

ระบบเฝ้าระวังอุทกภัยผ่านเซ็นเซอร์ตรวจวัดระดับความสูงของน้ำ กรณีศึกษา เขตเทศบาลนครศรีธรรมราช

ณัฐพล เวฬุบรรพ, อภิสิทธิ์ สกูลส่องบุญศิริ และ สุพัต รุ่งเรืองศิลป์*

หลักสูตรเทคโนโลยีสารสนเทศ สำนักวิชาสารสนเทศศาสตร์ มหาวิทยาลัยวลัยลักษณ์

อำเภอท่าศาลา จังหวัดนครศรีธรรมราช 80161

*suppat.ru@mail.wu.ac.th

บทคัดย่อ

ระบบเฝ้าระวังอุทกภัยผ่านเซ็นเซอร์ตรวจวัดระดับความสูงของน้ำ กรณีศึกษาเขตเทศบาลนครศรีธรรมราช พัฒนาขึ้นเพื่อเป็นต้นแบบที่ทางทีมวิจัยพัฒนาแทนการซื้ออุปกรณ์จากต่างประเทศซึ่งมีราคาสูง ช่วยป้องกันความเสี่ยงที่จะสร้างความเสียหายจากอุทกภัยที่อาจเกิดขึ้นแก่เจ้าหน้าที่ในการลงพื้นที่ปฏิบัติงานและประชาชนที่จะได้รับความเดือดร้อนจากภัยน้ำท่วม โดยกระบวนการหลักในการทำงานคือเพื่อติดตามระดับน้ำ บริเวณพื้นที่ต้นน้ำที่สำคัญภายในจังหวัดนครศรีธรรมราช ซึ่งจะใช้กรณีศึกษาจากเขตเทศบาลนครศรีธรรมราช เพื่อให้การดูแลและป้องกันประชาชนอย่างทั่วถึงในกรณีที่เกิดเหตุไม่คาดคิด ทั้งยังมีระบบการแจ้งเตือนด้วยสัญญาณไฟเพื่อเป็นจุดสังเกตแก่ประชาชน สามารถเฝ้าติดตามระดับน้ำผ่านเว็บไซต์ นอกจากนี้ระบบยังสามารถออกรายงานประวัติข้อมูลระดับน้ำในการตรวจสอบย้อนหลังเพื่อใช้ในการประเมินสถานการณ์ได้ การประยุกต์ใช้ระบบเฝ้าระวังอุทกภัยผ่านเซ็นเซอร์ตรวจวัดระดับความสูงของน้ำ โดยใช้อุปกรณ์ Arduino ในการจัดการข้อมูลที่ได้รับจากเซ็นเซอร์ที่ประกอบขึ้นเพื่อตรวจวัดระดับน้ำ ใช้ Raspberry Pi ในการประมวลผลเพื่อส่งค่าไปยังโฮสต์ที่อยู่บนคลาวด์ซึ่งใช้แสดงผลออกมาบนหน้าเว็บไซต์ โดยการพัฒนาระบบภายในงานวิจัยนี้จะเป็นการพัฒนาแบบต้นแบบและสามารถนำไปพัฒนาต่อยอดให้เกิดประโยชน์ต่อไป

คำสำคัญ: RS485 Firebase Internet of Things

Abstract

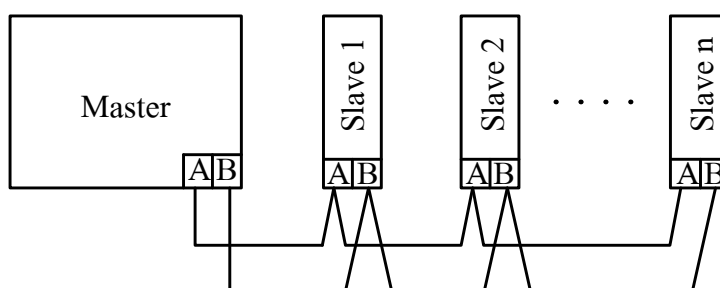
Early warning water level system a case study of Nakhon Si Thammarat city municipality is a prototype that making instead of the costly overseas product. It supports the government officers who check areas at risk of flooding and people who flooding brings massive problems. The main responsibility of officers is water level monitoring in the source or headwaters of a river or stream of Nakhon Si Thammarat. We use a case study of Nakhon Si Thammarat city municipality for build a prototype to look after and protect the risk for people in this area. Moreover, we build 3 warning led lights to be notice for people and this system can monitor water level on website. Furthermore, the system can generate the water level history report to check back and estimate the incident. Developed early warning water level system with an analog water pressure sensor is integrated with Arduino controller board (or slave). The analog water pressure sensor creates the water level (or output). The Raspberry Pi controller board is the Master to process data from Arduino controller board before sending the data to Firebase hosting (or data visualization on the website). We develop the system prototype and can further expand to make a benefit in the future.

