



# Dispositivo de biotelemedicina con mediciones de temperatura y pulso integrado para aplicaciones multifuncionales

Ian Lee, Angela Julio, Lyneth Lezcano, Anguie Vásquez, Eimy Caballero, Alejandra Villegas, Candy Zhong, Brigiditt Bethancourt, César Abrego



Técnico en Ingeniería con Especialización en Electrónica Biomédica - Facultad de Ingeniería Eléctrica - Universidad Tecnológica de Panamá

**Resumen:** Este paper describe más que nada la importancia del dispositivo de biotelemedicina con mediciones de temperatura y pulso se ha convertido en una herramienta fundamental en el campo de la salud y el bienestar. Su capacidad para obtener mediciones precisas y transmitirlos en tiempo real ha revolucionado la forma en que se monitorizan y se siguen los indicadores fisiológicos clave. Con este dispositivo, se logra una vigilancia continua y un seguimiento más efectivo de la salud, permitiendo una detección temprana de posibles problemas y un cuidado personalizado. En definitiva, este avance tecnológico representa un gran paso hacia un futuro más saludable y conectado.

**Palabras Claves:** Aplicaciones, biotelemedicina, desarrollo, integración, pulso, temperatura.

**Abstract:** This overview primarily emphasizes the significance of the biotelemetry device with temperature and pulse measurements as a fundamental tool in the field of health and well-being. Its ability to obtain precise measurements and transmit them in real-time has revolutionized the way key physiological indicators are monitored and tracked. With this device, continuous surveillance and more effective health monitoring are achieved, enabling early detection of potential issues and personalized care. Ultimately, this technological advancement represents a significant step towards a healthier and more connected future.

**Keywords:** Applications, biotelemetry, development, integration, pulse, temperature

\*Corresponding author: [gabriel.thompson@utp.ac.pa](mailto:gabriel.thompson@utp.ac.pa)



```

1 #include <BluetoothSerial.h>
2 #include <SPI.h>
3 #include <Wire.h>
4 #include <Adafruit_GFX.h>
5 #include <Adafruit_SSD1306.h>
6 #include <Adafruit_MX30014.h>
7 #define WHITE Wire
8
9 //Ritmo cardíaco
10
11
12 float factor = 0.75; // coeficiente para filtro pasa bajos
13 float maximo = 0.0; // para almacenar valor maximo
14 int minimoEntreLetidos = 300; // 300 seg. de tiempo minimo entre letidos
15 float valorAnterior = 500; // para almacenar valor previo
16 int letidos = 0; // contador de cantidad de letidos
17
18
19 float tp, th, countol, bpm;
20 BluetoothSerial SerialBT;
21 Adafruit_MX30014 mls = Adafruit_MX30014();
22 Adafruit_SSD1306 display = Adafruit_SSD1306(128, 32, WHITE);
23
24 void setup() {
25   display.begin(SSD1306_SWITCHCAPVCC, 0x3C); //inicializa led
26   mls.begin(); //Inicia sensor de temperatura
27   Serial.begin(9600); //velocidad de comunicación
28   SerialBT.begin("ESP32"); //Inicia bluetooth y le pone un nombre
29 }
30
31 void loop() {
32   if(countol > 0){
33     display.clearDisplay(); //limpia display
34     //Inicia lecturas
35     tp = mls.readObjectTempC();
36     th = mls.readObjectTempC();
37     //Impresión en display
38     display.setTextSize(1);
39     display.setTextColor(SSD1306_WHITE);
40     display.setCursor(0,0);
41     display.print("Paciente: "); display.println(tp);
42     display.print("Habitación: "); display.println(th);
43     display.print("Pulso: "); display.println(bpm);
44     display.display();
45     //Impresión en bluetooth
46     SerialBT.print("P"); SerialBT.println(tp);
47     SerialBT.print("H"); SerialBT.println(th);
48     SerialBT.print("B"); SerialBT.println(bpm);
49     countol--;
50   }
51   static unsigned long tiempoPM = millis();
52   static unsigned long entreLetidos = millis();
53   int valorLeido = analogRead(34);
54   float valorFiltrado = factor * valorAnterior + (1 - factor) * valorLeido;
55   float cambio = valorFiltrado - valorAnterior;
56   valorAnterior = valorFiltrado;
57   if ((cambio > maximo) || (millis() > entreLetidos + minimoEntreLetidos)) {
58     maximo = cambio;
59     entreLetidos = millis();
60     letidos++;
61   }
62   maximo = maximo * 0.97;
63   if (millis() > tiempoPM + 15000) {
64     bpm = letidos * 60;
65     letidos = 0;
66     tiempoPM = millis();
67   }
68   countol++;
69   Serial.println(analogRead(34));
70   delay(100);
71 }

```

Figure 5. Programación de los sensores

Para comenzar, se importaron las librerías necesarias y se declararon las variables correspondientes a la temperatura del paciente, la temperatura de la habitación y los pulsos por minuto.

