

การพัฒนาระบบควบคุมอัตโนมัติสำหรับการเปิด-ปิดเครื่องปรับอากาศโดยการจำลอง สัญญาณควบคุมจากรีโมทคอนโทรล

The Development of an Automated System for Air Conditioner On/Off Control Using Remote Signal Emulation

ธานินทร์ ดวงจันทร์ ศิริพงษ์ ฉายสินธ์*

ภาควิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ องครักษ์ 26120

Tanin Duangchan Siripong Chaysin*

¹Department of Electrical Engineering Faculty of Engineering Sirinakharinwirot University 26120

*Corresponding author Email: Siripong@g.swu.ac.th

(Received: June 25, 2026; Revised: July 7, 2026 ; Accepted: July 24, 2026)

บทคัดย่อ

เครื่องปรับอากาศที่ใช้ภายในอาคารหรือห้องจัดเก็บอุปกรณ์ บางประเภทไม่มีฟังก์ชันตั้งเวลาเปิด-ปิดอัตโนมัติ หรือมีข้อจำกัดในการใช้งานฟังก์ชันดังกล่าว ส่งผลให้ต้องอาศัยบุคลากรในการควบคุมการทำงานอยู่เสมอ งานวิจัยนี้มีวัตถุประสงค์เพื่อพัฒนาระบบควบคุมการเปิด-ปิดเครื่องปรับอากาศแบบอัตโนมัติโดยใช้ไมโครคอนโทรลเลอร์จำลองสัญญาณจากรีโมทคอนโทรลเดิมของเครื่องปรับอากาศ วิธีการดำเนินการเริ่มจากการใช้บอร์ด ESP8266 ร่วมกับตัวรับสัญญาณอินฟราเรดเพื่ออ่าน และบันทึกข้อมูลสัญญาณจากรีโมทคอนโทรล จากนั้นนำข้อมูลที่ได้นำมาประมวลผลและสร้างสัญญาณอินฟราเรดจำลองสำหรับสั่งงานเครื่องปรับอากาศ พร้อมเชื่อมต่อกับวงจรนาฬิกาจริง (Real-Time-Clock : RTC) เพื่อกำหนดเวลาเปิด-ปิดอัตโนมัติ ระบบต้นแบบได้รับการทดสอบกับเครื่องปรับอากาศภายในห้องจัดเก็บเครื่องมือวัดทางไฟฟ้า โดยกำหนดให้เครื่องปรับอากาศทำงานระหว่างเวลา 10.00-16.00 น. และทำการบันทึกอุณหภูมิภายในห้องอย่างต่อเนื่องเป็นระยะเวลา 15 วัน ผลการทดลองพบว่าอุปกรณ์สามารถส่งสัญญาณควบคุมเครื่องปรับอากาศได้ถูกต้องตามเวลาที่กำหนดทุกครั้ง คิดเป็นความสำเร็จร้อยละ 100 และข้อมูลอุณหภูมิแสดงให้เห็นว่าอุณหภูมิภายในห้องลดลงอย่างต่อเนื่องในช่วงเวลาที่เครื่องปรับอากาศถูกสั่งให้ทำงาน ผลการวิจัยยืนยันว่าระบบที่พัฒนาขึ้นสามารถทดแทนการควบคุมด้วยรีโมทคอนโทรลแบบปกติได้อย่างมีประสิทธิภาพ ช่วยลดการพึ่งพาการควบคุมโดยมนุษย์ และสามารถประยุกต์ใช้กับเครื่องปรับอากาศยี่ห้อและรุ่นอื่นที่ควบคุมด้วยสัญญาณรีโมทได้

คำสำคัญ: ไมโครคอนโทรลเลอร์ เครื่องปรับอากาศ สัญญาณรีโมทคอนโทรล สัญญาณจำลองรีโมทคอนโทรล

ABSTRACT

Some air conditioners used in buildings or equipment storage rooms do not provide an automatic on/off scheduling function, or their timer functions have operational limitations. As a result, personnel are required to manually control the operation of these air conditioners. This research aimed to develop an automatic air conditioner control system using a microcontroller to emulate the infrared signals transmitted by the original remote control. The proposed system was implemented by

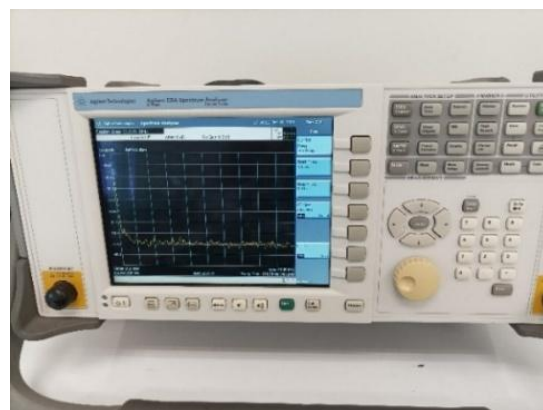
employing an ESP8266 development board together with an infrared receiver to capture and record signal data from the air conditioner remote control. The acquired signals were then analyzed and used to generate corresponding infrared command signals for controlling the air conditioner. In addition, a Real-Time Clock (RTC) module was integrated into the system to enable automatic scheduling of on/off operations. The prototype system was tested with an air conditioner installed in an electrical measuring instrument storage room. The operating schedule was configured to automatically turn the air conditioner on from 10:00 a.m. to 4:00 p.m., while room temperature data were continuously recorded over a period of 15 days. The experimental results demonstrated that the developed device successfully transmitted the control signals and operated the air conditioner according to the predefined schedule in every test, achieving a 100% success rate. Furthermore, the recorded temperature data indicated a continuous decrease in room temperature during the periods when the air conditioner was activated. The findings confirm that the proposed system can effectively replace conventional remote-control-based operation of air conditioners. The developed system reduces reliance on human intervention while maintaining reliable operational performance. Moreover, it can be adapted for use with other brands and models of air conditioners that are controlled via infrared remote signals.

Keyword: Microcontroller, air conditioner, remote control signal, remote control signal emulation.

