หลักสูตรการเขียนโปรแกรม Coding Language



บริษัท 168 เอ็ดดูเคชั่น จำกัด

เลขที่ 128 อาคารพญาไทพลาซ่า ชั้นที่ 11 ห้องเลขที่ 117 ถนนพญาไท แขวงทุ่งพญาไท เขตราชเทวี กทม. โทร. 02-129-3208



คำนำ

จากนโยบายการศึกษาของกระทรวงศึกษาธิการที่มุ่งเน้นให้นักเรียนมีการเรียนรู้ในภาษาที่ 3 หรือภาษาโค้ดดิ้ง (Coding) โดยเริ่มตั้งแต่ชั้นประถมศึกษาปีที่ 1 เป็นต้นไป เพื่อส่งเสริมให้เกิด I-Innovation หรือ นวัตกรรมที่จะสามารถนำไปใช้ได้จริงและเกิดประโยชน์แก่คนหมู่มาก ทำให้ ประเทศไทยมีบุคลากรที่สร้างนวัตกรรมหน้าใหม่ที่สามารถผลิตนวัตกรรมที่จะเปลี่ยนโลกได้ในทุก สาขาอาชีพ ซึ่งการพัฒนารูปแบบต่างๆ เหล่านี้จะต้องเป็นบุคคลที่มีความคิดสร้างสรรค์ มีความคิดที่ เป็นระบบและเข้าใจในภาษาดิจิตอล เพื่อจะสร้างผู้นำทางด้านเทคโนโลยีใหม่ๆ การเรียนรู้โค้ดดิ้ง (Coding) จึงจำเป็นและสำคัญมากต่อการศึกษาชาติและโรงเรียนต่างๆ จำเป็นต้องพัฒนาการเรียนการ สอนโค้ดดิ้งในทุกระดับชั้น

สำหรับหลักสูตรบอร์คไมโครคอนโทรเลอร์ Arduino เป็นบอร์คตระกูล AVR ที่มีการพัฒนา แบบ Open Source คือมีการเปิคเผยข้อมูลทั้งด้าน Hardware และ Software ตัว บอร์ค Arduino ถูก ออกแบบมาให้ใช้งานได้ง่าย ดังนั้นจึงเหมาะสำหรับผู้เริ่มต้นศึกษา ทั้งนี้ผู้ใช้งานยังสามารถคัดแปลง เพิ่มเติม พัฒนาต่อยอดทั้งตัวบอร์ค หรือโปรแกรมต่อได้อีกด้วย

Arduino คือ โครงการที่นำชิปไอซีไมโครคอนโทรลเลอร์ตระกูลต่างๆ มาใช้ร่วมกันในภาษา C ซึ่งภาษา C นี้เป็นลักษณะเฉพาะ คือมีการเขียนไลบารี่ของ Arduino ขึ้นมาเพื่อให้การสั่งงาน ใมโครคอนโทรลเลอร์ที่แตกต่างกัน สามารถใช้งานโค้คตัวเดียวกันได้ โดยตัวโครงการได้ออกบอร์ด ทดลองมาหลายๆรูปแบบ เพื่อใช้งานกับ IDE ของตนเอง สาเหตุหลักที่ทำให้ Arduino เป็นนิยมมาก เป็น เพราะซอฟแวร์ที่ใช้งานร่วมกันสามารถโหลดได้ฟรี ซึ่งในหลักสูตรผู้เรียนจะได้เรียนรู้เกี่ยวกับ รายละเอียดของตัวบอร์ด Arduino การต่อเชื่อมกับโปรแกรมเพื่อการใช้งานซึ่งจะทำให้ผู้เรียนสามารถ เรียนรู้ได้ครอบคลุมทั้งหมด

ซึ่งทางผู้จัดทำหวังเป็นอย่างยิ่งว่าสถานศึกษาที่ได้นำหลักสูตรการเรียนการสอน Coding เรื่อง หลักสูตร Arduino นี้ไปเรียนรู้จะสามารถทำการเรียนการสอนหลักสูตร Coding ได้เป็นอย่างดีและ สามารถประยุกต์ใช้รวมทั้งสร้างโครงงานได้เป็นอย่างดี

บริษัท 168 เอ็ดดูเคชั่น จำกัด

สารบัญ

เรื่อง	หน้า
หน่วยการเรียนรู้ที่ 1 ไมโครคอนโทรลเลอร์และบอร์ค Arduino	
- ไมโครคอนโทรลเลอร์ Arduino	1
- ที่มาของบอร์ค Arduino และทำความรู้จักบอร์ค Arduino	12
- ส่วนประกอบของบอร์ค Arduino และการเริ่มทำงานกับบอร์ค Arduino	28
หน่วยการเรียนรู้ที่ 2 พัฒนาการของบอร์ค Arduino Uno R3	36
หน่วยการเรียนรู้ที่ 3 การเขียนภาษา C สำหรับไมโครคอนโทรลเลอร์	46
หน่วยการเรียนรู้ที่ 4 การติดตั้งและลองใช้ Arduino IDE	54
หน่วยการเรียนรู้ที่ 5 การลงมือเขียนโปรแกรม	
- การใช้งานโปรแกรม Arduino IDE	70
- โครงสร้างโปรแกรมและคำสั่ง	83
- โครงสร้างเงื่อนไข คำสั่งในการติดต่อกับพอร์ต จอแอลซีดี และการสร้างโปรแกรมย่อย	91
- การติดตั้ง Library บน Arduino IDE	110
หน่วยการเรียนรู้ที่ 6 Arduino Web Editor เขียนโปรแกรมผ่านบราวเซอร์	115
หน่วยการเรียนรู้ที่ 7 จอภาพตอบสนองผู้ชม	
- เครื่องมือและส่วนประกอบ	122
- ตัวต้านทาน	133
- ไฟ LED	142
- การเลื่อนภาพ	148
- การเชื่อมต่อบอร์ด Arduino กับ Processing	151

หน่วยการเรียนรู้ที่ 8 จอภาพตอบสนองผู้ชม

- Temperature Sensor Module และ Temperature Humidity Sensor	153
- IR Receiver Module	166
- Tracking Sensor	173
- Buzzer Module	177
- Switch Button Module	180
- Light Cup Module	185
- Light Blocking Sensor	189
- Relay และ Motion Sensor Module	192
- Temperature Humidity Sensor และ LED 4-Digit Display Module	202
- เครื่องวัดอากาศ PM 2.5	207
หน่วยการเรียนรู้ที่ 9 การประยุกศ์ใช้โปรแกรมแลบวิว	221
หน่วยการเรียนรู้ที่ 10 การประยุกต์ใช้บอร์ค Arduino	
- ตัวอย่างการใช้งาน	243
- ตัวอย่างโครงงาน	248
- ตัวอย่างโครงงาน IOT	261

คำอธิบายรายวิชา

วิชา การเขียนโปรแกรม Arduino กลุ่มสาระการเรียนรู้วิทยาศาสตร์ ชั้นมัธยมศึกษา ภากเรียนที่ ๑-๒ เวลา ๔๐ ชั่วโมง จำนวน ๑ หน่วยกิต

สำหรับหลักสูตรบอร์คไมโครคอนโทรเลอร์ Arduino เป็นบอร์คตระกูล AVR ที่มีการพัฒนา แบบ Open Source คือมีการเปิคเผยข้อมูลทั้งด้าน Hardware และ Software ตัว บอร์ค Arduino ถูก ออกแบบมาให้ใช้งานได้ง่าย ดังนั้นจึงเหมาะสำหรับผู้เริ่มต้นศึกษา ทั้งนี้ผู้ใช้งานยังสามารถดัดแปลง เพิ่มเติม พัฒนาต่อยอดทั้งตัวบอร์ค หรือโปรแกรมต่อได้อีกด้วย

Arduino คือ โครงการที่นำชิปไอซีไมโครคอนโทรลเลอร์ตระกูลต่างๆ มาใช้ร่วมกันในภาษา C ซึ่งภาษา C นี้เป็นลักษณะเฉพาะ คือมีการเขียนไลบารี่ของ Arduino ขึ้นมาเพื่อให้การสั่งงาน ใมโครคอนโทรลเลอร์ที่แตกต่างกัน สามารถใช้งานโค้คตัวเคียวกันได้ โคยตัวโครงการได้ออกบอร์ค ทดลองมาหลายๆรูปแบบ เพื่อใช้งานกับ IDE ของตนเอง สาเหตุหลักที่ทำให้ Arduino เป็นนิยมมาก เป็น เพราะซอฟแวร์ที่ใช้งานร่วมกันสามารถโหลดได้ฟรี ซึ่งในหลักสูตรผู้เรียนจะได้เรียนรู้เกี่ยวกับ รายละเอียดของตัวบอร์ค Arduino การต่อเชื่อมกับโปรแกรมเพื่อการใช้งานซึ่งจะทำให้ผู้เรียนสามารถ เรียนรู้ได้กรอบคลุมทั้งหมด

ตัวชี้วัด

ງ໔.໑ਸ਼.๑/໑ ງ໔.໑ਸ਼.๔/໑ ງ໔.๒ਸ਼.๕/໑ ງ໔.๒ਸ਼.๖/໑ ງ໔.๑ਸ਼.๗/Ლ ງ໔.๑ਸ਼.๓/๒ ງ໔.๑ਸ਼.๓/๓ ງ໔.๑ਸ਼.๓/๔ う๔.๑ਸ਼.๓/๕ う๔.๑ਸ਼.๔/๕ う๔.๒ਸ਼.๑/໑ う๔.๒ਸ਼.๑/๒ う๔.๒ਸ਼.๑/๓ う๔.๒ਸ਼.๑/๔ う๔.๒ਸ਼.๒/๑ ๅ๔.๒ਸ਼.๒/๒

รวม ๒๔ ตัวชี้วัด

ตัวชี้วัดและสาระการเรียนรู้

วิชา การเขียนโปรแกรม Arduino กลุ่มสาระการเรียนรู้วิทยาศาสตร์ ชั้นมัธยมศึกษา ภากเรียนที่ ๑-๒ เวลา ๔๐ ชั่วโมง จำนวน ๑ หน่วยกิต

สาระที่ ๔ เทคโนโลยี

มาตรฐาน ว ๔.๑ เข้าใจแนวคิดหลักของเทคโนโลยีเพื่อการคำรงชีวิตในสังคมที่มีการเปลี่ยนแปลงอย่าง รวดเร็ว ใช้ความรู้และทักษะทางค้านวิทยาศาสตร์ คณิตศาสตร์ และศาสตร์อื่น ๆ เพื่อแก้ปัญหาหรือพัฒนา งานอย่างมีความคิดสร้างสรรค์ค้วยกระบวนการออกแบบเชิงวิศวกรรมเลือกใช้เทคโนโลยีอย่างเหมาะสม โดยคำนึงถึงผลกระทบต่อชีวิต สังคม และสิ่งแวคล้อม

ຄຳດັບ	ตัวชี้วัด	สาระการเรียนรู้แกนกลาง
ଭ	ว ๔.๑ ม.๑/๑	อธิบายแนวคิดหลักของเทคโนโลยีในชีวิตประจำวันและวิเคราะห์สาเหตุหรือ ปัจจัยที่ส่งผลต่อการเปลี่ยนแปลงของเทคโนโลยี
්	ີງ ໔.໑ ໗.໑/๒	ระบุปัญหาหรือความต้องการในชีวิตประจำวัน รวบรวม วิเคราะห์ข้อมูลและ แนวกิดที่เกี่ยวข้องกับปัญหา
ଜ	ີ ໔.໑ ໓.໑/໑	ออกแบบวิธีการแก้ปัญหา โดยวิเคราะห์ เปรียบเทียบ และตัดสินใจเลือกข้อมูลที่ จำเป็น นำเสนอแนวทางการแก้บัญหาให้ผู้อื่นเข้าใจ วางแผนและคำเนินการ แก้ปัญหา
હ	ી હ.૭ મુ.૭/હ	ออกแบบวิธีการแก้ปัญหา โดยวิเคราะห์ เปรียบเทียบ และตัดสินใจเลือกข้อมูลที่ จำเป็น นำเสนอแนวทางการแก้ปัญหาให้ผู้อื่นเข้าใจ วางแผนและคำเนินการ แก้ปัญหา
હ	ી ૡૼ.૦ પ્રી.૦/ૡૼ	ใช้ความรู้และทักษะเกี่ยวกับวัสดุ อุปกรณ์ เครื่องมือ กลไก ไฟฟ้า หรือ อิเล็กทรอนิกส์เพื่อแก้ปัญหาได้อย่างถูกต้อง เหมาะสมและปลอคภัย
રુ	ີ ໔.໑ ຆ.๒/ຓ	ออกแบบวิธีการแก้ปัญหา โดยวิเคราะห์ เปรียบเทียบ และตัดสินใจเลือกข้อมูลที่ จำเป็นภายใต้เงื่อนไขและทรัพยากรที่มีอยู่ นำเสนอแนวทางการแก้ปัญหาให้ ผู้อื่นเข้าใจ วางแผนขั้นตอนการทำงานและดำเนินการแก้ปัญหาอย่างเป็น ขั้นตอน
ග	ີງ ໔.໑ ໗.๒/໔	ทคสอบ ประเมินผล และอธิบายปัญหาหรือข้อบกพร่องที่เกิดขึ้น ภายใต้กรอบ เงื่อนไข พร้อมทั้งหาแนวทางการปรับปรุงแก้ไข และนำเสนอผลการแก้ปัญหา

r	ට ໔.0 ຏ.๒/๕	ใช้ความรู้ และทักษะเกี่ยวกับวัสคุ อุปกรณ์ เครื่องมือ กลไก ไฟฟ้า และ
		อิเล็กทรอนิกส์ เพื่อแก้ปัญหาหรือพัฒนางานได้อย่างถูกต้อง เหมาะสม และ
		ปลอคภัย
ග	ີງ ໔.໑ ໗.໑/໑	วิเคราะห์สาเหตุ หรือปัจจัยที่ส่งผลต่อการเปลี่ยนแปลงของเทคโนโลยี และ
		ความสัมพันธ์ของเทคโนโลยีกับศาสตร์อื่น โดยเฉพาะวิทยาศาสตร์ หรือ
		กณิตศาสตร์ เพื่อเป็นแนวทางการแก้ปัญหาหรือพัฒนางาน
ร	ີງ ໔.໑ ຆ.໑/໑	ออกแบบวิธีการแก้ปัญหา โดยวิเคราะห์เปรียบเทียบ และตัดสินใจเลือกข้อมูลที่
		จำเป็นภายใต้เงื่อนไขและทรัพยากรที่มีอยู่ นำเสนอแนวทางการแก้ปัญหาให้
		ผู้อื่นเข้าใจด้วยเทคนิคหรือวิธีการที่หลากหลาย วางแผนขั้นตอนการทำงานและ
		ดำเนินการแก้ปัญหาอย่างเป็นขั้นตอน
ಕ	ີງ ໔.໑ ຆ.໑/໔	ทคสอบ ประเมินผล วิเคราะห์ และให้เหตุผลของปัญหาหรือข้อบกพร่องที่
		เกิดขึ้นภายใต้ กรอบเงื่อนไข พร้อมทั้งหาแนวทางการปรับปรุงแก้ไข และ
		นำเสนอผลการแก้ปัญหา
ၜဝ	ට ໔.໑ ຏ.෨/៥	ใช้ความรู้ และทักษะเกี่ยวกับวัสดุ อุปกรณ์ เครื่องมือ กลไก ไฟฟ้าและ
		อิเล็กทรอนิกส์ให้ ถูกต้องกับลักษณะของงาน และปลอคภัย เพื่อแก้ปัญหาหรือ
		พัฒนางาน
99	า ๔.๑ ม.๔/๑	วิเคราะห์แนวคิดหลักของเทคโนโลยี ความสัมพันธ์กับศาสตร์อื่น โดยเฉพาะ
		วิทยาศาสตร์ หรือคณิตศาสตร์ รวมทั้งประเมินผลกระทบที่จะเกิดขึ้นต่อมนุษย์
		สังคม เศรษฐกิจ และสิ่งแวคล้อม เพื่อเป็นแนวทางในการพัฒนาเทคโนโลยี
බේෂ	ว ๔.๑ ม.๔/๕	ใช้ความรู้และทักษะเกี่ยวกับวัสคุ อุปกรณ์ เครื่องมือ กลไก ไฟฟ้าและ
		อิเล็กทรอนิกส์และเทคโนโลยีที่ซับซ้อนในการแก้ปัญหาหรือพัฒนางาน ได้
		อย่างถูกต้อง เหมาะสมและปลอดภัย

มาตรฐาน ว ๔.๒ เข้าใจและใช้แนวคิดเชิงคำนวณในการแก้ปัญหาที่พบในชีวิตจริงอย่างเป็นขั้นตอนและเป็น ระบบ ใช้เทคโนโลยีสารสนเทศและการสื่อสารในการเรียนรู้ การทำงาน และการแก้ปัญหาได้อย่างมี ประสิทธิภาพ รู้เท่าทัน และมีจริยธรรม

ลำดับ	ตัวชี้วัด	สาระการเรียนรู้แกนกลาง
9	າ ໔.ໄຫ ນ.໑/໑	ออกแบบอัลกอริทึมที่ใช้แนวคิดเชิงนามธรรมเพื่อแก้ปัญหาหรืออธิบายการ ทำงานที่พบในชีวิตจริง
්	ງ ໔.ໂຫ ນ.໑/ໂຫ	ออกแบบและเขียนโปรแกรมอย่างง่ายเพื่อแก้ปัญหาทางคณิตศาสตร์หรือ วิทยาศาสตร์
စာ	ີງ ໔.២ ນ.໑/໑	รวบรวมข้อมูลปฐมภูมิ ประมวลผล ประเมินผล นำเสนอข้อมูล และ สารสนเทศ ตามวัตถุประสงค์โดยใช้ซอฟต์แวร์ หรือบริการบนอินเทอร์เน็ต ที่หลากหลาย
હ	າ ໔.๒ ม.๑/໔	ใช้เทคโนโลยีสารสนเทศอย่างปลอดภัย ใช้สื่อและแหล่งข้อมูลตาม ข้อกำหนดและข้อตกลง
હ	າ ໔.២ ຆ.២/໑	ออกแบบอัลกอริทึมที่ใช้แนวคิดเชิงคำนวณในการแก้ปัญหา หรือการทำงาน ที่พบในชีวิตจริง
ઠ	ງ ໔.ໄອ ນ.ໄອ/ໄອ	ออกแบบและเขียนโปรแกรมที่ใช้ตรรกะและพึงก์ชันในการแก้ปัญหา
ෆ	ີງ ໔.ໂອ ນີ.ຫ∕ໂອ	รวบรวมข้อมูล ประมวลผล ประเมินผล นำเสนอข้อมูลและสารสนเทศตาม วัตถุประสงค์ โคยใช้ซอฟต์แวร์หรือบริการบนอินเทอร์เน็ตที่หลากหลาย
ග	າ ໔.២ ນ.໔/໑	ประยุกต์ใช้แนวคิคเชิงคำนวณในการพัฒนาโครงงานที่มีการบูรณาการกับ วิชาอื่นอย่างสร้างสรรค์ และเชื่อมโยงกับชีวิตจริง
r	ว ๔.๒ ม.๕/๑	รวบรวม วิเคราะห์ข้อมูล และใช้ความรู้ด้านวิทยาการคอมพิวเตอร์ สื่อคิจิทัล เทคโนโลยีสารสนเทศในการแก้ปัญหาหรือเพิ่มมูลค่าให้กับบริการหรือ ผลิตภัณฑ์ที่ใช้ ในชีวิตจริงอย่างสร้างสรรค์
હ	ງ ໔.២ ມ.៦/໑	ใช้เทคโนโลยีสารสนเทศในการนำเสนอ และแบ่งปั้นข้อมูลอย่างปลอดภัย มี จริยธรรม และวิเคราะห์การเปลี่ยนแปลงเทคโนโลยีสารสนเทศที่มีผลต่อการ คำเนินชีวิต อาชีพ สังคม และวัฒนธรรม

โครงสร้างรายวิชา

วิชา การเขียนโปรแกรม Arduino กลุ่มสาระการเรียนรู้วิทยาศาสตร์ ชั้นมัธยมศึกษา ภาคเรียนที่ ๑-๒ เวลา ๔๐ ชั่วโมง จำนวน ๑ หน่วยกิต

ลำดับ	ชื่อหน่วยการเรียนรู้	สาระสำคัญ (key Concept) นักเรียนจะ ได้อะไรจากการเรียนแต่ละหน่วย	มาตรฐานการ เรียนรู้/ ตัวชี้วัด	เวลา (ชม.)	น้ำหนัก คะแนน
6	ไมโครคอนโทรลเลอร์และ บอร์ค Arduino	- ไมโครคอนโทรลเลอร์ Arduino - ที่มาของบอร์ด Arduino และทำความ รู้จักบอร์ด Arduino - ส่วนประกอบของบอร์ด Arduino และ การเริ่มทำงานกับบอร์ด Arduino	ີງ ໔.໑ ໗.໑/໑ ີງ ໔.໑ ໗.໑/໑ ີງ ໔.໑ ໗.໑/໑ ີງ ໔.໑ ໗.๒/໑	ရာ	ရာ
lo	พัฒนาการของบอร์ด Arduino Uno R3	Arduino Uno R3 มีการพัฒนามาเป็น ครั้งที่ ๓ เพื่อเพิ่มศักยภาพทางการเรียนรู้ เรื่องอุปกรณีและ ใมโครคอนโทรลเลอร์	ි ໔.໑ ਸ਼.෨/໑ ॊ ໔.໑ ਸ਼.෨/෨ ॊ ໔.໑ ਸ਼.෨/៥ ॊ ໔.໑ ਸ਼.໔/໑	ை	ရာ
ာ	การเขียนภาษา C สำหรับ ไมโครคอนโทรลเลอร์	การเขียนโปรแกรมที่มีความยืดหยุ่น และสามารถพัฒนางานได้เร็วกว่าการ เขียนโปรแกรมด้วยภาษาแอสเซมบลี	ി ഭ.๒ ม.๑/๑ ി ഭ.๒ ม.๕/๑ ി ഭ.๒ ม.๕/๑ ി ഭ.๒ ม.๕/๑ ി ഭ.๒ ม.๖/๑	ရာ	સ્
હ	การติดตั้งและลองใช้ Arduino IDE	Arduino IDE เป็นโปรแกรมสำหรับ เขียนโค้ดเพื่อควบคุมการทำงานของ ไมโครคอนโทรลเลอร์ Arduino	ി ഭ.๒ ม.๑/๑ ി ഭ.๒ ม.๑/๒ ി ഭ.๒ ม.๑/๔ ി ഭ.๒ ม.๒/๑	စာ	હ
હ્ય	การลงมือเขียน โปรแกรม	- การใช้งานโปรแกรม Arduino IDE - โครงสร้างโปรแกรมและคำสั่ง - โครงสร้างเงื่อนไข คำสั่งในการติดต่อ	ാ ഭ.๒ ม.๑/๑ ാ ഭ.๒ ม.๑/๒ ാ ഭ.๒ ม.๑/๔	હ	હ

