CIENCIA, INGENIERÍA Y DESARROLLO TEC LERDO
Revista digital universitaria
Año: 2025. Volumen: 1. Numero:11 ISSN: 2448-623X

Optimización de un Prototipo de Respirador AMBU Automatizado: Compactación, Eficiencia Energética e Interfaz Gráfica Mejorada

Romero-Reyes, L.A.[1] ; Alvarado-Tovar, N.[2]; González-Vargas, L.A.[3] ; Serrano-Hernández, A.[4]; Martínez-Sifuentes, O.E.[5]

Datos de Adscripción:

✉[1] Karen Magdalena Romero Reyes. Instituto Tecnológico de la Laguna /TecNM, Departamento de Posgrado, m.kmromeror@lalaguna.tecnm.mx https://orcid.org/0009-0005-6951-7387

[2] Noé Alvarado Tovar. Instituto Tecnológico de la Laguna /TecNM, Departamento de Posgrado, nalvarado@lalaguna.tecnm.mx. https://orcid.org/0000-0002-0218-0813

[3]Luis Amado González Vargas. Instituto Tecnológico Superior de Lerdo/TecNM, Departamento de Posgrado, Luis.gv@lerdo.tecnm.mx. https://orcid.org/0009-0008-9094-6188

[4] Arturo Serrano Hernández. Instituto Tecnológico Superior de Lerdo/TecNM, División de Ingeniería Ambiental, Arturo.sh@itslerdo.edu.mx https://orcid.org/0009-0001-1302-5747

[5] Omar Eruviel Martínez Sifuentes. Instituto Tecnológico Superior de Lerdo/TecNM, División de Ingeniería Ambiental, 202310564@itslerdo.edu.mx

Resumen **- La optimización de equipos médicos es esencial para mejorar su eficiencia, precisión y facilidad de uso, especialmente en contextos donde la asistencia respiratoria automatizada puede marcar una diferencia crítica. Este artículo presenta el rediseño de un respirador basado en una bolsa AMBU, cuyo prototipo original incorporaba un Arduino Mega como unidad de control, una pantalla LCD con encoder rotativo, un sensor de presión con acondicionamiento por divisor de tensión, y un motor a pasos nema 34 con su respectiva etapa de potencia. Aunque funcional, el sistema mostraba limitaciones en capacidad de procesamiento, consumo energético, tamaño, peso y usabilidad. El proceso de reingeniería se enfocó en superar estas limitaciones mediante la integración de una Raspberry Pi 3, que proporcionó mayor capacidad de procesamiento y mejor gestión del control en tiempo real. El motor a pasos fue sustituido por uno más compacto y eficiente, reduciendo el consumo energético sin comprometer el rendimiento. Además, se reemplazó la interfaz LCD y el encoder por una pantalla táctil intuitiva, facilitando la configuración de parámetros y la visualización de datos en tiempo real. Se integró también un conversor analógico-digital (ADC) de alta resolución para mejorar la precisión en la lectura del sensor de presión. Los resultados obtenidos reflejan mejoras significativas en eficiencia energética, capacidad de respuesta, precisión del monitoreo y experiencia de usuario. Estas mejoras posicionan al nuevo prototipo como una alternativa más viable, compacta, eficiente y accesible, con potencial para su implementación en entornos clínicos o de emergencia que requieran asistencia respiratoria confiable y portátil.**

Palabras clave: Reingeniería, asistencia respiratoria, optimización energética, Raspberry Pi e interfaz táctil.

Abstract **- The optimization of medical equipment is crucial to enhance efficiency, precision, and usability, particularly in scenarios where automated respiratory support can be lifesaving. This study presents the redesign of a respirator based on an AMBU bag. The original prototype incorporated an Arduino Mega as the control unit, an LCD display with rotary encoder, a pressure sensor with signal conditioning via a voltage divider, and a NEMA 34 stepper motor with its power stage. While functional, the system exhibited notable limitations in processing capacity, energy consumption, size, weight, and overall usability. To overcome these constraints, the reengineering process integrated a Raspberry Pi 3, providing higher processing power and improved real-time control. The original stepper motor was replaced with a more compact and energy-efficient alternative, reducing power consumption without compromising performance. The LCD and rotary encoder interface were substituted with a touchscreen, enabling intuitive parameter adjustment and real-time data visualization. In addition, a high-resolution analog-to-digital converter (ADC) was incorporated to improve pressure sensor accuracy. The redesigned prototype demonstrated significant improvements in energy efficiency, response time, monitoring precision, and user experience. These advances make the system a more viable, compact, and accessible alternative, with strong potential for deployment in clinical and emergency contexts requiring reliable and portable respiratory support.**

Keywords: Respiratory assistance, energy optimization, raspberry Pi and touch interface.

I. INTRODUCCIÓN

La innovación en dispositivos médicos portátiles es fundamental en contextos donde la asistencia respiratoria inmediata puede significar la diferencia entre la vida y la muerte, especialmente en zonas con recursos limitados o en situaciones de emergencia. La pandemia de COVID-19 evidenció la urgente necesidad de contar con ventiladores mecánicos accesibles, eficientes, fáciles de operar y aptos para su implementación rápida y segura en ambientes clínicos diversos.