1. บทนำ

ปัจจุบันเครื่องวัดสัญญาณทางไฟฟ้าสมัยใหม่ในส่วนของการแสดงผลจะใช้จอแสดงผลแบบแอลซีดี (LCD) โดยจอประเภทนี้จะมีการติดตั้งฟิล์มเพื่อการหักเหของแสงทั้งด้านหน้าและด้านหลังของจอ (รูปที่ 1) ซึ่งฟิล์มเหล่านี้จะมีอายุการใช้งาน แต่หากการเก็บรักษาเครื่องวัดไว้ในที่มีอุณหภูมิที่ค่อนข้างสูงต่อเนื่องเป็นเวลานานอาจจะทำให้อายุการใช้งานของฟิล์มสั้นลงกว่าที่ควรจะเป็น (รูปที่ 2) ซึ่งปัญหาดังกล่าวมีโอกาสเกิดได้ดังที่ผู้วิจัยประสบปัญหาเนื่องจากห้องจัดเก็บเครื่องวัดสัญญาณทางไฟฟ้าอยู่ในตำแหน่งที่แดดส่องตลอดเวลาหลายชั่วโมงต่อวัน ทำให้ต้องเปิดเครื่องปรับอากาศเพื่อให้อุณหภูมิในห้องอยู่ในช่วงที่เหมาะสม และจากการทบทวนวรรณกรรมพบว่า ระบบควบคุมเครื่องปรับอากาศสามารถพัฒนาได้ด้วยเทคโนโลยีไมโครคอนโทรลเลอร์ [1] – [3] สัญญาณอินฟราเรด และระบบฝังตัว [4] - [5] ซึ่งช่วยเพิ่มความสะดวกและลดการ

พึ่งพามนุษย์ อย่างไรก็ตาม แนวทางเดิมยังมีข้อจำกัด เช่น ความซับซ้อนของสัญญาณรีโมทที่ไม่เป็นมาตรฐาน [6] – [7] การตั้งเวลาเปิด-ปิดที่ไม่ต่อเนื่องสามารถถูกยกเลิกได้เมื่อมีการกดปุ่มอื่นบนรีโมทคอนโทรล หรือข้อจำกัดของระบบที่ต้องพึ่งพาอินเทอร์เน็ตในบางกรณี [8] - [9]



รูปที่ 1 เครื่องวัดทางไฟฟ้าที่มีจอแสดงผลแบบแอลซีดี

ดังนั้นงานวิจัยนี้จึงพัฒนาแนวทางที่แตกต่าง โดยใช้ไมโครคอนโทรลเลอร์เพื่ออ่านและจำลองสัญญาณรีโมทคอนโทรลร่วมกับวงจรนาฬิกาเวลาจริง (RTC) เพื่อกำหนดเวลาการทำงานแบบอัตโนมัติ โดยไม่ต้องพึ่งพา รีโมทเดิมหรือระบบเครือข่ายอินเทอร์เน็ต ซึ่งสามารถแก้ไขข้อจำกัดของงานเดิม และเพิ่มความต่อเนื่องในการควบคุมเครื่องปรับอากาศได้อย่างมีประสิทธิภาพ ไม่กระทบกับระบบเดิมของเครื่องปรับอากาศ และการลดการพึ่งพาการควบคุมโดยมนุษย์ ซึ่งสะท้อนถึงแนวโน้มของการพัฒนาระบบอัตโนมัติในการจัดการพลังงาน และสิ่งแวดล้อมภายในอาคารอย่างมีประสิทธิภาพ [10]



รูปที่ 2 หน้าจอแสดงผลแบบแอลซีดีเสื่อมสภาพ

2. วัตถุประสงค์

งานวิจัยนี้มีวัตถุประสงค์เพื่อพัฒนาระบบควบคุมการเปิด-ปิด เครื่องปรับอากาศแบบอัตโนมัติ โดยใช้ไมโครคอนโทรลเลอร์จำลองสัญญาณรีโมทคอนโทรล และสามารถกำหนดเวลาการทำงานได้อย่างแม่นยำ โดยที่ไม่ต้องมติดัดแปลงส่วนควบคุมใดๆ เครื่องปรับอากาศ ทำให้สามารถเปิดและปิดเครื่องปรับอากาศได้โดยไม่ต้องใช้บุคคล หรือเครื่องรีโมทคอนโทรล ได้ทุกวันแม้เป็นวันหยุดทำการ

3. วิธีดำเนินการวิจัย

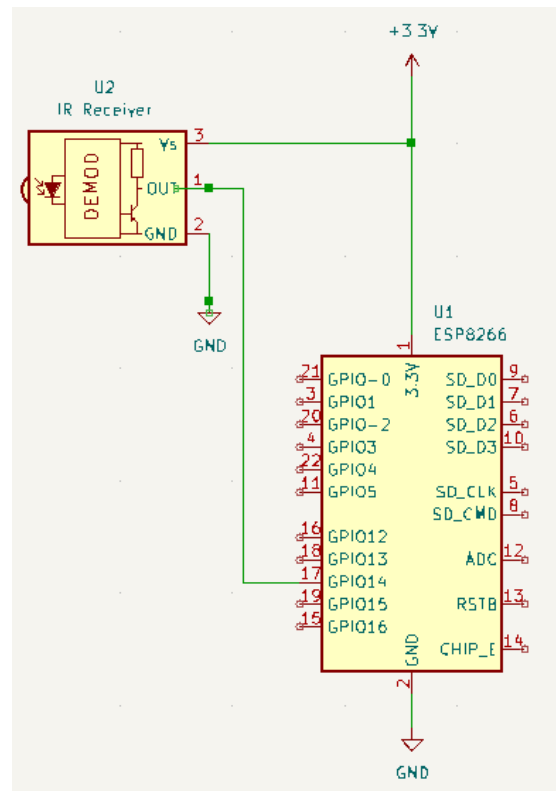
3.1 การเรียนรู้รูปแบบสัญญาณจากตัวรีโมทคอนโทรล

การวิจัยดำเนินการโดยใช้บอร์ดไมโครคอนโทรลเลอร์ ESP8266 ทำงานร่วมกับตัวรับสัญญาณอินฟราเรดความถี่ 38 kHz ดังบล็อกไดอะแกรมการทำงานดังรูปที่ 3.