Keywords: RS485, Firebase, Internet of Things

บทนำ

ในปัจจุบันภาคอุตสาหกรรมการใช้ PLC หรือ Programmable Logic Controller Circuit เป็นที่แพร่หลาย(R. Joshi et al., 2016) เนื่องจากมีความเสถียร แต่ยังมีจำกัดหลายอย่าง เช่น PLC มีราคาสูง ต้องเสียค่าใช้จ่ายสำหรับซอฟต์แวร์เพื่อใช้ในการเขียนโปรแกรม PLC และหากต้องการใช้งานในรูปแบบพิเศษบางอย่าง เช่น บันทึกข้อมูลในรูปแบบของ Data logger, การใช้งานอินเทอร์เน็ตของสรรพสิ่ง (IoT) จะต้องทำการเพิ่มอุปกรณ์เสริม ซึ่งทำให้มีค่าใช้จ่ายเพิ่มขึ้นมาในระบบ ทางทีมวิจัยจึงได้มีแนวคิดที่จะประยุกต์ใช้ Raspberry Pi มาใช้งานทดแทนซึ่งจะมีข้อได้เปรียบอย่างชัดเจนในเรื่องการใช้งานอินเทอร์เน็ตของสรรพสิ่ง

ทางทีมวิจัยจึงได้ใช้กรณีศึกษาของเทศบาลเมืองนครศรีธรรมราช จากปัญหาน้ำท่วมในปลายปี พ.ศ. 2559 ที่ผ่านมา ซึ่งจากการสอบถามเจ้าหน้าที่บริหารงานทั่วไป เทศบาลนครศรีธรรมราช พบว่าการดูแลและการจัดการระบบน้ำใน อำเภอเมือง จังหวัด นครศรีธรรมราช จะแบ่งการดูแลเป็น 2 หน่วยงานราชการ คือ เทศบาลนครศรีธรรมราชและกรมชลประทาน ซึ่งจะใช้เจ้าหน้าที่ในการตรวจวัดระดับน้ำจากซีตระดับวัดปริมาณน้ำ ในขณะที่กรมชลประทานใช้กล้อง CCTV ในการตรวจสอบระดับน้ำ ไม่ได้เป็นระบบแจ้งเตือนแบบอัตโนมัติ จึงเป็นที่มาในการกำหนดปัญหาของงานวิจัยนี้ โดยประยุกต์ใช้ Raspberry Pi มาใช้ในการอ่านค่าระดับน้ำแล้วแสดงผลพร้อมแจ้งเตือนผ่านระบบอินเทอร์เน็ตของสรรพสิ่ง โดยเลือกใช้เซ็นเซอร์วัดระดับน้ำ (Nugraheni & Vries, 2017) แต่ในการส่งค่าข้อมูลออกจากเซ็นเซอร์ลักษณะข้อมูลเป็นแบบอนาล็อกซึ่ง Raspberry Pi ไม่สามารถอ่านค่าได้โดยตรงจึงต้องทำการเพิ่มไมโครคอนโทรลเลอร์ (Arduino) เข้าไปเพื่อรับค่าอนาล็อกแล้วจึงทำการเก็บค่าแล้วส่งค่าผ่านพอร์ตสื่อสารให้กับ Raspberry Pi แต่เนื่องจากตำแหน่งติดตั้งเซ็นเซอร์ชุดเซ็นเซอร์วัดระดับน้ำ อยู่ในพื้นที่กลางแจ้งมีความชื้นสูงทำให้การติดตั้งบอร์ดต้องติดตั้งในตู้กันน้ำ แต่เนื่องจากอุณหภูมิแวดล้อมที่สูงส่งผลทำให้อุณหภูมิภายในตู้สูงขึ้นได้เกินกว่า 60 °C ไม่เหมาะสมที่จะติดตั้งบอร์ด Raspberry Pi ในบริเวณดังกล่าว การติดตั้งอุปกรณ์จึงต้องไปติดตั้งในอาคารซึ่งอยู่ห่างไปอีกหลายร้อยเมตร ทางทีมวิจัยได้จึงได้เลือกวิธีการส่งค่าโดยเลือกใช้การสื่อสารผ่านพอร์ต RS 485 ซึ่งสามารถส่งค่าผ่านสายไฟโดยมีความยาวสายสูงสุดได้ไกลถึง 1,200 เมตร และใช้จำนวนตัวนำเพียง 2 เส้น คือ A และ B โดยมีไดอะแกรมการต่อวงจรแสดงดังรูปที่ 1 โดยลักษณะการต่อสายแบบนี้เรียกว่า Daisy chain หรือ Straight line ซึ่งจากภาพจะเห็นได้ว่าการต่อสายสื่อสารแบบนี้จะต้องประกอบด้วย Master และ Slave โดยงานวิจัยนี้จะให้ Raspberry Pi เป็น Master และชุดเซ็นเซอร์วัดระดับน้ำเป็น Slave (Nishanth & John, 2017) และจากภาพจะเห็นได้ว่าการสื่อสารไม่ว่าจะมี Slave กี่ตัวก็จะใช้สายตัวนำที่เดินจาก Master เพียงแค่ 2 เส้นเท่านั้น ซึ่งจำนวน Slave สูงสุดที่สามารถต่อได้จะขึ้นอยู่กับโปรโตคอลที่ใช้ โดยงานวิจัยนี้เลือกใช้โปรโตคอล Modbus (Modicon Modbus Protocol Reference Guide, 1996) ซึ่งเป็นโปรโตคอลที่อุปกรณ์ (เช่น อินเวอร์เตอร์, บอร์ดควบคุมมอเตอร์) หรือ เซ็นเซอร์ (เช่น เซ็นเซอร์วัดอุณหภูมิ วัดแสง วัดพลังงานไฟฟ้า) ในอุตสาหกรรมส่วนใหญ่ที่เป็นพอร์ตสื่อสารแบบ RS 485 ใช้ จึงทำให้สามารถทำการเพิ่มอุปกรณ์หรือเซ็นเซอร์ในระบบได้ง่าย และโปรโตคอลนี้จะสามารถมี Slave ในระบบได้ 255 ตัว ด้วยเหตุนี้เองจึงเป็นเหตุผลที่ พอร์ตสื่อสาร RS 485 ที่ใช้โปรโตคอล MODBUS ถูกเลือกมาใช้งานวิจัย



