceja J., Rentería R., Ruelas R. y Ochoa G. (2017) **“Módulo ESP8266 y sus aplicaciones en el internet de las cosas”** [Revista de Ingeniería Eléctrica Vol.1 No.2, 24-36]

Osorio Londoño D. M., Delgado del Río A. (2016) **“Diseño e Implementación de un Sistema de Comunicación Inalámbrico para automatización y monitoreo”** [Tesis de grado. Facultad de Ingeniería. Universidad Autónoma de México]

“Documentación para ESP8266 Arduino Core. Instrucciones de instalación, funciones y referencia de clases”. [Internet] Disponible desde: <http://esp8266.github.io/Arduino/versions/2.3.0/> [Acceso 15 de abril de 2018]

“Espressif/ ESP8266 Developer Zone” [Internet] Disponible desde: <https://bbs.espressif.com/> [Acceso 20 de abril de 2018]

“Tareas en segundo plano en Android” [Internet] Disponible desde: <http://www.sgoliver.net/blog/tareas-en-segundo-plano-en-android-i-thread-y-async-task/> [Acceso 2 de abril de 2018]

“Lista de librerías de Arduino” [Internet] Disponible desde: <https://www.arduinolibraries.info/> [Acceso 13 de mayo de 2018]

“ESP8266 (servidor FTP): intercambio de archivos con el sistema de archivos SPIFFS” [Internet] Disponible desde: <https://diyprojects.io/esp8266-ftp-server-spiiffs-file-exchange-rapid-development-web-server/#.WyrSuNJKiIU/> [Acceso 10 de mayo de 2018]

J. Mattsson, B. Meyer **“RTC Módulo de Tiempo”** [Internet] Disponible desde: <https://www.arduinolibraries.info/> [Acceso 3 de mayo de 2018]

“Servicio de producción de PCB de alta calidad” [Internet] Disponible desde: <https://firstpcb.com/> [Acceso 13 de marzo de 2018]

P. Mousny **“Sistema de alarma de luz estroboscópica para personas con discapacidad auditiva”** [Internet] Disponible desde: <http://accessens.com/sites/uploads/2015/10/DocumentacionTLA-Spain-web.pdf/> [Acceso 6 de abril de 2018]

L. del Valle Hernández **“No de MCU Placa de desarrollo”** [Internet] Disponible desde: <https://programarfacil.com/podcast/nodemcu-tutorial-paso-a-paso/> [Acceso 8 de marzo de 2018]

“Eccentric Rotating Mass (ERM) Motor” [Internet] Disponible desde: http://sensorwiki.org/doku.php/actuators/eccentric_rotating_mass_erm_motor/ [Acceso 1 de junio de 2018]

E. J. Carletti **“Comunicación - Bus”** [Internet] Disponible desde: http://robots-argentina.com.ar/Comunicacion_busI2C.htm/ [Acceso 25 de mayo de 2018]

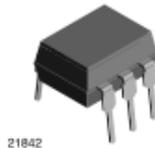
Capítulo 7: Anexos

4N25, 4N26, 4N27, 4N28

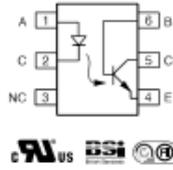
Vishay Semiconductors



Optocoupler, Phototransistor Output, with Base Connection



21842



FEATURES

- Isolation test voltage 5000 V_{RMS}
- Interfaces with common logic families
- Input-output coupling capacitance < 0,5 pF
- Industry standard dual-in-line 6 pin package
- Compliant to RoHS directive 2002/95/EC and in accordance to WEEE 2002/96/EC



RoHS COMPLIANT

APPLICATIONS

- AC mains detection
- Reed relay driving
- Switch mode power supply feedback
- Telephone ring detection
- Logic ground isolation
- Logic coupling with high frequency noise rejection

AGENCY APPROVALS

- UL1577, file no. E52744
- BSI: EN 60065:2002, EN 60950:2000
- FMKO: EN 60950, EN 60065, EN 60335

DESCRIPTION

The 4N25 family is an industry standard single channel phototransistor coupler. This family includes the 4N25, 4N26, 4N27, 4N28. Each optocoupler consists of gallium arsenide infrared LED and a silicon NPN phototransistor.

ORDER INFORMATION	
PART	REMARKS
4N25	CTR > 20 %, DIP-6
4N26	CTR > 20 %, DIP-6
4N27	CTR > 10 %, DIP-6
4N28	CTR > 10 %, DIP-6

ABSOLUTE MAXIMUM RATINGS ⁽¹⁾				
PARAMETER	TEST CONDITION	SYMBOL	VALUE	UNIT
INPUT				
Reverse voltage		V _R	5	V
Forward current		I _F	60	mA
Surge current	t ≤ 10 μs	I _{SM}	3	A
Power dissipation		P _{DISS}	100	mW
OUTPUT				
Collector emitter breakdown voltage		V _{CEO}	70	V
Emitter base breakdown voltage		V _{EBD}	7	V
Collector current		I _C	50	mA
	t ≤ 1 ms	I _C	100	mA
Power dissipation		P _{DISS}	150	mW



DRV2605L

SLOS854C –MAY 2014–REVISED SEPTEMBER 2014

DRV2605L 2 to 5.2 V Haptic Driver for LRA and ERM With Effect Library and Smart-Loop Architecture