และวงจรดังรูปที่ 4 เพื่ออ่านและถอดรหัสสัญญาณจากรีโมทคอนโทรลของเครื่องปรับอากาศ โดยใช้ไลบรารี IRremoteESP8266 [11] ช่วยในการประมวลผล ทดลองโดยใช้เครื่องปรับอากาศ SAMSUNG แบบติดผนัง ขนาด 12,000 BTU รุ่น AR13MRFN ในการทดสอบเรียนรู้รูปแบบสัญญาณจากตัวรีโมทคอนโทรล



รูปที่ 3 บล็อกไดอะแกรมการอ่านข้อมูลจากเครื่องรีโมทอินฟราเรด

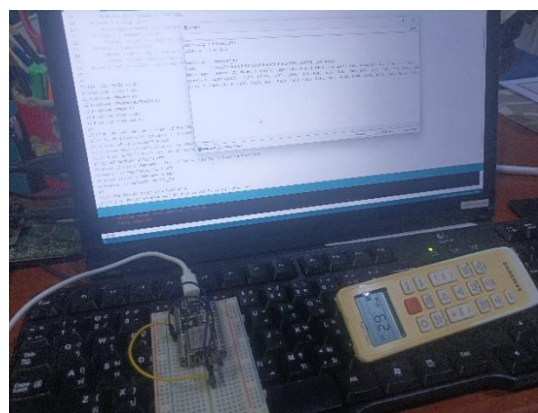


รูปที่ 4 วงจรเรียนรู้รูปแบบสัญญาณของตัวรีโมทคอนโทรล

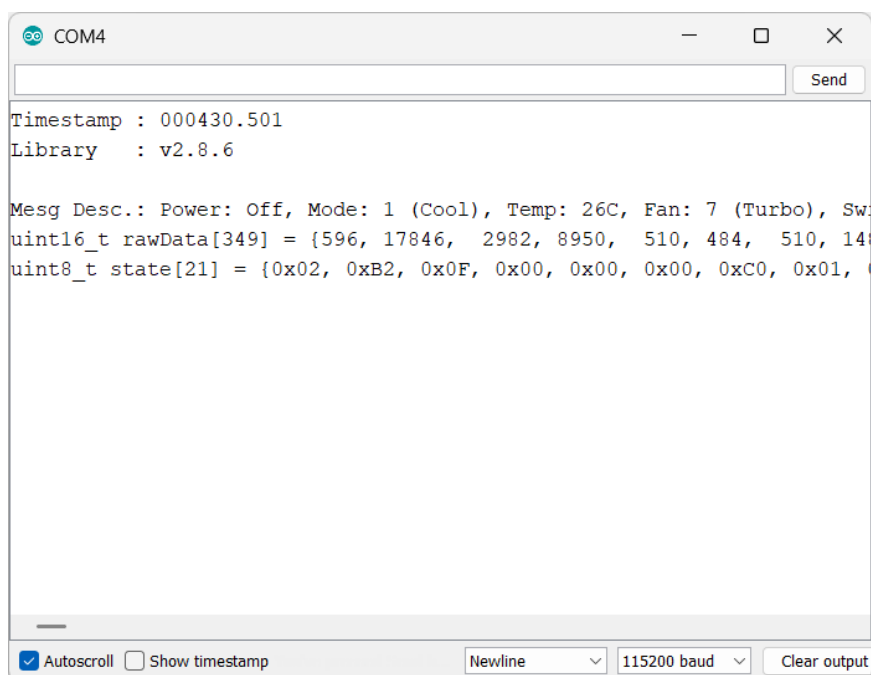
สัญญาณจากตัวรีโมทคอนโทรลเครื่องปรับอากาศจะส่งสัญญาณด้วยแสงอินฟราเรดโดยส่งข้อมูลแบบดิจิทัลที่กล่าสัญญาณด้วยความถี่ 38 kHz [12] โดยข้อมูลที่ส่งออกมาจะมีจำนวนข้อมูลดิบ (RAW Data) ที่มีจำนวนข้อมูลมาก และซับซ้อน [13] – [15] ไม่เหมือนกับข้อมูลดิบจาก

ตัวรีโมทคอนโทรลของอุปกรณ์ไฟฟ้าอื่นๆ เช่น เครื่องรับโทรทัศน์ที่ควบคุมการทำงานด้วยตัวรีโมทคอนโทรลโดยข้อมูลดิบจะมีจำนวนข้อมูลไม่มากไม่ซับซ้อน และสิ่งแตกต่างอีกอย่างของสัญญาณจากตัวรีโมทคอนโทรลคือ เครื่องปรับอากาศคือ ข้อมูลดิจิทัลที่ได้ในส่วนรายละเอียดของข้อมูลจะไม่เป็นมาตรฐานเดียวกันทุกผู้ผลิต โดยแต่ละผู้ผลิตก็จะมีรายละเอียดข้อมูลเฉพาะของตนเอง ดังนั้นวิธีการที่จะทราบข้อมูลของสัญญาณจากตัวรีโมทคอนโทรลจึงจะต้องใช้วิธีการเก็บข้อมูลโดยการอ่านค่าโดยตรงจากตัวรีโมทคอนโทรลด้วยการใช้ไมโครคอนโทรลเลอร์ดังรูปที่ 5 โดยนำตัวรีโมทคอนโทรลเครื่องปรับอากาศที่ต้องการจะอ่านข้อมูลมาเลือกกดปุ่มที่ต้องการทราบข้อมูลโดยการหันตัวรีโมทคอนโทรลไปที่ตัวตรวจรู้ การรับสัญญาณอินฟราเรดของไมโครคอนโทรลเลอร์ เมื่อไมโครคอนโทรลเลอร์ได้รับสัญญาณอินฟราเรดก็จะถอดรหัสให้เป็นข้อมูลดิบ ดังรูปที่ 6 โดยข้อมูลสัญญาณจากตัวรีโมทคอนโทรลจากรูปที่ 6 เป็นข้อมูลสัญญาณจากตัวรีโมทคอนโทรลเครื่องปรับอากาศซัมซุง เมื่อกดปุ่มเปิดให้เครื่องปรับอากาศทำงานที่อุณหภูมิ 25 °C ความเร็วลมระดับสูงสุด อุปกรณ์ควบคุม