		กับพอร์ต จอแอลซีดี และการสร้าง โปรแกรมย่อย - การติดตั้ง Library บน Arduino IDE	ാ ഭ.๒ ม.๒/๑ ാ ഭ.๒ ม.๕/๑ ാ ഭ.๒ ม.๖/๑		
રુ	Arduino Web Editor เขียน โปรแกรมผ่านบราวเซอร์	การใช้งาน Arduino Web Editor โปรแกรมจำลองการทำงาน_เพื่อนำไฟล์ ภาษาเครื่องที่ได้จากการแปลงไปจำลอง การทำงานด้วยเว็ปไซต์	ി ഭ.๒ ม.๑/๒ ി ഭ.๒ ม.๕/๑ ി ഭ.๒ ม.๖/๑	ை	હ
ಲಿ	จอภาพตอบสนองผู้ชม	- เครื่องมือและส่วนประกอบ - ตัวต้านทาน - ไฟ LED - การเลื่อนภาพ - การเชื่อมต่อบอร์ค Arduino กับ Processing	ි ໔.໑ ೫.๑/໔ ﺁ ໔.໑ ೫.๑/៥ ﺁ ໔.໑ ೫.๒/៥ ﺁ ໔.໑ ೫.๒/៥ ﺁ ໔.໑ ೫.๓/៥ ﺁ ໔.໑ ೫.๓/๕	æ	હ
હ	ทำความรู้จักเซนเซอร์	 Temperature Sensor Module IIAE Temperature Humidity Sensor IR Receiver Module Tracking Sensor Buzzer Module Switch Button Module Light Cup Module Light Blocking Sensor Relay IIAE Motion Sensor Module Temperature Humidity Sensor IIAE LED 4-Digit Display Module 	ට	ર્લ	Æ
હ	การประยุกต์ใช้โปรแกรมแลบ วิว	การประขุกต์ใช้โปรแกรมแลบวิว	ට	હ	භ

			ว ๔.๑ ม.๓/๕ ว ๔.๑ ม.๔/๕		
60	การประยุกต์ใช้บอร์ด Arduino	- ตัวอย่างการใช้งาน - ตัวอย่างโครงงาน - ตัวอย่างโครงงาน IoT	ට	રુ	හ
		กิจกรรม		G	ၜ၀
ข้อสอบกลางภาค		G	ဖာဝ		
ข้อสอบปลายภาค		G	ြာဝ		
รวม			৫০	900	

หน่วยการเรียนรู้ที่ ๑ ไมโครคอนโทรลเลอร์และบอร์ด Arduino

กลุ่มสาระการเรียนรู้วิทยาศาสตร์ สาระที่ ๔ เทคโนโลยี

มาตรฐาน ว ๔.๑ เข้าใจแนวคิดหลักของเทคโนโลยีเพื่อการดำรงชีวิตในสังคมที่มีการเปลี่ยนแปลงอย่าง รวดเร็ว ใช้ความรู้และทักษะทางด้านวิทยาศาสตร์ คณิตศาสตร์ และศาสตร์อื่น ๆ เพื่อแก้ปัญหาหรือพัฒนา งานอย่างมีความคิดสร้างสรรค์ด้วยกระบวนการออกแบบเชิงวิศวกรรมเลือกใช้เทคโนโลยีอย่างเหมาะสม โดยคำนึงถึงผลกระทบต่อชีวิต สังคม และสิ่งแวดล้อม

ตัวชี้วัด

ി ഭ്.െ ม.ം/െ	อธิบายแนวกิดหลักของเทกโนโลยีในชีวิตประจำวันและวิเกราะห์สาเหตุหรือปัจจัยที่
	ส่งผลต่อการเปลี่ยนแปลงของเทคโนโลยี
ി ഭ. െ പ്ര./ിഇ	ระบุปัญหาหรือความต้องการในชีวิตประจำวัน รวบรวม วิเคราะห์ข้อมูลและแนวคิดที่
	เกี่ยวข้องกับปัญหา
ว ๔.๑ ม.๑/๑	ออกแบบวิธีการแก้ปัญหา โดยวิเคราะห์ เปรียบเทียบ และตัดสินใจเลือกข้อมูลที่จำเป็น
	นำเสนอแนวทางการแก้ปัญหาให้ผู้อื่นเข้าใจ วางแผนและคำเนินการแก้ปัญหา
ີງ ໔.໑ [ູ] ນ.ໂຫ∕ຫ	ออกแบบวิธีการแก้ปัญหา โดยวิเคราะห์ เปรียบเทียบ และตัดสินใจเลือกข้อมูลที่จำเป็น
	ภายใต้เงื่อนไขและทรัพยากรที่มีอยู่ นำเสนอแนวทางการแก้ปัญหาให้ผู้อื่นเข้าใจ
	วางแผนขั้นตอนการทำงานและคำเนินการแก้ปัญหาอย่างเป็นขั้นตอน

สาระสำคัญ

- ๑. เรียนรู้และทำความเข้าใจไมโครคอนโทรลเลอร์ Arduino
- ๒. เรียนรู้ที่มาของบอร์ค Arduino และทำความรู้จักบอร์ค Arduino
- ๓. เรียนรู้ส่วนประกอบของบอร์ค Arduino และการเริ่มทำงานกับบอร์ค Arduino

สาระการเรียนรู้

<u>ความรู้</u>

- ด. การนำคอมพิวเตอร์ไปใช้ร่วมกับการเรียนรู้ทักษะทางวิชาการ
- ๒. เรียนรู้ โลกอัลกอริทึมที่เข้ามามีบทบาทในการเรียน
- ๑. เรียนรู้ ใมโครคอนโทรลเลอร์
- ๔. เรียนรู้ส่วนประกอบของบอร์ค Arduino

<u>ทักษะและกระบวนการ</u>

- ๑. ใช้กระบวนการเรียน ความรู้ ความเข้าใจเรื่องที่จะศึกษา
- ๒. ใช้กระบวนการคิดอย่างมีวิจารณญาณและการสร้างความกิดรวบขอดในการเรียนรู้
- ๓. สร้างเจตคติและค่านิยมที่ดีในการเรียน

<u>คุณลักษณะที่พึงประสงค์</u>

- ๑. นักเรียนมีความตั้งใจศึกษาเรื่องไมโครคอนโทรลเลอร์ นักเรียน
- ๒. นักเรียนมีความตั้งใจศึกษาเรื่องบอร์ด Arduino
- ๓. มีวินัย
- ๔. ใฝ่เรียนรู้
- ผู่งมั่นในการทำงาน
- ๖. มีจิตสาธารณะ

หน่วยการเรียนรู้ที่ 1 ไมโครคอนโทรลเลอร์และบอร์ด Arduino เรื่องย่อยที่ 1 ไมโครคอนโทรลเลอร์ Arduino

บทนำ

การศึกษา และเศรษฐกิจขุคใหม่ที่ต่างไปจากโลกใบเก่า 'การศึกษาในขุคดิจิทัล' จึงมีระบบ-กติกาใน ตัวเองที่เราต้องเรียนรู้ เพื่อปรับบทบาทหน้าที่และกฎกติกาในการบริหารจัดการการศึกษา ให้สอดคล้องกับ ความเปลี่ยนแปลงที่เกิดขึ้น ตั้งแต่บทบาทการนำของครูที่ต้องปรับเปลี่ยนมาเป็น 'โค้ช (Code)' เป็น 'ที่ ปรึกษา' เพื่อกระตุ้นสร้างแรงบันดาลใจให้แก่ผู้เรียนได้

อีกด้านหนึ่งของเทคโนโลยีระดับโลกอัลกอริทึมที่เข้ามามีบทบาทในการเรียนการสอนในหลาย สถาบัน สื่อให้เห็นถึงนวัตกรรมการศึกษาที่ก้าวผ่านห้องเรียนธรรมดาสู่ห้องเรียนที่เปิดกว้างทางด้าน กวามคิด อัลกอริทึม (Algorithm) คือ กระบวนการแก้ปัญหาที่สามารถอธิบายออกมาเป็นขั้นตอนที่ชัดเจน เมื่อนำเข้าอะไรแล้วจะต้องได้ผลลัพธ์เช่นไร กระบวนการนี้ประกอบด้วยจะประกอบด้วย วิธีการเป็นขั้นๆ และมีส่วนที่ต้องทำแบบวนซ้ำอีก จนกระทั่งเสร็จสิ้นการทำงาน Algorithm ไม่ใช่กำตอบแต่เป็นชุดคำสั่งที่ ทำให้ได้กำตอบ วิธีการในการอธิบาย Algorithm ได้แก่

- 1. Natural Language อธิบายแบบใช้ภาษาที่เราสื่อสารกันทั่วไป
- 2. Pseudocode อธิบายด้วยรหัสจำลองหรือรหัสเทียม
- 3. Flowchart อธิบายด้วยแผนผัง

การนำขั้นตอนวิธีไปใช้แก้ปัญหาไม่จำกัดเฉพาะการเขียนโปรแกรมคอมพิวเตอร์แต่สามารถใช้กับ ปัญหาอื่น ๆ ได้เช่นเดียวกัน ตัวอย่างเช่น

 ในการวางแผนการใช้ทรัพยากรทางธุรกิจขององค์กร หรือ Enterprise Resource Planning (ERP) เพื่อให้เกิดการใช้ทรัพยากรอย่างมีประโยชน์สูงสุด ซึ่งจำเป็นต้องวางแผนอย่างเป็นระบบ เป็น ขั้นตอน จึงจำเป็นต้องอาศัย Algorithm ด้วย เพื่อให้ทราบถึงขั้นตอนต่างๆ และสามารถลดขั้นตอนที่เกิน ความจำเป็นอีกทั้งยังสามารถปรับปรุง และเพิ่มเติมขั้นตอนใหม่ช่วยลดความสับสนขณะทำงานด้วย

2. ตัวอย่าง Algorithm ในการเข้ารหัส MD5 เช่นการสร้างรหัสผ่าน

 3. ตัวอย่าง Algorithm ในการแยกแยะประเภทข้อมูล SVM (Support Vector Machine) งณะที่ยุคคิจิทัลพัฒนาความก้าวหน้าไปด้วย 'กลไกการเคลื่อนไหวแบบแพลตฟอร์ม' ซึ่งประสาน รวมทั้งเทคโนโลยีไอที การสื่อสาร และการจัดการสารสนเทศยุคใหม่ในการขับเคลื่อนและพัฒนาการเรียนรู้ ให้แก่ผู้เรียนได้อย่างสร้างสรรค์ อีกทั้งยังปลอดจากการแบ่งชนชั้นและความเหลื่อมล้ำทั้งหลาย จึงเป็นปฐม บทของความเปลี่ยนแปลงที่ใครๆ ก็สามารถจับต้องได้

<u>ไมโครคอนโทรลเลอร์</u>

ไมโครคอนโทรลเลอร์ ถูกพัฒนามาจากไมโครโพรเซสเซอร์ ไมโครคอนโทรลเลอร์จึง ประกอบด้วยชุดควบคุมขนาดเล็กที่รวบรวมคุณสมบัติหลักของระบบคอมพิวเตอร์มารวมไว้ในบอร์ด เดียวกัน ไม่ว่าจะเป็นส่วนประกอบหลักอย่าง CPU หน่วยความจำและพอร์ตเชื่อมต่อกับอุปกรณ์ต่างๆ ทั้งหมดรวมอยู่ในชุดเดียวกัน แต่คุณสมบัติหลักๆ ของไมโครคอนโรลเลอร์ที่แตกต่างจากคอมพิวเตอร์ คือ มี พอร์ตติดต่อกับอุปกรณ์ภายนอกและอุปกรณ์ภายนอกเหล่านั้นไม่ใช่เพียงคีย์บอร์ด เมาส์ แฟรชไดรว์หรือ จอมอนิเตอร์ แต่คืออุปกรณ์ภายนอกที่เป็นชิ้นส่วนอิเล็กทรอนิกส์ หรือชุดอุปกรณ์อิเล็กทรอนิกส์ที่ถูก ประดิษฐ์ขึ้นมาเพื่อใช้งานในลักษณะเฉพาะ (โมดูล) เช่น โมดูลวัดค่า โมดูลสื่อสาร โมดูลการบันทึกค่าและ ส่งค่า การทำงานของมอเตอร์ การทำงานของรีเลย์ การแสดงแสง สี เสียง และจอลดทอนสัญญาณรบกวน และ GPS ไมโครคอนโทรลเลอร์ จะเป็นอุปกรณ์กวบคุมการทำงานของอุปกรณ์ภายนอกเหล่านี้ โดยส่ง สัญญาณกำสั่งและรับสัญญาณค่าต่างๆ จากอุปกณ์ภายนอกผ่านพอร์ตที่ถูกติดตั้งเชื่อมโยงกับระบบการ ประมวลผล กล่าวคือ ไมโครลคอนโทรลเลอร์จะส่งสัญญาณกำสั่งผ่านพอร์ตอินพุด (Input Port) ไปยัง อุปกรณ์ภายนอกและรับสัญญาณค่าต่างๆ จากอุปกรณ์ภายนอกผ่านพอร์ตเอาพุด (Output Port) แล้วนำ สัญญาณที่ได้รับไปทำการประมวลผล

เครื่องใช้ไฟฟ้าชนิดใดก็ตามหากต้องมีไมโกรกอนโทรลเลอร์เป็นอุปกรณ์ควบคุมการทำงานของ เครื่องใช้ไฟฟ้า โดยจะมีการเขียนโปรแกรมคำสั่งบันทึกลงไปใน CPU ของไมโครคอนโทรลเลอร์ เพื่อส่ง สัญญาณกำสั่งให้อุปกรณ์ในเครื่องใช้ไฟฟ้าทำงานหรือรับสัญญาณก่าต่างๆ ที่อุปกรณ์ภายนอกวัดได้เพื่อนำ สัญญาณก่าสั่งให้อุปกรณ์ในเครื่องใช้ไฟฟ้าทำงานหรือรับสัญญาณก่าต่างๆ ที่อุปกรณ์ภายนอกวัดได้เพื่อนำ สัญญาณก่าเหล่านั้นไปทำการประมวลผล แม้แต่ไฟฉายขนาดเท่านิ้วแม่มือยังต้องมีบอร์ด ใมโครคอนโทรลเลอร์ควบคุมการทำงาน เช่น ควบคุมระบบชาร์จแบตเตอรี่ ควบคุมโหมดแสงสว่าง ควบคุม โหมดสวิตช์ เป็นต้น แต่ไม่ได้หมายถึง ไมโครคอนโทรลเลอร์ คือ Arduino เพียงอย่างเดียวเพราะ ใมโครคอนโทรลเลอร์มีหลายรุ่น หลายยี่ห้อ หลายผู้ผลิต แต่ละรุ่นอาจเลือกใช้ PCU ที่แตกต่างกัน ตาม คุณสมบัติ หรือตามต้นทุนการผลิต ตามเทคโนโลยีที่ก้นพบ และตามแบบฉบับสถาปัตยกรรมที่มีการจด ลิงสิทธิ์ ไมโครคอนโทรลเลอร์จึงเป็นที่นิยมในนักเขียนโปรแกรมและรวมถึงสถาบันการศึกษาที่กำลัง พัฒนาต่อยอดการเรียนการสอน

1.โครงสร้างทั่วไป

โครงสร้างโดยทั่วไปของไมโครคอนโทรลเลอร์นั้นสามารถแบ่งออกมาได้เป็น 5 ส่วนใหญ่ๆ ดังต่อไปนี้

1. หน่วยประมวลผลกลางหรือซีพียู (CPU : Central Processing Unit)

2. หน่วยความจำ (Memory) สามารถแบ่งออกเป็น 2 ส่วน คือ หน่วยความจำที่มีไว้สำหรับเก็บ โปรแกรมหลัก (Program Memory) เปรียบเสมือนฮาร์คดิสก์ของเครื่องคอมพิวเตอร์ตั้งโต๊ะ คือข้อมูลใดๆ ที่ ถูกเก็บไว้ในนี้จะไม่สูญหายไปแม้ไม่มีไฟเลี้ยง อีกส่วนหนึ่งคือหน่วยความจำข้อมูล (Data Memory) ใช้เป็น เหมือนกับกระคาษทคในการคำนวณของซีพียู และเป็นที่พักข้อมูลชั่วคราวขณะทำงาน แต่หากไม่มีไฟเลี้ยง ข้อมูลก็จะหายไปคล้ายกับหน่วยความจำแรม (RAM) ในเครื่องคอมพิวเตอร์ทั่วๆ ไป แต่สำหรับ ไมโครคอนโทรลเลอร์สมัยใหม่หน่วยความจำข้อมูลจะมีทั้งที่เป็นหน่วยความจำแรม ซึ่งข้อมูลจะหายไปเมื่อ ไม่มีไฟเลี้ยงและเป็นอีอีพรอม (EEPROM : Electrically Erasable Programmable Read-Only Memory) ซึ่ง สามารถเก็บข้อมูลได้แม้ไม่มีไฟเลี้ยงก็ตาม

3. ส่วนติดต่อกับอุปกรณ์ภายนอกหรือพอร์ต (Port) มี 2 ลักษณะคือ พอร์ตอินพุต (Input Port) และ พอร์ตส่งสัญญาณหรือพอร์ตเอาต์พุต (Output Port) ส่วนนี้จะใช้ในการเชื่อมต่อกับอุปกรณ์ภายนอก ถือว่า เป็นส่วนที่สำคัญมากใช้ร่วมกันระหว่างพอร์ตอินพุตเพื่อรับสัญญาณ อาจจะด้วยการกดสวิตช์เพื่อนำไป ประมวลผลและส่งไปยังพอร์ตเอาต์พุตเพื่อแสดงผล เช่น การติดสว่างของหลอดไฟ เป็นต้น

 4. ช่องทางเดินของสัญญาณหรือบัส (BUS) คือเส้นทางการแลกเปลี่ยนสัญญาณข้อมูลระหว่าง ซีพียู หน่วยความจำและพอร์ต เป็นลักษณะของสายสัญญาณจำนวนมากอยู่ภายในตัวไมโครคอนโทรลเลอร์ โดย แบ่งเป็นบัสข้อมูล (Data Bus), บัสแอคเครส (Address Bus) และบัสควบคุม (Control Bus)
 5. วงจรกำเนิดสัญญาณนาฬิกา เป็นองก์ประกอบที่สำคัญมากอีกส่วนหนึ่ง เนื่องจากการทำงานที่ เกิดขึ้นในตัวไมโครคอนโทรลเลอร์จะขึ้นอยู่กับการกำหนดจังหวะ หากสัญญาณนาฬิกามีความถี่สูงจังหวะ การทำงานก็จะสามารถทำได้ถี่ขึ้นส่งผลให้ไมโครคอนโทรลเลอร์ตัวนั้นมีความเร็วในการประมวลผลสูง ตามไปด้วย



2. ตระกูลของไมโครคอนโทรลเลอร์

 Z-80 ไมโครคอนโทรลเลอร์ที่นิยมใช้กัน เริ่มตั้งแต่ตัวแรกที่เป็นลักษณะของ CPU ไม่ถึงขั้น เรียกว่า ไมโครคอนโทรลเลอร์ก็คือ ตระกูล Z80 เป็นลักษณะของ CPU เล็กๆ ที่ต้องอาศัย IO ต่างๆ เพิ่มเติม เข้ามามากจึงทำให้บอร์คมีขนาดค่อนข้างใหญ่ จัดได้ว่าเป็นการเริ่มต้นการเรียนรู้ที่ดีของยุคสมัยนั้นทำให้ได้ เรียนรู้ชุดคำสั่งที่เป็น Op Code



2. MCS-51 บริษัทที่สร้างไมโครคอนโทรลเลอร์ตระกูล MCS-51 เป็นบริษัทแรกคือบริษัท Intel ตระกูล MCS-51 เป็นตระกูลที่พัฒนาต่อจาก Z80 ทำให้การศึกษาเรียนรู้ไมโครคอนโทรลเลอร์ง่ายขึ้น กว่าเดิม ไม่ว่าจะเป็นการเขียนโปรแกรมในลักษณะของ Assembly Code แล้วโหลดลงบอร์ดเพื่อใช้งาน ตลอดจนสถาปัตยกรรมในการออกแบบ ไมโครคอนโทรลเลอร์รุ่นนี้จะช่วยลดอุปกรณ์รอบข้างลงไปได้มาก เหมาะที่จะนำไปใช้งานจริง



รูปไมโครคอนโทรลเลอร์ MCS-51

ตัวอย่างการต่อใช้งาน MCS-51 บนบอร์คทคลองจริงอุปกรณ์รอบข้างจะน้อยกว่า Z-80 มากทำให้ ออกแบบวงจรได้ง่ายขึ้น



รูปตัวอย่างการต่อใช้งาน MCS-51 บนบอร์คทคลอง

3. PIC บริษัท Microchip Technology เป็นผู้สร้างและผลิต PIC เป็นไมโครคอนโทรลเลอร์ยุคต่อมา ที่ได้รับความนิยมสูงอีกตระกูลหนึ่งตั้งแต่อดีตจนถึงปัจจุบัน คำว่า PIC ย่อมาจากคำว่า (Peripheral Interface Controller) ไมโครคอนโทรลเลอร์ตระกูลนี้มีการพัฒนาเทคโนโลยีขึ้นในทุกด้านทำให้ได้รับความนิยมกว่า ไมโครคอนโทรลเลอร์ยุคเก่า เพราะมีอุปกรณ์ต่อพ่วงน้อยประกอบกับมีหน่วยความจำ EEPROM ในตัวจึง ทำให้ง่ายต่อการบันทึกและจัดเก็บข้อมูลและ PORT ต่างๆ ได้มีการ latch ในตัว IC อยู่ แล้วจึงสามารถต่อ ออกมาใช้งานภายนอกได้โดยตรง มีกระแสและแรงดันที่เพียงพอและอีกความสามารถหนึ่งคือ สามารถ อัปโหลดโปรแกรม Boot Loader เข้าไปในไมโครคอนโทรลเลอร์ได้เลย จึงทำให้ง่ายต่อการอัปโหลด โปรแกรม IC เพิ่มเติมอย่างที่ต้องมีกับระบบไมโครคอนโทรลเลอร์รุ่นเก่าอย่าง MCS-51



รูปไมโครคอนโทรลเลอร์ PIC

ตัวอย่างการต่อใช้งาน PIC กับบอร์คทคลองจริง ฟังก์ชันการใช้งานก่อนข้างกรบและ โปรแกรมง่าย โดย CCS, HI-TECH C Compiler, C18 C Compile, C30 C Compiler, MPLAB เป็นต้น