En respuesta a esta necesidad, se desarrollaron múltiples prototipos de respiradores automatizados, empleando diferentes arquitecturas, metodologías de control y niveles de integración tecnológica. En México, se diseñó un ventilador de emergencia basado en un resucitador manual (bag-valve mask, BVM), también conocido como bolsa AMBU. Este sistema operaba en modos SIMV (Ventilación Mandatoria Intermitente Sincronizada) y AC (Asistido-Controlado), e integraba dos microcontroladores (Arduino Mega y Nano) dedicados a la operación de ventilación y la adquisición de datos provenientes de sensores de presión, temperatura, flujo y fracción inspirada de oxígeno (FiO_2), además de contar con servomotores para accionar las paletas compresoras deslizantes. El sistema fue validado en simuladores y modelos animales (Medina Tanco, 2020).

https://doi.org/10.62805/cid2025.CID054

This work is licensed under CC BY 4.0. To view a copy of this license, visit https://creativecommons.org/licenses/by/4.0/

A nivel internacional, también se llevaron a cabo desarrollos significativos. El Illinois RapidAlarm, por ejemplo, fue concebido como un sistema de monitoreo de bajo costo para ventiladores neumáticos de emergencia, utilizando sensores de presión y algoritmos para estimar parámetros clínicos (PIP, PEEP, frecuencia respiratoria) y detectar fallos. Su arquitectura se basaba en un microcontrolador ATmega328 y fue validada con pulmones artificiales y cerdos (Corey et al., 2020). Otro desarrollo, el proyecto MIT Emergency-Vent, propuso un sistema de código abierto que automatiza la compresión de bolsas BVM utilizando actuadores simétricos y control PID para minimizar riesgos de barotrauma, validado en estudios porcinos (Ort et al., 2021).

Se han explorado diversas configuraciones mecánicas para lograr una compresión eficiente de la bolsa, incluyendo mecanismos de yugo escocés, tornillos de bolas y sistemas de doble paleta. Estas configuraciones han sido evaluadas junto con estrategias de control como los controladores PI y LQR, cuyos resultados muestran mejoras notables en la estabilidad del sistema y la precisión en la entrega de volumen radial (Morales et al., 2021). Otros estudios como los de Du Pasquier et al. (2021), Robati et al. (2022), Ding et al. (2023), Jain et al. (2023) y Chauhan et al. (2024) han abordado el desarrollo de sistemas portátiles y de bajo costo, destacando la importancia de la automatización, monitoreo preciso y facilidad de uso.

En ese contexto, este trabajo presenta la optimización de un prototipo de respirador automatizado basado en una bolsa AMBU, cuyo diseño original utilizaba un Arduino Mega, una interfaz LCD con codificador rotativo, y un motor a pasos NEMA 34. El proceso de reingeniería propuesto se centró en la incorporación de una Raspberry Pi 3 como unidad central de procesamiento, un motor más compacto y eficiente, y una interfaz táctil intuitiva para facilitar la configuración de parámetros y la visualización de datos en tiempo real. Además, se integró un convertidor analógico-digital (ADC) de alta resolución para mejorar la precisión en la lectura del sensor de presión. Con estas mejoras, se busca posicionar al nuevo prototipo como una alternativa más viable, eficiente y accesible para su uso en entornos clínicos o de emergencia que requieran asistencia respiratoria confiable y portátil. En la sección de Anexos, se muestran a través de tablas las diferencias significativas entre los motores NEMA, los drivers utilizados y las plataformas de control.

II. PARTE TÉCNICA DEL ARTÍCULO

2.1 Prototipo de respirador anterior (utilizando Arduino Mega)

El primer prototipo de respirador automatizado fue concebido como una alternativa de bajo costo para la asistencia respiratoria en situaciones de emergencia, como lo fue la pandemia de COVID-19. Este dispositivo está diseñado para automatizar la compresión de una bolsa resucitadora tipo Ambu, utilizando un sistema mecánico activado por un motor a pasos, controlado electrónicamente a través de una placa Arduino Mega. La solución priorizó la simplicidad, replicabilidad y facilidad de programación, aunque su precisión en la medición de presión y sincronización temporal era limitada.

2.1.1 Estructura mecánica del prototipo de respirador anterior

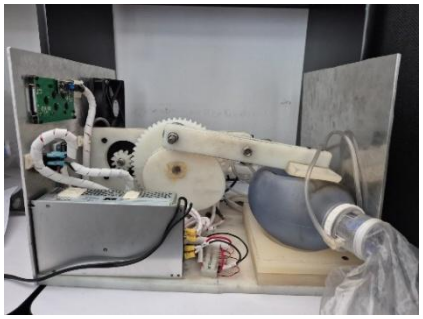
La estructura del prototipo anterior, véase la “Figura 1” y “Figura 2”, se diseñó principalmente con materiales accesibles, para los engranes, la palanca y la leva están hechos con Nylamid, el eje para la leva y placas de acero 1018 y los tornillos de acero inoxidable. En la base rectangular se monta un sistema de rieles para guiar el movimiento lineal. El mecanismo de compresión consiste en un brazo móvil que empuja la bolsa Ambu de manera repetitiva. Esta acción es realizada por un motor NEMA 34.

Dimensiones aproximadas del prototipo:
Altura: 31 cm
Largo: 49 cm
Ancho: 35 cm

Este diseño mecánico realizado permite ajustes manuales para modificar la frecuencia con la que se realizará el movimiento y se mantiene estable durante pruebas prolongadas.

Figura 1
Prototipo de respirador anterior vista frontal.

Figura 2
Prototipo de respirador anterior vista lateral.



2.1.2 Sistema de electrónico del prototipo de respirador anterior

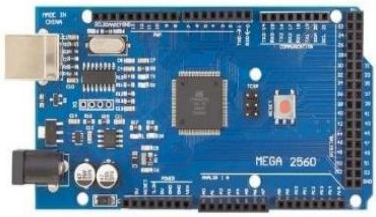
El sistema electrónico está compuesto por un microcontrolador Arduino Mega, encargado de generar las señales de control hacia el driver DM860H, el cual alimenta al motor NEMA 34. A su vez, un sensor de presión PN26998 permite monitorear los niveles de presión generados durante la ventilación. La interfaz de usuario incluye un display LCD 16x2 y un codificador (encoder) con botón para interactuar con la interfaz.

Componentes principales:
Arduino Mega: Este es un microcontrolador basado en el ATmega2560, se muestra en la “Figura 3”. Su arquitectura de 8 bits, junto con una frecuencia de reloj de 16 MHz, por estas características, fue adecuada para el proyecto. El dispositivo se encargaba de leer la señal del sensor de presión desde una entrada analógica, calcular valores estadísticos de presión

h t t p s : / / d o i . o r g / 1 0 . 6 2 8 0 5 / c i d 2 0 2 5 . C I D 0 5 4
This work is licensed under CC BY 4.0. To view a copy of this license, visit https://creativecommons.org/licenses/by/4.0/

(mínima, máxima, promedio) durante cada ciclo, controlar las señales de dirección y pulso hacia el driver del motor paso a paso y generar alertas por buzzer en caso de sobrepresión o fallas.