รูปที่ 1 ไดอะแกรมการต่อพอร์ตสื่อสาร RS 485

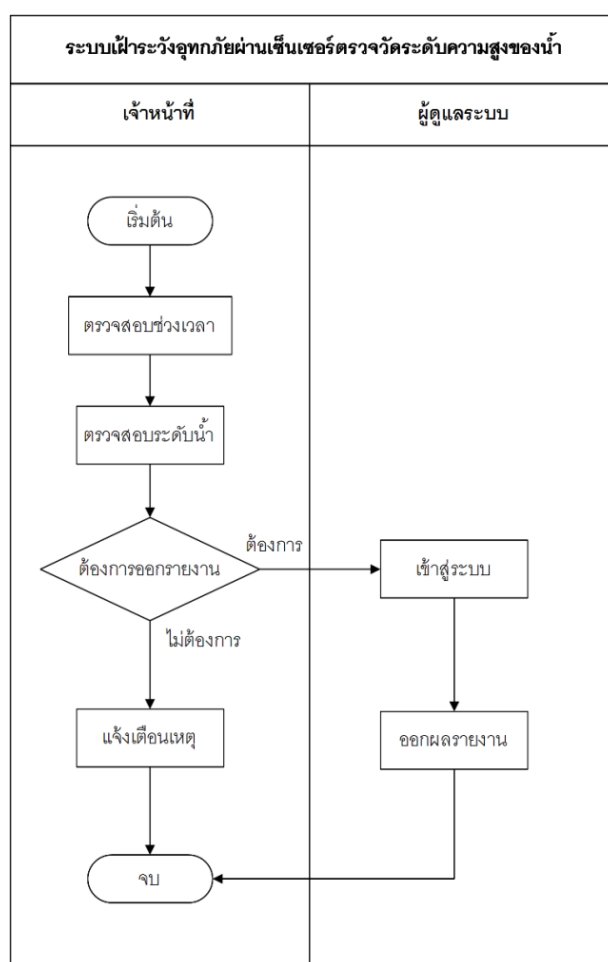
วิธีดำเนินการวิจัย

งานวิจัยนี้จัดทำขึ้นมาเป็นอุปกรณ์ต้นแบบเพื่อช่วยในการเฝ้าระวังและแจ้งเตือนระดับน้ำโดยใช้เซ็นเซอร์วัดระดับน้ำ ซึ่งเป็นต้นแบบรองรับสถานการณ์ของน้ำในช่วงที่เกิดอุทกภัยที่จะเกิดขึ้น โดยการใช้เซ็นเซอร์ส่งข้อมูลมาที่อุปกรณ์บอร์ด Raspberry Pi ผ่านโปรโตคอล Modbus มาใช้แทนวงจร PLC เพื่อมอนิเตอร์ข้อมูลน้ำเข้าโดยฟังก์ชันการทำงานมี ดังนี้

1. การรับข้อมูลน้ำเข้าจากอุปกรณ์เซ็นเซอร์ เป็นฟังก์ชันที่เก็บค่าข้อมูลของความสูงของระดับน้ำร่วมกับอุปกรณ์เซ็นเซอร์

2. การตรวจวัดระดับความสูงของน้ำในอุปกรณ์ต้นแบบเป็นฟังก์ชันในการคำนวณเพื่อหาความสัมพันธ์ของระดับน้ำและเกณฑ์การเฝ้าระวังเพื่อการเตือนภัยได้กำหนดให้มี 4 ระดับ ได้แก่ ปกติ (โดยกำหนดให้น้ำสูงน้อยกว่า 20 ซม.) เฝ้าระวัง (โดยกำหนดให้น้ำสูงเท่ากับ 20 ซม. แต่น้อยกว่า 30 ซม.) เตรียมความพร้อม (โดยกำหนดให้น้ำสูงเท่ากับ 30 ซม. แต่น้อยกว่า 40 ซม.) และ อพยพ (โดยกำหนดให้น้ำสูงตั้งแต่ 40 ซม.)
3. การแจ้งเตือนระดับน้ำหรือความสูงของน้ำ เป็นฟังก์ชันในการแจ้งเตือนระดับน้ำผ่านหลอดแอลอีดี 4 สถานะ คือ ไม่มีสีมีค่าเท่ากับสถานการณ์ปกติ สีเขียวมีค่าเท่ากับสถานการณ์เฝ้าระวัง สีเหลืองมีค่าเท่ากับสถานการณ์เตรียมความพร้อม และ สีแดงมีค่าเท่ากับสถานการณ์อพยพ เพื่อเฝ้าระวังและเตรียมตัวรับมือกับอุทกภัยที่จะเกิดขึ้นได้ทันเวลา
4. ออกรายงานเป็นฟังก์ชันที่ใช้ในการออกรายงานซึ่งสามารถตรวจสอบรายละเอียดของระดับน้ำที่ได้ในแต่ละช่วงเวลาที่ต้องการ โดยรายงานแสดงให้ทราบถึงสถานการณ์ของระดับน้ำ รายงานระดับความสูงของน้ำในแต่ละช่วงเวลาที่ต้องการ เป็นต้น