1 Features

- Flexible Haptic and Vibration Driver
 - LRA (Linear Resonance Actuator)
 - ERM (Eccentric Rotating Mass)
- I²C-Controlled Digital Playback Engine
 - Waveform Sequencer and Trigger
 - Real-Time Playback Mode through I²C
 - I²C Dual-Mode Drive (Open and Closed Loop)
- Smart-Loop Architecture⁽¹⁾
 - Automatic Overdrive and Braking
 - Automatic Resonance Tracking and Reporting (LRA Only)
 - Automatic Actuator Diagnostic
 - Automatic Level Calibration
 - Wide Support for Actuator Models
- Licensed Immersion TouchSense[®] 2200 features:
 - Integrated Immersion Effect Library
 - Audio-to-Vibe
- Drive Compensation Over Battery Discharge
- Wide Voltage Operation (2 V to 5.2 V)
- Efficient Differential Switching Output Drive
- PWM Input With 0% to 100% Duty-Cycle Control Range
- Hardware Trigger Input
- Fast Start-up Time
- 1.8 V Compatible, V_{DD}-Tolerant Digital Interface

⁽¹⁾ Patent pending control algorithm

2 Applications

- Mobile Phones and Tablets
- Watches and Wearable Technology
- Remote Controls, Mice, and Peripheral Devices
- Touch-Enabled Devices
- Industrial Human-Machine Interfaces
- Electronic Point of Sale (ePOS)

3 Description

The DRV2605L device is a low-voltage haptic driver which includes a haptic-effect library and provides a closed-loop actuator-control system for high-quality haptic feedback for ERM and LRA. This schema helps improve actuator performance in terms of acceleration consistency, start time, and brake time and is accessible through a shared I²C compatible bus or PWM input signal.

The DRV2605L device offers a licensed version of TouchSense 2200 software from Immersion which eliminates the need to design haptic waveforms because the software includes over 100 licensed effects (6 ERM libraries and 1 LRA library) and audio-to-vibe features.

Additionally, the real-time playback mode allows the host processor to bypass the library playback engine and play waveforms directly from the host through I²C.

The smart-loop architecture inside the DRV2605L device allows simple auto-resonant drive for the LRA as well as feedback-optimized ERM drive allowing for automatic overdrive and braking. This architecture creates a simplified input waveform interface as well as reliable motor control and consistent motor performance. The DRV2605L device also features automatic transition to an open-loop system in the event that an LRA actuator is not generating a valid back-EMF voltage. When the LRA generates a valid back-EMF voltage, the DRV2605L device automatically synchronizes with the LRA. The DRV2605L also allows for open-loop driving through the use of internally-generated PWM. Additionally, the audio-to-vibe mode automatically converts an audio input signal to meaningful tactile effects.

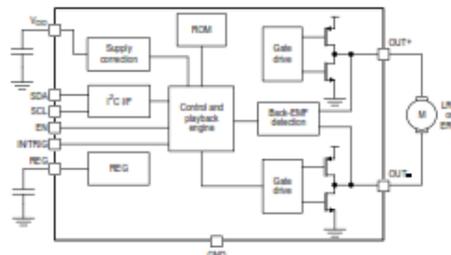
For an important notice regarding Immersion software, see the [Legal Notice](#) section.

Device Information⁽¹⁾

PART NUMBER	PACKAGE	BODY SIZE (MAX)
DRV2605L	DSBGA (9)	1.50 mm × 1.50 mm
DRV2605L	VSSOP (10)	3.00 mm × 3.00 mm

⁽¹⁾ For all available packages, see the orderable addendum at the end of the datasheet.

Simplified Schematic



An IMPORTANT NOTICE at the end of this data sheet addresses availability, warranty, changes, use in safety-critical applications, intellectual property matters and other important disclaimers. PRODUCTION DATA.

DS3231

Extremely Accurate I²C-Integrated RTC/TCXO/Crystal

General Description

The DS3231 is a low-cost, extremely accurate I²C real-time clock (RTC) with an integrated temperature-compensated crystal oscillator (TCXO) and crystal. The device incorporates a battery input, and maintains accurate timekeeping when main power to the device is interrupted. The integration of the crystal resonator enhances the long-term accuracy of the device as well as reduces the piece-part count in a manufacturing line. The DS3231 is available in commercial and industrial temperature ranges, and is offered in a 16-pin, 300-mil SO package.

The RTC maintains seconds, minutes, hours, day, date, month, and year information. The date at the end of the month is automatically adjusted for months with fewer than 31 days, including corrections for leap year. The clock operates in either the 24-hour or 12-hour format with an AM/PM indicator. Two programmable time-of-day alarms and a programmable square-wave output are provided. Address and data are transferred serially through an I²C bidirectional bus.

A precision temperature-compensated voltage reference and comparator circuit monitors the status of V_{CC} to detect power failures, to provide a reset output, and to automatically switch to the backup supply when necessary. Additionally, the RST pin is monitored as a pushbutton input for generating a µP reset.

Benefits and Features

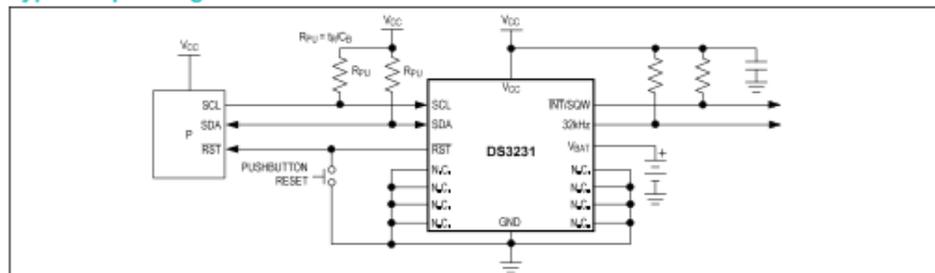
- **Highly Accurate RTC Completely Manages All Timekeeping Functions**
 - Real-Time Clock Counts Seconds, Minutes, Hours, Date of the Month, Month, Day of the Week, and Year, with Leap-Year Compensation Valid Up to 2100
 - Accuracy ±2ppm from 0°C to +40°C
 - Accuracy ±3.5ppm from -40°C to +85°C
 - Digital Temp Sensor Output: ±3°C Accuracy
 - Register for Aging Trim
 - RST Output/Pushbutton Reset Debounce Input
 - Two Time-of-Day Alarms
 - Programmable Square-Wave Output Signal
- **Simple Serial Interface Connects to Most Microcontrollers**
 - Fast (400kHz) I²C Interface
- **Battery-Backup Input for Continuous Timekeeping**
 - Low Power Operation Extends Battery-Backup Run Time
 - 3.3V Operation
- **Operating Temperature Ranges: Commercial (0°C to +70°C) and Industrial (-40°C to +85°C)**
- Underwriters Laboratories® (UL) Recognized

Applications

- Servers
- Telematics
- Utility Power Meters
- GPS

Ordering Information and Pin Configuration appear at end of data sheet.