รูปตัวอย่างการต่อใช้งาน PIC บนบอร์ดทดลอง

4. AVR AVR เป็นไมโครคอนโทรลเลอร์รุ่นต่อมาที่มีการพัฒนาต่อมาจาก MCS-51 โดยบริษัท ATMEL เนื่องมาจากว่า MCS-51 ยุคหลังนี้ไม่ค่อยมีคนใช้งานจริงและมีใช้งานแต่เฉพาะใน สถาบันการศึกษา ที่เป็นเช่นนี้ก็เพราะว่าการออกแบบวงจรที่ค่อนข้างยุ่งยากและต้องอาศัยการต่ออุปกรณ์ ร่วมเยอะนั้นเอง ดังนั้น AVR จึงเข้ามาเป็นที่นิยมในการทำงานด้านนี้ โดยคุณสมบัติหลักที่น่าสนใจก็คือ สามารถ Interface ผ่าน USB ได้โดยตรง ซึ่งไมโครคอนโทรลเลอร์ยุคเก่าทำได้โดยต่อผ่านพอร์ต RS-232 แต่ เนื่องด้วยคอมพิวเตอร์ยุคใหม่ พอร์ต RS-232 เริ่มหายาก ดังนั้น AVR จึงได้รับความนิยม



รูปไมโครคอนโทรลเลอร์ AVR

้ตัวอย่าง การต่อ AVR ใช้งานบนบอร์ดที่มีลักษณะใกล้เคียงกับไมโครคอมพิวเตอร์



รูปตัวอย่างการต่อใช้งาน AVR

5. Arduino Arduino เป็นไมโครคอนโทรลเลอร์บอร์คแบบสำเร็จรูปในยุคปัจจุบัน ซึ่งถูกสร้างจาก Controller ตระกูล ARM ของ ATMEL ข้อคีของไมโครคอนโทรลเลอร์บอร์คคือ เรื่องของ Open Source ที่ สามารถนำไปพัฒนาต่อเป็นอุปกรณ์ต่างๆ ได้และความสามารถในการเพิ่ม Boot Loader เข้าไปที่ตัว ARM จึงทำให้การ Upload Code เข้าตัวบอร์คสามารถทำได้ง่ายขึ้นและยังมีการพัฒนา Software ที่ใช้ในการ ควบคุมตัวบอร์คของ Arduino มีลักษณะเป็นภาษา C++ ที่นักเขียนโปรแกรมมีความคุ้นเคยในการใช้งาน อีก ทั้งตัวบอร์คสามารถนำโมดูลมาต่อเพิ่มเพื่อเพิ่มศักยภาพมากยิ่งขึ้นซึ่งทาง Arduino เรียกว่า shield

3. Arduino คืออะไร

Arduino เป็นบอร์คไมโครคอนโทรเลอร์ตระกูล AVR ที่มีการพัฒนาแบบ Open Source คือ มีการ เปิดเผยข้อมูลทั้งค้าน Hardware และ Software ตัวบอร์ค Arduino ถูกออกแบบมาให้ใช้งานได้ง่ายคังนั้นจึง เหมาะสำหรับผู้เริ่มต้นศึกษา ทั้งนี้นักเขียนโปรแกรมยังสามารถคัดแปลง เพิ่มเติม พัฒนาต่อยอคทั้งตัวบอร์ค หรือโปรแกรมต่อได้อีกค้วยที่สำคัญ Platform ของไมโครคอนโทรลเลอร์อย่าง Arduino ง่ายต่อการใช้งาน ซึ่งถือเป็นจุคมุ่งหมายของการสร้างไมโครคอนโทรลเลอร์สำหรับผู้คนทั่วโลก "โคยเฉพาะเค็กและเยาวชน ที่ มีความสนใจในเทคโนโลยี" ทำให้บอร์ค Arduino ถูกพัฒนาด้วยความร่วมมือจากผู้ที่สนใจทั่วโลกและยัง จำหน่ายในราคาที่เด็กและเยาวชนสามารถจับต้องได้



รูปไมโครคอนโทรลเลอร์Arduino Uno r3

อีกคุณสมบัติที่สำคัญของบอร์ค Arduino คือ เป็นบอร์คอิเล็กทรอนิกส์ขนาคเล็กขนาคเท่ากับบัตร ประชาชน (หรือเล็กกว่าด้วยซ้ำ) โคยส่วนประกอบหลักของบอร์ค คือ ไมโครคอนโทรลเลอร์ และนำมา ประกอบเข้ากับอุปกรณ์อิเล็กทรอนิกส์อื่นๆ และถูกพัฒนาให้ง่ายต่อการใช้งานมากขึ้นเรื่อยๆ ซึ่งทั้งนี้บอร์ค Arduino จะมีหลายรุ่นให้เลือกใช้โคยแต่ละรุ่นของ Arduino อาจมีความแตกต่างกันเริ่มจากขนาคของบอร์ค หรือคุณสมบัติซึ่งเป็นคุณสมบัติพิเศษอย่างอื่น เช่น จำนวนของ Pin แรงคันไฟฟ้าที่ใช้ แรงคันไฟฟ้าที่ Output หรือประสิทธิภาพของไมโครคอนโทรลเลอร์

ซอฟแวร์ Arduino

ซอฟแวร์ Arduino สร้างขึ้นจากภาษา Arduino นั่นก็คือ C/C++ ฉะนั้นจึงต้องใช้ภาษา C ในการเขียน โปรแกรมควบคุมไมโครคอนโทรลเลอร์ โดยการสร้างและเขียนโปรแกรมชุดคำสั่ง "Sketch Code" ฉะนั้น จึงต้องใช้ภาษา C ผ่านโปรแกรม Arduino IDE หรือ Arduino Wed Editor ซึ่งเป็นเครื่องมือสำหรับเขียน โปรแกรมชุดคำสั่งในการควบคุมไมโครคอนโทรลเลอร์ให้เป็นไปตามที่ด้องการ โดยทั้งนี้เครื่องมืออย่าง โปรแกรมArduino IDE และ Arduino Wed Editor จะสามารถรวบรวมและยืนยันโปรแกรมชุดคำสั่ง Sketch Code ได้ ซึ่งหาก Sketch Code มีความผิดพลาดหรือเขียนภาษา C ไม่เข้าใจ ระบบจะมีการแจ้งเตือนให้ นักเขียนโปรแกรมทำการแก้ไขในวรรคที่ผิด หรือบรรทัดที่ระบบไม่เข้าใจทันทีและขั้นตอนสุดท้ายในการ ทำงานของซอฟแวร์ Arduino คือ การ Upload Sketch Code ไปยังบอร์ด Arduino หมายถึง การอัปโหลด โปรแกรมชุดคำสั่ง (Sketch Code) ไปยังไมโครคอนโทรลเลอร์ของบอร์ด Arduino เพื่อให้บอร์ดส่ง-รับ สัญญาณ Data หรือสัญญาณไฟ 3V หรือ 5V ไปยังวงจรโมดูลอิเล็กทรอนิกส์และควบคุมการทำงานให้ เป็นไปตามที่นักเขียนโปรแกรมต้องการ Arduino IDE คือ เครื่องมือการเขียนโปรแกรมที่มีใช้งานได้กับ Arduino ได้ทุกรุ่นโดยภายในจะ ประกอบด้วยเครื่องมือที่จำเป็นสำหรับติดต่อ Arduino เช่น การก้นหา Arduino ที่ติดต่อกับเครื่อง กอมพิวเตอร์ การเลือกรุ่น Arduino ที่ต่ออยู่เพื่อตรวจสอบว่าขนาดของโปรแกรมที่เขียนหรือ Library ต่างๆ ซับพอร์ตกับ Arduino รุ่นนั้นๆ หรือไม่อีกทั้งยังมีโปรแกรมติดต่อผ่าน Serial Monitorโดยตรงสำหรับ กอมพิวเตอร์



รูปหน้าต่างโปรแกรม Arduino IDE

สำหรับการเชื่อมต่อบอร์ด Arduino กับโปรแกรม Arduino IDE จะคล้ายกับการเชื่อมต่อ Printter หรือเชื่อมต่อโทรศัพท์สมาร์ตโฟนกับเครื่องคอมพิวเตอร์โดยการเชื่อมต่อบอร์ด Arduino จะเชื่อมต่อโดย สาย USB กับพอร์ตของเครื่องคอมพิวเตอร์ จากนั้นเข้าไปเลือกพอร์ตในโปรแกรม Arduino IDE โดยเฉพาะ ระบบคอมพิวเตอร์ที่มีประสิทธิภาพระดับสูง เช่น Windows 10 และโปรแกรม USB ที่มีความเสถียรโดย นัก เขียนโปรแกรมไม่จำเป็นด้องค้นหาพอร์ตเพราะระบบจะค้นหาให้อัตโนมั โปรแกรม Arduino IDE เป็นโปรแกรม Open soft สามารถนำไปใช้งานได้โดยไม่มีค่าใช้จ่าย อีกทั้งยัง มี soft code ตัวอย่างให้ทดสอบกับเซ็นเซอร์ต่างๆ เช่นโปรแกรมไฟกระพริบโปรแกรมวัดอุณหภูมิ และ สามารถดาวน์โหลดได้โดยไม่มีค่าใช้จ่าย

หน่วยการเรียนรู้ที่ 1 ไมโครคอนโทรลเลอร์และบอร์ด Arduino เรื่องย่อยที่ 2 ที่มาของบอร์ด Arduino และทำความรู้จักบอร์ด Arduino

1. ที่มาของบอร์ด Arduino

โครงการ Arduino เดิมก่อตั้งมาด้วยผู้ร่วมก่อตั้ง 5 คน ใด้แก่ Massimo Banzi, David Cuartielles, David Mellis, Tom Igoe, และ Gianluca Martino โดยเริ่มโครงการมาตั้งแต่ช่วงปี 2005 ความหมายของคำว่า Arduino แปลว่า เพื่อนแท้ (Strong friend หรือ Brave friend) ในภาษาอิตาลีโดยผู้ก่อตั้งมีความตั้งใจให้ราคา ของอุปกรณ์นั้นถูกเมื่อเทียบกับไมโครคอนโทรลเลอร์ตระกูลอื่น ๆ เพื่อให้ทุกคนสามารถเข้าถึงได้โดยง่าย แพลตฟอร์ม Arduino ได้ออกแบบมาเพื่อให้ใช้งานง่ายนักเขียนโปรแกรมไม่จำเป็นต้องมีความรู้เกี่ยวกับ โครงสร้างสถาปัตยกรรมภายในซีพียูโดยรู้เพียงว่าบอร์ด Arduino ที่เลือกมาใช้งานนั้นมีช่องสัญญาณที่ใช้ งานอะไรบ้างและมีคุณสมบัติต่าง ๆ อะไรบ้างก็สามารถใช้งานได้แล้ว

Arduino ถูกใช้งานค้านต่าง ๆ มากมาย เนื่องจากการเขียนโค้คโปรแกรมควบคุมการทำงานของ Arduino มีความง่ายและยืดหยุ่นสามารถใช้งานในระดับสูงได้อีกด้วย เครื่องมือที่ใช้สำหรับเขียนโค้ค ควบคุมมีเวอร์ชันที่สามารถ Run ได้ในทุกระบบปฏิบัติการไม่ว่าจะเป็น Macintosh Windows หรือแม้กระทั้ง Linux ก็ตามทำให้ได้รับความนิยมเป็นอย่างสูง แพลตฟอร์ม Arduino ประกอบด้วย

- ฮาร์ตแวร์

- ซอฟต์แวร์

ฮาร์ตแวร์ (Hardware)

เป็นบอร์ดอิเล็กทรอนิกส์ขนาดเล็กที่มีไมโครคอนโทรลเลอร์เป็นชิ้นส่วนหลักประกอบร่วมกับ อุปกรณ์อิเล็กทรอนิกส์อื่น ๆ เพื่อให้ง่ายต่อการใช้งานหรือที่เรียกกันว่า "บอร์ด Arduino" โดยบอร์ด Arduino ก็มีหลายรุ่นให้เลือกใช้ตามความเหมาะสมของงานโดยในแต่ละรุ่นอาจมีความแตกต่างกันในเรื่องของขนาด ของบอร์ดหรือกุณสมบัติ เช่น จำนวนของขารับส่งสัญญาณ แรงดันไฟที่ใช้ ประสิทธิภาพของ MCU เป็นด้น ซอฟต์แวร์ (Software)

- ภาษาที่ใช้เขียนคำสั่งควบคุมบอร์ด Arduino เป็นภาษาสำหรับเขียนคำสั่งควบคุมที่มีไวยากรณ์แบบ เดียวกับภาษา C/C++

- Arduino IDE เป็นเครื่องมือสำหรับเขียนโค้คโปรแกรม การคอมไพล์โปรแกรม (การแปลงไฟล์ ภาษา C ให้เป็นภาษาเครื่อง) และอัปโหลคโปรแกรมลงบอร์ค



2. ทำความรู้จักบอร์ด Arduino

Arduino เป็นคอมพิวเตอร์ขนาดเล็กที่ทำหน้าที่เป็นสมองของระบบแบบฝังตัวโดย Arduino เป็น แพลตฟอร์มฮาร์ดแวร์ และมีสภาพแวคล้อมในการพัฒนาซอฟแวร์ และการทำต้นแบบที่ง่ายต่อการเรียนรู้ และจะเป็นพื้นฐานสำหรับโครงการที่นักพัฒนาจะพัฒนาต่อไป

ใมโครคอนโทรลเลอร์ คือ คอมพิวเตอร์ขนาดเล็กที่มีโปรเซสเซอร์และหน่วยความจำสามารถ ควบคุมการทำงานของอุปกรณ์มากมายที่พบได้ในชีวิตประจำวัน ไมโครคอนโทรลเลอร์บางรุ่นถูกออกแบบ มาเพื่อเชื่อมต่อกับคอมพิวเตอร์สำหรับการเขียนโปรแกรมโดยเฉพาะ ซึ่งสิ่งนี้เป็นวัตถุประสงค์เฉพาะของ Arduino

ใมโครคอนโทรลเลอร์ช่วยให้สามารถสร้างอุปกรณ์อิเล็กทรอนิกส์ได้ง่ายขึ้น เนื่องจากนักพัฒนา สามารถควบคุมพึงก์ชันต่างๆ ผ่านโปรแกรมบนไมโครคอนโทรลเลอร์ สามารถควบคุมทั้งอินพุตและ เอาต์พุต เช่น สามารถทำให้ไฟ LED ที่เชื่อมต่อกับขา Arduino กะพริบเป็นเวลา 5 วินาที แล้วดับ 3 วินาที ซึ่ง หลอด LED ถือเป็นตัวอย่างของเอาต์พุตและยังมีรูปแบบที่ซับซ้อนและอื่นๆ อีกมากมาย

2.1 ทำไมต้อง Arduino

ใมโครคอนโทรลเลอร์ที่เหมาะสมที่สุดสำหรับผู้เริ่มต้นคือ บอร์ด Basic Stamp และบอร์ด Arduino แต่ Basic Stamp ใช้ภาษา Basic ในการเขียนโปรแกรมที่แม้จะง่ายแต่ก่อนข้างจำกัด เมื่อเทียบกับภาษา C ที่ Arduino ใช้ โดยหลักการทำงานแล้วบอร์ด Arduino มีความคล้ายกับบอร์ด Stamp แต่บอร์ด Arduino แก้ปัญหาหลายอย่างของบอร์ด Stamp ออกไป หนึ่งในคุณลักษณะที่สำคัญสำหรับผู้เริ่มต้นคือ ค่าใช้ง่ายที่ ลดลง บอร์ด Arduino รุ่นสำหรับผู้เริ่มต้นมีราคาประมาณ 1 ใน 4 ของบอร์ด Stamp เลยทีเดียว แต่ถึงแม้จะมี ราคาถูกกว่าแต่บอร์ด Arduino กลับมีการประมวลผลที่ทรงพลังมากและมีหน่วยความจำมากกว่า อีกทั้ง Arduino ยังมีขนาดเล็กกว่าบอร์ด Stamp ซึ่งเป็นประโยชน์ในหลายโครงการ บอร์ด Stamp จะมีขนาดใหญ่ กว่าบอร์ด Arduino ดังภาพ


อีกหนึ่งข้อดีคือ สภาพแวคล้อมในการเขียนโปรแกรมของ Arduino เป็นแบบ Open Soft ที่สามารถ ติดตั้งได้บนระบบปฏิบัติการ Windows, Mac, OS X และ Linux

รุ่นต่างๆ ของบอร์ค Arduino มีคังต่อไปนี้

หมายเหตุ (บอร์คบางรุ่นใช้ชื่อเดียวกันแต่มีรายละเอียดของ spec. แตกต่างกัน เช่น บอร์ครุ่น Arduino Pro,

Arduino Nano, LilyPad Arduino)

Board	Micro-controller	Clock Speed	Flash Memory	SRAM	EEPROM
Arduino UNO	ATmega328	16 MHz	32 KB	2 KB	1 KB
Arduino Lenonardo	ATmega32u4	16 MHz	32 KB	2.5 KB	1 KB
Arduino DUE	AT91SAM3X8E	84 MHz	512 KB	96 KB	-
Arduino YUN	ATmega32u4	16 MHz	32 KB	2.5 KB	1 KB
Arduino Mega ADK	ATmega2560	16 MHz	256 KB	8 KB	4 KB
Arduino Ethernet	ATmega328	16 MHz	32 KB	2 KB	1 KB
Arduino Mega 2560	ATmega2560	16 MHz	256 KB	8 KB	4 KB
Arduino BT (Bluetooth)	ATmega328	16 MHz	32 KB	2 KB	1 KB
Arduino Micro	ATmega32u4	16 MHz	32 KB	2.5 KB	1 KB
Arduino Pro Mini	ATmega168	8 MHz	16 KB	1 KB	512 Bytes
Arduino Pro	ATmega168	8 MHz	16 KB	1 KB	512 Bytes
Arduino Pro	ATmega328	16 MHz	32 KB	2 KB	1 KB
Arduino Mini	ATmega328	16 MHz	32 KB	2 KB	1 KB
Arduino Nano	ATmega168	16 MHz	16 KB	1 KB	512 Bytes
Arduino Nano	ATmega328	16 MHz	32 KB	2 KB	1 KB

Arduino Fio	ATmega328P	8 MHz	32 KB	2 KB	1 KB
Arduino Robot	ATmega32u4	16 MHz	32 KB	2.5 KB	1 KB
Arduino Esplora	ATmega32u4	16 MHz	32 KB	2.5 KB	1 KB
LilyPad Arduino USB	ATmega32u4	8 MHz	32 KB	2.5 KB	1 KB
LilyPad Arduino Simple	ATmega328	8 MHz	32 KB	2 KB	1 KB
LilyPad Arduino SimpleSnap	ATmega328	8 MHz	32 KB	2 KB	1 KB
LilyPad Arduino	ATmega168V	8 MHz	16 KB	1 KB	512 Bytes
LilyPad Arduino	ATmega328V	8 MHz	16 KB	1 KB	512 Bytes

ตัวอย่างบอร์ค Arduino มีคังต่อไปนี้

1. Arduino Uno



รูปบอร์ด Arduino Uno R3



รูปบอร์ด Arduino Uno R2



รูปบอร์ด Arduino Uno SMD

<u>ข้อมูลจำเพาะ</u>

ชิปไอซีไมโครคอนโทรเลอร์	ATmega328
ใช้แรงคันไฟฟ้า	5V
รองรับการจ่ายแรงคันไฟฟ้า (ที่แนะนำ)	7 – 12V
รองรับการจ่ายแรงคันไฟฟ้า (ที่จำกัค)	6 – 20V
พอร์ต Digital I/O	14 พอร์ต (มี 6 พอร์ต PWM output)
พอร์ต Analog Input	6 พอร์ต
กระแสไฟที่จ่ายได้ในแต่ละพอร์ต	40mA
กระแสไฟที่จ่ายได้ในพอร์ต 3.3V	50mA
พื้นที่โปรแกรมภายใน	32KB พื้นที่โปรแกรม, 500B ใช้โดย Booloader
พื้นที่แรม	2KB
พื้นที่หน่วยความจำถาวร (EEPROM)	1KB
ความถี่คริสตัล	16MHz
ขนาด	68.6x53.4 mm
น้ำหนัก	25 กรัม

2. Arduino Due



รูปบอร์ด Arduino Due

คำว่า Due เป็นภาษาอิตาลี แปลว่า สอง เป็นรุ่นที่เพิ่มพอร์ตให้มากขึ้นเป็น 54 พอร์ตดิจิตอลอินพุต เอาต์พุต และ 12 พอร์ตอนาล็อกอินพุต 2 พอร์ตอนาล็อกเอาต์พุต เพิ่มพื้นที่โปรแกรมเป็น 512KB สามารถใช้ งานพื้นที่ ได้เต็ม ไม่มี Bootloader เนื่องจากสามารถใช้กับพอร์ต USB ได้ โดยตรงมีขนาดบอร์ด 101.52x53.3mm สามารถใช้ Shields ของ Arduino Uno ได้ แต่บางตัวจำเป็นต้องแก้ขาให้ถูกต้อง จากรูปจะ เห็นได้ว่าบอร์คได้เปลี่ยนมาใช้ชิปไอซีแบบ SMD จึงไม่นิยมนำมาใช้ในแบบ Standalone แต่นิยมนำมาใช้ใน งานที่จำเป็นต้องพื้นที่ โปรแกรมมากขึ้น ทำงานที่ซับซ้อนมากยิ่งขึ้น บอร์ค Arduino Due ใช้ชิปไอซีเบอร์ AT91SAM3X8E ซึ่งเป็นชิปไอซีที่ใช้เทค โนโลยี ARM Core สถาปัตยกรรม 32 บิต เร่งความถี่คริสตอลขึ้น ไปสูงถึง 84Mhz จึงทำให้สามารถใช้งานค้านการคำนวน หรือการประมวลผลอัลกอริทึมได้เร็วกว่า Arduino Uno มาก แต่เนื่องจากชิปไอซีทำงานที่แรงคัน 3.3V คังนั้นการนำไปใช้งานกับเซ็นเซอร์ควรระวังไม่ให้ แรงคัน 5V ใหลเข้าบอร์ค ควรใช้วงจรแบ่งแรงคันเพื่อช่วยให้ลอจิกลดแรงคันลงมาให้เหมาะสม

