

Pulsera inteligente: Watch Ur Health

A. G. Bravo Sánchez¹, R. E. Camas Maay^{1*}, M. A. Carrillo Fierros¹, R. F. Cervantes Rodríguez¹ y M. M. Álvarez Cervera¹

¹ Instituto Tecnológico Nacional de México Campus Mérida, Yucatán, México

* LE17080821@merida.tecnm.mx

Abstract— The COVID-19 disease continues to be a public health problem that society has integrated into its daily life. Prevention measures are a cornerstone to stop the growing increase in infections. Leading to an opportunity for technological development to create devices that contribute to the reinforcement of these measures. The "Smart Bracelet: Watch Ur Health" project is an innovative proposal for a device capable of warning its users from possible approaches of the hand to the face, which we know are a potential risk to get many infectious diseases, not limited only to COVID-19. This paper reports the development of facial approach detection algorithms and the implementation of a first prototype of the Smart Bracelet, integrating the functions of a common watch. Strategies are proposed for the improvement of functions in future developments.

Palabras clave— Contagio, COVID-19, Prevención, Pulsera.

I. INTRODUCCIÓN

En el marco de la pandemia de la enfermedad COVID-19 ocasionada por el virus SARS-CoV2, se han identificado numerosas vías de contagio silentes, las cuales en su mayoría resultan indiferentes para un amplio sector poblacional, pero que pueden resultar mortales. El contacto directo o indirecto con carga viral activa se considera como una de las principales vías de contagio de COVID-19 [1].

Tocarse la cara es uno de los comportamientos que nos diferencian del resto de los animales, inclusive el acto se realiza sin darse cuenta. Este hábito puede ser especialmente problemático durante el brote de una enfermedad, pues ayuda a que se propaguen padecimientos como el provocado por el nuevo coronavirus [2]. Con la finalidad de medir el impacto sanitario de esta acción, se han desarrollado diversos estudios para analizar el riesgo de contagio asociado al comportamiento de tocarse la cara de diversos grupos de personas, como el caso de estudiantes de medicina de la Universidad de Nueva Gales del Sur, que se analizó mediante una grabación de video. Utilizando hojas de puntuación estandarizadas, se registró la frecuencia de los contactos de las manos a la cara con áreas mucosas o no mucosas. Los resultados fueron que, de los 26 estudiantes observados, en promedio se tocaron 23 veces por hora la cara (552 veces al día). El 44% involucraron contacto con una membrana mucosa, mientras que el 56% de contactos involucraron áreas no mucosas. En relación a las veces que tocaban zonas de membrana mucosa, el 36% involucraba la boca, el 31% la nariz, el 27% los ojos, mientras que el 6% era una combinación de estas regiones [3].

El presente proyecto tiene como objetivo advertir al usuario sobre una aproximación facial, potencialmente contagiosa, por parte del brazo portador, mediante un reloj de pulsera inteligente. A diferencia de los que existen en el mercado, esta pulsera con reloj se adapta a personas zurdas y diestras, con diseños atractivos para incentivar su uso como accesorio independientemente de la edad; contando con un diseño ergonómico, hermético y de material adecuado para ser resistente al agua, cómodo en su uso y que puede ser desinfectado con facilidad por medios convencionales.

La idea principal de este dispositivo es la introducción de una vía de prevención que disminuya el riesgo de contagio debido al contacto de las manos con superficies o áreas contaminadas y posterior introducción al organismo.

II. METODOLOGÍA

A. Selección de componentes electrónicos

El dispositivo está compuesto principalmente por sensor de movimiento y posición, microcontrolador, reloj calendario, sistema de energía, pantalla, vibrador, botones e indicadores luminosos.

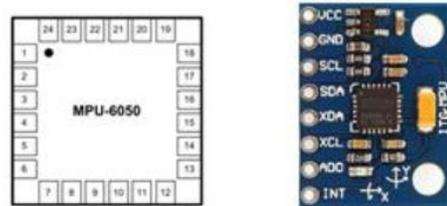


Figura 1. Sensor y módulo MPU-6050.