โดยทางทีมวิจัยได้ออกแบบแผนผังการทำงานของระบบแสดงดังรูปที่ 2 ดังนี้



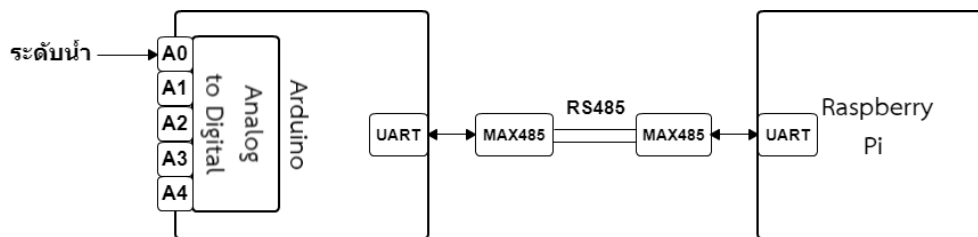
รูปที่ 2 กระบวนการทำงานของระบบ (Work Flow)

โดยใช้หลักการและเครื่องมือในการพัฒนา ดังต่อไปนี้

1. มาตรฐานในการสื่อสารข้อมูล (Data Communication Standards)

อุปกรณ์ Raspberry Pi นั้นไม่มีความสามารถในการรับค่าแบบอนาล็อก โดยตรง ซึ่งไม่เหมือนกับไมโครคอนโทรลเลอร์ เช่น Arduino, PIC, dsPIC, TMS, STM เป็นต้น ซึ่งรับค่าได้ทั้งแบบ ดิจิทัล และ อนาล็อก ดังนั้นหากต้องการนำข้อมูลแบบ อนาล็อก เข้าไป

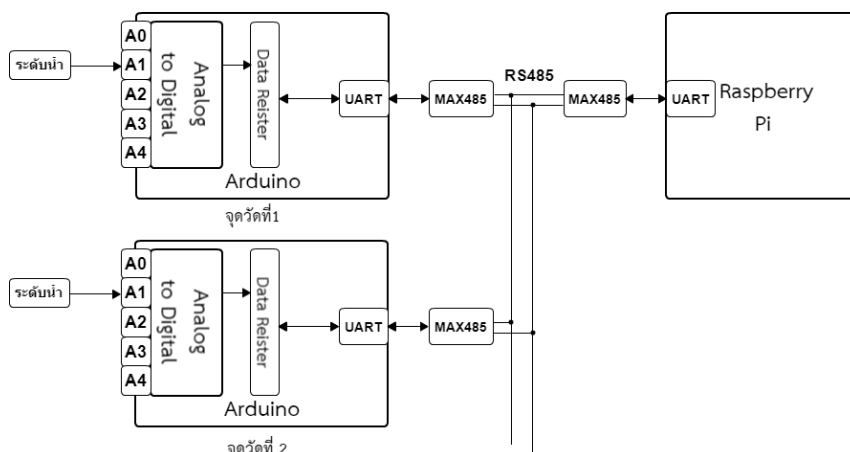
ประมวลผลต่อใน Raspberry Pi จะต้องมีส่วนกลางในการรับค่านอก แล้วทำการแปลงสัญญาณมาเก็บไว้ก่อนที่จะส่งค่านี้ผ่านพอร์ตสื่อสารให้กับ Raspberry Pi โดยงานวิจัยนี้จะใช้ Arduino เป็นตัวกลางในการรับค่านอก และเลือกใช้สื่อสารผ่านพอร์ต RS 485 ซึ่งเป็นมาตรฐานการส่งข้อมูลแบบอนุกรมที่ใช้เพื่อเพิ่มระยะทางในการรับและส่งข้อมูลให้ได้ระยะทางไกลขึ้น โดยจะทำการเปลี่ยนระดับสัญญาณ Logic TTL (0, 5V หรือ 0, 3.3V) ให้เป็นสัญญาณแบบ Differential เพื่อให้สามารถส่งข้อมูลได้ไกลขึ้น และถูกรบกวนจากสัญญาณภายนอกน้อยลง เช่น สัญญาณแม่เหล็กไฟฟ้าที่เกิดจากสายส่งไฟฟ้าได้ยากขึ้น โดยการส่งสัญญาณผ่านโปรโตคอล RS 485 จะมีรูปแบบการรับส่งแบบ Half-duplex ซึ่งจะใช้จำนวนสายในการส่งข้อมูล เพียง 2 เส้น ซึ่งในทางปฏิบัติจะใช้สาย (2 Core และ Shield) ในงานวิจัยนี้การเขียนโปรแกรมจะกำหนดให้ มี Master 1 ตัวเพื่อกอจจัดการสื่อสารในระบบเครือข่ายและให้อุปกรณ์ที่เหลือเป็น Slave โดย Slave แต่ละตัวจะมี Address ของตัวเอง เวลาที่ Master ต้องการจะสื่อสารกับ Slave ทำได้โดยส่ง Address ที่ต้องการจะสื่อสารออกไป แล้วตามด้วยฟังก์ชัน ซึ่ง Slave ทุกตัวจะรับข้อมูลได้เหมือนกันและจะตรวจสอบดูว่า Address นั้นเป็นของตัวเองหรือไม่ ถ้าเป็น Address ของตัวเองก็จะทำการตอบข้อมูลกลับ ตามที่ Master ต้องการส่วนการต่อใช้ในงานวิจัยนี้จะใช้ Raspberry Pi เป็น Master และ Arduino เป็น Slave แสดงดังรูปที่ 3 (Faisal et al., 2017)