ชิปไอซีไมโครคอนโทรเลอร์	AT91SAM3X8E
ใช้แรงคันไฟฟ้า	3.3V
รองรับการจ่ายแรงคันไฟฟ้า (ที่แนะนำ)	7 – 12V
รองรับการจ่ายแรงคันไฟฟ้า (ที่จำกัด)	6 – 16V
พอร์ต Digital I/O	54 พอร์ต (มี 12 พอร์ต PWM output)
พอร์ต Analog Input	2 พอร์ต
กระแสไฟฟ้ารวมที่จ่ายได้ในทุกพอร์ต	130mA
กระแสไฟที่จ่ายได้ในพอร์ต 3.3V	800mA
กระแสไฟที่จ่ายได้ในพอร์ต 5V	800mA
พื้นที่โปรแกรมภายใน	512KB พื้นที่โปรแกรม
พื้นที่แรม	2KB
พื้นที่หน่วยความจำถาวร (EEPROM)	96KB
ความถี่คริสตัล	84MHz
ขนาด	101.52x53.3 mm
น้ำหนัก	36 กรัม

3. Arduino Leonardo



รูปบอร์ค Arduino Leonard

บอร์ค Arduino Leonard เป็นบอร์คที่เลือกใช้ชิปไอซีเบอร์ ATmega32u4 ที่รองรับการเชื่อมต่อกับ พอร์ต USB ได้โคยตรง ทำให้บอร์คสามารถเขียนโปรแกรมเพื่อจำลองตัวเองให้เป็นเมาส์ หรือคีย์บอร์คได้ ทำงานที่แรงคัน 5V ทำให้ไม่มีปัญหากับเซ็นเซอร์ หรือ Shields ที่ใช้งานกับ Arduino Uno

ชิปไอซีไมโครคอนโทรเถอร์	ATmega32u4
ใช้แรงคันไฟฟ้า	5V
รองรับการจ่ายแรงคันไฟฟ้า (ที่แนะนำ)	7 – 12V
รองรับการจ่ายแรงคันไฟฟ้า (ที่จำกัด)	6 – 20V
พอร์ต Digital I/O	20 พอร์ต (มี 7 พอร์ต PWM output)
พอร์ต Analog Input	12 พอร์ต
กระแสไฟฟ้ารวมที่ง่ายได้ในทุกพอร์ต	40mA
กระแสไฟที่จ่ายได้ในพอร์ต 3.3V	50mA
พื้นที่โปรแกรมภายใน	32KB แต่ 4KB ถูกใช้โคย Bootloader
พื้นที่แรม	2.5KB
พื้นที่หน่วยความจำถาวร (EEPROM)	1KB
ความถี่คริสตัล	16MHz
ขนาด	68.6x53.3 mm
น้ำหนัก	20 กรัม

4. Arduino MEGA ADK



รูปบอร์ด Arduino MEGA ADK R3

บอร์ค Arduino MEGA ADK ใช้ชิปไมโครคอนโทรเลอร์เบอร์ ATmaega2560 มีชิปไอซี USB Host เบอร์ MAX3421e มาให้บนบอร์ค ใช้สำหรับเชื่อมต่อกับโทรศัพท์มือถือแอนครอยผ่าน OTG มีพอร์ค คิจิตอลอินพุตเอาต์พุตจำนวน 54 พอร์ต มีอนาล็อกอินพุต 16 พอร์ต ทำงานที่ความถี่ 16MHz บอร์ค Arduino MEGA ADK จะแตกต่างกับบอร์ค Arduino Due ตรงที่ชิปบนบอร์คนั้นฉลาคไม่เท่า และใช้ความถี่ต่ำกว่า คังนั้นจึงไม่เหมาะจะที่นำไปใช้กับงานคำนวณ แต่เหมาะสำหรับงานที่ใช้การเชื่อมต่อกับโทรศัพท์มือถือ แอนครอยมากกว่า

ชิปไอซีไมโครคอนโทรเลอร์	ATmega2560
ใช้แรงคันไฟฟ้า	5V
รองรับการจ่ายแรงคันไฟฟ้า (ที่แนะนำ)	7 – 12V
รองรับการจ่ายแรงคันไฟฟ้า (ที่จำกัค)	6 – 20V
พอร์ต Digital I/O	54 พอร์ต (มี 15 พอร์ต PWM output)
พอร์ต Analog Input	16 พอร์ต
กระแสไฟฟ้ารวมที่จ่ายได้ในทุกพอร์ต	40mA
กระแสไฟที่จ่ายได้ในพอร์ต 3.3V	50mA
พื้นที่โปรแกรมภายใน	256KB แต่ 8KB ถูกใช้โคย Bootloader
พื้นที่แรม	8KB
พื้นที่หน่วยความจำถาวร (EEPROM)	4KB
ความถี่คริสตัล	16MHz
ขนาด	101.52x53.3 mm
น้ำหนัก	36 กรัม

5. Arduino Mega 2560



รูปบอร์ด Arduino Mega 2560 R3

บอร์ค Arduino Mega 2560 จะเหมือนกับ Arduino MEGA ADK ต่างกันตรงที่บนบอร์คไม่มี USB Host เพื่ออัปโหลดโปรแกรมยังต้องทำผ่านโปรโตคอล UART อยู่บนบอร์คใช้ชิปไอซีไมโครคอนโทรเลอร์ เบอร์ ATmega2560

ชิปไอซีไมโครคอนโทรเลอร์	ATmega2560
ใช้แรงดันไฟฟ้า	5V
รองรับการจ่ายแรงคันไฟฟ้า (ที่แนะนำ)	7 – 12V
รองรับการจ่ายแรงคันไฟฟ้า (ที่จำกัค)	6-20V
พอร์ต Digital I/O	54 พอร์ต (มี 15 พอร์ต PWM output)
พอร์ต Analog Input	16 พอร์ต
กระแสไฟฟ้ารวมที่จ่ายได้ในทุกพอร์ต	40mA
กระแสไฟที่จ่ายได้ในพอร์ต 3.3V	50mA
พื้นที่โปรแกรมภายใน	256KB แต่ 8KB ถูกใช้โคย Bootloader
พื้นที่แรม	8KB
พื้นที่หน่วยความจำถาวร (EEPROM)	4KB
ความถี่คริสตัล	16MHz

6. Arduino Micro



รูปบอร์ด Arduino Micro

บอร์ด Arduino Micro ออกแบบให้มีขนาดที่เล็กและทันสมัยกว่าบอร์ด Arduino Mini หรือ Arduino Nano เนื่องจากบนบอร์ดใช้ชิปไอซีไมโครคอนโทรเลอร์เบอร์ ATmega32u4 ซึ่งมีพอร์ต USB สามารถ เชื่อมต่อกับคอมพิวเตอร์ได้โดยตรง และมีดิจิตอลอินพุตเอาต์พุตมากถึง 20 พอร์ต มีพื้นที่เก็บโปรแกรม ขนาด 32KB แต่ต้องใช้พื้นที่สำหรับ Bootloader ไป 4KB มีขนาดเพียง 48x18mm เนื่องจากบอร์ดใช้ชิปไอซี ตัวเดียวกับ Arduino Leonardo ทำให้สามารถทำให้บอร์ดจำลองตัวเองเป็นเมาส์หรือคีย์บอร์ดเชื่อมต่อกับ กอมพิวเตอร์ได้

ชิปไอซีไมโครคอนโทรเลอร์	ATmega32u4
ใช้แรงคันไฟฟ้า	5V
รองรับการจ่ายแรงคันไฟฟ้า (ที่แนะนำ)	7 – 12V
รองรับการจ่ายแรงคันไฟฟ้า (ที่จำกัค)	6 – 20V
พอร์ต Digital I/O	20 พอร์ต (มี 7 พอร์ต PWM output)
พอร์ต Analog Input	12 พอร์ต
กระแสไฟที่จ่ายได้ในแต่ละพอร์ต	40mA
กระแสไฟที่จ่ายได้ในพอร์ต 3.3V	50mA
พื้นที่โปรแกรมภายใน	32KB พื้นที่โปรแกรม , 4KB ใช้โดย
	Booloader
พื้นที่แรม	2.5KB
พื้นที่หน่วยความจำถาวร (EEPROM)	1KB
ความถี่คริสตัล	16MHz
ขนาด	48x18 mm
น้ำหนัก	13 กรัม

7. Arduino Nano



รูปบอร์ค Arduino Nano

บอร์ด Arduino Nano ออกแบบมาให้มีขนาดเล็กและใช้กับงานทั่วๆ ไปใช้ชิปไอซี ใมโครคอนโทรเลอร์เบอร์ ATmega168 หรือเบอร์ ATmega328 (มีรุ่น 2.3 กับ 3 ควรตรวจสอบก่อนซื้อ) โปรแกรมผ่านโปรโตคอล UART มีชิปUSB to UART มาให้ ใช้ Mini USB เชื่อมต่อกับคอมพิวเตอร์ มี พอร์ตดิจิตอลอินพุตเอาต์พุต 14 พอร์ต มีพอร์ตอนาลีอกอินพุต 8 พอร์ต บนบอร์คยังมีเรกกูเลเตอร์ สามารถ ง่ายไฟได้ตั้งแต่ 7 – 12V เพื่อให้บอร์ดทำงานได้ (ง่ายไฟที่งา VIN)กรณีมีแหล่งง่ายไฟ 5V อยู่แล้วก็ง่ายเง้า ได้เลยที่งา 5V

ชิปไอซีไมโครคอนโทรเลอร์	ATmega168 หรือ ATmega328
ใช้แรงคันไฟฟ้า	5V
รองรับการจ่ายแรงคันไฟฟ้า (ที่แนะนำ)	7 – 12V
รองรับการจ่ายแรงคันไฟฟ้า (ที่จำกัค)	6 – 20V
พอร์ต Digital I/O	14 พอร์ต (มี 6 พอร์ต PWM output)
พอร์ต Analog Input	6 พอร์ต
กระแสไฟที่จ่ายได้ในแต่ละพอร์ต	40mA
กระแสไฟที่จ่ายได้ในพอร์ต 3.3V	50mA
พื้นที่โปรแกรมภายใน	16KB หรือ 32KB พื้นที่โปรแกรม, 500B ใช้โดย
	Booloader
พื้นที่แรม	1 หรือ 2KB
พื้นที่หน่วยความจำถาวร (EEPROM)	512B หรือ 1KB
ความถี่คริสตัล	16MHz
ขนาด	45x18 mm
น้ำหนัก	5 กรัม



รูปบอร์ด Arduino Mini

บอร์ค Arduino Mini มีขนาดเล็กกว่าบอร์ค Arduino อื่นๆอยู่มาก แต่ยังคงความสามารถไว้เท่ากับ บอร์ค Arduino Uno R3 แถมยังมีพอร์ต A6 และ A7 เพิ่มขึ้นมาทำให้บอร์คมีอนาล็อกอินพุตเพิ่มมากขึ้น จาก เดิมมี 6 พอร์ต เพิ่มเป็น 8 พอร์ต เนื่องจากบอร์ค Arduino Mini เน้นที่ขนาดเล็ก คังนั้นจึงไม่สามารถทำการ อัปโหลคโปรแกรมได้โดยตรง หากต้องการอัปโหลคโปรแกรมบอร์คจำเป็นต้องซื้อโมคูล USB to UART มาใช้แยกตางหาก แต่ข้อคีของการไม่สามารถอัปโหลคโปรแกรมได้โดยตรงคือหากโครงการอยู่ตัวแล้ว ความเสี่ยงที่จะถูกนำมาเขียนโปรแกรมเข้าไปใหม่ก็จะลคน้อยลง บอร์ค Arduino Mini ยังคงมีรูปแบบ กล้ายๆกับ Arduino เดิม คือใช้ชิป ATmega328 ที่ความถี่ 16MHz ภายในบอร์คสามารถใช้แหล่งจ่ายไฟ 7 – 12V มาจ่ายได้ หากมีแหล่งจ่ายไฟ 5V ก็สามารถนำมาจ่ายได้เลย

ชิปไอซีไมโครคอนโทรเลอร์	ATmega328
ใช้แรงคันไฟฟ้า	5V
รองรับการจ่ายแรงคันไฟฟ้า (ที่แนะนำ)	7 – 12V
รองรับการจ่ายแรงคันไฟฟ้า (ที่จำกัค)	6 – 20V
พอร์ต Digital I/O	14 พอร์ต (มี 6 พอร์ต PWM output)
พอร์ต Analog Input	8 พอร์ต
กระแสไฟที่จ่ายได้ในแต่ละพอร์ต	40mA
พื้นที่โปรแกรมภายใน	32KB พื้นที่โปรแกรม, 2KB ใช้โคย Booloader
พื้นที่แรม	2KB
พื้นที่หน่วยความจำถาวร (EEPROM)	1KB
ความถี่คริสตัล	16MHz
ขนาด	30x18 mm
น้ำหนัก	ไม่ระบุ



รูปบอร์ค Arduino Pro Mini

บอร์ด Arduino Pro Mini เป็นบอร์ดที่แตกต่างจากบอร์ด Arduino Mini คือย้ายช่อง A4 A5 A6 A7 ออกมาภายในบอร์ด เพื่อให้บอร์ดมีขนาดที่เล็กลงกว่าเดิมอีกและมีให้เลือกใช้ทั้ง 5V และ 3.3V ก่อนใช้งาน จึงควรตรวจสอบให้แน่ใจเสียก่อน บอร์ด Arduino Pro Mini ได้ใช้ไอซีเบอร์ ATmega328 เช่นเดิม แต่มี ขนาดใหญ่ขึ้นเล็กน้อยทำให้ไอซีดูเต็มบอร์ดมากขึ้นและในโมเดลที่ใช้แรงดันไฟ 3.3V ลดความถื่ลงเป็น 8MHz ใช้พื้นที่ Booloader น้อยลง เหลือเพียง 500B การโปรแกรมยังคงต้องใช้โมดูล USB to UART ในการ เชื่อมต่อเพื่อโปรแกรมเช่นเดิม

ชิปไอซีไมโครคอนโทรเลอร์	ATmega328
ใช้แรงคันไฟฟ้า	3.3V หรือ 5V
รองรับการจ่ายแรงดันไฟฟ้า	3.35 – 12V (ในโมเคลใช้ไฟ 3.3V) หรือ 5 – 12V (ในโมเคลใช้
	ใฟ 5V)
พอร์ต Digital I/O	14 พอร์ต (มี 6 พอร์ต PWM output)
พอร์ต Analog Input	6 พอร์ต
กระแสไฟที่จ่ายได้ในแต่ละพอร์ต	40mA
พื้นที่โปรแกรมภายใน	32KB พื้นที่โปรแกรม, 500B ใช้โดย Booloader
พื้นที่แรม	2KB
พื้นที่หน่วยความจำถาวร (EEPROM)	1KB
ความถี่คริสตัล	8MHz (ในโมเคลใช้ไฟ 3.3V) หรือ 16MHz (ในโมเคลใช้ไฟ
	5V)

ตัวอย่างบอร์ครุ่นอื่นๆ



รูปบอร์ค LilyPad Arduino USB











รูปบอร์ด LilyPad Arduino SimpleSnap



หน่วยการเรียนรู้ที่ 1 ไมโครคอนโทรลเลอร์และบอร์ด Arduino เรื่องย่อยที่ 3 ส่วนประกอบของบอร์ด Arduino และการเริ่มทำงานกับบอร์ด Arduino

1. ส่วนประกอบบอร์ด Arduino

ปัจจุบันบอร์คไมโครคอนโทรลเลอร์ Arduino ได้พัฒนาไปถึงรุ่น Arduino Uno ซึ่งตอนนี้รุ่นใหม่ ล่าสุด คือ Arduino Uno R3 หมายถึง บอร์ค Arduino รุ่นที่มีการแก้ไขปรับปรุงเป็นครั้งที่ 3 "Third Revision" โดยบอร์ค Arduino รุ่นคั้งเคิมที่คนไทยรู้จักคือ Arduino Uno R2 แต่วิวัฒนาการของ Arduino นั้นเริ่มตั้งแต่ ปี ค.ศ. 2005

สำหรับบอร์ค Arduino Uno R3 นั้นแบ่งออกเป็น 2 เวอร์ชัน ตามชนิคของชิปไมโครคอนโทรลเลอร์ กือ Arduino Uno R3 แบบใช้ชิป MCU แบบ DIP หรือ Dual Inline Package ซึ่งชิปไมโครคอนโทรลชนิคนี้ จะสามารถถอคเปลี่ยนได้ โคยชิปจะเสียบเข้าไปใน Socket ที่บัคกรีติคกับบอร์ค ส่วนบอร์ค Arduino Uno R3 อีกเวอร์ชันหนึ่ง คือ Arduino Uno R3 SMD จะใช้ชิป MCU แบบ SMD หรือ Surface Mount Device สำหรับชิปไมโครคอนโทรลเลอร์ชนิคนี้จะบัคกรีติคกับบอร์ค ซึ่งไม่สามารถถอคเปลี่ยนได้ และแน่นอน บอร์ค Arduino ทั้ง 2 เวอร์ชัน มีต้นทุนที่ถูกกว่า ชิปชนิค DIP ที่มาสามารถถอคเปลี่ยนได้แต่นอกเหนือจากนี้ บอร์ค Arduino ทั้ง 2 เวอร์ชัน มีคุณสมบัติเหมือนกันทั้งหมด



รูป Arduino Uno R3 MCU DIP



รูป Arduino Uno R3 MCU SMD

2. ส่วนประกอบของบอร์ด Arduino UNO R3 MCU SMD



รูปส่วนประกอบของบอร์ค Arduino UNO R3

จากรูปมีรายละเอียดส่วนประกอบเบื้องต้นดังนี้

- 1. ปุ่มรีเซ็ต (Reset Button)
- 2. ช่องสำหรับเสียบสาย USB สำหรับต่อกับ Computer (USBPort)
- 3. ช่องเสียบสำหรับแรงคันไฟฟ้ากระแสตรง 7-12 โวลต์
- 4. ช่องเสียบงาแหล่งจ่ายแรงคันไฟฟ้ากระแสตรง 3.3 โวลต์, 5 โวลต์ และกราวค์ (GND)
- 5. ช่องเสียบบาอินพุตสัญญาณแอนะล็อก (Analog) ตั้งแต่บา A0-A5
- 6. ไมโครคอนโทรลเลอร์ MCU: Atmega328 เป็น MCU ที่ใช้บนบอร์ค Arduino
- 7. พอร์ต ICSP : Atmega328 เป็นพอร์ตที่ใช้โปรแกรม Bootloader
- 8. พอร์ต I/O มีตั้งแต่ขา D0 ถึง D13
 - 0,1 Serial
 - 2 Interrupt 0
 - 3 Interrupt 1
 - 10 (SPI) ss
 - 11–(SPI) MOSI
 - 12 (SPI) MOSI

3. ความรู้เรื่องระบบตัวเลขและรหัส

แนวกิดเกี่ยวกับจำนวนและการนับมีมาแต่โบราณมนุษย์สามารถนับจำนวนสิ่งต่าง ๆ โดยมีความกิด ว่าเมื่อเพิ่มสิ่งใดก็จะ ได้สิ่งนั้นมากขึ้นและถ้าเอาสิ่งนั้นออกไปจะทำให้สิ่งนั้นลดลง เมื่อสังคมเจริญขึ้นก็ใช้ วิธีการจับคู่สิ่งนั้น เช่น อาจใช้นิ้วมือแทนสิ่งของ เช่น หนึ่งนิ้วแทนสัตว์หนึ่งตัวและอาจใช้ก้อนหินบันทึก จำนวนสิ่งนั้น หรืออาจใช้ขีดเขียนบนพื้นดิน หรือสลักบนต้นไม้ ต่อมาจึงใช้สัญลักษณ์แทนจำนวน ขึ้นและพบว่าการนับและการเขียนในสมัยโบราณนั้นจะไม่มี "ศูนย์"

มีหลักฐานยืนยันเมื่อปี ค.ศ. ที่ 9 ใช้เรียกเป็นตัวเลขอาระบิกเป็นชื่อที่เรียกตัวเลขที่ประดิษฐ์ขึ้นใน ประเทศอินเดียยุคโบราณโดยนักคณิตศาสตร์ชาวอินเดียนประมาณ 500 ปี ก่อนคริสตศักราชจากนั้นจึงไป ถ่ายทอดไปยังประเทศในแถบยุโรปในยุคกลางทั้ง ๆ ที่เริ่มต้นที่ประเทศอินเดีย แต่เริ่มรู้จักในฝั่งตะวันตกใน ชื่อว่า ตัวเลขอาระบิก ทั้งนี้เนื่องจากในยุโรปได้ใช้ตำราของอาหรับ (Arabic texts) เช่น " Thecalculation with Hindu numerals" เป็นต้น ทำให้ชาวยุโรปพากันเรียกตัวเลขนี้ว่าตัวเลขอาระบิกและในปัจจุบันจึงเรียก ตัวเลขนี้ว่าตัวเลขอนดูอาระบิก ด้วยตัวเลขอาระบิกนี้จึงเรียกในภาษาอาระบิกว่า Hindunumerals ต่อมา เรียกว่า Hindu-Arabic numerals หรือ Indian numerals อันเป็นสัญลักษณ์ที่ใช้แทนจำนวนกันทั่วโลก

ตัวเลขที่ใช้อยู่ในปัจจุบันซึ่งเรียกว่าตัวเลขฮินดูอาระบิกนี้ใช้แทนด้วยตัวเลข 10 ตัว คือ 1, 2, 3, 4,5, 6, 7, 8, 9 และ 0 ตัวเลขเหล่านี้ได้มีขึ้นในยุโรปในศตวรรษที่ 12 โดย Leonardo Pissano ซึ่งเป็นนักคณิตศาสตร์ ชาวอิตาเลียน เขาได้รับการศึกษาจากประเทศในแถบแอฟริกาเหนือแล้วนำความรู้กลับมาใช้ในประเทศ อิตาลี ระบบตัวเลขฮินดูเป็นระบบที่มีค่าประจำตำแหน่งค่าของตัวเลขขึ้นกับตำแหน่งของตัวเลขนั้น เช่น "2" ใน 205 มีค่าเป็น 10 เท่าของ "2" ใน 25 ดังนั้นจึงต้องมี "0" เพื่อบอกตำแหน่งหรือหลักสิบ ใน"205" ด้วย การก้นพบตัวเลข 0 เกิดขึ้นหลังจากมีสัญลักษณ์แทนจำนวนเต็ม 9 จำนวนแล้วตัวเลข 0ใช้แทนตำแหน่งว่าง เปล่าเรียกว่า "ตัวรั้งตำแหน่ง" (Place holder)