El sensor que se seleccionó es de movimiento IMU MPU650, sensor microelectromecánico que integra en sí las capacidades de acelerómetro y giroscopio presentado en la figura 1 [4]. Una de las razones principales para escogerlo, fue el tipo de información que brinda en sus lecturas. Los sensores ultrasónicos e infrarrojos se centran en detectar movimiento cuando la emisión de energía es interrumpida. Esto plantea un inconveniente para la finalidad de prevención del proyecto, ya que se pretende informar al usuario antes de concretar la aproximación facial. Utilizando tecnología infrarroja o ultrasónica, la ventana de tiempo para adquirir información de un posible acercamiento facial del sensor, y la respuesta del microcontrolador para avisar al usuario, se vuelve más reducida. Por ejemplo, un proyecto similar al propuesto es el collar "Pulse" de la NASA, éste

utiliza un sensor infrarrojo y tiene la desventaja de alertar falsos acercamientos de la mano a la cara [5].

Uno de los ejes para la elección del microcontrolador fue la compatibilidad para ser programado, y el acceso a múltiples librerías de uso libre, para utilizar los módulos seleccionados como el MPU-6050. Se seleccionó la tarjeta de evaluación Dragonfruit Rev 2, la cual realiza los procesos a través del Microcontrolador (MCU) ATmega32U4. El MCU contiene un bootloader integrado con la capacidad de poder usarse con el IDE, conectándolo al PC, asimismo tiene la capacidad de programar y de cargar bootloader a otros modelos de microcontroladores compatible para realizar tareas específicas como hacer sonar una alarma cuando ocurra un evento [6].

Para satisfacer la funcionalidad de reloj de la pulsera se utilizaron dos botones para ajustar la hora, una pantalla de 0.91" OLED (Organic Light-Emitting Diode) 128x32 SSD1306 y un Módulo RTC (Real Time Clock). Entre otros componentes seleccionados para el proyecto se encuentran los indicadores: 2 leds, una batería recargable y un motor vibrador que es el elemento de alarma de la pulsera.

B. Programación embebida

Una vez seleccionados los componentes, se realiza la programación del software embebido. Como el microcontrolador está contenido en la tarjeta DragonFruit Rev 2, se siguen las indicaciones del fabricante y se utiliza la librería <<MPU6050.h>> para la obtención de datos del sensor de movimiento (acelerómetro y giroscopio incluido). Es importante que al iniciar se calibre el sensor MPU6050 ya que probablemente no se encuentre en una posición horizontal debido a que al ser soldado en el módulo puede estar desnivelado agregando un error en cada componente. Para solucionar este problema, se configura en el módulo MPU6050 OFFSETS y de esta forma se compensan los errores [7].

Los datos que se consideraron útiles para el desarrollo del proyecto son:

- Del acelerómetro
 - Aceleración en el eje x (ax_m_s2).
 - Aceleración en el eje y (ay_m_s2).
 - Aceleración en el eje z (az_m_s2).
 - Ángulo de inclinación con respecto al eje x ($accel_ang_x$).
 - Ángulo de inclinación con respecto al eje y ($accel_ang_y$).
- Del giroscopio
 - Velocidad angular en el eje x (gx_deg_s).
 - Velocidad angular en el eje y (gy_deg_s).
 - Velocidad angular en el eje z (gz_deg_s).

Se desarrolló el código para la obtención de los 8 datos antes listados, se imprimieron en el monitor serie y serial plotter con una velocidad de 57600 baudios. En primera instancia se utilizó como referencia el brazo derecho y se fueron muestreando las posiciones del sensor (ubicado en la

parte posterior de la muñeca derecha) como se muestra en la figura 2.



Figura 2. Posiciones del sensor MPU-6050 en la muñeca derecha para detectar estado en reposo y acercamiento a la cara.

Los datos generados por el sensor en el monitor serie fueron graficados en Excel y para su mejor visualización se agruparon como se muestra en la figura 3 y 4.

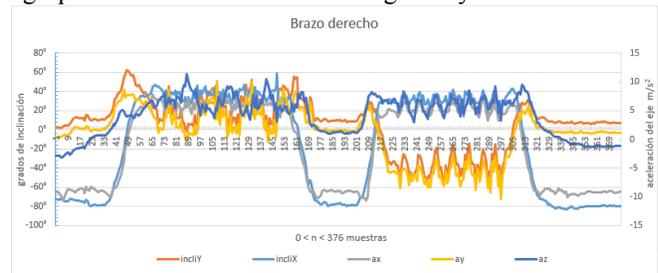


Figura 3. Inclinación en X y Y del sensor, más la aceleración en los 3 direcciones (vectores).