หรือไม่ ถ้าเป็น Address ของตัวเองก็จะทำการตอบข้อมูลกลับ ตามที่ Master ต้องการส่วนการต่อใช้ในงานวิจัยนี้จะใช้ Raspberry Pi เป็น Master และ Arduino เป็น Slave แสดงดังรูปที่ 3 (Faisal et al., 2017)

รูปที่ 3 การใช้ Raspberry Pi เป็น Master และ Arduino เป็น Slave ผ่าน RS485

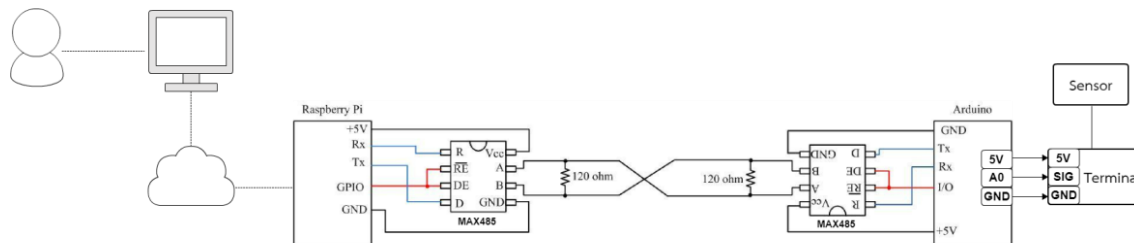
การเขียนโปรแกรมควบคุมการทำงานดังกล่าวจะต้องมีการกำหนดข้อตกลงในการสื่อสาร เพื่อให้ Master และ Slave สามารถติดต่อสื่อสารกันได้ถูกต้อง โครงงานนี้จึงเลือกใช้ Open Protocol ที่เป็นมาตรฐาน และเป็นที่ยอมรับใช้ในอุตสาหกรรม ซึ่งก็คือการกำหนดข้อตกลงในการสื่อสารผ่านโปรโตคอล MODBUS ซึ่งเป็นที่แพร่หลายและทำให้สามารถนำโครงงานนี้ไปประยุกต์ใช้งานอื่นๆ ได้อีกด้วยเนื่องจากเซ็นเซอร์ต่างๆ เช่น เซ็นเซอร์ตรวจวัดอุณหภูมิ เซ็นเซอร์วัดอัตราการไหล เซ็นเซอร์ตรวจวัดแสง ฯลฯ ที่ทำออกมาจำหน่ายจะนิยมส่งค่าที่ปริมาณที่วัดได้(output) เป็น RS 485 ที่ใช้โปรโตคอล Modbus จึงทำให้ระบบนี้สามารถเพิ่มเซ็นเซอร์ต่างๆ เข้าไปได้ทันทีโดยไม่ต้องทำการเพิ่มฮาร์ดแวร์ในการรับค่าจากเซ็นเซอร์แสดงดังรูปที่ 4