Figura 3
Microcontrolador Arduino Mega. (s. f.). Talos Electronics. Obtenido de: https://www.taloselectronics.com/products/arduino-mega-atmega2560-compatible

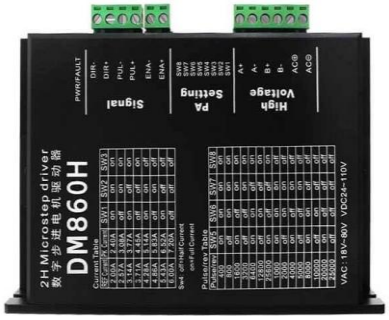


Motor a pasos NEMA 34: Es un actuador robusto de gran tamaño mostrado en la "Figura 4", comúnmente utilizado en aplicaciones industriales y CNC. Fue aprovechado para comprimir de manera efectiva la bolsa de resucitación manual (Bolsa Ambu). La elección de este motor se debió a la necesidad de fuerza en la compresión.

Figura 4
Motor a pasos Nema34. (s. f.). Electronilab. Obtenido de: https://electronilab.co/tienda/motor-paso-a-paso-nema-34-5-5a-637-oz-in-200-pasos-vuelta/

Driver de motor DM860H: Es un controlador de alto rendimiento para motores paso a paso; se muestra en la "Figura 5". Este modelo soporta una alimentación de hasta 80 VDC y corrientes de salida de hasta 7.2 A, lo cual lo hace adecuado para motores grandes como el empleado. Sus principales características para el prototipo fueron las entradas de control tipo PUL/DIR/ENA y la protección contra sobrecorriente, sobrevoltaje y sobretemperatura.

Figura 5
Manejador de motor a pasos DM860H. (s. f.). Unit Electronics. Obtenido de: https://uelectronics.com/producto/dm860h-driver-para-motor-a-pasos/



Sensor de presión PN26998: Es un sensor industrial de presión fabricado por IFM, se muestra en la "Figura 6", diseñado para aplicaciones neumáticas y de automatización. Opera en un rango de presión de –12.5 a 250 mbar, ideal para sistemas de ventilación pulmonar, especialmente cuando se requiere medir presiones bajas y diferenciales. Este sensor ofrece una salida analógica de corriente de 4 a 20 mA.

Figura 6
Sensor de presión PN2698. (s. f.). IFM. Obtenido de: https://www.ifm.com/mx/es/product/PN2698

Display LCD 16x2 con interfaz I2C: La visualización de parámetros se realizaba mediante un LCD 16x2, es decir, una pantalla de 2 líneas con 16 caracteres por línea, comunicada mediante interfaz I2C para reducir el número de pines utilizados en el microcontrolador.
Este display era utilizado para mostrar la frecuencia respiratoria seleccionada, la presión medida del ciclo actual (en mbar) y la presión mínima, máxima y promedio acumuladas durante cada ciclo de respiración. Se ilustra la pantalla en la "Figura 7".

Figura 7
Pantalla LCD 16x2. (s. f.). ElectroCrea. Obtenido de: https://electrocrea.com/products/display-16x2-azul-1

Fuente de alimentación externa: Provee 24V, 12V y 5V para motor y lógica de control, se muestra en la "Figura 8".

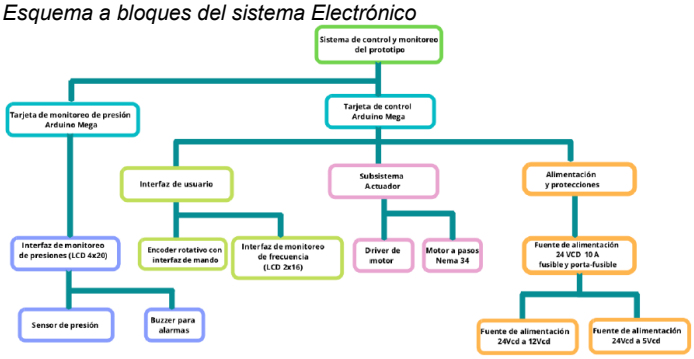
Figura 8
Fuente de alimentación.

https://doi.org/10.62805/cid2025.CID054
This work is licensed under CC BY 4.0. To view a copy of this license, visit https://creativecommons.org/licenses/by/4.0/

Año: 2025. Volumen: 1. Numero:11 ISSN: 2448-623X

Todos los componentes eléctricos utilizados en el proyecto fueron ordenados detalladamente en el diagrama de bloques de la “Figura 9”, donde se aprecian cómo son las conexiones de cada uno de ellos.

Figura 9
Esquema a bloques del sistema Electrónico



2.2 Propuesta de optimización: prototipo de respirador nuevo (utilizando Raspberry pi 3)

A partir del análisis del prototipo original, se identificaron áreas de mejora en cuanto a la capacidad de procesamiento y precisión en las mediciones. Por lo tanto, se propuso un nuevo diseño optimizado que sustituye el Arduino Mega por una Raspberry Pi 3, permitiendo implementar funciones avanzadas como adquisición y registro de datos en tiempo real, visualización gráfica, y procesamiento digital de señales. Además, se mejoró el tiempo de ejecución del ciclo respiratorio mediante un control más eficiente del motor paso a paso, ahora usando un NEMA 23 con un driver TB6600.

2.2.1 Estructura mecánica prototipo de respirador nuevo

A diferencia del diseño original que se basaba en una estructura de Nylamid y mecanismos voluminosos, el nuevo prototipo optimizado incorpora un diseño más compacto, con una estructura basada en perfiles de aluminio, componentes impresos en 3D, y manteniendo una leva de Nylamid. Esto facilita tanto su transporte como su ensamblaje y mantenimiento.