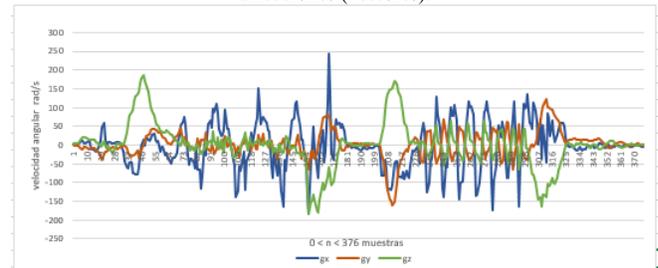


Figura 4. Velocidad angular en los vectores X, Y, Z del sensor.

Este experimento se reprodujo con 2 sujetos de prueba de diferente estatura simulando los mismos movimientos. De los datos obtenidos anteriormente, se tomaron como filtro para establecer rangos de movilidad y los valores que no serían tomados en cuenta, definiendo rangos de valores entre los cuales, la mano se encuentra más cerca de completar un contacto con el rostro. Estos fueron tomados de la oscilación de valores cuando el brazo se encontraba en posiciones relacionadas con el segundo y tercer panel de la figura 2. Dichos rangos para el modo en mano derecha se encuentran listados a continuación:

- $22.5 < accel_ang_x < 62$
- $-5 < accel_ang_y < 70$
- $4.5 < ax_m_s2 < 9$
- $-0.3 < ay_m_s2 < 9.5$
- $3 < az_m_s2$

Con el brazo izquierdo se tomó la posición de la figura 2 “en forma de espejo”, se usó la parte superior de la muñeca izquierda y se recolectaron los 8 datos para analizarlo en Excel.

Nuevamente, los rangos establecidos se basaron en los valores de oscilación en las posiciones más cercanas al rostro y se especifican en el algoritmo para mano izquierda:

- $30 < \text{accel_ang_x} < 75$
- $-80 < \text{accel_ang_y} < 10$
- $5 < \text{ax_m_s2} < 9$
- $\text{ay_m_s2} < 2$
- $2.4 < \text{az_m_s2} < 9$

Teniendo los rangos del sensor donde el MPU6050 detecta que la mano (derecha o izquierda según la información del giroscopio) se está acercando al rostro, se integró la función de “alarma” digital, indicando con una salida en HIGH cuando el movimiento esté dentro de los valores establecidos. De esta forma se activa el motor vibrador, actuando cada vez que la mano portadora de la pulsera se intenta acercar a la cara.

Se agregó un interruptor para controlar en qué mano se utilizará la pulsera, así como un indicador led para señalar si se está utilizando el modo para mano izquierda. De igual forma se anexó un sistema para bloquear la alarma, basado en las mediciones del giroscopio, específicamente de gx_deg_s : cuando la muñeca gira sobre su eje repetidamente más de 15 veces (valor medido para comodidad del usuario mediante pruebas físicas) en un movimiento intuitivo que asemeja a la acción de abrir y cerrar con una llave (figura 5), el microcontrolador envía la instrucción de activar o desactivar la función de alarma. Lo anterior con el fin de que el usuario pueda elegir dinámicamente si quiere recibir o no advertencias en un momento determinado según su entorno desactivar la alarma.



Figura 5. Movimiento para el sistema de bloqueo y desbloqueo.

C. Programación de pantalla OLED y RTC.

Para programar el OLED es necesario usar la librería `<<Adafruit_SSD1306.h>>`, seguida de `<<Wire.h>>` para la comunicación I2C con el Arduino. Siempre que se desee imprimir en la pantalla hay que seleccionarla con `oled.display()`, de otra forma no se envía la información que se quiere mostrar; los comandos principales/básicos a utilizar son para borrar pantalla (`clear()`), seleccionar un pixel (donde se puede empezar a escribir con `setCursor()`), seleccionar el tamaño del texto (`setTextSize()`) y el color (`setTextColor()`), este último solo puede ser blanco porque el módulo OLED seleccionado es monocromo. Con el comando `oled.print(“texto”)` se imprime en la pantalla [8].

Para la programación del RTC (Real Time Clock) se utiliza la librería `<<RtcDS1302.h>>` seguida de `<<ThreeWire.h>>` para la comunicación SPI con el Arduino. Se inicializa el RTC y se adquiere la fecha y hora del instante en el que se carga el programa, en este caso se

obtiene del computador, se establece la variable día y mes obteniéndolas a partir de la información del RTC de la siguiente forma: `HoraFecha = RtcDateTime(__DATE__, __TIME__);` [9]

Una vez definido el funcionamiento del RTC y del OLED, lo siguiente fue unificar ambos en un solo código para mostrar la hora y fecha en la pantalla OLED. Al código del primer prototipo se le agregó la combinación de la aplicación del RTC y OLED; adicionalmente se agregaron funciones específicas en la pantalla OLED, como la barra de batería, cambio de hora-fecha y mensaje de alerta. Se simuló en Proteus y después se implementó en el prototipo.