ทิศทางการกระจายลมทำงาน โดยการกดปุ่มหนึ่งครั้งจะมีจำนวนข้อมูลที่ส่งออกมาเท่ากับ 349 ข้อมูล ทุกครั้งที่ทำการกดปุ่มตัวรีโมทคอนโทรลของเครื่องปรับอากาศ ค่าต่างๆที่แสดงผลบนหน้าจอของตัวรีโมทคอนโทรลจะถูกแปลงเป็นข้อมูลดิจิทัล ส่งออกไปด้วยสัญญาณแสงอินฟราเรด เช่น ให้เครื่องปรับอากาศทำงานทำงานในโหมดทำความเย็น ที่อุณหภูมิ 25 ความเร็วลมระดับสูงสุด อุปกรณ์ควบคุมทิศทางการกระจายลมทำงานทั้งแนวนอนและแนวตั้ง เป็นต้น



รูปที่ 5 การอ่านข้อมูลโดยตรงจากตัวรีโมท

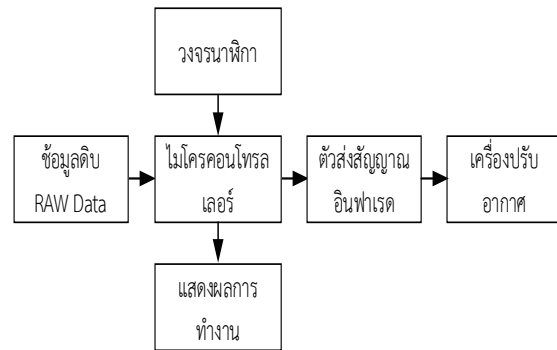


รูปที่ 6 ข้อมูลสัญญาณจากตัวรีโมทคอนโทรลเครื่องปรับอากาศ SAMSUNG

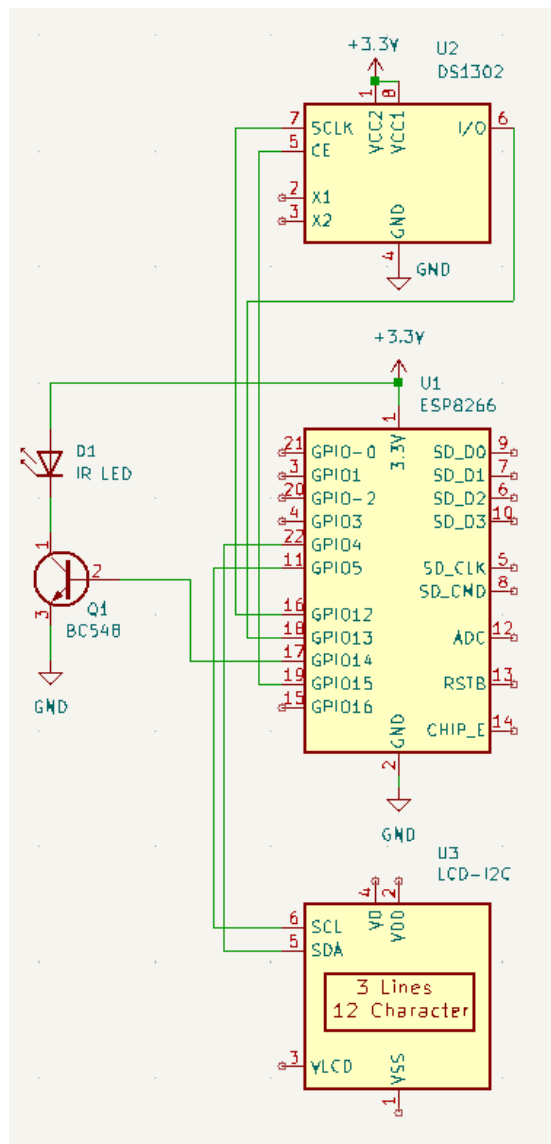
ข้อมูลตามรูปที่ 6 จะพบว่าจำนวนข้อมูลที่ส่งออกมาจะมีจำนวนมากถึง 349 ข้อมูลนั้น ก็เนื่องมาจากทุกครั้งที่มีการกดปุ่มที่ตัวรีโมทคอนโทรลของเครื่องปรับอากาศ ข้อมูลที่ส่งออกมาจะรวมการกำหนดค่าทุกอย่างตามที่แสดงอยู่บนหน้าจอแอลซีดีของตัวรีโมทคอนโทรล ดังนั้นเมื่อได้ข้อมูลดิบนี้มาก็จะนำไปใช้เป็นชุดข้อมูลในการเขียนโปรแกรมสั่งการทำงานให้เครื่องปรับอากาศเปิดทำงาน โดยใช้ไมโครคอนโทรลเลอร์ที่เขียนโปรแกรมให้ทำหน้าที่แทนตัวรีโมทคอนโทรลของเครื่องปรับอากาศ ทำให้สามารถสั่งงานเครื่องปรับอากาศให้ทำงานตามข้อมูลดิบที่ได้อ่านค่าเก็บไว้