El sistema está conformado por una base rectangular de aluminio, sobre la cual se monta un brazo móvil conectado a un motor NEMA 23 encargado de comprimir la bolsa Ambu. Al igual que el prototipo anterior, en este nuevo su mecanismo de compresión es vertical, reduciendo el espacio requerido para su operación, en la “Figura 10” y “Figura 11” podemos apreciar una vista clara del prototipo nuevo.

Dimensiones del nuevo prototipo:
Altura: 27 cm
Largo: 40 cm
Ancho: 25 cm

Figura 10
Prototipo de respirador nuevo vista frontal.

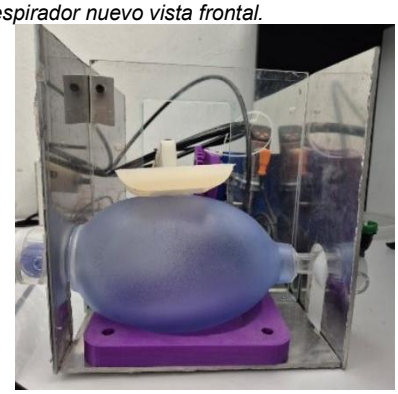
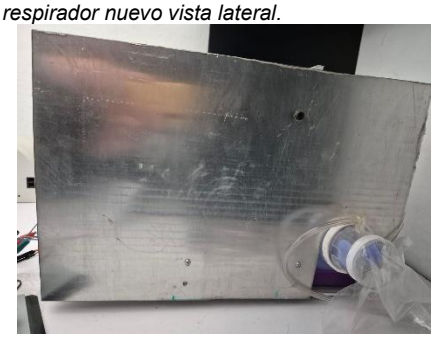


Figura 11
Prototipo de respirador nuevo vista lateral.



2.2.2 Sistema electrónico de prototipo de respirador nuevo

El nuevo sistema se basa en una Raspberry Pi 3 que ejecuta un script en Python, controlando directamente los pines GPIO para generar las señales necesarias hacia el driver del motor. Para la adquisición de señales analógicas, se incorpora el conversor ADS1115, el cual ofrece alta resolución (16 bits), superando ampliamente al conversor de 10 bits del Arduino. Esta mejora permite detectar con mayor precisión los valores del sensor de presión, componentes principales:

Raspberry Pi 3: Es una computadora de placa reducida, basada en un procesador ARM Cortex-A53 de 64 bits, con cuatro núcleos a 1.2 GHz; se ilustra una imagen en la “Figura 12”. Este dispositivo reemplaza al Arduino como unidad de control central. Se conecta vía I2C al ADS1115 y controla directamente los pines digitales del driver TB6600 mediante GPIO.
Su capacidad de procesamiento multitarea permite una resolución de tiempo mucho más fina y reduce el tiempo por ciclo completo.

Figura 12
Microcomputadora Raspberry pi 3. (s. f.). Newark. Obtenido de: https://mexico.newark.com/raspberry-pi/raspberrypi3-modb/sbc-raspberry-pi-1gb-cortex-a53/dp/42AH1370

https://doi.org/10.62805/cid2025.CID054
This work is licensed under CC BY 4.0. To view a copy of this license, visit https://creativecommons.org/licenses/by/4.0/

Año: 2025. Volumen: 1. Numero:11 ISSN: 2448-623X

Driver de motor TB6600: Es un driver profesional de motor a pasos, ampliamente utilizado en CNC y robótica por su estabilidad y facilidad de uso, se muestra en la "Figura 13". En este proyecto fue configurado para controlar un motor NEMA 23, trabajando con las señales de control PUL (pulso), DIR (dirección) y ENA (habilitación) provenientes de la Raspberry Pi. Es compatible con señales de 3.3 V – 5 V (lógicas aceptadas por Raspberry, protección contra cortocircuitos, sobre temperatura y sobre corriente).

Figura 13

Manejador de motor a pasos TB6600. (s. f.). Unit Electronics. Obtenido de: https://uelectronics.com/producto/tb6600-driver-para-motor-a-pasos-4a-nema/

Motor paso a paso NEMA 23: Es un motor más compacto que el NEMA 34, pero capaz de proporcionar un torque suficiente para comprimir la bolsa Ambu de forma controlada. Se muestra una imagen en la "Figura 14". Su menor tamaño y peso permiten una optimización de las dimensiones del respirador, haciéndolo más portátil y eficiente. Se logró ajustar con mayor precisión el mecanismo de compresión.

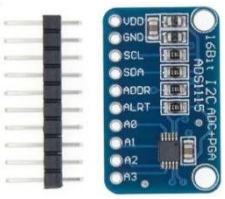
Figura 14

Motor a pasos Nema 23. (s. f.). Talos Electronics. Obtenido de: https://www.taloselectronics.com/products/nema-23-motor-a-pasos-2a

ADS1115 Conversor Analógico Digital de 16 bits: Es un ADC (Analog-to-Digital Converter) externo que se comunica con la Raspberry Pi mediante el protocolo I2C que proporciona una resolución de 16 bits, es decir, 32,768 campos por polaridad, lo que permite capturar variaciones mínimas de voltaje.
Este ADC fue fundamental para convertir la señal analógica del sensor PN26998 a un valor digital preciso, se puede apreciar una imagen de referencia en la "Figura 15".

Figura 15

Conversor analógico digital ADS1115. (s. f.). Unit Electronics. Obtenido de: https://uelectronics.com/producto/ads1115-convertidor-analogico-digital-adc-16-bit-i2c/



Pantalla táctil Raspberry Pi 7'': Se desarrolló una interfaz gráfica en Tkinter, una librería de Python para la creación de interfaces de usuario. Esta interfaz se despliega en una pantalla táctil de 7 pulgadas para Raspberry Pi.

La pantalla mostrada en la "Figura 16" permite al usuario seleccionar la frecuencia respiratoria mediante un control deslizante, controlar el sistema entero por medio de sus botones dinámicos y visualizar en tiempo real los valores de la presión.