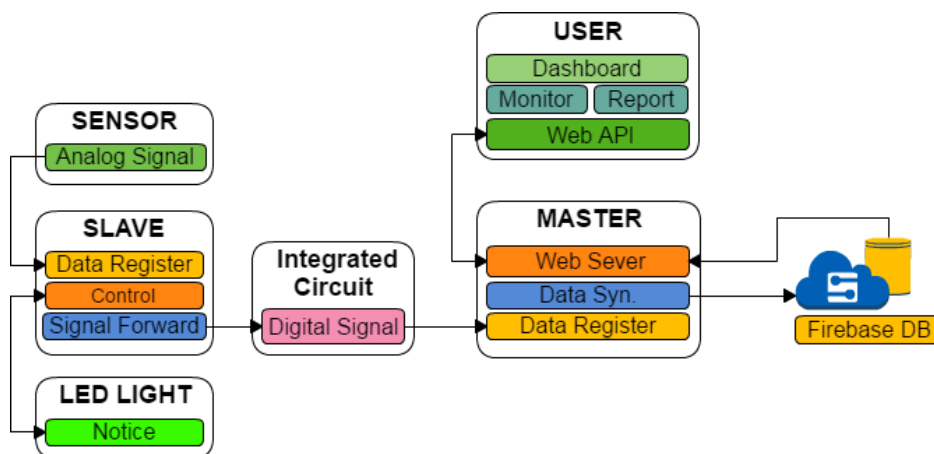
รูปที่ 4 การต่อแบบ Slave หลายตัว

2. สถาปัตยกรรมซอฟต์แวร์

การออกแบบสถาปัตยกรรมซอฟต์แวร์ ในการใช้งานระบบเฝ้าระวังอุทกภัยผ่านเซ็นเซอร์ตรวจวัดระดับความสูงของน้ำ กรณีศึกษา เขตเทศบาลนครศรีธรรมราช สามารถออกแบบสถาปัตยกรรมที่เหมาะสมแสดงดังรูปที่ 5 และ 6 ได้ดังนี้



รูปที่ 5 โครงสร้างแผนผังวงจรโดยรวมของระบบ



รูปที่ 6 สถาปัตยกรรมซอฟต์แวร์

3. Firebase Database

Firebase เป็น Database ที่ถูกออกแบบมาให้เป็น API และ Cloud Storage (Alsalemi et al., 2017) ซึ่งการใช้งานในส่วนนี้จะต้องมีการเชื่อมต่ออินเทอร์เน็ต มีลักษณะเด่นก็คือการทำงานที่เป็น Real Time Database เป็น NoSQL Cloud Database ที่เก็บข้อมูลในรูปแบบของ JSON และมีการ Sync ข้อมูลแบบ Real Time กับทุกอุปกรณ์ที่เชื่อมต่อแบบอัตโนมัติแสดงดังรูปที่ 7