จากความก้าวหน้าทางเทคโนโลยีได้มีการพัฒนาเครื่องคอมพิวเตอร์ขึ้นมาใช้งาน โดยเครื่อง กอมพิวเตอร์เป็นเครื่องใช้ไฟฟ้าซึ่งถูกออกแบบมาเพื่อที่จะรับรู้สภาวะเพียงสองสภาวะเท่านั้น เพื่อป้องกัน การผิดพลาดของข้อมูลคือ สภาวะมีแรงคันไฟฟ้าโดยเรียกว่าลอจิก 1 และสภาวะไม่มีแรงคันไฟฟ้าโดย เรียกว่าลอจิก 0 ในระบบคอมพิวเตอร์ที่มีความเร็วต่ำ แรงคันไฟฟ้าสำหรับลอจิก 1 จะมีค่าเท่ากับ 5 โวลต์ และแรงคันไฟฟ้าสำหรับลอจิก 0 จะมีค่าเท่ากับ 0 โวลต์ แต่สำหรับคอมพิวเตอร์ที่มีความเร็วสูงค่าระดับ แรงคันลอจิก 1 จะลดลงเพื่อลดความร้อนที่เกิดขึ้นกับตัวประมวลผลกลาง ดังนั้นการประมวลผลต่าง ๆ ของคอมพิวเตอร์จึงใช้งานในระบบตัวเลขฐาน 2 หรือที่เรียกว่าเลข ใบนารี่ (Binary) ตัวเลขที่ใช้ระบบเลขฐาน 10 คือ 0, 1, 2, ... 9 และในระบบเลขฐาน 2 มีเพียงเลข 0 กับ 1 เท่านั้น

4. การเริ่มทำงานกับบอร์ด Arduino

เตรียมบอร์ค Arduino และสาย USB ที่ใช้งานร่วมกันได้ เพื่อเชื่อมต่อสื่อสารกับคอมพิวเตอร์ผ่าน USB (สำหรับการอัพโหลคโปรแกรมหรือส่งข้อความโต้ตอบ) นอกจากนี้บอร์คยังสามารถใช้พลังงานไฟฟ้า ผ่าน USB ได้อีกด้วยโคย Uno จะใช้สาย USB-B ดังรูป



5. การต่อวงจรจริง

เพื่อทคลองวงจรจริง (ซึ่งต้องติคตั้งไคร์ฟเวอร์ USB ก่อน) การใช้งานโปรแกรม Arduino IDE มี ขั้นตอนดำเนินการดังนี้

1. เรียกใช้โปรแกรมด้วยการดับเบิลคลิกไฟล์ Arduino.exe ในโฟลเดอร์โปรแกรมที่ได้แตก ไฟล์ไว้

2. ตั้งค่าบอร์ดให้ตรงกับที่ใช้งานซึ่งมีการตั้งค่าอยู่ด้วยกัน 4 รายการดังนี้

- ชนิดของบอร์ค Arduino ที่ใช้งาน (กำลังเชื่อมต่ออยู่)

- กอมพอร์ตที่กำลังเชื่อมต่อ เลือกให้ตรงกับที่บอร์ด Arduino กำลังเชื่อมต่อ (ดูจาก

Device manager)

- ชนิดของเครื่องโปรแกรม เลือก AVRISP mkII



รูป การตั้งค่าบอร์ดที่ใช้งาน

3. ตั้งค่าในโปรแกรม Arduino IDE ให้แสดงตำแหน่งของไฟล์ภาษาเครื่อง (HEX file) หลังจาก การคอมไพล์ผ่าน โดยการตั้งค่าให้คลิกเลือก File --> Preferences คลิกเลือกแสดงผลทั้ง 3 รายการดังรูป

Settings Network		
Sketchbook location:		
C:\Users\OFFBKK\Documents	Varduino	Browse
Editor language:	System Default v (requires restart of Arduino)	
Editor font size:	16	
Interface scale:	Automatic 100 * % (requires restart of Arduino)	
Theme: 2	Default theme v (requires restart of Arduino)	
Show verbose output during:		
Compiler gs:	None 🗸	
Display line numbers		
Enable Code Folding		
Verify code after upload		

รูป ตำแหน่งของไฟล์ภาษาเครื่อง (HEX file)

4. ต่อวงจรจริง โดยมีอุปกรณ์ดังต่อไปนี้





รูป ต่อวงจรจริง

5. เขียนโค้คโปรแกรมที่ต้องการในหน้าต่างโปรแกรม



 6. ในกรณีที่ต้องการคอม ไพล์เพื่อดูผลของการเขียน โปรแกรมว่าถูก ไวยกรณ์หรือ ไม่ หรือต้องการ ไฟล์ภาษาเครื่อง (HEX file) ไปจำลองการทำงานใน โปรแกรมจำลองการทำงาน เมื่อเขียน โค้ด โปรแกรม เสร็จสิ้นแล้วให้คลิกที่ปุ่มคอม ไพล์



รูป คลิกปุ่มคอมไพล์หลังเขียนโค้คโปรแกรมเสร็จสิ้น

7. เลือก Board และ Port ที่เมนู Tools คังรูป

Tools	Help		
	Auto Format	Ctrl+T	
	Archive Sketch		
	Fix Encoding & Reload		
	Manage Libraries	Ctrl+Shift+I	
	Serial Monitor	Ctrl+Shift+M	
	Serial Plotter	Ctrl+Shift+L	
	WiFi101 / WiFiNINA Firmware Updater		
	Board: "Arduino/Genuino Uno"		>
	Port: "COM4"		>

8. คลิกที่ปุ่ม Upload ไปยังบอร์ค คังรูป



รูป สัญลักษณ์อัปโหลดโปรแกรม

หน่วยการเรียนรู้ที่ ๒ พัฒนาการของบอร์ด Arduino Uno R3

กลุ่มสาระการเรียนรู้วิทยาศาสตร์ สาระที่ ๔ เทคโนโลยี

มาตรฐาน ว ๔.๑ เข้าใจแนวคิดหลักของเทคโนโลยีเพื่อการดำรงชีวิตในสังคมที่มีการเปลี่ยนแปลงอย่าง รวดเร็ว ใช้ความรู้และทักษะทางด้านวิทยาศาสตร์ คณิตศาสตร์ และศาสตร์อื่น ๆ เพื่อแก้ปัญหาหรือพัฒนา งานอย่างมีความคิดสร้างสรรค์ด้วยกระบวนการออกแบบเชิงวิศวกรรมเลือกใช้เทคโนโลยีอย่างเหมาะสม โดยคำนึงถึงผลกระทบต่อชีวิต สังคม และสิ่งแวดล้อม

ตัวชี้วัด

ວ ໔.໑ ຆ.ຓ/໑	วิเคราะห์สาเหตุ หรือปัจจัยที่ส่งผลต่อการเปลี่ยนแปลงของเทคโนโลยี และ
	ความสัมพันธ์ของเทคโนโลยีกับศาสตร์อื่น โดยเฉพาะวิทยาศาสตร์ หรือคณิตศาสตร์
	เพื่อเป็นแนวทางการแก้ปัญหาหรือพัฒนางาน
ີງ ໔.໑ ມ.໑/໑	ออกแบบวิธีการแก้ปัญหา โดยวิเคราะห์ เปรียบเทียบ และตัดสินใจเลือกข้อมูลที่จำเป็น
	ภายใต้เงื่อนไขและทรัพยากรที่มีอยู่ นำเสนอแนวทางการแก้ปัญหาให้ผู้อื่นเข้าใจค้วย
	เทคนิคหรือวิธีการที่หลากหลาย วางแผนขั้นตอนการทำงานและคำเนินการแก้ปัญหา
	อย่างเป็นขั้นตอน
ີ] ໔.໑ [ູ] ຟ.໑/໕	ใช้ความรู้ และทักษะเกี่ยวกับวัสคุ อุปกรณ์ เครื่องมือ กลไก ไฟฟ้า และอิเล็กทรอนิกส์
	เพื่อแก้ปัญหาหรือพัฒนางานได้อย่างถูกต้อง เหมาะสม และปลอดภัย
ી હ.૭ મ.હ/૭	วิเกราะห์แนวกิดหลักของเทคโนโลยี ความสัมพันธ์กับศาสตร์อื่น โดยเฉพาะ
	วิทยาศาสตร์ หรือคณิตศาสตร์ รวมทั้งประเมินผลกระทบที่จะเกิดขึ้นต่อมนุษย์ สังคม
	เศรษฐกิจ และสิ่งแวคล้อม เพื่อเป็นแนวทางในการพัฒนาเทคโนโลยี

สาระสำคัญ

เรียนรู้บอร์คไมโครคอนโทรลเลอร์ Arduino Uno R3 ที่ใช้ ATmegaต๒๘ (แผ่นข้อมูล) และในเรื่อง คุณลักษณะ การแปลงสัญญาณ Analog (อนาล็อก) และ สัญญาณ Digital (ดิจิตอล) รวมถึงการทำงานของ อินพุด / เอาต์พุด

สาระการเรียนรู้

<u>ความรู้</u>

เรียนรู้ที่มา ชนิดของชิปไมโครคอนโทรลเลอร์และเข้าใจการพัฒนา Arduino Uno R3

<u>ทักษะ/กระบวนการ</u>

ผู้เรียนสังเกตและจำแนกความแตกต่างในสิ่งที่กำลังเรียน

๒. ผู้เรียนรับฟังกวามกิดเห็น ได้ ตอบกำถามวิพากษ์วิจารณ์จากผู้อื่นที่มีต่อกวามกิดของตน เน้นการ ปรับเปลี่ยนกวามกิดเดิมของตนตามเหตุผลหรือข้อมูลที่ดี

๓. ผู้เรียนมีใจรักและชื่นชอบในสิ่งที่กำลังเรียน

<u>คุณลักษณะที่พึงประสงค์</u>

- ๑. นักเรียนเข้าใจพัฒนาการของบอร์ค Arduino Uno R3
- ๒. นักเรียนเข้าใจชิปไมโครคอนโทรลเลอร์ Arduino Uno
- ๑. รักชาติ ศาสน์ กษัตริย์
- ๔. ซื่อสัตย์
- ๕. มีวินัย
- ใฝ่เรียนรู้
- ๗. อยู่อย่างพอเพียง
- ๘. มุ่งมั่นในการทำงาน

หน่วยการเรียนรู้ที่ 2 พัฒนาการของบอร์ด Arduino Uno R3



รูป บอร์ด Arduino Uno R3



Arduino Uno R3 MCU DIP (บอร์คสำหรับหุ่นยนต์ในหลักสูตรนี้)

1. โครงสร้างของบอร์ด Arduino Uno R3

Arduino Uno เป็นบอร์ดไมโครคอนโทรลเลอร์ที่ใช้ ATmega328 (แผ่นข้อมูล) มีอินพุด / เอาต์พุด 14 อินพุด (6 สามารถใช้เป็นเอาท์พุท PWM), 6 อินพุดแบบอนาลีอก, ดัวเรโซเนเตอร์เซรามิก 16 MHz, การ เชื่อมต่อ USB, แจ็กไฟ, ส่วนหัว ICSP และปุ่มรีเซ็ด มันมีทุกอย่างที่จำเป็นในการสนับสนุน ใมโครคอนโทรลเลอร์; เพียงเชื่อมต่อกับคอมพิวเตอร์ด้วยสายเคเบิล USB หรือใช้อะแดปเตอร์หรือ แบตเตอรี่ AC-to-DC เพื่อเริ่มต้นใช้งาน Uno แตกต่างจากบอร์ดก่อนหน้าทั้งหมด เนื่องจากไม่ได้ใช้ชิป ควบคุม USB แบบอนุกรมของ FTDI แต่มีคุณลักษณะของ Atmega16U2 (Atmega8U2 ถึงเวอร์ชั่น R2) ซึ่ง ใด้รับการตั้งโปรแกรมเป็นตัวแปลงสัญญาณแบบ USB-to-serialRevision 2 ของบอร์ด Uno มีตัวด้านทานดึง สาย HWB 8U2 ไปยังพื้นทำให้ง่ายต่อการใส่ลงในโหมด DFURevision 3 ของบอร์ด มีคุณสมบัติใหม่ ดังต่อไปนี้:1.0 pinout: เพิ่มหมุด SDA และ SCL ที่อยู่ใกล้กับหมุด AREF และอีก 2 หมุดใหม่ที่วางอยู่ใกล้ กับขา RESET IOREF ที่อนุญาดให้โล่ปรับให้เข้ากับแรงดันไฟฟ้าที่จัดหามาจากบอร์ด ในอนาคตโล่จะเข้า กันได้กับทั้งบอร์ดที่ใช้ AVR ซึ่งทำงานร่วมกับ 5V และด้วย Arduino Due ที่ทำงานกับ 3.3V ที่สองคือ ขาทิ่ ไม่ได้เชื่อมต่อซึ่งสงวนไว้สำหรับวัตถุประสงค์ในอนาคตวงจร RESET ที่แข็งแกร่งขึ้น Atmega 16U2 เปลี่ยน 8U2 "Uno" หมายถึงภาษาอิตาลีและมีชื่อว่า Arduino 1.0 Uno และเวอร์ชัน 1.0 จะเป็นเวอร์ชันอ้างอิง ของ Arduino ก้าวไปข้างหน้า Uno เป็นชุดบอร์ด USB Arduino รุ่นล่าสุดและเป็นโมเดลอ้างอิงสำหรับ แพลตฟอร์ม Arduino;

Arduino Uno R3 คำว่า Uno เป็นภาษาอิตาลี ซึ่งแปลว่าหนึ่ง เป็นบอร์ค Arduino รุ่นแรกที่ผลิต ออกมา มีขนาค ประมาณ 68.6x53.4 mm. เป็นบอร์คมาตรฐานที่นิยมใช้งานมากที่สุค เนื่องจากเป็นขนาคที่ เหมาะสำหรับการเริ่มต้นเรียนรู้ Arduino และมี Shields ให้เลือกใช้งานได้มากกว่าบอร์ค Arduino รุ่นอื่นๆ ที่ ออกแบบมาเฉพาะมากกว่า โคยบอร์ค Arduino Uno ได้มีการพัฒนาเรื่อยมาตั้งแต่ R2 R3 และรุ่นย่อยที่ เปลี่ยนชิปไอซี เป็นแบบ SMD เป็นบอร์ค Arduino ที่ได้รับความนิยมมากที่สุค เนื่องจากราคาไม่แพง และ ส่วนใหญ่โปรเจค และ Library ต่างๆ ที่พัฒนาขึ้นมา Support จะอ้างอิงกับบอร์คนี้เป็นหลัก และข้อคีอีก อย่างคือ กรณีที่ MCU เสียผู้ใช้งานสามารถซื้อมาเปลี่ยนเองได้ง่ายArduino Uno R3 มี MCU ที่เป็น Package DIP

นอกจากนั้นแล้ว Arduino UNO R3 เอง ก็แบ่งย่อยออกเป็น 2 แบบ ตามชนิดของชิป Microcontroller ที่ใช้ ได้แก่ Arduino UNO R3 แบบธรรมดา จะใช้ชิปMCU แบบ DIP (Dual Inline Package) ซึ่งเป็นชิปที่ เสียบเข้ากับ Socket อีกทีสามารถถอดเปลี่ยนได้ และ Arduino UNO R3 SMD จะใช้ชิปMCU แบบ SMD (Surface Mount Device) ซึ่งเป็นชิปที่ถูกบัดกรีติดลงบนบอร์ดเลย บอร์ดลักษณะนี้จะไม่สามารถถอดเปลี่ยน ชิปได้ ซึ่งบอร์ดแบบ SMD จะมีต้นทุนที่ถูกกว่าจึงทำให้ราคาถูกกว่าบอร์ดแบบDIP แต่ทั้งสองบอร์ดก็มี ฟังก์ชัน สเปกและการใช้งานที่เหมือนกันทุกประการ



Arduino UNO R3 Specification

Microcontroller	ATmega328P
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
PWM Digital I/O Pins	6
Analog Input Pins	6
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328P)of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
Length	68.6 mm
Width	53.4 mm
Weight	25 g

2. ส่วนประกอบหลัก Arduino Uno R3

- ไมโครคอนโทรลเลอร์

Arduino Uno R3 จะมีชิป Microcontroller ซึ่งผลิต โดยบริษัท Atmel โดยส่วนนี้จะเป็นเหมือน สมองของบอร์ดที่ใช้สำหรับการประมวลผลรวมถึงการควบคุมหมุด I/O



แบบ SMD

แบบ DIP

- Header Socket

Header Socket จะเชื่อมต่อมาจากขาของไมโครคอนโทรลเลอร์ ซึ่งจะเรียงอยู่บริเวณขอบของ บอร์ค Arduino สำหรับต่อสายไฟเพื่อรับสัญญาณ Input หรือสัญณาณ Output ทั้งนี้จะมีป้ายกำกับพร้อมทั้ง หมายเลขหมุคปรากฏอยู่เพื่อความสะควกต่อการใช้งานโคย Header จะแบ่งออกเป็น 3 กลุ่มหลัก คือ Digital Pin, Analog in Pin และ Power Pin



- Digital Pin

Digital Pin คือ หมุดสำหรับรับและส่งสัญญาณที่เป็น Digital โดยมี 2 สถานะ คือ On (0V) และ Off (5V)



- Analog in Pin

Analog in Pin คือ หมุคสำหรับรับค่าสัญญาณที่เป็น Analog



- Power Pin

Power Pin คือ หมุดสำหรับจ่ายแรงคันไฟฟ้าให้กับอุปกรณ์อิเล็กทรอนิกส์หรือชุดโมดูลเซ็นเซอร์ ต่างๆ ซึ่งจะมีทั้ง 5V และ 3.3V นอกจากนี้ยังมีหมุด Vin ที่จะให้ก่ากวามต่างศักย์เท่ากับไฟที่ต่อมาจาก พอร์ตอะแคปเตอร์ (External Power Jack)



- USB Socket

USB Socket คือ พอร์ตที่ใช้ในการเชื่อมต่อสาย USB ในการ Upload Sketch Code ไปยังชิป ใมโครคอนโทรลเลอร์ รวมถึงจ่ายแรงคันไฟให้กับบอร์ค Arduino



- External Power Jack

External Power Jack คือ ช่องสำหรับเสียบ Power Adapter เพื่อต่อไฟจากแหล่งจ่ายภายนอก



- LED On Board

L LED

L LED คือ ไฟ LED ที่ติดอยู่กับบอร์ดซึ่งเชื่อมต่อกับหมุด D13 ซึ่งจะใช้ในการทดสอบการทำงาน ของบอร์ดหรือใช้สำหรับ Sketch Code ที่กำหนดให้โมดูลทำงานแล้วไฟ LED ติด



- Power LED

Power LED คือ ไฟ LED ที่ติดอยู่กับบอร์คซึ่งจะเป็นไฟแสดงสถานะเมื่อไฟเลี้ยงบอร์ค เช่น หาก ไฟ LED ควงนี้ขึ้นเป็นสีเขียว แสดงว่าบอร์ค Arduino พร้อมทำงานแล้ว



- RX, TX LED

RX, TX LED คือ ไฟ LED ที่แสดงสถานะของการรับและส่งข้อมูลในขณะที่บอร์คกำลังทำการส่ง หรือรับข้อมูล



- Reset Button

Reset Button คือ ปุ่มสำหรับ Reset โปรแกรม Sketch Code ที่บอร์ค Arduino กำลังทำงานเพื่อให้ บอร์คหยุคทำงานเดิม จากนั้นจะเริ่มต้นทำงานใหม่ตั้งแต่บรรทัดแรก



นักเขียนโปรกแรมคงสงสัยว่าทำไมถึงมี Uno ยี่ห้ออื่นมากมาย และบอร์คไมโครคอนโทรลเลอร์บาง ยี่ห้อตั้งชื่อรุ่นเป็น Uno R4 ก็มี อีกทั้งบอร์คยี่ห้ออื่นๆที่ไม่ใช่ Arduino ยังมีรูปร่างและลักษณะเหมือนกันกับ บอร์ค Arduino ด้วย สาเหตุเป็นเพราะว่าบอร์คไมโครคอนโทรลเลอร์เป็นแฟลตฟอร์มแบบ Open-Source ฉะนั้นลายวงจรรวมถึง PCB Design ของบอร์ค Arduino จึงคล้ายกัน

หน่วยการเรียนรู้ที่ ๓ การเขียนภาษา C สำหรับไมโครคอนโทรลเลอร์

กลุ่มสาระการเรียนรู้วิทยาศาสตร์ สาระที่ ๔ เทคโนโลยี

มาตรฐาน ว ๔.๒ เข้าใจและใช้แนวกิคเชิงกำนวณในการแก้ปัญหาที่พบในชีวิตจริงอย่างเป็นขั้นตอนและ เป็นระบบ ใช้เทคโนโลยีสารสนเทศและการสื่อสารในการเรียนรู้ การทำงาน และการแก้ปัญหาได้อย่างมี ประสิทธิภาพ รู้เท่าทัน และมีจริยธรรม

ตัวชี้วัด

ງ ໔.២ ໗.໑/໑	ออกแบบอัลกอริทึมที่ใช้แนวคิคเชิงนามธรรมเพื่อแก้ปัญหาหรืออธิบายการทำงานที่
	พบในชีวิตจริง
ງ ໔.ໂອ ນ.໑/ໂອ	ออกแบบและเขียนโปรแกรมอย่างง่ายเพื่อแก้ปัญหาทางคณิตศาสตร์หรือวิทยาศาสตร์
ງ ໔.๒ ໗.๔/๑	ประยุกต์ใช้แนวคิดเชิงคำนวณในการพัฒนาโครงงานที่มีการบูรณาการกับวิชาอื่นอย่าง
	สร้างสรรค์ และเชื่อมโยงกับชีวิตจริง
ງ ໔.២ ໗.໕/໑	รวบรวม วิเคราะห์ข้อมูล และใช้ความรู้ด้านวิทยาการคอมพิวเตอร์ สื่อดิจิทัล
	เทคโนโลยีสารสนเทศในการแก้ปัญหาหรือเพิ่มมูลค่าให้กับบริการหรือผลิตภัณฑ์ที่ใช้
	ในชีวิตจริงอย่างสร้างสรรค์
ງ ໔.២ ມ.๖/໑	ใช้เทคโนโลยีสารสนเทศในการนำเสนอ และแบ่งปั้นข้อมูลอย่างปลอคภัย มีจริยธรรม
	และวิเคราะห์การเปลี่ยนแปลงเทคโนโลยีสารสนเทศที่มีผลต่อการดำเนินชีวิต อาชีพ
	สังคม และวัฒนธรรม

สาระสำคัญ

ผู้เรียนรู้จักโครงสร้างภาษาซีที่ใช้กับบอร์ค Arduino และส่วนของวงจรเชื่อมต่อกับอุปกรณ์ภายนอก เพื่อใช้ในการควบคุมงานต่าง ๆ

สาระการเรียนรู้

<u>ความรู้</u>

๑. ผู้เรียนได้ศึกษาการเขียนโปรแกรมควบคุมการทำงานไมโครคอนโทรลเลอร์ด้วยภาษาซี

๒. ผู้เรียน ได้ศึกษาวงจร ไม โครคอน โทรลเลอร์และมีส่วนของวงจรเชื่อมต่อกับอุปกรณ์ภายนอกเพื่อ ใช้ในการควบคุมงานต่าง ๆ <u>ทักษะ/กระบวนการ</u>