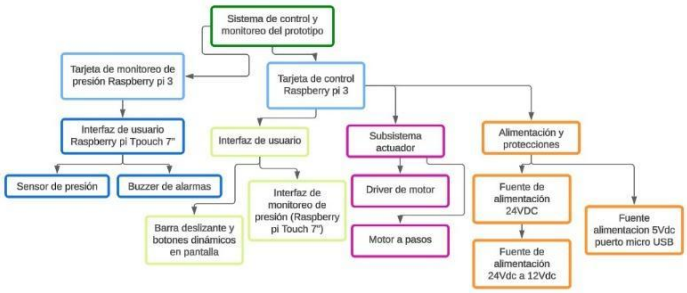
Figura 16

Pantalla táctil Raspberry Pi de 7''. (s. f.). 300ohms. Obtenido de: https://www.330ohms.com/products/raspberry-pi-7-touchscreen-display

Los dispositivos electrónicos antes mencionados tienen ahora una nueva organización en cuestión de conexiones, en la "Figura 17" se observa un diagrama de bloques elaborado para entender de una manera simple el enlace de cada componente.

Figura 17

Diagrama de bloques prototipo nuevo



2.2.3 Metodología para realizar la optimización de prototipo de respirador nuevo

La optimización del sistema se llevará a cabo mediante una metodología orientada a evaluar, rediseñar y mejorar el desempeño del prototipo de respirador automatizado.

En primer lugar, se analizará el funcionamiento del sistema actual basado en el microcontrolador Arduino Mega. Este análisis incluirá la medición del tiempo de ejecución por ciclo respiratorio, la resolución efectiva del sensor de presión PN26998 (con salida analógica), así como la capacidad de respuesta del microcontrolador de 8 bits ante las exigencias del procesamiento en tiempo real.

https://doi.org/10.62805/cid2025.CID054
This work is licensed under CC BY 4.0. To view a copy of this license, visit https://creativecommons.org/licenses/by/4.0/

Posteriormente, se desarrollará una nueva interfaz de usuario en lenguaje Python, que será capaz de ejecutar múltiples tareas en paralelo. Esta incluirá la adquisición continua de datos del sensor de presión, la generación de señales hacia el driver del motor, el cálculo estadístico de los datos recogidos y su visualización gráfica en tiempo real mediante una pantalla táctil de 7". Para ello, se utilizarán bibliotecas como Tkinter (interfaz gráfica) y el bus I2C para la comunicación con el conversor ADS1115.

Se optó por realizar pruebas controladas para comparar el prototipo original con el sistema optimizado, pruebas a realizar:

Prueba 1: Medición del tiempo de ejecución por ciclo respiratorio. Esta prueba tiene como objetivo cuantificar la duración total de un ciclo completo de ventilación en ambos sistemas.

1.Se programan ambos sistemas para ejecutar 10 ciclos consecutivos de respiración con parámetros idénticos.

2.En Arduino Mega se utiliza la función millis() para medir el tiempo de cada ciclo, mientras que en Raspberry Pi 3 se emplea time.perf_counter() de Python por su mayor resolución.

3.En la toma de medición el tiempo comienza a contar al poner a trabajar el respirador por lo que se incluyeron los procesos de adquisición de datos, procesamiento estadístico (mínimo, máximo, promedio), generación de señales hacia el driver del motor, y actualización de la interfaz de usuario (LCD o pantalla táctil).

4.Los tiempos medidos se promedian y se calcula la desviación estándar en cada sistema para evaluar su viabilidad operativa.

Prueba 2: Evaluación de la precisión y estabilidad de las lecturas del sensor de presión. Esta prueba se enfoca en validar la fidelidad con la que cada sistema podía registrar y representar los valores reales de presión. Para ello, se realiza en un mismo bloque tanto la calibración inicial como la toma de mediciones experimentales.

1.Se utiliza una fuente neumática controlada (compresor con válvula reguladora de presión) y un manómetro como patrón de referencia para establecer tres puntos de presión conocidos: 40 mbar, 80 mbar y 120 mbar.

2.En cada uno de estos niveles de presión, se mantiene la salida del compresor estable durante al menos 30 segundos, comparando simultáneamente las lecturas entregadas al sensor de presión PN26998 con las del manómetro patrón.

3.Se verifica que en ninguna de las tres mediciones la diferencia entre el sensor y el patrón supere los ±0.5 mbar, asegurando así una correcta calibración para ambos sistemas.

4.En base a lo obtenido en la calibración se toman las mismas lecturas, procesando los datos entregados por el ADC interno de 10 bits en el caso del Arduino, y por el módulo ADS1115 de 16 bits en la Raspberry Pi.

5.Se analizan las mediciones mediante el cálculo de la desviación estándar, lo que permite evaluar la estabilidad de lectura de cada plataforma.

Prueba 3: Toma de mediciones físicas de los prototipos. Esta última prueba tiene como objetivo calcular las dimensiones de ambos respiradores y verificar la reducción del dispositivo actual.

1.Utilizando un flexómetro se mide el largo, alto y ancho de los prototipos.

2.Se calcula el volumen con las medidas obtenidas y se realiza una resta para obtener la diferencia en volumen.

III. RESULTADOS Y DISCUSIÓN

La optimización del sistema de control del respirador permitió obtener mejoras significativas en múltiples aspectos del funcionamiento general del prototipo. Los resultados más destacados se agrupan en tres categorías principales: eficiencia temporal, precisión en la lectura de presión y capacidad de visualización e interacción.

3.1 Comparación del tiempo de ejecución por ciclo respiratorio

Se realizaron mediciones del tiempo total que tarda el sistema en ejecutar un ciclo completo de ventilación (envío de señales, lectura de sensor y muestreo de datos en la pantalla). Se realizaron 10 ciclos consecutivos y se midió el tiempo de cada uno, se calculó el promedio y la desviación estándar. A continuación, se encuentran los tiempos obtenidos en ambos microcontroladores.