Typical Operating Circuit



Underwriters Laboratories is a registered certification mark of Underwriters Laboratories Inc.



1.

Overview

Espressif's ESP8266EX delivers highly integrated Wi-Fi SoC solution to meet users' continuous demands for efficient power usage, compact design and reliable performance in the Internet of Things industry.

With the complete and self-contained Wi-Fi networking capabilities, ESP8266EX can perform either as a standalone application or as the slave to a host MCU. When ESP8266EX hosts the application, it promptly boots up from the flash. The integrated high-speed cache helps to increase the system performance and optimize the system memory. Also, ESP8266EX can be applied to any micro-controller design as a Wi-Fi adaptor through SPI / SDIO or I2C / UART interfaces.

ESP8266EX integrates antenna switches, RF balun, power amplifier, low noise receive amplifier, filters and power management modules. The compact design minimizes the PCB size and requires minimal external circuitries.

Besides the Wi-Fi functionalities, ESP8266EX also integrates an enhanced version of Tensilica's L106 Diamond series 32-bit processor and on-chip SRAM. It can be interfaced with external sensors and other devices through the GPIOs. Software Development Kit (SDK) provides sample codes for various applications.

Espressif Systems' Smart Connectivity Platform (ESCP) enables sophisticated features including fast switch between sleep and wakeup mode for energy-efficient purpose, adaptive radio biasing for low-power operation, advance signal processing, spur cancellation and radio co-existence mechanisms for common cellular, Bluetooth, DDR, LVDS, LCD interference mitigation.

1.1. Wi-Fi Protocols

- 802.11 b/g/n/e/i support.
- Wi-Fi Direct (P2P) support.
- P2P Discovery, P2P GO (Group Owner) mode, GC(Group Client) mode and P2P Power Management.
- Infrastructure BSS Station mode / P2P mode / SoftAP mode support.
- Hardware accelerators for CCMP (CBC-MAC, counter mode), TKIP (MIC, RC4), WAPI (SMS4), WEP (RC4), CRC.
- WPA/WPA2 PSK, and WPS driver.
- Additional 802.11i security features such as pre-authentication, and TSN.
- Open interface for various upper layer authentication schemes over EAP such as TLS, PEAP, LEAP, SIM, AKA, or customer specific.
- 802.11n support (2.4 GHz).
- Supports MIMO 1x1 and 2x1, STBC, A-MPDU and A-MSDU frame aggregation and 0.4µs guard interval.



- WMM power low U-APSD,
- Multiple queue management to fully utilize traffic prioritization defined by 802.11e standard,
- UMA compliant and certified,
- 802.11h/RFC1042 frame encapsulation,
- Scattered DMA for optimal CPU off load on Zero Copy data transfer operations,
- Antenna diversity and selection (software managed hardware),
- Clock/power gating combined with 802.11-compliant power management dynamically adapted to current connection condition providing minimal power consumption,
- Adaptive rate fallback algorithm sets the optimum transmission rate and Tx power based on actual SNR and packet loss information,
- Automatic retransmission and response on MAC to avoid packet discarding on slow host environment,
- Seamless roaming support,
- Configurable packet traffic arbitration (PTA) with dedicated slave processor based design provides flexible and exact timing Bluetooth co-existence support for a wide range of Bluetooth Chip vendors,
- Dual and single antenna Bluetooth co-existence support with optional simultaneous receive (Wi-Fi/Bluetooth) capability,



1.2. Main Technical Specifications

Table 1-1. Main Technical Specifications

Categories	Items	Parameters
Wi-Fi	Standards	802.11 b/g/n/ac
	Protocols	802.11 b/g/n/ac
	Frequency Range	2.4G ~ 2.5G (2400M ~ 2483.5M)
	Tx Power	802.11 b: +20 dBm
		802.11 g: +17 dBm
		802.11 n: +14 dBm
	Rx Sensitivity	802.11 b: -91 dbm (11 Mbps)
802.11 g: -75 dbm (54 Mbps) 802.11 n: -72 dbm (MCS7)		
Antenna	PCB Trace, External SMA Connector, Ceramic Chip	
Hardware	CPU	Tensilica L106 32-bit micro controller
	Peripheral Interface	UART/SPI/I2C/FSMC/RTC/Remote Control
		GPIO/ADC/PWM/LED Light & Button
	Operating Voltage	2.5V ~ 3.6V
	Operating Current	Average value: 80 mA
	Operating Temperature Range	-40°C ~ 125°C
	Storage Temperature Range	-40°C ~ 125°C
	Package Size	QFN32-pin (5 mm x 5 mm)
External Interface	-	
Software	Wi-Fi Mode	Station/SoftAP/SoftAP+Station
	Security	WPA/WPA2
	Encryption	WEP/TKIP/AES
	Firmware Upgrade	UART Download / OTA (via network)
	Software Development	Supports Cloud Server Development / Firmware and SDK for fast on-chip programming
	Network Protocols	IPv4, TCP/UDP/HTTP/FTP
	User Configuration	AT Instruction Set, Cloud Server, Android/iOS App