A continuación, se procedió a crear los objetos que se encargarían de realizar cada una de las mediciones mencionadas.

Cada medio segundo, se enviaron los valores de la temperatura del paciente, la temperatura de la habitación y los pulsos del paciente a la pantalla LCD y al dispositivo Bluetooth.

Finalmente, se observó en el código una fórmula utilizada para capturar la señal del pulsímetro y convertirla en pulsaciones por minuto (bpm).

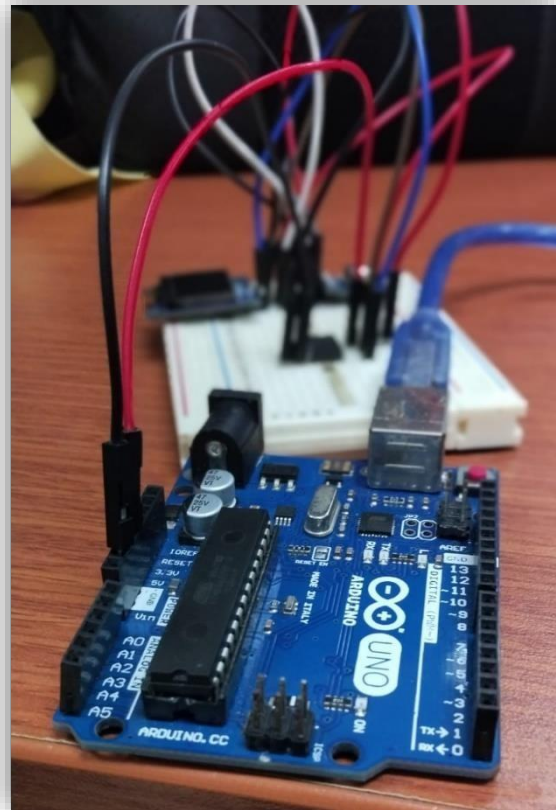


Figura 6. Circuito conectado al Arduino

Por último, se creó una página web mediante Javascript, con el framework de REACT, este framework nos permite crear páginas webs más dinámicas debido al cambio de mediciones que se presentan, en la página web se muestran 3 cuadros como se ve en la figura #7: la temperatura del paciente, el pulso del paciente y la temperatura de la habitación.



Figura 7. Captura de la webpage creada





**Figura 8. Personalización del proyecto**

Utilizamos un software de diseño asistido por ordenador (CAD, por sus siglas en inglés) para crear un modelo tridimensional del proyecto que queremos realizar. Esto nos permite visualizar y especificar todos los detalles del diseño, como las dimensiones, formas y componentes necesarios.

Una vez que hemos finalizado el modelo 3D, utilizamos una impresora 3D para convertir ese diseño virtual en un objeto físico. La impresora 3D crea capas sucesivas de material, generalmente plástico, para construir el objeto de acuerdo con las especificaciones del modelo.

Después de haber soldado los componentes a la PCB, colocamos la placa dentro del "casing" o carcasa que hemos preparado previamente. La carcasa más que nada es un contenedor físico diseñado para proteger y contener todos los componentes internos del proyecto.



**Figura 9. Proyecto en su respectiva carcasa**

Una vez que hemos colocado la placa con los componentes dentro del casing, utilizamos tornillos u otros métodos de fijación para asegurar que todos los elementos estén correctamente posicionados y sujetos. Esto garantiza que no haya movimientos no deseados en el interior del casing y ayuda a mantener la integridad y estabilidad del proyecto.

### ***E. CONFLICTO DE INTERESES***

Se cambia el tamaño del proyecto para así mejorar la potencia del mismo, ya que en un inicio se quería hacer lo mas minimalista posible. Pero por motivos de tener un proyecto más complejo se optó por la mejor opción. Cambiamos un **attiny85** por un **esp32**.

### ***F. Discusión final y Conclusiones***

En este estudio, se ha presentado el diseño y desarrollo de un prototipo de biotelemedicina integrado que combina varios sensores.

Este dispositivo ofrece numerosas aplicaciones y beneficios tanto en entornos clínicos como en la vida cotidiana. Proporciona a los médicos y profesionales de la salud la capacidad de realizar un seguimiento remoto y continuo de los pacientes, permitiendo una atención personalizada y una detección temprana de posibles problemas de salud.

Además, para las personas que desean mantener un estilo de vida saludable y activo, este dispositivo brinda una herramienta invaluable para monitorear la actividad física y evaluar la eficacia del entrenamiento.

La integración de la biotelemedicina en la atención médica y el autocuidado ha abierto un mundo de posibilidades para mejorar la calidad de vida y la prevención de enfermedades. Gracias a este dispositivo, se logra una vigilancia constante y un seguimiento más efectivo de la salud, lo que permite una toma de decisiones más informada y una atención más oportuna.