Para el Arduino Mega, los tiempos medidos (en ms) fueron: 3400, 3340, 3290, 3250, 3300, 3350, 3280, 3310, 3380, 3240
Para la Raspberry Pi 3, los tiempos fueron: 22.0, 23.1, 22.5, 21.8, 22.4, 22.3, 22.1, 22.2, 22.7, 22.1

Primero se realizó un promedio utilizando "(1)", en donde t simboliza los tiempos medidos de ambos dispositivos y N el número de lecturas realizadas.

$$Promedio = \frac{\sum \quad ti}{N} \tag{1}$$

Con las Ecuaciones "(2)" y "(3)", se logró obtener un promedio de tiempo de 3314 ms para el microcontrolador Arduino y 22.32 ms para la Raspberry.

$$Promedio\ tiempo\ Arduino = \frac{33140}{10} = 3314\ ms \tag{2}$$

$$Promedio\ tiempo\ Raspberry = \frac{223.2}{10} = 22.32\ ms \tag{3}$$

En seguida, con ayuda de la fórmula de la desviación estándar se logró obtener el margen de error de cada microcontrolador, donde en "(4)" y "(5)" resultó una desviación estándar de aproximadamente 52.18 ms para Arduino y de 0.357 ms para Raspberry.

$$\sigma\ Arduino = \sqrt{\frac{1}{N}\sum_{i=1}^{N} (ti - t)^2} \approx 52.18\ ms \tag{4}$$

https://doi.org/10.62805/cid2025.CID054
This work is licensed under CC BY 4.0. To view a copy of this license, visit https://creativecommons.org/licenses/by/4.0/

$$\sigma\ Raspberry = \sqrt{\frac{1}{N}\sum_{i=1}^{N} (ti - t)^2} \approx 0.357\ ms \quad (5)$$

En la tabla 1 se pueden observar los resultados que se obtuvieron en las pruebas.

Tabla 1
Resultados de la prueba de velocidad de procesamiento

Microcontrolador	*Tiempo por ciclo (ms)*	*Error*
Arduino Mega	3314	±52.18
Raspberry pi 3	22.32	±0.357

Estos datos presentados en la tabla 1 representan una mejora en la parte de velocidad de ejecución y una variabilidad 146 veces menor. Esta diferencia se debe principalmente a la arquitectura superior de la Raspberry Pi, que permite ejecutar múltiples procesos en paralelo y operar a velocidades de reloj mucho más altas, lo cual es fundamental en aplicaciones médicas donde la estabilidad y la respuesta rápida son críticas.

3.2 Precisión en la lectura de presión

Para esta prueba, se utilizaron valores de presión simulada de 40 mbar, 80 mbar y 120 mbar, calibrando los sensores con un manómetro haciendo comparaciones de las lecturas mostradas en el sensor y en el patrón. Se midieron los valores entregados por los conversores analógico-digitales de cada sistema y se estimó su error.
Gracias a la incorporación del ADS1115, se logró una resolución de 16 bits (frente a los 10 bits del Arduino), lo cual permitió captar pequeñas variaciones de presión con mayor fidelidad. Durante las pruebas de simulación de presión con un compresor y una válvula reguladora, se obtuvieron los resultados mostrados en la tabla 2.

Tabla 2
Resultados de las pruebas de resolución

Presión simulada	*Lectura ADC (Arduino)*	*Error estimado*	*Lectura ADC (Raspberry)*	*Error estimado*
40 mbar	227	±0.256mbar	7240	±0.08mbar
80 mbar	351	±0.256mbar	11229	±0.08mbar
120 mbar	469	±0.256mbar	15217	±0.08mbar

Para calcular los respectivos errores de cada microcontrolador, primero se obtuvo la resolución de voltaje que se define como el valor mínimo de voltaje que puede detectar el ADC, para esto se hizo uso de "(6)".

$$\Delta V = \frac{V_{ref}}{N} \quad (6)$$

donde:
V_{ref} se entiende por el voltaje de referencia (5V)
N es el número de campos cada dispositivo

En base a los resultados obtenidos en "(7)" y "(8)" se calculó un error aproximado de 4.88 mV para Arduino y 0.153 mV para Raspberry.

$$\Delta V\ Arduino = \frac{5\ V}{1024} \approx 4.88mV \quad (7)$$

$$\Delta V\ Raspberry = \frac{5\ V}{32767} \approx 0.153mV \quad (8)$$

Obteniendo la resolución de cada dispositivo se pasa a la ecuación "(9)" para la calcular el error en presión con respecto al voltaje, donde se utiliza el valor 262.5 debido al rango neto del sensor de presión PN26998 (-12.5 a 250 mbar)

$$Error = \left(\frac{\Delta V}{5000\ mV}\right) x\ 262.5 \quad (9)$$

Como resultado de los cálculos empleados en "(10)" y "(11)" se obtuvo un error de presión de aproximadamente 0.256 mbar en Arduino y de 0.008 mbar en Raspberry.

$$Error\ Arduino = \left(\frac{4.88}{5000\ mV}\right) x\ 262.5 \approx 0.256mbar \quad (10)$$

$$Error\ Raspberry = \left(\frac{0.153}{5000\ mV}\right) x\ 262.5 \approx 0.008mbar \quad (11)$$

3.3 Comparación de Dimensiones Físicas entre Prototipos

Uno de los objetivos clave en el rediseño del respirador fue la reducción del tamaño y el volumen físico del equipo, sin sacrificar funcionalidad ni confiabilidad mecánica. Se presenta en la tabla 3 una comparativa entre ambos diseños.