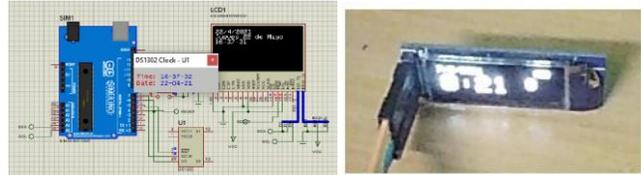


Figura 6. Simulación e implementación del OLED SSD1306 y RTC DS1302. El RTC marca la hora y fecha, el OLED lo imprime en su pantalla.

D. Optimización del código y ahorro de energía.

Para tener bajo consumo de energía se activa la pantalla cuando el usuario lo requiera mediante un movimiento específico (figura 7), manteniendo la lectura de las velocidades angulares con el sensor de movimiento del MPU, si detecta movimiento entonces se aplica la codificación principal del acelerómetro relacionado con la posición de la mano, la activación de la alarma en caso de que la mano esté cerca de la cara.

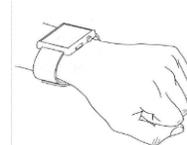


Figura 7. Representación ilustrativa de movimiento para ver la hora y activar el modo despierto del OLED.

E. Implementación del prototipo

Con base en las funciones descritas de cada componente se realizó la integración de la pulsera inteligente “Watch Ur Health”. Se combinan los bloques de código de cada componente electrónico en uno solo, con el propósito de registrar movimientos de la mano, activar la alarma vibratoria, aplicar sistemas de bloqueo de alarma, registrar la hora y mostrarla en la pantalla OLED. En la figura 8 se muestra el diagrama a bloques del sistema.

III. RESULTADOS

El primer producto alcanzado, figura 9, fue el desarrollo de un algoritmo funcional de detección de acercamiento facial, con base en las pruebas realizadas con el sensor de movimiento MPU6050 y la identificación de patrones en señales adquiridas como ángulos de inclinación y aceleraciones en cada eje, logrando eliminar en gran medida los “falsos sondeos” donde la mano del usuario con el sensor

no se esté acercando al rostro. Este algoritmo emite una salida digital para activar la alarma de elección para el dispositivo, un motor vibrador.

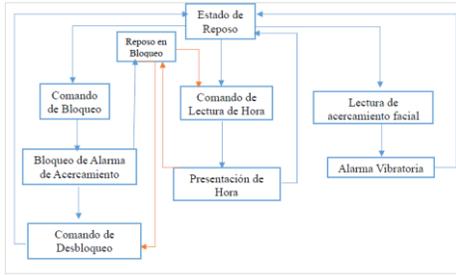


Figura 8. Modelo lógico de funcionamiento general de la pulsera inteligente "Watch Ur Health".

Fue posible adquirir datos sobre la temperatura del usuario de la Pulsera Prototipo gracias a las características del sensor. Otro resultado destacable es la configuración de un RTC DS1302 como reloj en tiempo real, asociado a una pantalla OLED SSD1306 que muestra la fecha, hora, el estado de la batería y la temperatura detectada por el MPU.

Se ensambló un primer prototipo de la Pulsera utilizando como núcleo el controlador ATMEGA 32U4, una batería tipo LiPo de 3.7 V a 110 mAh como fuente de poder para permitir portabilidad. Con pruebas de uso se determinó un tiempo medio de vida de la batería de aproximadamente 3 horas, lo cual no es ideal para este tipo de dispositivos; se pretende extender la duración de la batería hasta 8 horas en un desarrollo futuro. Se probó en 5 personas ambidiestras para cada mano en cada persona en 30 min detectando y alarmando el acercamiento de la pulsera a la cara más del 80% de los movimientos.

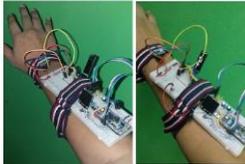


Figura 9. Prototipo de la pulsera inteligente "Watch Ur Health". Apagado (izquierda) y con pantalla encendida (derecha)."