SBOS193D – MARCH 2001 – REVISED JANUARY 2006

High-Side, Bidirectional CURRENT SHUNT MONITOR

FEATURES

- COMPLETE BIDIRECTIONAL CURRENT MEASUREMENT CIRCUIT
- WIDE SUPPLY RANGE: 2.7V to 40V
- SUPPLY-INDEPENDENT COMMON-MODE VOLTAGE: 2.7V TO 60V
- RESISTOR PROGRAMMABLE GAIN SET
- LOW QUIESCENT CURRENT: 75µA (typ)
- MSOP-8 PACKAGE

APPLICATIONS

- CURRENT SHUNT MEASUREMENT:
Automotive, Telephone, Computers, Power Systems, Test, General Instrumentation
- PORTABLE AND BATTERY-BACKUP SYSTEMS
- BATTERY CHARGERS
- POWER MANAGEMENT
- CELL PHONES

DESCRIPTION

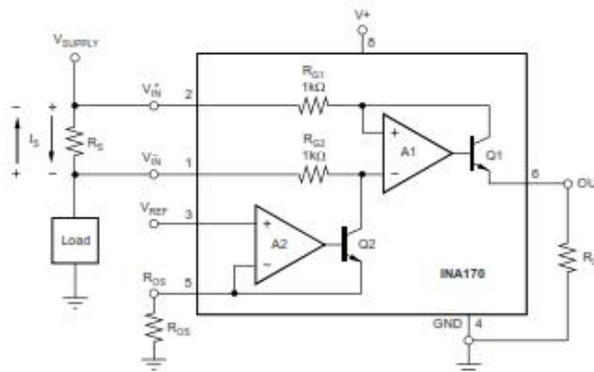
The INA170 is a high-side, bidirectional current shunt monitor featuring a wide input common-mode voltage range, low quiescent current, and a tiny MSOP-8 package.

Bidirectional current measurement is accomplished by output offsetting. The offset voltage level is set with an external resistor and voltage reference. This permits measurement of a bidirectional shunt current while using a single supply for the INA170.

Input common-mode and power-supply voltages are independent. Input voltage can range from +2.7V to +60V on any supply voltage from +2.7V to +40V. Low 10µA input bias current adds minimal error to the shunt current.

The INA170 converts a differential input voltage to a current output. This current develops a voltage across an external load resistor, setting any gain from 1 to over 100.

The INA170 is available in an MSOP-8 package, and is specified over the extended industrial temperature range, -40°C to +85°C with operation from -55°C to +125°C.



Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

All trademarks are the property of their respective owners.

PRODUCTION DATA information is current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.



Copyright © 2001-2006, Texas Instruments Incorporated



LM1117

SNOS412N – FEBRUARY 2000 – REVISED JANUARY 2016

LM1117 800-mA Low-Dropout Linear Regulator

1 Features

- Available in 1.8 V, 2.5 V, 3.3 V, 5 V, and Adjustable Versions
- Space-Saving SOT-223 and WSON Packages
- Current Limiting and Thermal Protection
- Output Current 800 mA
- Line Regulation 0.2% (Maximum)
- Load Regulation 0.4% (Maximum)
- Temperature Range
 - LM1117: 0°C to 125°C
 - LM1117I: -40°C to 125°C

2 Applications

- Post Regulator for Switching DC-DC Converter
- High Efficiency Linear Regulators
- Battery Chargers
- Portable Instrumentation
- Active SCSI Termination Regulator

3 Description

The LM1117 is a low dropout voltage regulator with a dropout of 1.2 V at 800 mA of load current.

The LM1117 is available in an adjustable version, which can set the output voltage from 1.25 to 13.8 V with only two external resistors. In addition, it is available in five fixed voltages, 1.8 V, 2.5 V, 3.3 V, and 5 V.

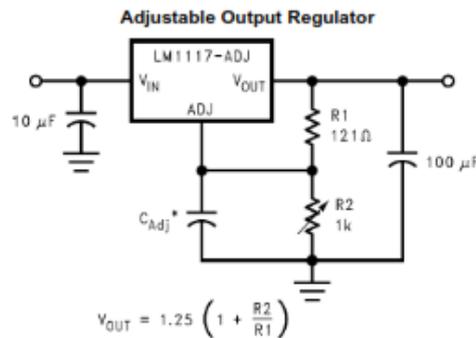
The LM1117 offers current limiting and thermal shutdown. Its circuit includes a Zener trimmed bandgap reference to assure output voltage accuracy to within ±1%.

A minimum of 10-μF tantalum capacitor is required at the output to improve the transient response and stability.

Device Information⁽¹⁾

PART NUMBER	PACKAGE	BODY SIZE (NOM)
LM1117, LM1117I	SOT-223 (4)	6.50 mm × 3.50 mm
	TO-220 (3)	14.966 mm × 10.16 mm
	TO-252 (3)	6.56 mm × 6.10 mm
	WSON (8)	4.00 mm × 4.00 mm
	TO-263 (3)	10.16 mm × 6.41 mm

(1) For all available packages, see the orderable addendum at the end of the data sheet.



* C_{Adj} is optional, however it will improve ripple rejection.

An IMPORTANT NOTICE at the end of this data sheet addresses availability, warranty, changes, use in safety-critical applications, intellectual property matters and other important disclaimers. PRODUCTION DATA.

	SPECIFICATION Vibration Motor		
	TOTAL PAGE	07	http://www.bestargroups.com

Customer		Model Name	BMV1003H02
Customer P/N		Product No.	210813
Date	16 May. 2016	Issue No.	BS/TET01.031A
Page	01 of 07	Issue Date	2016/05/16

Approval:

- 1.Characteristics
- 2.Dimension
- 3.Reliability Test
- 4.Packing
- 5.Notice
- 6.History change record

Drawn by	Checked by	Approved by	Customer approved
常 珩	陶红仲	张秀琴	

BeStar Holding Co., Ltd.