3.2 การสร้างอุปกรณ์สำหรับเปิด-ปิดการทำงานของเครื่องปรับอากาศ

หลังจากได้ข้อมูลสัญญาณตัวรีโมทคอนโทรลที่ต้องการใช้เพื่อควบคุมการทำงานของเครื่องปรับอากาศ ขั้นตอนต่อไปเป็นการจำลองสัญญาณควบคุมจากรีโมทคอนโทรล โดยข้อมูลสัญญาณตัวรีโมทคอนโทรลจะนำไปใช้ในโปรแกรมส่งสัญญาณรีโมทคอนโทรลด้วยบอร์ดไมโครคอนโทรลเลอร์ ESP8266 ที่เขียนโปรแกรมการทำงานให้ส่งข้อมูลโดยใช้ LED อินฟราเรดที่ส่งสัญญาณแสงในย่านความถี่ 38kHz และโปรแกรมจะถูกเขียนให้เชื่อมต่อกับแผงวงจรนาฬิกาเวลาจริง เนื่องจากความต้องการให้เครื่องปรับอากาศเปิดทำงาน และปิดทำงานตามเวลาที่กำหนด แต่เนื่องจากสถานที่ใช้งานจริงมีข้อจำกัดในการใช้งานอินเทอร์เน็ตจึงไม่สามารถออกแบบให้ใช้เวลานาฬิกาจริงได้จากการใช้ไทม์เซิร์ฟเวอร์ (NTP Sever) ผ่านเครือข่ายอินเทอร์เน็ตได้ แผงวงจรนาฬิกาเวลาจริงจึงเป็นตัวเลือกในการใช้งาน และเพิ่มการแสดงผลการทำงานด้วยจอ LCD ขนาดเล็ก เพื่อแสดงให้เห็นว่าวงจรยังคงทำงานอยู่ ดังบล็อกไดอะแกรมการทำงานและวงจรการทำงานทั้งหมดแสดงตามรูปที่ 7 และรูปที่ 8 ตามลำดับ



รูปที่ 7 บล็อกไดอะแกรมการใช้ไมโครคอนโทรลเลอร์เพื่อส่งข้อมูลให้กับเครื่องปรับอากาศแทนเครื่องรีโมท



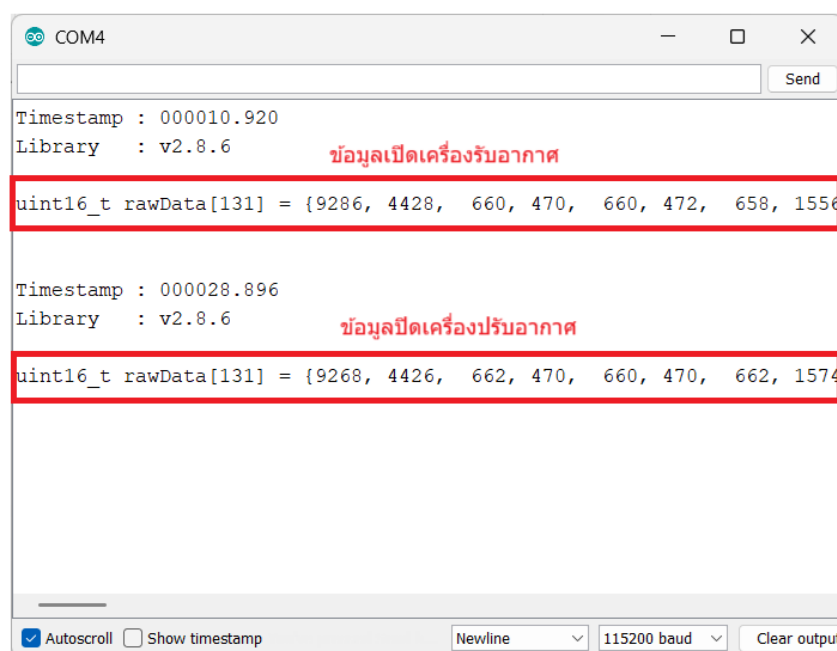
รูปที่ 8 วงจรส่งสัญญาณควบคุมแทนตัวรีโมทคอนโทรล

4. ผลการวิจัย

ขั้นตอนแรกผู้วิจัยได้ทำการเรียนรู้รูปแบบสัญญาณจากตัวรีโมทคอนโทรลด้วยการเขียนโปรแกรมที่ติดตั้งในไมโครคอนโทรลเลอร์ ESP8266 ตามวงจรในรูปที่ 4 อ่านข้อมูลออกมาจำนวน 2 ชุดคือ ข้อมูลชุดที่ 1. เปิดเครื่องปรับอากาศในโหมดทำความเย็นที่อุณหภูมิ 25 องศาเซลเซียส ความเร็วพัดลมส่งลมเย็นที่ความเร็วลมระดับสูงสุด ข้อมูลชุดที่ 2. ปิดเครื่องปรับอากาศในโหมดทำความเย็นที่อุณหภูมิ 25 องศาเซลเซียส พัดลมส่งลมเย็นที่ความเร็วลมระดับสูงสุด ซึ่งผลของเรียนรู้รูปแบบสัญญาณจากตัวรีโมทคอนโทรลเป็นดังรูปที่ 9 ต่อจากนั้นผู้วิจัยได้นำชุดข้อมูลสำหรับเปิด-ปิด เครื่องปรับอากาศที่ได้ในขั้นตอนแรกมาเป็นส่วนหนึ่งในโปรแกรมที่ติดตั้งในไมโครคอนโทรลเลอร์ ESP8266 ตามโปรแกรมรูปที่ 10 และสร้างเป็นอุปกรณ์เปิด-ปิดเครื่องปรับอากาศด้วยไมโครคอนโทรลเลอร์ตามรูปที่ 11 จากนั้นนำไปวางไว้ในห้องจัดเก็บเครื่องมือวัดทางไฟฟ้าดังรูปที่ 12 โดยจัดวางให้ทิศทางการส่งสัญญาณอินฟราเรดให้ตรงกับตัวรับสัญญาณของเครื่องปรับอากาศ โดยโปรแกรมที่จัดทำขึ้นกำหนดให้เปิดการทำงานเครื่องปรับอากาศในเวลา 10

นาฬิกา โหมดทำความเย็นที่อุณหภูมิ 25 องศาเซลเซียส พัดลมส่งลมเย็นที่ระดับความแรงลมระดับสูงสุด และปิดการทำงานที่เวลา 16 นาฬิกาของทุกวัน