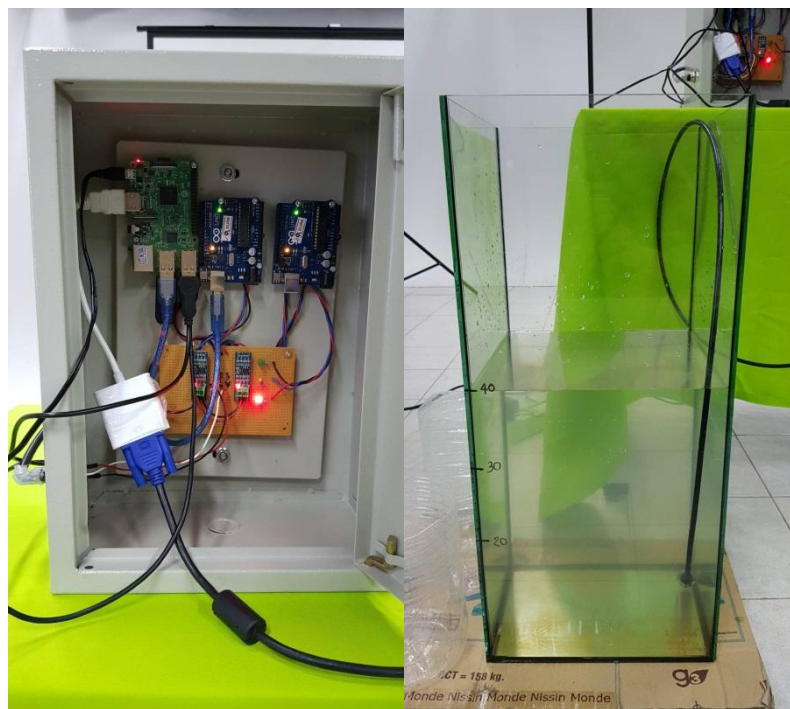


รูปที่ 7 โมดูล Firebase Database

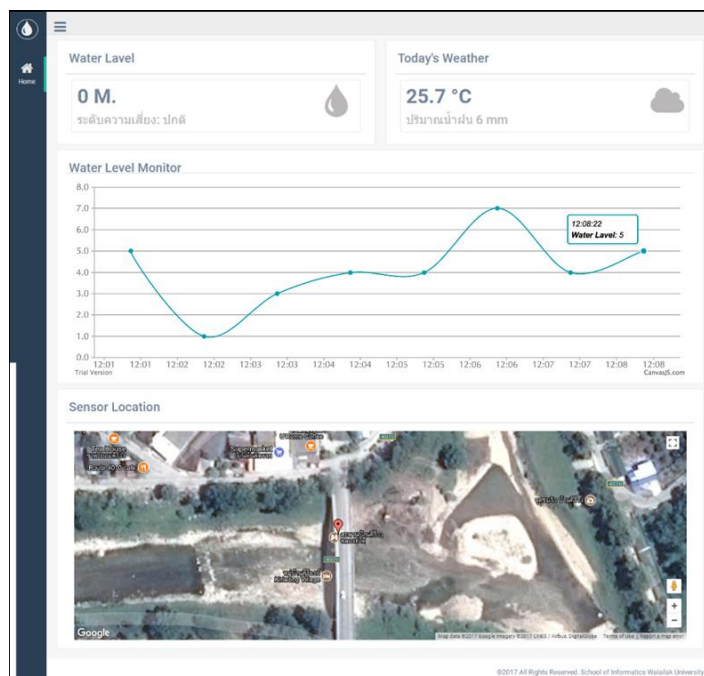
ผลและอภิปรายผล

การออกแบบหน้าจอเฝ้าระวังของระบบเฝ้าระวังอุทกภัยผ่านเซ็นเซอร์ตรวจวัดระดับความสูงของน้ำกรณีศึกษา เขตเทศบาลนครศรีธรรมราช แสดงดังรูปที่ 8 และ 9 เป็นการแสดงผลข้อมูลของระดับน้ำ ซึ่งการแสดงผลหลักจะแสดงในรูปแบบของกราฟความสัมพันธ์ระหว่างระดับน้ำกับช่วงเวลา โดยกราฟข้อมูลดังกล่าวจะเป็นข้อมูล Real time ณ เวลาปัจจุบันที่ตรวจวัดค่าได้ ส่วนของ

การแสดงผลที่ถัดลงมาจะเป็นการแสดงผลข้อมูลเพิ่มเติม อาทิเช่น ข้อมูลอุณหภูมิ ปริมาณน้ำฝน และแผนที่แสดงจุดที่ติดตั้งเซ็นเซอร์โดยจะใช้วิธีการอ้างอิงจาก Google Map

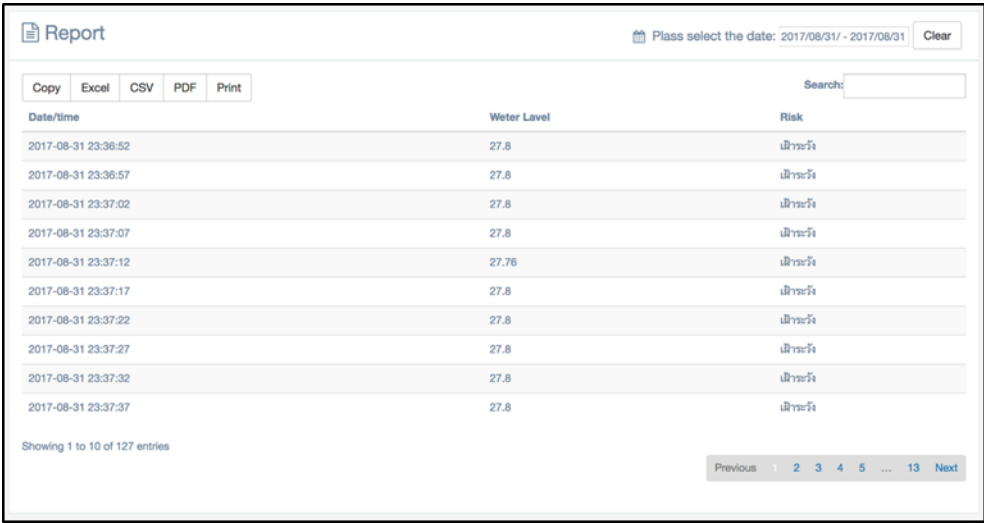


รูปที่ 8 ไฟล์แดงแจ้งเตือนให้อพยพ (โดยกำหนดให้น้ำสูงตั้งแต่ 40 ซม.)



รูปที่ 9 หน้าจอแสดงส่วนของการเฝ้าระวัง

รูปที่ 10 แสดงสรุปรายงานของระดับน้ำ ประกอบด้วย วันที่ ช่วงเวลา ระดับของน้ำ และความเสี่ยงที่จะเกิดอุทกภัย ซึ่งสามารถเลือกช่วงวันเวลาเมื่อต้องการข้อมูลมาออกรายงานได้



Date/time	Water Level	Risk
2017-08-31 23:36:52	27.8	เฝ้าระวัง
2017-08-31 23:36:57	27.8	เฝ้าระวัง
2017-08-31 23:37:02	27.8	เฝ้าระวัง
2017-08-31 23:37:07	27.8	เฝ้าระวัง
2017-08-31 23:37:12	27.76	เฝ้าระวัง
2017-08-31 23:37:17	27.8	เฝ้าระวัง
2017-08-31 23:37:22	27.8	เฝ้าระวัง
2017-08-31 23:37:27	27.8	เฝ้าระวัง
2017-08-31 23:37:32	27.8	เฝ้าระวัง
2017-08-31 23:37:37	27.8	เฝ้าระวัง

รูปที่ 10 สรุปรายงานของระดับน้ำ

ทางผู้วิจัยได้ทำแบบสอบถามเพื่อประเมินการใช้งานของผู้ใช้ โดยให้ผู้ใช้ทดลองใช้งานมีทั้งหมด 23 คนเพื่อดูความพึงพอใจของผู้ใช้ โดยวัดทั้งหมด 3 ด้านได้แก่ ความสามารถในการทำงาน การออกแบบหน้าจอและการทดสอบการใช้งาน โดยกำหนดตัวเลือกเป็น 5 = มากที่สุด, 4 = มาก, 3 = ปานกลาง, 2 = น้อย และ 1 = น้อยที่สุด ได้ผลดังตารางที่ 1