No.199 HuangHe West Road,New district,ChangZhou,JiangSu Province,P.R.China

Tel: +86 519 88222567 **Fax:** +86 519 88222551

E-mail: wu@be-star.com <http://www.bestargroups.com>

文件号: BS/QDTE045B



	6	5	4	3	2																																																	
H	<h2 style="margin: 0;">BMV1003H02</h2>					H																																																
G	<h3 style="margin: 0;">1.Characteristics</h3>					G																																																
F	<h4 style="margin: 0;">1.1 Operating Conditions</h4>					F																																																
E						E																																																
D	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 5%;"></th> <th style="width: 45%;">Parameter</th> <th style="width: 50%;">Specifications</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">1.</td> <td>Rated Voltage</td> <td>3VDC</td> </tr> <tr> <td style="text-align: center;">2.</td> <td>Operating Voltage</td> <td>2.7~3.3VDC</td> </tr> <tr> <td style="text-align: center;">3.</td> <td>Rotation</td> <td>CW(clockwise) or CCW (counter clockwise)</td> </tr> <tr> <td style="text-align: center;">4.</td> <td>* Current at 25℃</td> <td>Max. 90mA</td> </tr> <tr> <td style="text-align: center;">5.</td> <td>* Rated Speed</td> <td>Min. 9000 r/min</td> </tr> <tr> <td style="text-align: center;">6.</td> <td>Starting Current</td> <td>Max. 120mA</td> </tr> <tr> <td style="text-align: center;">7.</td> <td>Starting Voltage</td> <td>Max. 2.3VDC</td> </tr> <tr> <td style="text-align: center;">8.</td> <td>Terminal resistance at 25℃</td> <td>31±15%Ω (single posture) 59±15%Ω (compose posture)</td> </tr> <tr> <td style="text-align: center;">9.</td> <td>**Insulation resistance</td> <td>Min.10MΩ</td> </tr> <tr> <td style="text-align: center;">10.</td> <td>Bracket Deflection strength</td> <td>Min. 9.8N</td> </tr> <tr> <td style="text-align: center;">11.</td> <td>Mechanical noise</td> <td>Max. 50dB</td> </tr> <tr> <td style="text-align: center;">12.</td> <td>Operating Temperature</td> <td>-20...+60℃</td> </tr> <tr> <td style="text-align: center;">13.</td> <td>Storage Temperature</td> <td>-30...+70℃</td> </tr> <tr> <td colspan="3" style="padding: 2px;">*Value applying 3VDC</td> </tr> <tr> <td colspan="3" style="padding: 2px;">**At DC 100V between lead wire and case.</td> </tr> </tbody> </table>						Parameter	Specifications	1.	Rated Voltage	3VDC	2.	Operating Voltage	2.7~3.3VDC	3.	Rotation	CW(clockwise) or CCW (counter clockwise)	4.	* Current at 25℃	Max. 90mA	5.	* Rated Speed	Min. 9000 r/min	6.	Starting Current	Max. 120mA	7.	Starting Voltage	Max. 2.3VDC	8.	Terminal resistance at 25℃	31±15%Ω (single posture) 59±15%Ω (compose posture)	9.	**Insulation resistance	Min.10MΩ	10.	Bracket Deflection strength	Min. 9.8N	11.	Mechanical noise	Max. 50dB	12.	Operating Temperature	-20...+60℃	13.	Storage Temperature	-30...+70℃	*Value applying 3VDC			**At DC 100V between lead wire and case.			D
	Parameter	Specifications																																																				
1.	Rated Voltage	3VDC																																																				
2.	Operating Voltage	2.7~3.3VDC																																																				
3.	Rotation	CW(clockwise) or CCW (counter clockwise)																																																				
4.	* Current at 25℃	Max. 90mA																																																				
5.	* Rated Speed	Min. 9000 r/min																																																				
6.	Starting Current	Max. 120mA																																																				
7.	Starting Voltage	Max. 2.3VDC																																																				
8.	Terminal resistance at 25℃	31±15%Ω (single posture) 59±15%Ω (compose posture)																																																				
9.	**Insulation resistance	Min.10MΩ																																																				
10.	Bracket Deflection strength	Min. 9.8N																																																				
11.	Mechanical noise	Max. 50dB																																																				
12.	Operating Temperature	-20...+60℃																																																				
13.	Storage Temperature	-30...+70℃																																																				
*Value applying 3VDC																																																						
**At DC 100V between lead wire and case.																																																						
C	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%;"></td> <td style="width: 15%;"></td> <td style="width: 15%;"></td> <td style="width: 15%;"></td> <td style="width: 15%;">Date:</td> <td style="width: 15%;">2016/05/16</td> <td rowspan="3" style="text-align: center; vertical-align: middle;"> BMV1003H02 Vibration Motor </td> </tr> <tr> <td></td> <td></td> <td></td> <td></td> <td>Drawn by:</td> <td>管 琦</td> </tr> <tr> <td style="text-align: center;">A</td> <td style="text-align: center;">2016/05/16</td> <td style="text-align: center;">管琦</td> <td></td> <td>Checked by:</td> <td>陶红华</td> </tr> <tr> <td style="text-align: center;">Rev</td> <td style="text-align: center;">Date</td> <td style="text-align: center;">Drawn</td> <td style="text-align: center;">Note</td> <td>Approved by:</td> <td>张秀琴</td> <td></td> </tr> </table>									Date:	2016/05/16	BMV1003H02 Vibration Motor					Drawn by:	管 琦	A	2016/05/16	管琦		Checked by:	陶红华	Rev	Date	Drawn	Note	Approved by:	张秀琴		C																						
				Date:	2016/05/16	BMV1003H02 Vibration Motor																																																
				Drawn by:	管 琦																																																	
A	2016/05/16	管琦		Checked by:	陶红华																																																	
Rev	Date	Drawn	Note	Approved by:	张秀琴																																																	
B	BeStar Holding Co., Ltd.					B																																																
A	www.bestargroups.com ws@be-star.com					A																																																
6	5	4	3	2	1	A																																																

文件号: BS/QDTE045B

The PDF and information therein are the property of BeStar Holding Co., Ltd. and shall not be used in whole or in part without its written consent.