- ๑. ใช้กระบวนการแก้ปัญหาเรื่องของการวิเคราะห์ เก็บข้อมูลประเมินทางเลือก และ สรุปความรู้ที่ได้
- ๒. ใช้กระบวนการสร้างความตระหนักเรื่องของการสังเกต และวิเคราะห์

<u>คุณลักษณะที่พึงประสงค์</u>

- ๑. ผู้เรียนมีใจรักเรื่องการเขียนและพัฒนาโปรแกรมภาษาซี
- ๒. รักชาติ ศาสน์ กษัตริย์
- ด. ซื่อสัตย์
- ๔. มีวินัย
- ๕. ใฝ่เรียนรู้
- ๖. อยู่อย่างพอเพียง
- 🔿. ນຸ່່งນັ້ນໃนการทำงาน

หน่วยการเรียนรู้ที่ 3 การเขียนภาษา C สำหรับไมโครคอนโทรลเลอร์

การเขียนโปรแกรมควบคุมการทำงานไมโครคอนโทรลเลอร์ด้วยภาษาซี เป็นการเขียนโปรแกรมที่ มีความยืดหยุ่นมากกว่าและสามารถพัฒนางานได้เร็วกว่าการเขียนโปรแกรมด้วยภาษาแอสเซมบลี การเขียน โปรแกรมเพื่อควบคุมการทำงานของ Arduino ยึดหลักวิธีการเขียนตามไวยากรณ์ภาษาซี ดังนั้นเมื่อสามารถ เขียนโปรแกรมควบคุมการทำงานไมโครคอนโทรลเลอร์ Arduino นี้ได้ก็สามารถนำความรู้ไปเขียน โปรแกรมภาษาซีไมโครคอนโทรลเลอร์อื่น ๆ ได้

การใช้งานไมโครคอนโทรลเลอร์นอกจากจะต้องมีวงจรไมโครคอนโทรลเลอร์และมีส่วนของวงจร เชื่อมต่อกับอุปกรณ์ภายนอกเพื่อใช้ในการควบคุมงานต่าง ๆ ตามต้องการที่เรียกว่าฮาร์คแวร์ (Hardware) แล้วจำเป็นต้องมีชุดคำสั่งหรือโปรแกรมไว้สำหรับสั่งงานให้ไมโครคอนโทรลเลอร์ทำงานตามที่ต้องการซึ่ง เรียกว่าซอฟท์แวร์ (Software) ในบทนี้เป็นการเรียนรู้หลักการเขียนโปรแกรมควบคุมการทำงานของ ไมโครคอนโทรลเลอร์

1. ขั้นตอนในการพัฒนาโปรแกรม

โปรแกรมคอมพิวเตอร์ที่นำมาใช้งานได้นั้น ไม่สามารถเริ่มต้นจากการเขียนกำสั่งด้วย ภาษาคอมพิวเตอร์ได้ทันทีจะต้องมีการวิเคราะห์ วางแผน และปฏิบัติตามกระบวนการทำงาน ซึ่งแบ่ง ออกเป็น 5 ขั้นตอน คือ

 วิเคราะห์ปัญหาโดยจะเริ่มจากการวิเคราะห์ผลลัพธ์ที่ต้องการ (Output) แล้วย้อนกลับไปยังข้อมูล ที่นำเข้าสู่ระบบ (Input) ตลอดจนข้อมูลอื่น ๆ ที่เกี่ยวข้องในการที่จะนำไปใช้ในการประมวลผล

 2. ออกแบบวิธีการแก้ปัญหา เมื่อทราบผลลัพธ์ที่ต้องการและข้อมูลที่นำเข้าสู่ระบบแล้ว ต้อง กำหนดการวางแผนในการแก้ปัญหา โดยใช้วิธีเขียนลำดับขั้นตอนการแก้ปัญหาที่เรียกว่า อัลกอริทึม (Algorithm) และใช้เครื่องมือสำหรับช่วยในการเขียนอัลกอริทึมเช่น การเขียนรหัสจำลอง (Pseudo code) การเขียนผังงาน (Flowchart) เป็นต้น

 เขียนโปรแกรม เลือกภาษาคอมพิวเตอร์ที่เหมาะสมโดยพิจารณาจากความสามารถของผู้เขียน โปรแกรมและประสิทธิภาพของภาษาคอมพิวเตอร์นั้น ๆ ให้เหมาะสมกับระบบงานที่ต้องการแล้วเขียน ชุดคำสั่งเป็นภาษาคอมพิวเตอร์ตามอัลกอริทึมที่ได้ออกแบบไว้

4. ทคสอบและแก้ไขโปแกรมภายหลังจากเขียนโปรแกรมเสร็จสิ้น จะต้องทำการทคสอบโปรแกรม เพื่อหาข้อผิดพลาค (Error) ซึ่งข้อผิดพลาคที่พบในขั้นตอนการทคสอบโปรแกรมนั้นจะต้องนำมาปรับปรุง แก้ไขโปรแกรมเพื่อให้สามารถทำงานได้ตามต้องการ 5. จัดทำเอกสารประกอบ เมื่อโปรแกรมผ่านการทดสอบแล้วก็จะต้องจัดทำเอกสารประกอบซึ่งมี รายละเอียดของวิธีการใช้งานโปรแกรม วิธีการติดตั้งโปรแกรม ตลอดจนขั้นตอนในการพัฒนาโปรแกรม รวมถึงอัลกอริทึมและโปรแกรมต้นฉบับ (Source code) เพื่อประโยชน์ในกรณีที่ต้องการแก้ไขหรือปรับปรุง โปรแกรมภายหลัง

ขั้นตอนวิธี หรือ Algorithm (ภาษาไทย : อัลกอริทึม) หมายถึง กระบวนการแก้ปัญหาที่สามารถเข้าใจ ใด้ มีลำคับหรือวิธีการในการแก้ไขปัญหาใดปัญหาหนึ่งอย่างเป็นขั้นเป็นตอนและชัคเจนเมื่อนำเข้าอะไรแล้ว จะต้องได้ผลลัพธ์เช่นไร ซึ่งแตกต่างจากการแก้ปัญหาแบบสามัญสำนึกหรือฮิวริสติก(Heuristic) โดยทั่วไป ขั้นตอนวิธีจะประกอบด้วยวิธีการเป็นขั้น ๆ และมีส่วนที่ต้องทำแบบวนซ้ำหรือเวียนเกิดโดยใช้ตรรกะ และ/ หรือในการเปรียบเทียบในขั้นตอนต่าง ๆ จนกระทั่งเสร็จสิ้นการทำงาน

ในการทำงานอย่างเดียวกันนักเขียนโปรแกรมอาจจะเลือกขั้นตอนวิธีที่ต่างกันเพื่อแก้ปัญหาได้โดยที่ ผลลัพธ์ที่ได้ในขั้นสุดท้ายจะออกมาเหมือนกันหรือไม่ก็ได้ และจะมีความแตกต่างที่จำนวนและชุดคำสั่งที่ใช้ ต่างกันซึ่งส่งผลให้เวลาและขนาดหน่วยความจำที่ต้องการต่างกัน หรือเรียกได้อีกอย่างว่ามีความซับซ้อน ต่างกัน

การนำขั้นตอนวิธีไปใช้ไม่จำกัดเฉพาะการเขียนโปรแกรมคอมพิวเตอร์ แต่สามารถนำไปใช้กับ ปัญหาอื่น ๆ ได้เช่น การออกแบบวงจรไฟฟ้า, การทำงานเครื่องจักรกล, หรือแม้กระทั่งปัญหาในธรรมชาติ เช่น วิธีของสมองมนุษย์ในการกิดเลข หรือวิธีการขนอาหารของแมลง

ผังงาน (Flowchart) คือ รูปภาพ (Image) หรือสัญลักษณ์(Symbol) ที่ใช้เขียนแทนขั้นตอนคำอธิบาย ข้อความ หรือคำพูดที่ใช้ในอัลกอริทึม (Algorithm) เพราะการนำเสนอขั้นตอนของงานให้เข้าใจตรงกัน ระหว่างผู้เกี่ยวข้องด้วยคำพูดหรือข้อความทำได้ยากกว่า

ผังงานในการเขียนโปรแกรมเป็นรูปทรงเลขาคณิต ที่บรรจุรายละเอียคกระบวนการประมวลผลโดย มีรูปทรงในการใช้งานหลัก ๆ (เฉพาะงานไมโครคอนโทรลเลอร์) ดังนี้

รูปทรง	ชนิดการประมวลผล	ตัวอย่างเส้นทาง	
	จุดเริ่มต้นหรือจุดสิ้นสุด ของโปรแกรม		
	กระบวนการประมวลผล		→
	การตัดสินใจ		+

ผังงานหลักที่ใช้งานไมโครคอนโทรลเลอร์

รูปทรง	ชนิดการประมวลผล	ตัวอย่าง	เส้นทาง
	ชุดกระบวนการที่เตรียม ไว้แล้ว (โปรแกรมย่อย)		→
	จุดเชื่อมในหน้าเดียวกัน 	-	•
	จุดเชื่อมในหน้าอื่น		-

ตัวอย่าง การเขียนลำดับขั้นขั้นตอนวิธี และการแปลงเป็นผังงาน

ยกตัวอย่างในชีวิตประจำวันเพื่อให้ง่ายต่อการจินตนาการ โจทย์การทำข้าวผัดกุ้ง

ขั้นที่ 1 เขียนลำคับขั้น (Algorithm)

1. กระเทียมสับ, ต้นหอมหั่น, หอมหัวใหญ่หั่นแว่น, น้ำมันพืช, กระทะ, เตาไฟ, ถ้วย และจาน	8. รอให้ด้านที่ 2 สุก
2. นำกระทะตั้งไฟเพื่ออุ่นกระทะ	9. ใส่หอมหัวใหญ่หั่นแว่น
3. ใส่น้ำมันพืช	10. ผัครอให้สุก
4. ใส่กระเทียมสับ	9. ใส่ข้าวสวยสุก
5. ใส่กุ้ง	10. ผัดให้เข้ากัน
6. รอให้หนึ่งด้านสุก	11. ปรุงรส
7. กลับด้ำน	12. ใส่ต้นหอมหั่น
	13. ตักออกใส่จาน (เสร็จ)
ขั้นที่ 2 แปลงลำคับขั้นเป็นผังงาน (Flowchart)





้**ตัวอย่าง** การเขียนโปรแกรมไมโครคอนโทรลเลอร์



ตัวอย่างจงเขียนโปรแกรมไฟกระพริบ LED ที่ต่ออยู่ที่ขา D9

ขั้นตอนที่ 3 แปลงผังงานเป็นโปรแกรม

```
sketch_sep03b§
 1 #define LED
 2 void setup()
 3
   {
 4
     pinMode (LED, OUTPUT);
 5
 6
  }
 7
8 void loop() {
9
    digitalWrite(LED, HIGH);
     delay(1000);
10
11
     digitalWrite(LED, LOW);
12
     delay(1000);
13
14 }
15
```

2. ภาษาซีสำหรับไมโครคอนโทรลเลอร์

ไมโครคอนโทรลเลอร์ไม่ว่าจะเป็นตระกูลใดก็ตามจะทำงานได้ก็ต่อเมื่อมีชุดคำสั่งที่สั่งให้ทำงาน ตามที่ต้องการที่เรียกว่าโปรแกรม โดยคำสั่งหรือโปรแกรมที่ไมโครคอนโทรลเลอร์เข้าใจและสามารถ ทำงานได้อยู่ในรูปของรหัสลอจิก o และ 1 หากนำลอจิกมาจับกลุ่มก็เป็นเลขฐาน 16 ที่เรียกว่าภาษาเครื่องซึ่ง ภาษาเครื่องเป็นภาษาที่มนุษย์ไม่สามารถเข้าใจได้เนื่องจากเป็นเลขฐาน 16 ทั้งหมด ดังนั้นในการเขียน โปรแกรมจึงจำเป็นต้องใช้ภาษาที่มนุษย์สามารถเข้าใจได้ โดยภาษาที่มนุษย์เข้าใจได้และใกล้เคียงกับ ภาษาเครื่องมากที่สุดคือ ภาษาแอสเซมบลี แต่เนื่องจากการพัฒนางานโดยใช้ภาษาแอสเซมบลีเป็นไปได้ยาก และซับซ้อน เพื่อให้ง่ายและรวดเร็วต่อการพัฒนาโปรแกรมใช้งานไมโครคอนโทรลเลอร์ภาษาที่เหมาะสม คือ ภาษาซี

<u>โครงสร้างของภาษาซี</u>

ภาษาซีเป็นภาษาที่ได้รับความนิยมสูงเป็นภาษาโครงสร้างง่ายต่อการทำความเข้าใจง่ายต่อการนำ ไปพัฒนาต่อ สามารถเขียนโปรแกรมแยกเป็นส่วน ๆ โดยแต่ละส่วนสามารถเรียกใช้งานได้จากส่วนอื่นของ โปรแกรมทำให้สามารถแบ่งงานให้หลายคนไปพัฒนาได้ การเขียนโปรแกรมเป็นส่วน ๆ เรียกว่าฟังก์ชัน โครงสร้างของภาษาซีมีส่วนประกอบ 2 ส่วนคือ ส่วนหัวโปรแกรมและส่วนตัวโปรแกรมส่วนตัวโปรแกรม จะมีฟังก์ชันหลักชื่อว่า main() เพื่อเป็นส่วนหลักในการทำงาน และอาจมีฟังก์ชันอื่นที่ผู้ใช้เขียนขึ้นเพื่อใช้ งานเรียกว่าฟังก์ชันรอง



โครงสร้างภาษาซี

หน่วยการเรียนรู้ที่ ๔ การติดตั้งและลองใช้ Arduino IDE

กลุ่มสาระการเรียนรู้วิทยาศาสตร์ สาระที่ ๔ เทคโนโลยี

มาตรฐาน ว ๔.๒ เข้าใจและใช้แนวคิคเชิงคำนวณในการแก้ปัญหาที่พบในชีวิตจริงอย่างเป็นขั้นตอนและ เป็นระบบ ใช้เทคโนโลยีสารสนเทศและการสื่อสารในการเรียนรู้ การทำงาน และการแก้ปัญหาได้อย่างมี ประสิทธิภาพ รู้เท่าทัน และมีจริยธรรม

ตัวชี้วัด

ว ๔.๒ ม.๑/๑	ออกแบบอัลกอริทึมที่ใช้แนวคิคเชิงนามธรรมเพื่อแก้ปัญหาหรืออธิบายการทำงานที่
	พบในชีวิตจริง
ງ ໔.ໂຫ ນ.໑/ໂຫ	ออกแบบและเขียน โปรแกรมอย่างง่ายเพื่อแก้ปัญหาทางคณิตศาสตร์หรือวิทยาศาสตร์
റ ๔.๒ ม.๑/๔	ใช้เทคโนโลยีสารสนเทศอย่างปลอคภัย ใช้สื่อและแหล่งข้อมูลตามข้อกำหนดและ
	ข้อตกลง
ງ ໔.ໂອ ນ.ໂອ/໑	ออกแบบอัลกอริทึมที่ใช้แนวคิดเชิงคำนวณในการแก้ปัญหา หรือการทำงานที่พบใน
	ชีวิตจริง

สาระสำคัญ

ผู้เรียนได้รู้จัก Arduino IDE ซึ่งเป็นโปรแกรมสำหรับเขียนโค้ดเพื่อควบคุมการทำงานของ ไมโครคอนโทรลเลอร์

สาระการเรียนรู้

<u>ความรู้</u>

ผู้เรียนได้รู้จักโปรแกรมการเขียนคำสั่ง Arduino IDE และทดลองเขียนเขียนคำสั่งอย่างง่าย

<u>ทักษะ/กระบวนการ</u>

๑. ผู้เรียนใช้ทักษะการกิดวิเกราะห์วิจารณ์ สร้างทางเลือกที่หลากหลาย ประเมินและเลือกทางเลือกใน การเขียนกำสั่งอย่างมีระบบ

๒. ผู้เรียนใช้กระบวนการสังเกต จำแนกความแตกต่าง และหาลักษณะร่วม เพื่อวิเคราะห์สิ่งที่กำลัง ศึกษาอยู่ขณะนั้น

๓. ผู้เรียนสามารถแยกระหว่างโปรแกรมที่เขียนคำสั่ง Arduino IDE กับ โปรแกรมอื่นๆได้

<u>คุณลักษณะที่พึงประสงค์</u>

- ๑. ผู้เรียนเข้าใจและมีใจรักในการศึกษาโปรแกรม Arduino IDE เป็นอย่างดี
- ๒. รักชาติ ศาสน์ กษัตริย์
- ๓. ซื่อสัตย์
- ๔. มีวินัย
- ๕. ใฝ่เรียนรู้
- ๖. อยู่อย่างพอเพียง
- ๗. มุ่งมั่นในการทำงาน

หน่วยการเรียนรู้ที่ 4 การติดตั้งและลองใช้ Arduino IDE

1. การติดตั้งโปรแกรม Arduino IDE

Arduino IDE เป็นโปรแกรมสำหรับเขียนโค้ดเพื่อควบคุมการทำงานของไมโครคอนโทรลเลอร์ ซึ่ง สามารถดาวน์โหลดใช้งานได้โดยไม่มีค่าใช้จ่ายและมีเวอร์ชันที่รองรับระบบปฏิบัติการหลายแบบ ซึ่งใน หลักสูตรนี้จะขอแนะนำเฉพาะโปรแกรมที่รองรับระบบปฏิบัติการวินโดวส์เท่านั้น โดยขั้นตอนมีดังนี้

คาวน์โหลดโปรแกรมโดยเข้าไปที่เว็ปไซด์ https://www.arduino.cc/en/Main/Software คลิกที่กำ
 ว่า SOFTWARE ดังรูป



รูปหน้าต่างเว็ปไซต์สำหรับดาวน์โหลดโปรแกรม Arduino IDE

2. เลือกดาวน์โหลดไฟล์ที่เป็นไฟล์ zip (เพื่อให้ง่ายต่อการอัพเดตหรือถอนโปรแกรมออกใน อนาคต) โดยกลิกตรงกำว่า Windows ZIP file for non admin install ดังรูป



ARDUINO 1.8.9

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other opensource software. This software can be used with any Arduino board. Refer to the <u>Getting Started</u> page for Installation instructions. Windows Installer, for Windows XP and up Windows ZIP file for non admin install

Windows app Requires Win 8.1 or 10

Mac OS X 10.8 Mountain Lion or newer

Linux 32 bits Linux 64 bits Linux ARM 32 bits Linux ARM 64 bits

Release Notes Source Code Checksums (sha512)

รูปบริเวณที่สำหรับคลิกเพื่อดาวน์โหลดไฟล์ zip

3. กลิกตรงกำว่า JUST DOWNLOAD เพื่อดาวน์โหลดไฟล์



รูป หน้าต่างเว็ปไซต์สำหรับกลิกเพื่อคาวน์โหลด

4. กัคลอกไฟล์ zip มายังไคร์ฟ D จากนั้นแยกไฟล์ แล้วคับเบิ้ลคลิกไฟล์ arduino.exe

ดังรูป



รูป คัดลอกทั้งโฟลเคอร์มาไว้ที่ไคร์ฟ D

with WinZin Print E-mail	New folder	= - FI 🙆
Name	Date modified	Туре
and time 180	21/0/2562 21/10	Die Geblee
arduino-1.8.9	31/0/2302 21:19	Mie Tie File
arduno-18.9-windows	21/0/2302 21:19	Adaba Assabat D
	31/0/2302 19:40	Adobe Acrobat D
42 DH 1 DBD 3	51/6/2302 21:10	rauate microsoft
🧖 ส่วจัมนขอไฟส์กวอ	arduino-18 🖂 🖳 🔀	
San Internet in the office		
เอกสาร D:\wan\arduin	o\arduino-1.8.9-windows.zip	
กำลังแยก 8d2xda	12%	
	10.4	
เวลาที่ผ่านไป	00:00:02	
	00.00.01	
ถูกต่าเนินการ	5%	
-		
สวาวนา ปันน ข้าวมาต้		
Unish	ระเย	
		_

5. เปิดไฟล์ติดตั้งขึ้นมาจะปรากฎหน้าต่าง สำหรับติดตั้งให้กลิก I Agree

6. คลิก _{Next} เพื่อไปสู่ขั้นตอนถัคไป

7. เลือกที่ติดตั้งโปรแกรมจากนั้น กด Install



💿 Arduino Setup: Installation Folder	-		×			
Setup will install Arduino in the following folder. folder, dick Browse and select another folder. C installation.	To install i Click Install	n a differer to start th	nt e			
Destination Folder						
C:\Program Files (x86)\Arduino Browse						
Space required: 420.6MB						
Space available: 52.7GB						
Cancel Nullsoft Install System v3.0	< Back	Inst	all			

< Back

Next >

Nullsoft Install System v3.0

Cancel



13. จะปรากฎไอคอน Arduino บนหน้าจอ คอมพิวเตอร์



14. ทคสอบเปิคโปรแกรมจะได้หน้าต่างดังรูป



แถบเครื่องมือในโปรแกรม Arduino IDE มีแถบรายการคำสั่งต่าง ๆ ดังนี้

- 1. แถบเครื่องมือไฟล์ (File) มีเครื่องมือช่วยในคำสั่งต่าง ๆ ซึ่งแต่ละปุ่มมีหน้าที่การใช้งานคังนี้
 - 1. New : เป็นแถบสร้างไฟล์ใหม่เพื่อเริ่มการเขียนโปรแกรมใหม่
 - 2. Open : เป็นแถบเปิดไฟล์งานที่เก็บในโฟลเดอร์
 - 3. Sketchbook : เป็นแถบเปิดไฟล์งานล่าสุด
 - 4. Example : เป็นแถบในการเลือกเปิดไฟล์ตัวอย่างที่อยู่ในโปรแกรม
 - 5. Close : เป็นแถบปิดไฟล์งานล่าสุด
 - 6. Save : เป็นแถบบันทึกไฟล์งานที่สร้างขึ้นปัจจุบัน
 - 7. Save As : เป็นแถบบันทึกไฟล์งานโดยเปลี่ยนชื่อไฟล์
 - 8. Page Setup : เป็นแถบตั้งก่าหน้ากระคาษของไฟล์งานปัจจุบัน
 - 9. Print : เป็นแถบสั่งพิมพ์งานออกทางเครื่องพิมพ์
- 10. Preference : เป็นแถบกำหนดค่าการทำงานของโปรแกรม
- 11. Quit : เป็นแถบใช้ออกจากโปรแกรม

00	sketch_sep02c	Arduino 1.8.9)		_		×
File	Edit Sketch	Tools Help					
	New	Ctrl+N					Ø
	Open	Ctrl+O					_
	Open Recent		>				
	Sketchbook		>				^
	Examples		>	ode here, to run on	ice:		
	Close	Ctrl+W					
	Save	Ctrl+S					
	Save As	Ctrl+Shift+S					
	Page Setup	Ctrl+Shift+P		de here, to run rep	eated	ly:	
	Print	Ctrl+P					
	Preferences	Ctrl+Comma					
	Quit	Ctrl+Q					
			-				
					Arc	luino Leo	nardo