ในการทดสอบผู้วิจัยได้ใช้วิธีการเก็บค่าอุณหภูมิภายในห้องด้วยอุปกรณ์จัดเก็บค่าอุณหภูมิแบบอัตโนมัติ (Temperature Data Logger) จับเก็บข้อมูลในรูปแบบไฟล์ CSV จัดเก็บข้อมูลทุก 5 วินาที ข้อมูลที่จัดเก็บใช้เพื่อตรวจสอบว่าชุดอุปกรณ์ที่ได้จัดทำสามารถทำงานได้ตามวัตถุประสงค์หรือไม่เนื่องจากไม่มีบุคลากรคอยตรวจสอบการทำงานได้ทุกวัน โดยถ้าในช่วงเวลาที่กำหนดให้เครื่องปรับอากาศทำงานคือ 10 นาฬิกา ถึง 16 นาฬิกา ถ้าอุณหภูมิภายในห้องลดลงนั้นแสดงว่าอุปกรณ์ที่จัดทำสามารถสั่งการให้เครื่องปรับอากาศทำงานได้ตามวัตถุประสงค์ จากการเก็บข้อมูล 15 วันต่อเนื่อง ดังรูปที่ 13. เมื่อนำข้อมูลอุณหภูมิที่จัดเก็บมาตรวจสอบพบว่าอุณหภูมิภายในห้องจะลดลงในช่วงเวลาที่สั่งการให้เครื่องปรับอากาศทำงาน ซึ่งจะลดลงต่อเนื่องจนถึงค่าที่กำหนดไว้ และจะเริ่มกลับมามีอุณหภูมิสูงขึ้นหลังจากเลยช่วงเวลาที่สั่งการให้เครื่องปรับอากาศหยุดทำงาน



```
COM4
Timestamp : 000010.920
Library : v2.8.6
ข้อมูลเปิดเครื่องปรับอากาศ
uint16_t rawData[131] = {9286, 4428, 660, 470, 660, 472, 658, 1556

Timestamp : 000028.896
Library : v2.8.6
ข้อมูลปิดเครื่องปรับอากาศ
uint16_t rawData[131] = {9268, 4426, 662, 470, 660, 470, 662, 1574

Autoscroll Show timestamp Newline 115200 baud Clear output
```

รูปที่ 9 รูปแบบข้อมูลจากตัวรีโมทคอนโทรล

```
#include DS1302.h

// กำหนดขาต่อ I2C กับจอ OLED
#define OLED_RESET -1 //ขา reset เป็น -1 ถ้าใช้ร่วมกับขา Arduino reset
Adafruit_SSD1306 OLED(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET);

//กำหนดขาต่อกับ DS1302 rtc([CE/RST], [I/O], [CLOCK]);
DS1302 rtc(15, 13, 12);
Time t;

const uint16_t kIrLed = 14; // Pin ที่ต่อหลอด Infrared แฉก pin : 14 (D5), pin : 4 (D2)
IRsend irsend(kIrLed); // Lib ณ IR

//Array ข้อมูลสัญญาณ IR ที่บันทึกได้จาก IRrecvDumpV2
uint16_t ac_on[131] = {9272, 4440, 638, 494, 636, 494, 660, 1574, 636, 494,
638, 1598, 636, 494, 662, 468, 636, 494, 636, 494, 636, 1598, 636, 494,
636, 1598, 636, 1598, 638, 494, 636, 1598, 636, 494, 636, 492, 638, 494,
636, 494, 636, 494, 660, 470, 636, 494, 636, 494, 636, 494, 662, 468,
636, 492, 638, 494, 636, 494, 636, 494, 662, 468, 636, 494, 636, 494,
638, 492, 636, 494, 636, 494, 636, 494, 636, 494, 636, 494, 660, 470,
636, 494, 660, 470, 636, 1600, 636, 1598, 638, 494, 636, 1598, 636, 494,
636, 494, 636, 1600, 636, 494, 638, 492, 638, 492, 638, 494, 636, 494,
636, 494, 636, 1598, 638, 494, 636, 494, 662, 468, 636, 494, 636, 494,
636, 494, 638, 1598, 638, 492, 638, 1598, 636}; // UNKNOWN 5EBC98B1
uint16_t ac_off[131] = {9266, 4444, 734, 396, 734, 396, 656, 1578, 658,
474, 656, 474, 656, 474, 734, 396, 656, 474, 732, 398, 656, 1578, 658,
472, 734, 1504, 732, 1502, 658, 472, 734, 1502, 656, 472, 658, 474, 732,
396, 658, 472, 734, 396, 658, 474, 734, 396, 734, 396, 658, 474, 656,
474, 658, 472, 658, 472, 658, 472, 658, 474, 656, 472, 734, 396, 656,
474, 658, 474, 656, 472, 658, 474, 732, 398, 734, 396, 734, 396, 734,
396, 734, 396, 658, 1578, 678, 1558, 656, 1578, 658, 472, 658, 1578,
734, 398, 656, 474, 734, 1500, 734, 396, 734, 396, 658, 472, 658, 474,
732, 398, 656, 474, 656, 1580, 656, 472, 658, 474, 734, 396, 658, 472,
658, 474, 656, 472, 734, 1502, 658, 474, 734, 1500, 658}; // UNKNOWN
2B373187

void setup() {
  irsend.begin();
  //กำหนดการทำงานในโหมด run, และยกเลิกการป้องกันการเขียนข้อมูล
  rtc.halt(false);
  rtc.writeProtect(false);
  Serial.begin(9600);
}
```

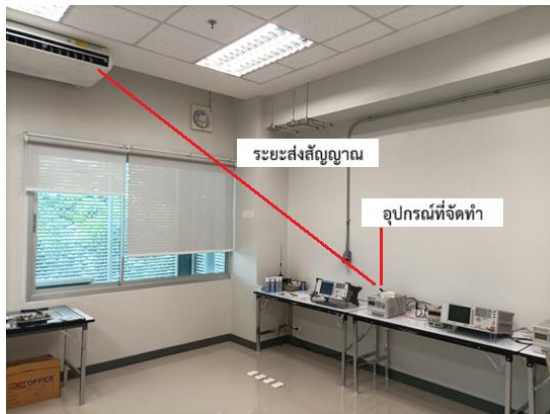
รูปที่ 10 โปรแกรมสั่งการเครื่องปรับอากาศ