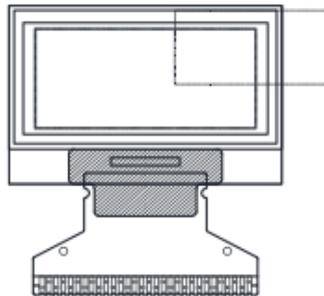


www.vishay.com

OLED-128O064D-BPP3N00000

Vishay

128 x 64 Graphic OLED



FEATURES

- Type: graphic
- Display format: 128 x 64 dots
- Built-in controller: SSD1306BZ
- Duty cycle: 1/64
- +3 V power supply
- Interface: 6800, 8080, serial, and I²C
- Material categorization: for definitions of compliance please see www.vishay.com/doc?99912



RoHS COMPLIANT

MECHANICAL DATA		
ITEM	STANDARD VALUE	UNIT
Module dimension	26,7 x 19,26 x 1,65	mm
Viewing area	23,938 x 12,058	
Active area	21,738 x 10,858	
Dot size	0,148 x 0,148	
Dot pitch	0,17 x 0,17	
Mounting hole	n/a	

ABSOLUTE MAXIMUM RATINGS				
ITEM	SYMBOL	STANDARD VALUE		UNIT
		MIN.	MAX.	
Supply voltage for logic ⁽¹⁾	V _{DD}	0	4	V
Supply voltage for display ⁽¹⁾⁽²⁾	V _{CC}	0	15	
Operating temperature	T _{OP}	-40	+80	°C
Storage temperature	T _{STG}	-40	+80	

Notes

- ⁽¹⁾ All the above voltages are on the basis of "V_{SS} = 0 V".
- ⁽²⁾ When this module is used beyond the above absolute maximum ratings, permanent breakage of the module may occur. Also, for normal operations, it is desirable to use this module under the conditions according to section 6 "Electrical Characteristics". If this module is used beyond these conditions, malfunctioning of the module can occur and the reliability of the module may deteriorate.

ELECTRICAL CHARACTERISTICS						
ITEM	SYMBOL	CONDITION	STANDARD VALUE			UNIT
			MIN.	TYP.	MAX.	
Supply voltage for logic	V _{DD}	-	2,8	3,0	3,3	V
Supply voltage for display	V _{CC}	-	10	12	15	
Input high voltage	V _{IH}	-	0,8 V _{DD}	-	V _{DD(OH)}	
Input low voltage	V _{IL}	-	0	-	0,2 V _{CC}	
Output high voltage	V _{OH}	-	0,9 V _{DD}	-	V _{DD(OH)}	
Output low voltage	V _{OL}	-	0	-	0,1 V _{CC}	
50 % check board operating current	I _{CC}	V _{CC} = 12 V	9	10	12	mA

OPTIONS				
EMITTING COLOR				
YELLOW	GREEN	RED	BLUE	WHITE
-	-	-	Yes	-

Revision: 14-Dec-16

1

Document Number: 37902

For technical questions, contact: displays@vishay.com

THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE, THE PRODUCTS DESCRIBED HEREIN AND THIS DOCUMENT ARE SUBJECT TO SPECIFIC DISCLAIMERS, SET FORTH AT www.vishay.com/doc?91000