IV. DISCUSIÓN

Si bien este prototipo tiene en cuenta el collar "pulse" de NASA [5] como antecedente (un collar de sensor infrarrojo), posee características adicionales ya que además de anticipar la aproximación hacia el rostro, con una simple alerta, y con un estímulo sensorial, tiene mayor rango de captación y un menor tiempo de respuesta, detecta la temperatura del usuario y le proporciona las características de un reloj de pulsera.

Dado los resultados obtenidos de este primer prototipo, tenemos razones para avanzar a la siguiente etapa de recolección de datos a distintas personas con distintas alturas, mejorando los rangos para cada usuario y buscando la eliminación de fallos.

V. CONCLUSIONES

El diseño del prototipo de pulsera representa una alternativa de fácil acceso e implementación, alertando a su usuario de la aproximación de la mano con el rostro en un rango de aproximadamente 15cm (+/- 5cm) de distancia y a su vez, funcionalidades de reloj y calendario. Durante el desarrollo de este dispositivo se pudo apreciar un buen tiempo de respuesta y la detección de patrones de movimiento y ángulos de inclinación de los movimientos específicos para llevar la mano cerca de la cara.

Las funciones de bloqueo y desbloqueo aumentan el tiempo de uso de la batería y eliminan las alertas innecesarias. A la par, la función de desplegar la pantalla, un nivel adecuado de brillo y contraste, y desactivación tras un breve instante, se presentó como una tarea compleja pero provechosa para el ahorro de la batería. Los retos en el desarrollo del prototipo, minimizan el contacto del usuario con botones, y permiten el acceso a las funciones de la pulsera de forma dinámica y sencilla. Un circuito electrónico en conjunto con la impresión por medio de manufactura aditiva de la carcasa es parte fundamental de la implementación, en la cual el prototipo pasará a una versión más cercana a otras pulseras inteligentes en el mercado.

REFERENCIAS

- [1] Organización Mundial de la Salud. (2020, marzo 29). Vías de transmisión del virus de la COVID-19: repercusiones para las recomendaciones relativas a las precauciones en materia de prevención y control de las infecciones. [En línea]. Disponible en: <https://www.who.int/es/news-room/commentaries/detail/modes-of-transmission-of-virus-causing-covid-19-implications-for-ipc-precaution-recommendations>. [Accedido: 11-febrero-2021]
- [2] E. Duarte. (2020, marzo 10). Coronavirus: por qué es tan difícil dejar de tocarse la cara como recomiendan para evitar el contagio del covid-19. [En línea]. Disponible: <https://www.bbc.com/mundo/noticias-51768647>. [Accedido: 11-febrero-2021]
- [3] Y. L. A. Kwok et al., "Face touching: a frequent habit that has implications for hand hygiene.", *American journal of infection control*, vol. 43, no. 2, pp 112-114, 2015. doi: 10.1016/j.ajic.2014.10.015
- [4] InvenSense Inc. (2013, august 19). MPU-6000 and MPU-6050 Product Specification. [Online]. Available: <https://invensense.tdk.com/wp-content/uploads/2015/02/MPU-6000-Datasheet1.pdf>. [Accesed: 5-marzo-2021]
- [5] La Nación. (2020, julio 1). Así es Pulse, el collar de la NASA que evita que te toques la cara (ideal para esta época). [En línea]. Disponible: <https://www.elpais.com.uy/eme/lifestyle/asi-pulse-collar-nasa-evita-te-toques-cara-ideal-epoca.html>. [Accedido: 5-marzo-2021]
- [6] Atmel Corporation. (2015). ATmega16U4/ATmega32U4. [Online]. Available: https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7766-8-bit-AVR-ATmega16U4-32U4_Datasheet.pdf. [Accesed: Mar 11, 2021]
- [7] Naylamp mechatronics (2016). TUTORIAL MPU6050, ACELERÓMETRO Y GIROSCOPIO. [En línea]. Disponible en: https://naylampmechatronics.com/blog/45_tutorial-mpu6050-acelerometro-y-giroscopio.html. [Accedido: 11-abril-2021]
- [8] V. García. (2018, febrero 3). Pantalla OLED 0.96. [En línea]. Disponible en: <https://www.diarioelectronicohoy.com/blog/pantalla-oled-0-96>. [Accedido 11-abril-2021]
- [9] M. Miller. (2019). Makuna/Rtc. [Online]. Available: <https://github.com/Makuna/Rtc/blob/master/src/RtcDS1302.h>. [Accesed: april 11, 2021]