เมื่อนำข้อมูลมาพล็อตกราฟ พบว่ากราฟอุณหภูมิภายในห้องจะเริ่มลดลงในช่วงเวลาที่สั่งการให้เครื่องปรับอากาศทำงาน และกราฟอุณหภูมิภายในห้องจะเริ่มเพิ่มขึ้นหลังช่วงเวลาที่สั่งการให้เครื่องปรับอากาศหยุดทำงานครบทั้ง 15 วันที่จัดเก็บข้อมูลดังรูปที่ 14 อุปกรณ์ที่จัดทำสามารถทำงานได้ตามวัตถุประสงค์ หลังจากนั้นผู้วิจัยจึงได้จัดทำชิ้นงานเพิ่มเติมโดยนำไปใช้ในห้องปฏิบัติการที่ใช้เครื่องปรับอากาศ TAZAKI แบบแขวนได้ฟ้า ขนาด 40,000 BTU รุ่น FULE40BI-AF1 ดังรูปที่ 15 เพื่อทดสอบซ้ำตามหลักการเดิมคือ การเรียนรู้รูปแบบสัญญาณจากตัวรีโมทคอนโทรล และนำชุดข้อมูลสำหรับเปิด-ปิด เครื่องปรับอากาศที่ได้จากการเรียนรู้มาเป็นส่วนหนึ่งในโปรแกรมที่ติดตั้งในไมโครคอนโทรลเลอร์ ESP8266 แล้วจัดสร้างเป็นอุปกรณ์เปิด-ปิด

เครื่องปรับอากาศด้วยไมโครคอนโทรลเลอร์ชิ้นที่ 2 โดยผลการทำงานที่ได้ก็เป็นตามวัตถุประสงค์เช่นกัน



รูปที่ 11 อุปกรณ์ต้นแบบ



รูปที่ 12 การทดสอบเครื่องต้นแบบแทนรีโมทคอนโทรล

File

Home

Insert

Draw

Page Layout

Formulas

Data

Review

View



Paste



Cut



Copy



Format Painter

Clipboard

Font

Calibri

11

A⁺

A⁻































E108

:

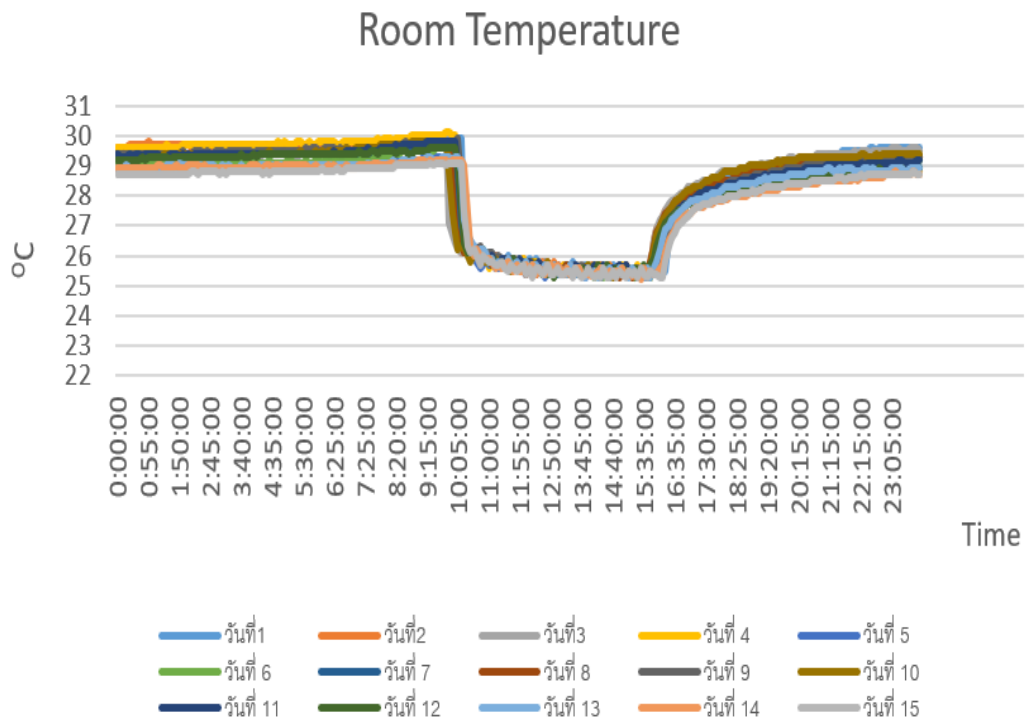






	A	B	C	D	E	F
1	Temperature *C	Time & Date				
102	29.2	1/6/2026 9:40				
103	29.2	1/6/2026 9:45				
104	29.2	1/6/2026 9:50				
105	29.2	1/6/2026 10:00				
106	29.3	1/6/2026 10:05				
107	29.3	1/6/2026 10:15				
108	29.3	1/6/2026 10:20				
109	27.8	1/6/2026 10:25				
110	26.8	1/6/2026 10:30				
111	26.5	1/6/2026 10:35				
112	26.3	1/6/2026 10:40				
113	26	1/6/2026 10:50				

รูปที่ 13. ข้อมูลอุณหภูมิที่จัดเก็บ



รูปที่ 9 ผลการทดสอบการปรับอุณหภูมิในการสั่งงานเครื่องปรับอากาศ

5. สรุป

จากการเก็บข้อมูล 15 วันต่อเนื่อง พบว่ากราฟอุณหภูมิภายในห้องจะเริ่มลดลงหลังจากสั่งการให้เปิดเครื่องปรับอากาศ และอุณหภูมิภายในห้องจะเริ่มเพิ่มขึ้นหลังจากสั่งการให้ปิดเครื่องปรับอากาศดังรูปที่ 14 ซึ่งแสดงให้เห็นว่า ระบบสามารถเปิดและปิด

เครื่องปรับอากาศได้ตามเวลาที่กำหนดโดยอัตโนมัติ จึงไม่จำเป็นต้องมีผู้ปฏิบัติงานคอยเปิดหรือปิดเครื่องทุกวัน ช่วยลดภาระงานและลดความผิดพลาดจากการลืมเปิดหรือปิดเครื่องเป็นการลดการพึ่งพาการควบคุมโดยมนุษย์ วิธีการควบคุมโดยใช้วิธีจำลองสัญญาณอินฟราเรดจากรีโมทคอนโทรลเดิม จึงไม่จำเป็นต้องแก้ไขวงจรภายใน