www.vishay.com

OLED-128O064D-BPP3N00000

Vishay

INTERFACE PIN FUNCTION							
PIN NO.	SYMBOL	FUNCTION					
1	NC (GND)	Reserved pin (supporting pin) The supporting pins can reduce the influences from stresses on the function pins. These pins must be connected to external ground.					
2	C2N	Positive terminal of the flying inverting capacitor negative terminal of the flying boost capacitor. The charge-pump capacitors are required between the terminals. They must be floated when the converter is not used.					
3	C2P						
4	C1P						
5	C1N						
6	V _{BAT}	Power supply for DC/DC converter circuit This is the power supply pin for the internal buffer of the DC/DC voltage converter. It must be connected to external source when the converter is used. It should be connected to V _{DD} when the converter is not used.					
7	NC	NC					
8	V _{SS}	Ground of logic circuit This is a ground pin. It also acts as a reference for the logic pins. It must be connected to external ground.					
9	V _{DD}	Power supply for logic circuit. This is a voltage supply pin. It must be connected to external source.					
10	BS0	Communicating protocol select These pins are MCU interface selection input. See the following table:					
11	BS1		FC	3-wire SPI	4-wire SPI	8-bit 68XX parallel	8-bit 80XX parallel
		BS0	0	1	0	0	0
		BS1	1	0	0	0	1
12	BS2						
		BS2	0	0	0	1	1
13	CS#	Chip select This pin is the chip select input. The chip is enabled for MCU communication only when CS# is pulled low.					
14	RES#	Power reset for controller and driver This pin is reset signal input. When the pin is low, initialization of the chip is executed.					
15	D/C#	Data / command control This pin is data / command control pin. When the pin is pulled high, the input at D7 to D0 is treated as display data. When the pin is pulled low, the input at D7 to D0 will be transferred to the command register. For detail relationship to MCU interface signals, please refer to the timing characteristics diagrams. When the pin is pulled high and serial interface mode is selected, the data at SDIN is treated as data. When it is pulled low, the data at SDIN will be transferred to the command register. In FC mode, this pin acts as SA0 for slave address selection.					
16	R/W#	Read / write select or write This pin is MCU interface input. When interfacing to a 68XX-series microprocessor, this pin will be used as read / write (R/W) selection input. Pull this pin to "high" for read mode and pull it to "low" for write mode. When 80XX interface mode is selected, this pin will be the write (WR#) input. Data write operation is initiated when this pin is pulled low and the CS# is pulled low.					
17	E/RD#	Read / write enable or read This pin is MCU interface input. When interfacing to a 68XX-series microprocessor, this pin will be used as the enable (E) signal. Read / write operation is initiated when this pin is pulled high and the CS# is pulled low. When connecting to an 80XX-microprocessor, this pin receives the read (RD#) signal. Data read operation is initiated when this pin is pulled low and CS# is pulled low.					
18 to 25	D0 to D7	Host data input / output bus These pins are 8-bit bi-directional data bus to be connected to the microprocessor's data bus. When serial mode is selected, D1 will be the serial data input SDIN and D0 will be the serial clock input SCLK. When FC mode is selected, D2 and D1 should be tied together and serve as SDA _{OUT} and SDA _{IN} in application and D0 is the serial clock input SCL.					
26	I _{REF}	Current reference for brightness adjustment This pin is segment current reference pin. A resistor should be connected between this pin and V _{SS} . Set the current lower than 12.5 μA.					
27	V _{COMH}	Voltage output high level for COM signal This pin is the input pin for the voltage output high level for COM signals. A capacitor should be connected between this pin and V _{SS} .					
28	V _{CC}	Power supply for OEL panel This is the most positive voltage supply pin of the chip. A stabilization capacitor should be connected between this pin and V _{SS} when the converter is used. It must be connected to external source when the converter is not used.					
29	V _{LSS}	Ground of analog circuit This is an analog ground pin. It should be connected to V _{SS} externally.					
30	NC (GND)	Reserved pin (supporting pin) The supporting pins can reduce the influences from stresses on the function pins. These pins must be connected to external ground.					

Revision: 14-Dec-16

2

Document Number: 37902

For technical questions, contact: dispt_hvs@vishay.com

THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE. THE PRODUCTS DESCRIBED HEREIN AND THIS DOCUMENT ARE SUBJECT TO SPECIFIC DISCLAIMERS, SET FORTH AT www.vishay.com/doc?91000

G5LE

PCB Power Relay

Cubic, Single-pole 10A Power Relay

- Ideal for a wide variety of applications such as home appliances, OA equipments, vending machines, etc.
- Ambient Operating Temperature 85°C
- UL class-B coil insulation for standard model.
- UL, CSA, EN standards approved and conforms to Electrical Appliance and Material Safety Law (300 V max.).



RoHS Compliant

Model Number Legend

G5LE-□□□
1 2 3

- Number of Poles**
1: 1-pole
- Contact Form**
None: SPDT (1c)
A: SPST-NO (1a)
- Enclosure rating**
None: Flux protection
4: Fully sealed

Application Examples

- Home appliances
- OA equipments
- Vending machines

Ordering Information

Terminal Shape	Classification	Enclosure rating	Contact form	Flux protection		Fully sealed		Minimum packing unit
				Model	Rated coil voltage	Model	Rated coil voltage	
PCB terminals	Standard		SPDT (1c)	G5LE-1	5 VDC	G5LE-1A	5 VDC	100 pcs/ tray
					12 VDC		12 VDC	
					24 VDC		24 VDC	
					5 VDC		5 VDC	
			SPST-NO (1a)	G5LE-1A	12 VDC	G5LE-1A4	12 VDC	
					24 VDC		24 VDC	

Note. When ordering, add the rated coil voltage to the model number.
Example: G5LE-1 0C5

Rated coil voltage

However, the notation of the coil voltage on the product case as well as on the packing will be marked as □□VDC.

Ratings

Coil

Rated voltage	Rated current (mA)	Coil resistance (Ω)	Must operate voltage (V)	Must release voltage (V)	Max. voltage (V)	Power consumption (mW)
			% of rated voltage			
5 VDC	75.4	63				Approx. 400
12 VDC	33.3	360	75% max.	10% min.	170% at 23°C	
24 VDC	16.7	1,440				

Note 1. The rated current and coil resistance are measured at a coil temperature of 23°C with a tolerance of ±10%.

2. The operating characteristics are measured at a coil temperature of 23°C.

3. The "Max. voltage" is the maximum voltage that can be applied to the relay coil.

Contacts

Item	Load	Resistive load		Inductive load (cosφ = 0.4)	
		Single Ag-alloy (Cd free)			
Contact type					
Contact material					
Rated load		10 A at 120 VAC, 8 A at 30 VDC	5 A at 120 VAC, 4 A at 30 VDC		
Rated carry current		10 A			
Max. switching voltage		250 VAC, 125 VDC (30 VDC when UL/CSA standard is applied)			
Max. switching current		10 A	5 A		

Characteristics

Contact resistance *1	100 mΩ max.
Operate time	10 ms max.
Release time	5 ms max.
Insulation resistance *2	100 MΩ min.
Dielectric strength	Between coil and contacts 2,000 VAC, 50/60 Hz for 1 min
	Between contacts of the same polarity 750 VAC, 50/60 Hz for 1 min
Impulse withstand voltage	between coil and contacts 4,500 V (12-50 μs)
Vibration resistance	Destruction 10 to 55 to 10 Hz, 0.75 mm single amplitude (1.5 mm double amplitude)
	Malfunction 10 to 55 to 10 Hz, 0.75 mm single amplitude (1.5 mm double amplitude)
Shock resistance	Destruction 1,000 m/s ²
	Malfunction 100 m/s ²
Durability	Mechanical 10,000,000 operations/min. (at 18,000 operations/hr)
	Electrical 100,000 operations/min. (at 1,800 operations/hr)
Failure rate (F level) (reference value) *3	100 mA at 5 VDC
Ambient operating temperature	•25°C to 85°C (with no icing or condensation)
Ambient operating humidity	35% to 85%
Weight	Approx. 12 g

Note. The data given above are initial values.