รูป แถบเครื่องมือไฟล์ (File)

- 2. แถบเครื่องมือแก้ไข (Edit) มีเครื่องมือช่วยในคำสั่งต่าง ๆ ซึ่งแต่ละปุ่มมีหน้าที่การใช้งานดังนี้
 - 1. Undo : เป็นแถบย้อนกลับไปคำสั่งล่าสุด
 - 2. Redo : เป็นแถบข้อนกลับคืนเมื่อกดแถบ Undo มากเกินไป
 - 3. Cut : เป็นแถบตัดข้อความที่ต้องการคัดลอก
 - 4. Copy : เป็นแถบสำหรับคัคลอกข้อความ
 - 5. Paste : เป็นแถบสำหรับวางข้อความที่ตัดหรือคัดลอกมา
 - 6. Select All : เป็นแถบเลือกข้อความทั้งหมด
 - 7. Comment/UnComment : เป็นแถบสำหรับเติมเครื่องหมาย // เพื่อสร้างหมายเหตุหรือคำอธิบายลง

ในโปรแกรม

- 8. Increase Indent : เป็นแถบสำหรับเลื่อนเคอร์เซอร์
- 9. Decrease Indent : เป็นแถบสำหรับยกเลิกการเลื่อนเคอร์เซอร์
- 10. Find.. : เป็นแถบค้นหา
- 11. Find Next : เป็นแถบค้นหาต่อไป
- 12. Find Previous : เป็นแถบย้อนกลับการค้นหา



รูป แถบเครื่องมือแก้ไข (Edit)

- 3. แถบเครื่องมือสเกตช์ (Sketch) มีเครื่องมือช่วยในคำสั่งต่าง ๆ ซึ่งแต่ละปุ่มมีหน้าที่การใช้งานคังนี้
 - 1. Verify/Compile : เป็นแถบสำหรับการคอมไพล์โปรแกรม
 - 2. Upload : เป็นแถบสำหรับการอัปโหลดโปรแกรมที่เขียนขึ้นลงบอร์ด Arduino
 - 3. Show Sketch folder : เป็นแถบใช้ในการสั่งเปิดโฟลเดอร์ที่เก็บโปรแกรม
- 4. Import Library : เป็นแถบคำสั่งเรียกใช้ไลบรารีเพิ่มเติม
- 5. Add File : เป็นแถบเพิ่มไฟล์ให้กับสเกตซ์บุ๊กปัจจุบัน



รูป แถบเครื่องมือสเกตช์ (Sketch)

4. แถบเครื่องมือ (Tools) มีเครื่องมือช่วยในคำสั่งต่างๆ ซึ่งแต่ละปุ่มมีหน้าที่การใช้งานคังนี้

1. Auto Format : เป็นแถบจัครูปแบบ โปรแกรมแบบอัต โนมัติ

2. Archive Sketch : เป็นแถบสั่งบีบอัคไฟล์โปรแกรมทั้งโฟลเคอร์ หลักและโฟลเคอร์ย่อยของไฟล์ งานปัจจุบันไฟล์ที่สร้างใหม่ให้มีชื่อเดียวกับไฟล์งานปัจจุบัน

3. Board : เป็นแถบสำหรับเลือกฮาร์ดแวร์ของบอร์ด Arduino

4. Serial Port : เป็นแถบสำหรับเลือกหมายเลขพอร์ตของคอมพิวเตอร์ที่ต่อกับบอร์ค Arduino



รูป แถบเครื่องมือ (Tools)

5. แถบเครื่องมือตัวช่วย (Help) ถ้าต้องการความช่วยเหลือหรือข้อมูลที่เกี่ยวข้องกับ โปรแกรมให้ เลือกเมนูนี้ และเลือกแถบเมนูย่อยของ โปรแกรม โปรแกรมจะเปิดไฟล์เว็บเพจที่เกี่ยวข้องกับหัวข้อที่เลือก



รูป แถบเครื่องมือตัวช่วย (Help)

6. ปุ่ม Serial Monitor เป็นปุ่มสำหรับการติดต่อสื่อสารบอร์ด Arduino กับคอมพิวเตอร์ เมื่อคลิกปุ่มนี้ กอมพิวเตอร์ทำการเปิดหน้าต่างใหม่ แล้วนำค่าในการรับส่งข้อมูลมาแสดง ที่หน้าต่างที่เปิดขึ้นนี้



รูป หน้าต่างแสดงการติดต่อสื่อสารบอร์ด Arduino กับคอมพิวเตอร์

4.3 ส่วนประกอบของโปรแกรม Arduino (IDE)

โครงสร้างโปรแกรมภาษา C บน Arduino จะมีลักษณะแบบเดียวกับ C ทั่วๆไป โดยมีโครงสร้าง ดังต่อไปนี้

- 1. Preprocessor directives
- 2. ฟังก์ชัน setup()
- 3. ฟังก์ชัน loop()



1. Preprocessor directives

ส่วนที่ 1 นอกจากจะเป็นส่วนของ Preprocessor directives แล้วยังมี ส่วนของการกำหนดค่า (Global declarations) คือ การกำหนดชนิดตัวแปรแบบนอกฟังก์ชัน

ส่วน Preprocessor directives จะเป็นส่วนที่คอมไพลเลอร์จะมีการประมวลผลและทำตามกำสั่ง ก่อนที่จะมีการคอมไพล์โปรแกรม ซึ่งจะเริ่มต้นด้วยเครื่องหมายใดเร็กทีฟ (directive) หรือเครื่องหมาย สี่เหลี่ยม # แล้วจึงตามด้วยชื่อกำสั่งที่ต้องการเรียกใช้หรือกำหนด โดยปกติแล้วส่วนนี้จะอยู่ในส่วนบนสุด หรือส่วนหัวของโปรแกรมและต้องอยู่นอกฟังก์ชันหลักใดๆ ก็ตาม

<u>#include</u> เป็นคำสั่งที่ใช้อ้างอิงไฟล์ภายนอก เพื่อเรียกใช้ฟังก์ชันหรือตัวแปรที่มีการสร้างหรือ กำหนดไว้ในไฟล์นั้นรูปแบบการใช้งานคือ #include <ชื่อไฟล์.h> ตัวอย่างเช่น

#include <Wire.h>
#include <Time.h>

จากตัวอย่าง จะเห็นว่าได้มีการอ้างอิงไฟล์ Wire.h และไฟล์ Time.h ซึ่งเป็นไลบารี่พื้นฐานที่มีอยู่ใน Arduino ทำให้เราสามารถใช้ฟังก์ชันเกี่ยวกับเวลาที่ไลบารี่ Time มีการสร้างไว้ให้ใช้งานได้

การอ้างอิงไฟล์จากภายในหรือการอ้างอิงไฟล์ไลบารี่ที่มีอยู่แล้วใน Arduino หรือไลบารี่ที่นักเขียน โปรแกรมเพิ่มเข้าไปเองจะใช้เครื่องหมาย <> ในการคร่อมชื่อไฟล์ไว้ เพื่อให้โปรแกรมคอมไพลเลอร์เข้าใจ ว่าควรไปหาไฟล์เหล่านี้จากในโฟลเดอร์ไลบารี่ แต่หากต้องการอ้างอิงไฟล์ที่อยู่ในโฟลเดอร์โปรเจค จะต้อง ใช้เครื่องหมาย "" คร่อมแทน ซึ่งคอมไพล์เลอร์จะวิ่งไปหาไฟล์นี้โดยอ้างอิงจากไฟล์โปรแกรมที่คอมไพล์ เลอร์อยู่

เช่น

#include "mywork.h"

จากตัวอย่างค้านบนคอมไพล์เลอร์จะวิ่งไปหาไฟล์ mywork.h ภายในโฟลเคอร์โปรเจกทันที หากไม่ พบก็จะแจ้งเป็นข้อผิดพลาดออกมา

#define เป็นคำสั่งที่ใช้ในการแทนข้อความที่กำหนดไว้ด้วยข้อความที่กำหนดไว้ ซึ่งการใช้คำสั่งนี้ ข้อคื คือจะไม่มีการอ้างอิงกับตัวโปรแกรมเลย

รูปแบบ

#define NAME VALUE

ตัวอย่างเช่น

#define LEDPIN 12

จากตัวอย่าง ไม่ว่าคำว่า LEDPIN จะอยู่ส่วนใดของโค้คโปรแกรมก็ตามคอมไพล์เลอร์จะแทนคำว่า LEDPIN ด้วยเลข 12 แทน ซึ่งข้อดีคือเราไม่ต้องสร้างเป็นตัวแปรขึ้นมาเพื่อเปลืองพื้นที่แรม และยังช่วยให้ โปรแกรมทำงานเร็วขึ้นอีกด้วยเพราะซีพียูไม่ต้องไปขอข้อมูลมาจากแรมหลายต่อ

<u>ส่วนของการกำหนดค่า (Global declarations)</u>

ส่วนนี้จะเป็นส่วนที่ใช้ในการกำหนดชนิดตัวแปรแบบนอกฟังก์ชันหรือประกาศฟังก์ชัน เพื่อให้ ฟังก์ชันที่ประกาศสามารถกำหนดหรือเรียกใช้ได้จากทุกส่วนของโปรแกรม เช่น

```
int pin_led = A1;
int pin_sensor = 8;
```

2. ฟังก์ชัน void setup()

พึงก์ชัน void setup() เป็นพึงก์ชันแรกที่ถูกเรียกใช้นิยมใช้กำหนดค่าหรือเริ่มต้นใช้งานไลบารี่ต่างๆ เช่น ในพึงก์ชัน setup() จะมีคำสั่ง pinMode() เพื่อกำหนดให้ขาใดๆ ก็ตามเป็นดิจิตอลอินพุตหรือเอาต์พุต โดย OUTPUT เป็นค่าคงที่เมื่อใช้ขาดิจิตอลบน Arduino ต้องระบุโหมดเป็น OUTPUT หรือ INPUT เสมอ ตามหลักของภาษา C (ที่ Arduino ใช้) คำสั่งหนึ่งๆ จะลงท้ายด้วยเครื่องหมายอัฒภาค (;) เช่น

pinMode(trigPin, OUTPUT);

pinMode(echoPin, INPUT);

3. ฟังก์ชัน void loop ()

เป็นพึงก์ชันที่ทำงานหลังจากพึงก์ชัน setup() ได้ทำงานเสร็จสิ้นไปแล้วและมีการวนรอบแบบไม่รู้ จบ เมื่อพึงก์ชัน loop() งานครบตามกำสั่งแล้วพึงก์ชัน loop() ก็จะถูกเรียกขึ้นมาใช้อีก

ตัวอย่าง

```
1 #include <Servo.h>
 2 Servo servoR:
 3 Servo servoL;
 4 Servo servol80;
5 int S1 = A3;
 6 int sensor;
7 int pos = 0;
8
9 void setup()
10 {
11 Serial.begin(9600);
12 servoR.attach(4);
13
   servoL.attach(7);
14
   servol80.attach(5);
15
   pinMode(S1, INPUT);
    servol80.write(90);
16
17 }
18
19 void loop()
20 {
21
      FD();
      sensor = analogRead(S1);
22
23
     Serial.print("distant = ");
24
     Serial.println(sensor);
25
     if (sensor > 500)
26
     {
27
      look R();
28
      look L();
29
      }
30 }
```

จากตัวอย่าง คือการเขียนโปรแกรมรถเคลื่อนที่ และจะเห็นว่ามีการประกาศตัวแปรแบบนอกฟังก์ชัน ทำ ให้สามารถกำหนดหรือเรียกใช้จากในฟังก์ชันใดๆก็ตาม ได้ โดยนำเข้าไลบารี่ Servo.h ประกาศ Servor และตัว แปรอีก 3 ตัว เพื่อใช้ควบคุมรถในฟังก์ชัน setup() ได้มี การกำหนด servo 3 ตัว และขา A3 เป็น INPUT ใน ฟังก์ชัน loop() มีการใช้ชุดคำสั่งย่อย FD(); (นักเขียน โปรแกรมสามารถเขียนชุดคำสั่งย่อยขึ้นมาได้โดยตั้งชื่อ ชุดคำสั่งขึ้นมา ตัวอย่างเช่น void FD() {"เขียนกำสั่ง" }) จากนั้นประกาศตัวแปร sensor เพื่อรับค่าจากขา analog S1 จากนั้นให้แสดงค่าระยะออกทาง Serial Monitor และตามด้วยค่าเซ็นเซอร์ที่วัดได้ นำค่าเซ็นเซอร์ ที่วัดได้เข้าสู่เงื่อนไข

โปรแกรมจะวนกลับมาทำงานฟึงก์ชัน loop() ซ้ำไปเรื่อยๆ โดยอัตโนมัติจนกว่านักเขียนโปรแกรม จะถอดสายไฟออกจากบอร์ด Arduino

<u>การติดตั้งไดร์ฟเวอร์ USB</u>

บอร์ด Arduino ปัจจุบันมีหลายรุ่นซึ่งแต่ละรุ่นอาจใช้ชิพสำหรับติดต่อสื่อสารผ่านพอร์ต USB แตกต่างกันไป ซึ่งบางรุ่นมีใดร์ฟเวอร์มาให้พร้อมในตัวโปรแกรม Arduino IDE แล้ว เช่น บอร์ดที่ใช้ชิพ CP210x, FTDI, หรือบอร์ดที่ใช้ไมโครคอนโทรลเลอร์เป็นตัวสื่อสารผ่าน USB ดังนั้นการติดตั้งผู้ใช้งาน สามารถเรียกใช้ได้เลยโดยไม่ต้องไปหาดาวน์โหลดไดร์ฟเวอร์อีก โดยไดร์ฟเวอร์จะอยู่ใน โฟลเดอร์C:\arduino-1.8.9\drivers ดังรูป

> WIN10 (C:) > Program Files (x86) > arduino-1.8.9 > drivers						
Name	Date modified	Туре				
amd64	06/09/2562 15:44	File folder				
CP210x_6.7	06/09/2562 15:44	File folder				
CP210x_6.7.4	06/09/2562 15:44	File folder				
FTDI USB Drivers	06/09/2562 15:44	File folder				
📊 ia64	06/09/2562 15:44	File folder				
license	06/09/2562 15:44	File folder				
x86	06/09/2562 15:44	File folder				

ยกเว้นบอร์คที่ใช้ชิพ CH340,CH341 (บอร์ค Arduino Nano 3.0 เถือกใช้ชิพเบอร์นี้ในบางผู้ผลิต) ซึ่งผู้ใช้งานจะต้องไปหาดาวน์โหลดไดร์ฟเวอร์เพื่อมาติดตั้งเอง การดาวน์โหลดสามารถดาวน์โหลด ไดร์ฟเวอร์ได้ที่ http://www.winchiphead.com/download/CH341/CH341SER.ZIP

การตรวจสอบการทำงานของไคร์ฟเวอร์ด้วยการเชื่อมต่อบอร์ค Arduino เข้ากับคอมพิวเตอร์ด้วย สาย USB ทำการเปิด Device Manager สังเกตพอร์ตสื่อสารที่ปรากฏขึ้นดังรูป



ตัวอย่างการลองใช้โปรแกรม Arduino IDE

ตัวอย่าง เ



ตัวอย่างที่ 2

💿 I	esson_1-3 Arduino 1.8.9	- 🗆 🗙				
File Edit Sketch Tools Help						
	📀 🛅 🛃 🔛 Upload	2				
les	son_1-3	•				
1	<pre>int Num1 = 15;</pre>	^				
2	int Num2 = 25;	COM8				
3						
4						
5	void setup()	ถูกตูอง				
6	{	ถูกตุอง				
7	<pre>Serial.begin(115200);</pre>	ถูกด้อง				
8	}	ากกล่อง				
9		กกต้อง				
10	void loop()	ถูกต้อง				
11	{	ถูกต้อง				
12	if (Num1 == Num2)	ถูกตู้อง				
13	{	ถูกดูอง				
14	serial.println("ใผถูกตอง");	ถูกดูอง				
15	delay(1000);	ถูกตุอง				
16	}	ายอดอา				
17	else if (Num1 < Num2)	(INDO				
18	{					
19	Serial.println("ຖູກທອນ");	Autoscroll Show timestamp				
20	delay(1000);					
21	}					
22	else					
23	i Comial println(" ").					
24	Seriar.printin();					
25	1					
27	1					
61						

หน่วยการเรียนรู้ที่ ๕ การลงมือเขียนโปรแกรม

กลุ่มสาระการเรียนรู้วิทยาศาสตร์ สาระที่ ๔ เทคโนโลยี

มาตรฐาน ว ๔.๒ เข้าใจและ ใช้แนวคิคเชิงคำนวณในการแก้ปัญหาที่พบในชีวิตจริงอย่างเป็นขั้นตอนและ เป็นระบบ ใช้เทคโนโลยีสารสนเทศและการสื่อสารในการเรียนรู้ การทำงาน และการแก้ปัญหาได้อย่างมี ประสิทธิภาพ รู้เท่าทัน และมีจริยธรรม

ตัวชี้วัด

ງ ໔.ໄຫ [ູ] ມ.໑/໑	ออกแบบอัลกอริทึมที่ใช้แนวกิคเชิงนามธรรมเพื่อแก้ปัญหาหรืออธิบายการทำงานที่
	พบในชีวิตจริง
ີ ໔.ໂຫ ມ.໑/ໂຫ	ออกแบบและเขียนโปรแกรมอย่างง่ายเพื่อแก้ปัญหาทางคณิตศาสตร์หรือวิทยาศาสตร์
า ๔.๒ ม.๑/๔	ใช้เทคโนโลยีสารสนเทศอย่างปลอดภัย ใช้สื่อและแหล่งข้อมูลตามข้อกำหนดและ
	ข้อตกลง
ງ ໔.ໂຫ ມ.ໂຫ/໑	ออกแบบอัลกอริทึมที่ใช้แนวคิคเชิงคำนวณในการแก้ปัญหา หรือการทำงานที่พบใน
	ชีวิตจริง
ງ ໔.២ ໗.໕/໑	รวบรวม วิเคราะห์ข้อมูล และใช้ความรู้ด้านวิทยาการคอมพิวเตอร์ สื่อดิจิทัล
	เทคโนโลยีสารสนเทศในการแก้ปัญหาหรือเพิ่มมูลค่าให้กับบริการหรือผลิตภัณฑ์ที่ใช้
	ในชีวิตจริงอย่างสร้างสรรค์
ງ ໔.២ ມ.៦/໑	ใช้เทคโนโลยีสารสนเทศในการนำเสนอและแบ่งปั้นข้อมูลอย่างปลอคภัย มึจริยธรรม
	้ และวิเคราะห์การเปลี่ยนแปลงเทคโนโลยีสารสนเทศที่มีผลต่อการคำเนินชีวิต อาชีพ
	สังคม และวัฒนธรรม

สาระสำคัญ

<u>ความรู้</u>

ผู้เรียนได้เขียนกำสั่งในโปรแกรมและได้ทดลองรันกำสั่ง และผู้เรียนยังได้ประเมินภาษาซีได้อย่าง เข้าใจมากยิ่งขึ้น <u>ทักษะ/กระบวนการ</u>

ผู้เรียนใช้ทักษะการกิดวิเกราะห์ในปัญหาที่พบ

๒. ผู้เรียนใช้ทักษะการคิดวิเคราะห์วิจารณ์ ตอบคำถามและให้โอกาสผู้เรียนแสดงความคิดเห็นเป็น กลุ่มหรือรายบุคคล

<u>คุณลักษณะที่พึงประสงค์</u>

- ๑. ผู้เรียนมีใจรักและพร้อมที่จะเป็นผู้พัฒนาการเขียนภาษาเพื่อควบคุมไมโครคอนโทรลต่อไป
- ๒. รักชาติ ศาสน์ กษัตริย์
- ด. ซื่อสัตย์
- ๔. มีวินัย
- ๕. ใฝ่เรียนรู้
- ๖. อยู่อย่างพอเพียง
- ๗. มุ่งมั่นในการทำงาน

หน่วยการเรียนรู้ที่ 5 การลงมือเขียนโปรแกรม เรื่องย่อยที่ 1 การใช้งานโปรแกรม Arduino IDE

💿 sketch_sep09b | Arduino 1.8.9 \times Sketch Help เมนู 0 sketch_sep09b แทบเครื่องมือ setup() 4 to run once: put your setup code here, 4 5 แก้ไข Code 6 void loop to run repeat 7 your main code here. 8 9) ข้อความแสดงสถานะ รายละเอียดสถานะ แสดงค่าพอร์ตที่เชื่อมต่อบอร์ด

1. เมนูการใช้งานโปรแกรม Arduino IDE

2. การต่อวงจรเข้าบอร์ด Arduino เบื้องต้น

2.1 การต่อและการเขียนโปรแกรม RGB color module

RGB color module คือ วงจรแสดงสี RGB R คือ Red (สีแดง) G คือ Green (สีเขียว) และ B คือ Blue (สีน้ำเงินหรือฟ้า) ซึ่งไฟทั้ง 3 สี จะอยู่ในหลอด LED เดียวกัน โดยการเขียนโปรแกรม Sketch จะกำหนด เงื่อนไขให้ฟังทั้ง 3 สีติดสลับกันวนไปเริ่มจากต่อวงจร RGB color module

รูปด้านถ่างคือ ชุดวงจร RGB color module โดยขาอุปกรณ์จะมีอยู่ด้วยกัน 4 ขา คือ (ไถ่จากซ้ายไป ขาว) - ขาที่ 1 คือ B - ขาที่ 2 คือ G - ขาที่ 3 คือ R - ขาที่ 4 คือ – หรือ GNO (กราวด์)



ทดลองใช้ Arduino ควบคุมหลอด LED RGB



รูป จำลองการต่อหลอด LED RGB

1. ขา R เชื่อมต่อกับหมุด -9	3. ขา B เชื่อมต่อกับหมุด -11
A 1 0	A 1 A

2. ขา G เชื่อมต่อกับหมุด -10 4. ขา Cathode เชื่อมต่อกับหมุด GND

โค้ดโปรแกรมควบคุม Arduino

```
1 int RGBPin[] = { 9, 10, 11 };
 2
 3 void setup() {
 4
     pinMode(RGBPin[0], OUTPUT);
 5
    pinMode(RGBPin[1], OUTPUT);
 6
     pinMode(RGBPin[2], OUTPUT);
 7 }
 8
 9 void SetRGB(int r, int g, int b) {
10
    analogWrite(RGBPin[0], r);
    analogWrite(RGBPin[1], g);
11
12
    analogWrite(RGBPin[2], b);
13 }
14
15 void loop() {
     SetRGB(255, 0, 0); // Red
16
17
     delay(500);
     SetRGB(0, 255, 0); // Green
18
19
    delay(500);
    SetRGB(0, 0, 255); // Blue
20
21
    delay(500);
22
     SetRGB(255, 255, 0); // Yellow
23
     delay(500);
     SetRGB(0, 255, 255); // Sky blue
24
25
    delay(500);
    SetRGB(255, 0, 255); // Pink
26
27
    delay(500);
    SetRGB(255, 255, 255); // White
28
29
     delay(500);
30 }
```