เครื่องปรับอากาศ ทำให้ไม่กระทบต่อการรับประกันหรือการทำงานเดิมของเครื่อง ระบบในช่วงจรรยาภิภาเวลางจริงสำหรับกำหนดเวลาจึงสามารถทำงานได้อย่างอิสระ ไม่ต้องเชื่อมต่อเครือข่ายอินเทอร์เน็ต ผลการทดสอบตลอด 15 วันพบว่าสามารถสั่งงานเครื่องปรับอากาศได้ถูกต้องทุกครั้ง คิดเป็นอัตราความสำเร็จร้อยละ 100 แสดงถึงความน่าเชื่อถือของระบบมีความแม่นยำและเชื่อถือได้สูง ข้อมูลอุณหภูมิแสดงให้เห็นว่าอุณหภูมิภายในห้องลดลงอย่างต่อเนื่องในช่วงเวลาที่ระบบสั่งให้เครื่องปรับอากาศทำงาน ช่วยรักษาสภาพแวดล้อมที่เหมาะสมสำหรับอุปกรณ์หรือเครื่องมือที่ไวต่อความร้อน การควบคุมอุณหภูมิให้เหมาะสมช่วยลดการเสื่อมสภาพของจอ LCD และอุปกรณ์วัดทางไฟฟ้าที่เก็บอยู่ในห้อง ทำให้อายุการใช้งาน และลดค่าใช้จ่ายด้านการซ่อมบำรุง ระบบใช้ไมโครคอนโทรลเลอร์ ESP8266 และอุปกรณ์อิเล็กทรอนิกส์พื้นฐาน ทำให้ต้นทุนการพัฒนาไม่สูง และสามารถสร้างหรือปรับปรุงเพิ่มเติมได้ง่าย วิธีการนี้สามารถประยุกต์ใช้กับเครื่องปรับอากาศหลายยี่ห้อที่สามารถอ่านและบันทึกสัญญาณรีโมทคอนโทรลของเครื่องปรับอากาศรุ่นนั้นได้ ระบบก็สามารถนำไปประยุกต์ใช้กับเครื่องปรับอากาศยี่ห้อและรุ่นอื่นที่ควบคุมด้วยสัญญาณอินฟราเรดได้เช่นกัน ในอนาคตการพัฒนาต่อไปคือ ให้ระบบสามารถเลือกโหมดการทำงานระหว่างการทดสอบสัญญาณและการควบคุมตามเวลาที่กำหนดได้อย่างยืดหยุ่นมากยิ่งขึ้น

6. เอกสารอ้างอิง

- [1] H. Du, "Study of air conditioning remote control based on embedded system," in *Proc. 2011 IEEE 3rd Int. Conf. Commun. Softw. Networks (ICCSN)*, Xi'an, China, 2011, pp. 550–553.
- [2] W. Song, N. Feng, Y. Tian and S. Fong, "An IoT-Based Smart Controlling System of Air Conditioner for High Energy Efficiency" in *Proc. 2017 IEEE Int. Conf. on Internet of Things (iThings) and IEEE Green Comp. and Commun. (GreenCom) and IEEE Cyber, Phys. and Social Comp. (CPSCoM) and IEEE Smart Data (SmartData)*, Exeter, UK, 2017, pp. 442–449.
- [3] B. E. Medina and L. T. Manera, "Retrofit of air conditioning systems through an Wireless Sensor and Actuator Network An IoT-based application for smart buildings", in *Proc 2017 IEEE 14th Int. Conf. on Networking, Sensing and Control (ICNSC)*, Calabria, Italy, 2017, pp. 49–53.
- [4] M. D. Permata, A. N. N. Chamim and R. O. Wiyagi, "Web-based air conditioner controller," *Journal of Applied Social and Informatics Science*, vol. 1, pp. 28–35, 2022.
- [5] Z. Shi, J. Chu, L. Geng and P. Zheng and D. Liu, "Design of Smart Home Control System Based on Embedded System," in *Proc. 2020 IEEE 9th Joint International Information Technology and Artificial Intelligence Conference (ITAIC)*, Chongqing, China, 2020, pp. 2209–2213.
- [6] Y. Lian, H. Chen and X. Li, "Infrared remote control protocol analysis and implementation," *Int. J. Control Autom.*, vol. 8, no. 5, pp. 125–132, 2015.
- [7] D. Depau. (2021). Reverse-engineering an AC IR remote protocol [Online]. Available: <https://blog.depau.eu>
- [8] A. Kumar, R. Singh and P. Sharma, "IoT-based smart air conditioner control system," *Int. J. Eng. Res. Technol.*, vol. 9, no. 6, pp. 45–50, Jun. 2020.

- [9] K. Lee and H. Kim, “Energy-efficient HVAC control system using automation techniques,” *Energy and Buildings*, vol. 158, pp. 1–10, 2018.
- [10] S. Ahmed, M. Rahman, and N. Islam, “Smart energy management system for buildings,” *Sustain. Comput. Inform. Syst.*, vol. 22, pp. 187–197, Jun. 2019.
- [11] D. Conran. (2023). IRremoteESP8266 Library [Online]. Available: github.com
- [12] Vishay Semiconductors. (2024). Data Formats for IR Remote Control (Document No. 80071) [Online]. Available: [vishay.com](https://www.vishay.com)
- [13] Ch. Arunprakash and K.R.K. Sastry, “Implementation of Air Conditioner Infrared Protocol on a Reconfigurable Hardware”, in *Proc. 2018 8th Int. Conf. on Cloud Computing, Data Science & Engineering (Confluence)*, Noida, India, pp. 787-790.
- [14] S. Bhawal. (2019). Understanding IR Protocol of Air Conditioner’s Remotes [Online]. Available: [instructables.com](https://www.instructables.com)
- [15] H. Bhosale, (2024). Understanding IR Protocols and decoding the IR codes for Bluestar AC [Online]. Available: medium.com