Tabla 3
Dimensiones de ambos prototipos

Prototipo	*Altura (cm)*	*Largo (cm)*	*Ancho (cm)*	*Volumen (cm3)*
Anterior	31	49	35	53165
Nuevo	27	40	25	27000

Esta disminución significativa fue posible gracias al uso de materiales más ligeros y al rediseño del sistema de compresión. Para obtener los cálculos del prototipo anterior se trabajó "(12)", al igual que al nuevo empleando la ecuación "(13)".

$$V1 = 31\ x\ 49\ x\ 35 = 53165\ cm^3 \quad (12)$$

$$V2 = 27\ x\ 40\ x\ 25 = 27000\ cm^3 \quad (13)$$

https://doi.org/10.62805/cid2025.CID054
This work is licensed under CC BY 4.0. To view a copy of this license, visit https://creativecommons.org/licenses/by/4.0/

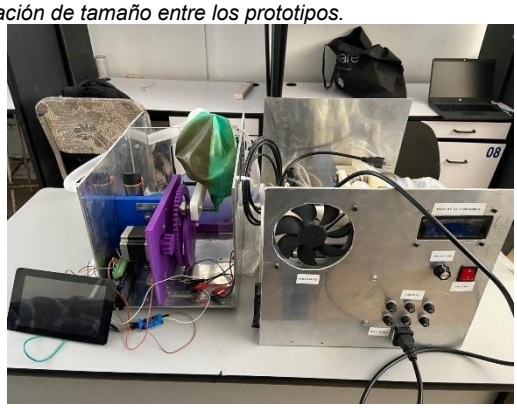
En la ecuación "(14)" se observa la reducción absoluta de volumen que se logró ahorrar $26165\,cm^3$, en otras palabras, hablando en términos porcentuales se puede deducir con "(15)" que se redujo el tamaño un 49.19%.

$$\Delta V = 53165 - 27000 = 26165 cm^3 \quad (14)$$

$$Reducción\ \% = \left(\frac{26165}{53185}\right) x100 = 49.19\% \quad (15)$$

En este nuevo diseño logró reducir su volumen total en casi 50% de su tamaño original, sin comprometer la funcionalidad, como se observa en la "Figura 18", es posible ver una diferencia gigante entre ambos prototipos, resaltando el prototipo nuevo (dispositivo de la izquierda) por tener un menor tamaño que el prototipo anterior.

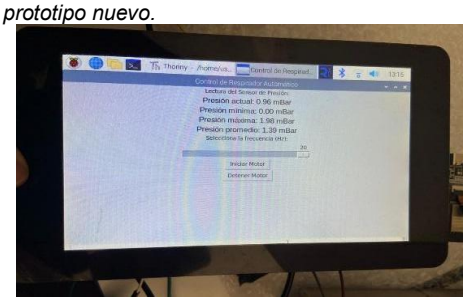
Figura 18
Comparación de tamaño entre los prototipos.



Además de resaltar el avance tecnológico que se logró observar en cuestión de manipulación y control del prototipo anterior, donde solamente se podían observar datos limitados con el componente LCD, véase la "Figura 19". Con la optimización del prototipo se enriqueció la composición tecnológica adaptando una pantalla táctil para mejor manejo por parte del usuario, véase la "Figura 20".

Figura 19
Interfaz de prototipo anterior.

Figura 20
Interfaz de prototipo nuevo.



IV. CONCLUSIONES

El proceso de rediseño y optimización del respirador automatizado basado en bolsa Ambu demostró mejoras significativas en términos de rendimiento, precisión, eficiencia y portabilidad. A partir del análisis comparativo entre el sistema original basado en Arduino Mega y el nuevo sistema optimizado con Raspberry Pi 3.

Los resultados experimentales muestran una diferencia grande entre ambos sistemas, mientras que el Arduino Mega requería un promedio de 3314 ms para completar un ciclo completo de lectura, procesamiento y control motor, la Raspberry Pi 3 fue capaz de ejecutar el mismo ciclo en tan solo 22.32 ms, lo que representa una excelente mejora en velocidad. Esta diferencia se atribuye principalmente a la arquitectura superior de la Raspberry Pi, en comparación con el microcontrolador de 8 bits de Arduino. La reducción en la variabilidad del tiempo (±0.357 ms en Raspberry frente a ±52.18 ms en Arduino) resalta también una mayor estabilidad del sistema.
El empleo del conversor analógico-digital ADS1115 de 16 bits permitió incrementar la precisión de lectura del sensor de presión PN26998, reduciendo el error de medición de ±0.256 mbar en Arduino a tan solo ±0.008 mbar para la Raspberry. Esto se traduce en una capacidad de monitoreo clínico más precisa, permitiendo capturar variaciones mínimas en la presión.

Se logró una disminución volumétrica de casi 50%, al pasar de 53,165 cm³ a 27,000 cm³, gracias al cambio de materiales pasando de robustos a pequeños, usando un motor más compacto (NEMA 23 en lugar del NEMA 34). Este rediseño mecánico no solo mejora la portabilidad del sistema, sino que también facilita su transporte.

La migración del sistema de visualización desde un LCD básico de 16x2 a una pantalla táctil de 7" con interfaz gráfica no solo mejoró la experiencia del usuario, sino que permitió ajustar parámetros con mayor facilidad.

La propuesta de optimización no solo logró superar las limitaciones del prototipo original, sino que elevó el nivel técnico del sistema, resultando en una alternativa más eficiente, precisa, compacta y escalable de un modelo de respirador automatizado basado en bolsa Ambu.

https://doi.org/10.62805/cid2025.CID054
This work is licensed under CC BY 4.0. To view a copy of this license, visit https://creativecommons.org/licenses/by/4.0/

V. AGRADECIMIENTOS

Expresamos nuestro sincero agradecimiento al Tecnológico Nacional de México, Instituto Tecnológico de la Laguna e Instituto Tecnológico Superior de Lerdo, por el respaldo académico y las facilidades otorgadas para la realización de este proyecto. Su compromiso con la formación de investigadores fue un pilar fundamental para el desarrollo exitoso de este trabajo.
Asimismo, extendemos nuestro reconocimiento a todas las personas que participaron en la ejecución del proyecto. Su colaboración y asesoría técnica fueron esenciales para el cumplimiento de los objetivos planteados.

VI. ANEXOS

Las siguientes tablas muestran la comparación significativa entre los motores NEMA, los drivers utilizados y las plataformas de procesamiento.