*1. Measurement conditions: 5 VDC, 1 A, voltage drop method.

*2. Measurement conditions: The insulation resistance was measured with a 500 VDC megohmmeter at the same locations as the dielectric strength was measured.

*3. This value was measured at a switching frequency of 120 operations/min.

G5LE



南京拓微集成电路有限公司
NanJing Top Power ASIC Corp.

TP4056 1A Standalone Linear Li-Ion Battery Charger with Thermal Regulation in SOP-8

DESCRIPTION

The TP4056 is a complete constant-current/constant-voltage linear charger for single cell lithium-ion batteries. Its SOP package and low external component count make the TP4056 ideally suited for portable applications. Furthermore, the TP4056 can work within USB and wall adapter.

No blocking diode is required due to the internal PMOSFET architecture and have prevent to negative Charge Current Circuit. Thermal feedback regulates the charge current to limit the die temperature during high power operation or high ambient temperature. The charge voltage is fixed at 4.2V, and the charge current can be programmed externally with a single resistor. The TP4056 automatically terminates the charge cycle when the charge current drops to 1/10th the programmed value after the final float voltage is reached.

TP4056 Other features include current monitor, under voltage lockout, automatic recharge and two status pin to indicate charge termination and the presence of an input voltage.

FEATURES

- Programmable Charge Current Up to 1000mA
- No MOSFET, Sense Resistor or Blocking Diode Required
- Complete Linear Charger in SOP-8 Package for Single Cell Lithium-Ion Batteries
- Constant-Current/Constant-Voltage
- Charges Single Cell Li-Ion Batteries Directly from USB Port
- Preset 4.2V Charge Voltage with 1.5% Accuracy
- Automatic Recharge
- two Charge Status Output Pins
- C/10 Charge Termination
- 2.9V Trickle Charge Threshold (TP4056)
- Soft-Start Limits Inrush Current
- Available Radiator in 8-Lead SOP Package, the Radiator need connect GND or impending

ABSOLUTE MAXIMUM RATINGS

- Input Supply Voltage(V_{CC}): -0.3V~8V
- TEMP: -0.3V~10V
- CE: -0.3V~10V
- BAT Short-Circuit Duration: Continuous
- BAT Pin Current: 1200mA
- PROG Pin Current: 1200uA
- Maximum Junction Temperature: 145°C
- Operating Ambient Temperature Range: -40°C~85°C
- Lead Temp.(Soldering, 10sec): 260°C

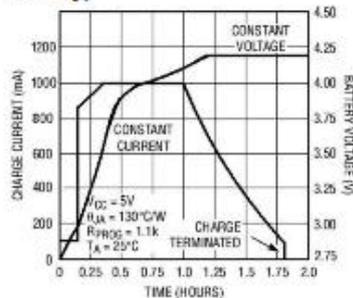
APPLICATIONS

- Cellular Telephones, PDAs, GPS
- Charging Docks and Cradles
- Digital Still Cameras, Portable Devices
- USB Bus-Powered Chargers,Chargers

PACKAGE/ORDER INFORMATION

TEMP1	<p>TP4056 XXXX</p>	8 CE
PROG2		7 CHRG
GND3		6 STDBY
Vcc4		5 BAT
SOP-8		
photo		
ORDER PART NUMBER		
TP4056-42-SOP8-PP		
PART MARKING TP4056		

Complete Charge Cycle (1000mAh Battery)

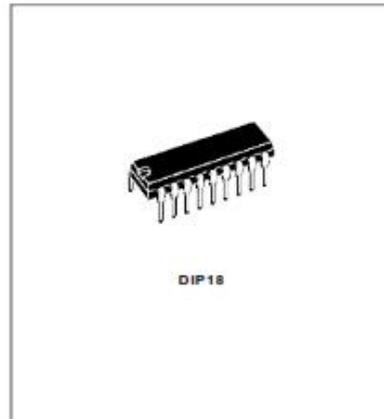




ULN2801A
 ULN2802A - ULN2803A
 ULN2804A - ULN2805A

EIGHT DARLINGTON ARRAYS

- EIGHT DARLINGTONS WITH COMMON EMITTERS
- OUTPUT CURRENT TO 500 mA
- OUTPUT VOLTAGE TO 50 V
- INTEGRAL SUPPRESSION DIODES
- VERSIONS FOR ALL POPULAR LOGIC FAMILIES
- OUTPUT CAN BE PARALLELED
- INPUTS PINNED OPPOSITE OUTPUTS TO SIMPLIFY BOARD LAYOUT



DESCRIPTION

The ULN2801A-ULN2805A each contain eight darlington transistors with common emitters and integral suppression diodes for inductive loads. Each darlington features a peak load current rating of 600mA (500mA continuous) and can withstand at least 50V in the off state. Outputs may be paralleled for higher current capability.

Five versions are available to simplify interfacing to standard logic families: the ULN2801A is designed for general purpose applications with a current limit resistor; the ULN2802A has a 10.5kΩ input resistor and zener for 14-25V PMOS; the ULN2803A has a 2.7kΩ input resistor for 5V TTL and CMOS; the ULN2804A has a 10.5kΩ input resistor for 6-15V CMOS and the ULN2805A is designed to sink a minimum of 350mA for standard and Schottky TTL where higher output current is required.

All types are supplied in a 18-lead plastic DIP with a copper lead from and feature the convenient input-opposite-output pinout to simplify board layout.

PIN CONNECTION (top view)