ตารางที่ 1 หัวข้อและผลประเมินแบบสอบถาม

หัวข้อแบบสอบถาม	$\bar{x}$	SD	Min	Max
1. ความสามารถในการทำงาน				
1.1 ความสามารถใช้งานหลากหลายอุปกรณ์ (Responsive Web Design)	4.00	0.67	3	5
1.2 แสดงข้อมูลแบบทันที (Real Time)	4.26	0.81	2	5
1.3 สีหลอดไฟสัมพันธ์กับแสดงผลหน้าจอ	4.48	0.79	2	5
1.4 สีหลอดไฟสื่อความหมาย	4.52	0.59	3	5
1.5 กราฟที่ใช้แสดงผลเหมาะสม	3.96	0.64	3	5
2. การออกแบบหน้าจอ				
2.1 ขนาดตัวอักษร และสีที่นำเสนอ	3.91	0.51	3	5
2.2 การใช้ภาษาในการสื่อสาร	4.22	0.52	3	5
2.3 ความสวยงาม และง่ายต่อการใช้งาน	4.09	0.68	3	5
3. การทดสอบการใช้งาน				
3.1 ความเร็วในการประมวลผล	3.83	0.72	3	5
3.2 ความถูกต้องในการประมวลผล	4.22	0.67	3	5
3.3 ตรงต่อความต้องการของผู้ใช้	4.09	0.6	3	5

สรุปผลการวิจัย

การพัฒนาาระบบเฝ้าระวังอุทกภัยผ่านเซ็นเซอร์ตรวจวัดระดับความสูงของน้ำ กรณีศึกษาเขตเทศบาลนครศรีธรรมราช สามารถติดตั้งเซ็นเซอร์เพื่อวัดระดับน้ำได้โดยการจุ่มอุปกรณ์เซ็นเซอร์ลงภายในน้ำ ระบบสามารถจัดเก็บข้อมูล แสดงผลข้อมูลของระดับน้ำในรูปแบบของกราฟผ่านเว็บไซต์เพื่อต่อการเข้าถึงข้อมูลเมื่อมีการเชื่อมต่ออินเทอร์เน็ต และระบบสามารถแจ้งเตือนผู้ใช้ด้วยสัญญาณไฟแอลอีดี โดยสามารถนำไปประยุกต์ใช้เพื่อให้ง่ายต่อการสังเกตและเฝ้าระวัง รวมถึงระบบสามารถออกรายงานเพื่อเตรียมความพร้อมในการรับมืออุทกภัยได้ ระหว่างพัฒนาระบบได้พบปัญหา Arduino บางรุ่นไม่รองรับการใช้งานในด้านการสื่อสารที่ใช้จากเดิมจะใช้ Arduino D1 ที่ทำหน้าที่เป็นตัว Slave ไม่สามารถใช้งานได้เนื่องจากอุปกรณ์ไม่รองรับการทำงาน จึงต้องเปลี่ยนมาใช้ Arduino UNO จึงจะสามารถทำงานได้อย่างปกติและการประยุกต์ใช้ Raspberry Pi ในการสื่อสารโปรโตคอล Modbus ซึ่งพบว่าการเชื่อมต่อโปรโตคอลผ่าน GPIO Port จะไม่รองรับการเชื่อมต่อจะต้องใช้ USB Port ในการสื่อสารจึงจะทำงานได้อย่างปกติ ในด้านค่าความถูกต้องของเซ็นเซอร์พบว่ามีความผิดพลาดอยู่ที่ $\pm 2\%$ จากการวัดการเพิ่มขึ้นของน้ำในแต่ละช่วงความสูงทั้งหมด 40 สเกล โดยเทียบกับเครื่องมือวัดความสูงมาตรฐาน ซึ่งค่าความผิดพลาดอาจเกิดขึ้นจากความหนาแน่นของน้ำร่วมด้วย

บรรณานุกรม

- Alsalemi A. et al. (2017). Real-Time Communication Network Using Firebase Cloud IoT Platform for ECMO Simulation, 2017 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), Exeter, 2017, pp. 178-182.
- Faisal, S. M. S. R., Ahmed, I. U., Rashid, H., Das, R., Karim, M. M. & Reza, S. M. T. (2017). Design and development of an autonomous floodgate using arduino uno and motor driver controller, 2017 4th International Conference on Advances in Electrical Engineering (ICAEE), Dhaka, 2017, pp. 276-280.
- Joshi, R., Jadav, H. M., Mali, A. & Kulkarni, S. V. (2016). IOT application for real-time monitor of PLC data using EPICS, 2016 International Conference on Internet of Things and Applications (IOTA), Pune, 2016, pp. 68-72.
- Modicon Modbus Protocol Reference Guide. (1996). [เว็บไซต์]. สืบค้นเมื่อ 20 มีนาคม 2562, จาก http://modbus.org/docs/PI_MBUS_300.pdf.
- Nishanth, P. R. & John, N. P. (2017). Development of gateway for industrial controllers, 2017 Innovations in Power and Advanced Computing Technologies (i-PACT), Vellore, 2017, pp. 1-5.
- Nugraheni, D. M. K. & Vries, D. de. (2017). The effectiveness of SMS as verification of flood early warning messages from users' perception, 2017 1st International Conference on Informatics and Computational Sciences (ICICoS), Semarang, 2017, pp. 77-82.