Tabla 4
Comparativa entre motores paso a paso NEMA 23 vs NEMA 34.

Característica	NEMA 23	NEMA 34
Tamaño de cara (mm)	56 x 56	86 x 86
Paso por vuelta	1.8°(200 pasos/rev)	1.8°(200pasos/rev)
Corriente nominal	2.8–6 A	4–6.2 A
Torque	Hasta 3 Nm	Hasta 12 Nm
Peso aproximado	0.5–1.8 kg	2–5 kg

Tabla 5
Comparativa entre drivers DM860H vs TB6600.

Característica	DM860H	TB6600
Voltaje alimentación	24-110Vdc/18-80 Vac	9–40 Vdc
Corriente de salida	2.1–7.2 A (ajustable)	0.7–4 A (ajustable)
Frecuencia de pulso	Hasta 200 kHz	Hasta 20 kHz
Micro-stepping	Hasta 25,600 pasos/rev	Hasta 6,400 pasos/rev

Tabla 6
Comparativa entre Arduino Mega vs Raspberry Pi 3

Característica	Arduino Mega	Raspberry Pi 3
Tipo de plataforma	Microcontrolador AVR (8 bits)	Microcomputadora ARM (64 bits)
Frecuencia de reloj	16 MHz	1.2 GHz (quad-core)
RAM	8 KB	1 GB
Entradas analógicas	16 ADC	No (requiere ADC externo)
Conectividad	USB, I2C, SPI	Wi-Fi, Bluetooth, Ethernet

VII. REFERENCIAS

Chauhan, R., Bloria, S. D., Thappa, P., Sharma, R., Sarna, R., Meena, S. C., Luthra, A., Panda, N. B., Mohindra, S., Singh, N., & Patel, S. (2024). Assessment of the efficacy of an automated AMBU bag operating device (RC device) in patients requiring mechanical ventilation: *A pilot study. Indian Journal of Neurotrauma, 21(1), 12–16.* https://www.thieme-connect.de/products/ejournals/pdf/10.1055/s-0043-1777675.pdf

Corey, R. M., Widloski, E. M., Null, D., Ricconi, B., Johnson, M. A., White, K. C., Amos, J. R., Pagano, A., Oelze, M. L., Switzky, R. D., Wheeler, M. B., Bethke, E. B., Shipley, C. F., & Singer, A. C. (2020). Low-Complexity System and Algorithm for an Emergency Ventilator Sensor and Alarm. *IEEE transactions on biomedical circuits and systems*, *14*(5), 1088–1096. https://doi.org/10.1109/TBCAS.2020.3020702

Ding, B., Xu, F., Wang, J., Pan, C., Pang, J., Chen, Y., & Li, K. (2023). *Design and evaluation of portable emergency ventilator prototype with novel titration methods*. Biomedical Signal Processing and Control, 83. https://doi.org/10.1016/j.bspc.2023.104619

Du Pasquier, C., Hewing, L., Steffen, N., Lumpe, T. S., Tatchatos, N., & Shea, K. (2021). Systematic design, control and parametric testing of an automated resuscitator bag mechanical ventilator. *Journal of Mechanical Design, 143(5), 1-12.* https://www.research-collection.ethz.ch/bitstream/handle/20.500.11850/488070/md_143_12_125001.pdf

Española, J., Calliung, E., Dadios, E., Culaba, A., Sybingoo, E., Bandala, A., Vicerra, R. R., Madrazo, A. B., Lim, L. G., Billones, R. K., Ligutan, D. D., Palingcod, J., & Castillo, C. J. P. (2020). *Calibration and testing of the integrated ventilator scalar measurement module for a bag-valve-mask-based emergency ventilator.* 2020 IEEE 12th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM), Manila, Philippines, 2020, pp. 1-5. https://ieeexplore.ieee.org/document/9400118

Jain, S., Rajput, K. S., Sikarwar, D. S., & Batham, D. (2023). *PIMV: Portable Invasive Mechanical Ventilator. In Smart Engineering Technology and Management (Proceedings of ISCMCTR – 2023) (pp. 447–451).* Bharti Publications.

Medina Tanco, G. (2020). **Proyecto para la producción de respirador básico para la emergencia 2020 de COVID-19* [*Reporte técnico]. Laboratorio de Instrumentación Espacial (LINX), Instituto de Ciencias Nucleares, Universidad Nacional Autónoma de México.

Morales, S., Palomino, S., Terreros, R., Ulloque, V., Bazán-Lavanda, N., Palacios-Matos, M., Valdivia-Silva, J., Vela, E. A., & Canahuire, R. (2021). *Pressure and Volume Control of a Non-invasive Mechanical Ventilator: a PI and LQR Approach.* 2021 9th International Conference on Control, Mechatronics and Automation (IC CMA). IEEE.https://ieeexplore.ieee.org/document/9646186

Ort, T., Hanumara, N., Antonini, A., Araki, B., Abu-Khalaf, M., Detienne, M., Hagan, D., Jung, K., Ramirez, A., Shaligram, S., Unger, C., Kwon, A., Slocum, A., Nabzdyk, C., Varelmann, D., Connor, J., Rus, D., & Slocum, A. (2021). *MIT Emergency-Vent: An Automated Resuscitator Bag for the COVID-19 Crisis. Annual International Conference of the IEEE Engineering in Medicine and Biology Society.* IEEE Engineering in Medicine and Biology Society. Annual International Conference, 2021, 4998–5004. https://doi.org/10.1109/EMBC46164.2021.9630882

Robati, M., Nazari, M. A., Sadighi, A., & Naseri, H. (2022). *MoMed: An affordable mechanical ventilator.* 2022 30th International Conference on Electrical Engineering (ICEE), 345–350. https://ieeexplore.ieee.org/document/9827389

https://doi.org/10.62805/cid2025.CID054

This work is licensed under CC BY 4.0. To view a copy of this license, visit https://creativecommons.org/licenses/by/4.0/