2.2 การเขียนโปรแกรมใช้งานพอร์ตทำหน้าที่เอาท์พุทพอร์ตเบื้องต้น

ทฤษฎีพื้นฐาน

บอร์คArduino รุ่นที่ใช้ไอซีไมโครคอนโทรลเลอร์ ATMEGA328, ATMEGA168, ATMEGA8 จะมี พอร์ตคิจิทัลให้สามารถเชื่อมต่อใช้งานได้ 14 ขาด้วยกันโคยเริ่มจากขา D0, D1,...D13 แต่ในขณะใช้งานที่มี การติดต่อสื่อสารกับคอมพิวเตอร์ผ่านทางพอร์ตอนุกรม (สาย USB ที่กำลังเชื่อมต่อกับคอมพิวเตอร์) ขา Do และ D1 จะถูกใช้งานรับส่งข้อมูลอยู่ ดังนั้นจึงห้ามใช้ในขณะที่ใช้งานพอร์ตอนุกรม



รูป ขาพอร์ตใช้งานของไอซี Arduino เบอร์ ATMEGA328

ฟ้งก์ชันที่ใช้

ในตัวอย่างนี้ประกอบด้วยฟังก์ชันกำหนดโหมดการทำงานให้กับขาพอร์ต สามารถกำหนดได้ทั้ง Pin ดิจิทัลโดยใส่เพียงตัวเลขของขา (0, 1, 2,...13) และ Pin แอนาลอกที่ต้องการให้ทำงานในโหมดดิจิทัลแต่ การใส่ Pin ต้องใส่ A นำหน้าซึ่งใช้ได้เฉพาะ A0, A1,...A5 ส่วน Pin A6 และ A7 ไม่สามารถใช้งานใน โหมดดิจิทัลได้ รูปแบบของฟังก์ชันเป็นดังนี้

pinMode(pin, mode)

pin: หมายเลข Pin ที่ต้องการเซต โหมด mode: INPUT, OUTPUT, INPUT_PULLUP

และพึงก์ชันส่งค่าลอจิกคิจิทัลไปยังขาพอร์ต ค่า HIGH เป็นการส่งลอจิก 1 และค่า LOW เป็นการส่ง ลอจิก 0 ออกไปยังขาพอร์ต ซึ่งพึงก์ชันนี้จะทำงานได้จะต้องมีการใช้ฟังก์ชัน pinMode ก่อน

digitalWrite(pin, value)

pin: หมายเลขขาที่ต้องการเขียนลอจิกออกพอร์ต

value: HIGH or LOW

ฟังก์ชันที่สามคือ ฟังก์ชั่นหน่วงเวลาหรือฟังก์ชันหยุดค้าง การใช้งานสามารถกำหนดตัวเลขของ เวลาที่ต้องการหยุดค้างโดยตัวเลขที่ใส่เป็นตัวเลขของเวลาหน่วยเป็นมิลลิวินาที ตัวเลขของเวลาที่ใส่ได้ สูงสุดคือ 4,294,967,295 ซึ่งเป็นขนาดของตัวแปร unsigned long

delay(ms)		

ms: ตัวเลขที่หยุดค้างของเวลาหน่วยมิลลิวินาที (unsigned long)

<u>การทดลองที่ 1</u> เขียนโปรแกรมหลอดไฟ LED วิ่ง 4 ตัว ติด-ดับ เรียงกันไปโดยให้ติดครั้งละ 1 ตัวและที่ เหลือดับผังงาน

อุปกรณ์ที่ใช้ในการทดลอง







หลอดไฟ LED จำนวน 4 ควง

สายจัมเปอร์ ผู้-ผู้ จำนวน 6 เส้น

Bread Board จำนวน 1 อัน

การต่อวงจร



รูป การต่อวงจรไมโครคอนโทรลเลอร์ Arduino ที่ใช้บอร์คโมคูล Arduino สำเร็จรูปลงบอร์คทคลอง

คำสั่งควบคุม Arduino

```
1
   int LED1 = 2;
 2
   int LED2 = 3;
 3
   int LED3 = 4;
 4
   int LED4 = 5;
 5
   void setup()
 6
 7
   Serial.begin(9600);
 8
   pinMode(LED1,OUTPUT);
 9
   pinMode(LED2,OUTPUT);
10
   pinMode (LED3, OUTPUT);
11
   pinMode(LED4,OUTPUT);
12
13
   void loop()
14
15
   digitalWrite(LED1,HIGH);
   digitalWrite(LED2,LOW);
16
17
   digitalWrite(LED3,LOW);
18
   digitalWrite(LED4,LOW);
   delay(500);
19
20
   digitalWrite(LED1,LOW);
21
   digitalWrite(LED2,HIGH);
22
   digitalWrite(LED3,LOW);
   digitalWrite(LED4,LOW);
23
   delay(500);
24
25
   digitalWrite(LED1,LOW);
26
   digitalWrite(LED2,LOW);
27
   digitalWrite(LED3, HIGH);
   digitalWrite(LED4,LOW);
28
   delay(500);
29
30
   digitalWrite(LED1,LOW);
31
   digitalWrite(LED2,LOW);
32
   digitalWrite(LED3,LOW);
33
   digitalWrite(LED4,HIGH);
34
   delay(500);
35
```

รายละเอียดโค้ดโปรแกรม

- บรรทัดที่ 1-4 กำหนดชื่อ LED1, 2, 3, 4 ให้แทน 2, 3, 4, 5 (เป็นบาที่ LED ต่ออยู่กับ Arduino)
- บรรทัดที่ 7 กำหนดอัตราเร็วของการสื่อสารของพอร์ตอนุกรม (กำหนด Baud rate)
- บรรทัดที่ 8-11 กำหนด โหมดให้กับขาพอร์ตทา งานเป็นเอาต์พุทพอร์ต
- บรรทัคที่ 15 ส่งลอจิก 1 ออกพอร์ตที่ใช้ชื่อว่า LED1
- บรรทัคที่ 16 ส่งลอจิก 0 ออกพอร์ตที่ใช้ชื่อว่า LED2
- บรรทัคที่ 17 ส่งลอจิก 0 ออกพอร์ตที่ใช้ชื่อว่า LED3
- บรรทัคที่ 18 ส่งลอจิก 0 ออกพอร์ตที่ใช้ชื่อว่า LED4
- บรรทัคที่ 19 หน่วงเวลา 200 mS (1/5 วินาที)

<u>การทคลองที่ 2</u> เขียนโปรแกรมไฟวิ่ง LED 4 ตัวติดดับเรียงกันไปโดยให้ติดครั้งละ 1 ตัวและที่เหลือดับ เช่นเดียวกับการทคลองที่หนึ่ง แต่ใช้วิธีการเขียนฟังก์ชันรองชนิดรับค่าแต่ไม่ส่งคืนค่าขึ้นใช้งานเองผังงาน

คำสั่งควบคุม Arduino

```
1
   char LED pin[] = {2,3,4,5};
    void send2port(byte data);
 2
 3
    void setup()
 4
 5 Serial.begin(9600);
 6
   for(char i=0;i<4;i++)
 7
 8 pinMode(LED pin[i],OUTPUT);
 9
10
    - }
   void loop()
12
   {
13 send2port(0B1000);
14 delay(500);
15 send2port(0B0100);
16 delay(500);
17 send2port(0B0010);
18 delay(500);
19 send2port(0B0001);
20 delay(500);
21
    - 3
22 void send2port(byte data)
23
   - {
24 if (data & 1 ){digitalWrite(LED_pin[0],HIGH);} else {digitalWrite(LED_pin[0],LOW);}
25 if (data & 2 ){digitalWrite(LED_pin[1],HIGH);} else {digitalWrite(LED_pin[1],LOW);}
26 if (data & 4 ){digitalWrite(LED_pin[2],HIGH);} else {digitalWrite(LED_pin[2],LOW);}
27 if (data & 8 ){digitalWrite(LED pin[3],HIGH);} else {digitalWrite(LED pin[3],LOW);}
28 }
```

รายละเอียดโค้ดโปรแกรม

- บรรทัคที่ 1 ประกาศตัวแปรแบบอาร์เรย์ชื่อ LED_pin และกำหนคค่าเป็น 2, 3, 4, 5 (เป็นขาที่ LED ต่ออยู่ กับ Arduino)

- บรรทัคที่ 2 ประกาศรูปแบบ (Prototype) ของพึงก์ชันรอง ซึ่งเป็นพึงก์ชำนให้ทำหน้าที่รับค่าที่ส่งมาจาก ฟังก์ชันหลักแล้วคำนวณเพื่อส่งลอจิกออกพอร์ต

- บรรทัดที่ 5 กำหนดอัตราเร็วของการสื่อสารของพอร์ตอนุกรม (กำหนด Baud rate)

- บรรทัคที่ 6-9 กำหนดโหมดให้กับขาพอร์ตทำงานเป็นเอาต์พุทพอร์ต โดยใช้กำสั่ง for ในการวนทำซ้ำ - บรรทัคที่ 14 หน่วงเวลา 200 mS (1/5 วินาที)

- บรรทัดที่ 22-28 ฟังก์ชันรองที่รับค่าเข้ามาจากฟังก์ชันหลักผ่านตัวแปร data แล้วนำค่าเข้ามาตรวจสอบว่า ควรจะต้องส่งลอจิก 1 หรือลอจิก 0 ออกทางพอร์ตใด

<u>การทดลองที่ 3</u> การทดลองต่อ Motor Drive Module L298N (ขับมอเตอร์) กับมอเตอร์ล้อ

สิ่งที่เป็นพลังขับเคลื่อนหลักให้กับหุ่นยนต์นั้นคือ มอเตอร์ ซึ่งต้องการการควบคุมจาก ชุคไดรว์ มอเตอร์ (Motor Driver) ที่จะมาควบคุม ทั้ง ทิศทาง และ ความเร็วซึ่งตัวขับมอเตอร์ในมี่นี้คือ Module L298N ตัวขับมอเตอร์มี 2 ตัวแบบแยกอิสระ สามารถควบคุมความเร็วมอเตอร์ได้ ใช้ไฟ 5 โวลต์ สามารถรับ ไฟเข้า 7-35 โวลต์ได้ มีวงจรเรกูเลตในตัว ขับกระแสสูงสุด ข้างละ 2A Module L298N เป็นชุดขับมอเตอร์ ชนิด H-Bridge ซึ่งส่วนใหญ่จะถูกนำไปใช้ในการควบคุมทิศทางและความเร็วของมอเตอร์

Out 1: ช่องต่อขั้วไฟของมอเตอร์ A Out 2: ช่องต่อขั้วไฟของมอเตอร์ A Out 3: ช่องต่อขั้วไฟของมอเตอร์ B Out 4: ช่องต่อขั้วไฟของมอเตอร์ B 12V: ช่องจ่ายไฟเลี้ยงมอเตอร์ 12V (ต่อได้ตั้งแต่ 5V ถึง 35V) GND: ช่องต่อไฟลบ (Ground) 5V: ช่องจ่ายไฟเลี้ยงมอเตอร์ 5V (หากมีการต่อไฟเลี้ยงที่ช่อง 12V แล้ว ช่องนี้จะทำหน้าที่จ่ายไฟออก เป็น 5V Output สามารถต่อไฟจากช่องนี้ไปเลี้ยงบอร์ค Arduino ได้ ENA: ช่องต่อสัญญาณ PWM สำหรับมอเตอร์ A IN1: ช่องต่อสัญญาณลอจิคเพื่อควบคุมทิศทางของมอเตอร์ A IN2: ช่องต่อสัญญาณลอจิคเพื่อควบคุมทิศทางของมอเตอร์ A IN3: ช่องต่อสัญญาณลอจิคเพื่อควบคุมทิศทางของมอเตอร์ B IN4: ช่องต่อสัญญาณลอจิคเพื่อควบคุมทิศทางของมอเตอร์ B ENB: ช่องต่อสัญญาณ PWM สำหรับมอเตอร์ B

อุปกรณ์ที่ใช้ในการทดลอง







Motor Drive Module L298N 1 ตัว

มอเตอร์จำนวน 1 อัน

ล้อ จำนวน 1 ล้อ

สายจัมเปอร์ ผุ้ – ผู้ จำนวน 5 เส้น

การต่อวงจร

- 1. สายจัมเปอร์สีน้ำเงินต่อที่ช่องต่อ GND (กราวน์)
- 2. สายจัมเปอร์สีแคงต่อที่ช่องไฟ 5 โวลต์
- 3. สายจัมเปอร์สีส้มต่อที่ขา IN3
- 4. สายจัมเปอร์สีน้ำตาลต่อที่งา IN4
- 5. สายจัมเปอร์สีขาวต่อที่ขา ENB



คำสั่งควบคุม Arduino



หน่วยการเรียนรู้ที่ 5 การลงมือเขียนโปรแกรม เรื่องย่อยที่ 2 โครงสร้างโปรแกรมและคำสั่ง

1. โครงสร้างของภาษาซีสำหรับ Arduino

โครงสร้างภาษาซีสำหรับ Arduino ถูกจัดใหม่ให้ง่ายต่อนักเขียนโปรแกรมเบื้องต้นซึ่งผู้ออกแบบได้ จัดวางให้ผู้ใช้งานได้ใช้งานง่ายซึ่งโครงสร้างหลัก ๆ จะมีเพียง 2 ส่วนเท่านั้นคือ

1. setup เป็นส่วนที่เก็บฟังก์ชันที่ทำงานครั้งเดียว

2. loop เป็นส่วนที่เก็บฟังก์ชันที่เมื่อทำงานครบแล้วจะวนกลับมาทำซ้ำใหม่ตั้งแต่ต้น

แต่ถ้าต้องการเขียนโปรแกรมขั้นสูงสามารถเขียนในส่วนหัวโปรแกรมและส่วนของฟังก์ชันรองที่ เขียนขึ้นใช้งานเองเพื่อให้ใช้งานสะดวกมากยิ่งขึ้นได้เช่นเดียวกับภาษาซีมาตรฐาน



โครงสร้างภาษาซีสำหรับ Arduino

ตัวอย่างโปรแกรมที่เขียนด้วยโครงสร้างแบบพื้นฐาน

<pre>void setup() { Serial.begin(9600); pinMode(2,OUTPUT); pinMode(3,OUTPUT); pinMode(4,OUTPUT); </pre>	พังก์ชันที่ทำงานครั้งเดียว setup()
<pre>pinMode(5,OUTPUT);</pre>	
}	
vola loop()	
digitalWrite(2 HICE);	
digitalWrite(2,fildf);	
digitalWrite(4.LOW)	
digitalWrite (5,LOW) :	
delay(200);	
digitalWrite(2,LOW);	
<pre>digitalWrite(3,HIGH);</pre>	
<pre>digitalWrite(4,LOW);</pre>	
<pre>digitalWrite(5,LOW);</pre>	e de de é de atum
delay(200);	พงกชนททางานครงเดยว Setup()
digitalWrite(2,LOW);	
digitalWrite(3,LOW);	
digitalWrite (4, HIGH);	
delaw(200):	
digitalWrite(2.LOW)	
digitalWrite(3.LOW):	
digitalWrite (4,LOW);	
digitalWrite(5,HIGH);	
delay(200);	
F	

ตัวอย่างโปรแกรมที่เขียนด้วยโครงสร้างแบบขั้นสูง (ที่ให้ผลแบบเดียวกับขั้นพื้นฐาน)



2. ตัวแปรในภาษาซี

การประกาศตัวแปรคือ การจองพื้นที่ในหน่วยความจำเพื่อนำไปใช้งานในฟังก์ชัน โดยการใช้ชื่อตัว แปรแทนการกำหนดเป็นก่าแอดเครสงองหน่วยความจำรูปแบบงองการประกาศตัวแปรเป็นดังนี้

2.1 หลักการตั้งชื่อตัวแปรในภาษาซี

หลักการตั้งชื่อตัวแปรมีข้อกำหนดหลัก ๆ อยู่ 4 ประการด้วยกันคือ

- 1. ชื่อที่ตั้งต้องไม่ซ้ำกับกำสงวนของภาษาซี (กำที่ภาษาซีมีใช้งานอยู่แล้ว)
- 2. การใช้ตัวอักษรใหญ่กับตัวอักษรเล็กถือว่าเป็นคนละตัว
- 3. ตัวแรกของชื่อตัวแปรต้องเป็นตัวอักษรเท่านั้นตัวถัดไปเป็นตัวเลขได้
- 4. ชื่อตัวแปรห้ามเว้นวรรค
- 2.2 ชนิดของตัวแปรในภาษาซีสำหรับไมโครคอนโทรลเลอร์ Arduino

ชนิด	บบ	เาค	າເລາມາແຫ
ואאת	บิต	ไบต์	ואַניזראַנ
boolean	8	1	true, false
char	8	1	-128 ถึง +127
unsigned char	8	1	0 ถึง 255
byte	8	1	0 ถึง 255
int	16	2	-32768 ถึง +32767
unsigned int	16	2	0 ถึง 65535
long	32	4	-2147483648 ถึง +2147483647
unsigned long	32	4	0 ถึง 4294967295
float	32	4	±3.4E±38 (~7 digits)
double	64	8	±1.7E±308 (~15 digits)

ชนิดของตัวแปรในภาษาซีสำหรับไมโกรกอนโทรลเลอร์ Arduino
2.3 พอยน์เตอร์

พอยน์เตอร์ เป็นตัวชี้ตำแหน่งข้อมูลของตัวแปรอื่นที่เก็บในหน่วยความจำโดยตัวพอยน์เตอร์เป็นตัว เก็บตำแหน่งแทนการเก็บข้อมูลในการใช้งานพอยน์เตอร์จะใช้เกรื่องหมาย * นำหน้าและใช้เครื่องหมาย & เมื่อต้องการก่าตำแหน่งของตัวแปรอื่นโดยมีรูปแบบดังนี้



2.4 อาร์เรย์

อาร์เรย์เป็นการเพิ่มความสามารถในการเก็บข้อมูลของตัวแปรให้สามารถเก็บเป็นชุดได้ โดยใช้ชื่อ ตัวแปรเดิมได้ การใช้งานตัวแปรอาร์เรย์จะใช้เครื่องหมาย[] ต่อท้ายตัวแปรโดยภายในวงเล็บเป็นตัวเลข ซึ่ง สามารถใช้งานได้หลายมิติ มีรูปแบบดังนี้

ູສູປແบນ

ชนิดของตัวแปร ชื่อตัวแปร[ตัวเลข];	// เป็นอาร์์เรย์ 1 มิติ
ชนิดของตัวแปร ชื่อตัวแปร[ตัวเลข,ตัวเลข];	// เป็นอาร์์เรย์ 2 มิติ
ชนิดของตัวแปร ชื่อตัวแปร[ตัวเลข,ตัวเลข,ตัวเลข];	// เป็นอาร์เรย์ 3 มิติ



int x[4]; x[0] x[1] x[2] x[3]

ตัวอย่างอาร์เรย์ 2 มิติ

int x[3,3];

x[0]	x[1]	x[2]	x[3]
x[0]	x[1]	x[2]	x[3]
x[0]	x[1]	x[2]	x[3]
x[0]	x[1]	x[2]	x[3]

ตัวอย่างอาร์เรย์ 3 มิติ

x[0,0,0]	x[0,1,0]	x[0,2,0]
x[1,0,0]	x[1,1,0]	x[1,2,0]
x[2,0,0]	x[2,1,0]	x[2,2,0]

2.5 ตัวดำเนินการในภาษาซี

ตัวดำเนินการในภาษาซีแบ่งตามลักษณะการกระทำได้ 5 กลุ่มด้วยกันกือ

- 1. ตัวกระทำทางคณิตศาสตร์
- 2. ตัวกระทำทางลอจิกระดับบิต
- 4. ตัวกระทำเปรียบเทียบ
- 5. ตัวกระทำประสม

้โดยในแต่ละลักษณะมีรายละเอียดดังต่อไปนี้

ตัวกระทำทางคณิตศาสตร์

เครื่องหมาย	การกระทำ	ตัวอย่าง	คำอชิบาย
+	บวก	x=y+z;	x เท่ากับค่าในตัวแปร y บวกกับค่าในตัวแปร z
-	ດນ	x=y-z;	x เท่ากับค่าในตัวแปร y ลบด้วยค่าในตัวแปร z
*	ፈመ	x=y*z;	x เท่ากับค่าในตัวแปร y คูณด้วยค่าในตัวแปร z
/	หาร	x=y/z;	x เท่ากับค่าในตัวแปร y หารด้วยค่าในตัวแปร z
%	หารเอาเศษ	x=y%z;	x เท่ากับเศษของการหารระหว่างตัวแปร y กับตัวแปร z

ตัวกระทำทางลอจิก

เครื่องหมาย	การกระทำ	ตัวอย่าง	คำอธิบาย
&	แอนค์	x=y&z	x เท่ากับค่าในตัวแปร y แอนด์กับค่าในตัวแปร z
	ออร์	x=y z;	x เท่ากับค่าในตัวแปร y ออร์กับค่าในตัวแปร z
^	เอ็กซ์คลูซิฟออร์	x=y^z;	x เท่ากับค่าในตัวแปร y เอ็กซ์คลูซิฟออร์กับ z
~	วันคอมพลีเมนต์	x=~y;	x เท่ากับค่าตรงข้ามของค่าในตัวแปร y
<<	เลื่อนไปทางซ้าย	x=x<<1;	เลื่อนข้อมูลใน x ไปทางซ้ายไป 1 บิต

ตัวกระทำบูลีน

เครื่องหมาย	การกระทำ	คำอธิบาย
&&	แอนด์	เชื่อมเงื่อนไข 2 เงื่อนไขด้วยกา ว่า "และ"
	ออร์	เชื่อมเงื่อนไข 2 เงื่อนไขด้วยกา ว่า "หรือ"
!	อินเวิร์ส	ใช้ตรวจสอบตัวแปรว่าเท่ากับศูนย์หรือไม่เช่น if (!x)

ตัวกระทำการเปรียบเทียบ

เครื่องหมาย	การกระทำ	ตัวอย่าง	คำอธิบาย
>	ນາກກວ່າ	x>10	x มากกว่า 10
<	น้อยกว่า	x<10	x น้อยกว่า 10
>=	มากกว่าหรือเท่ากับ	x>=10	x มากกว่าหรือเท่ากับ 10
<=	น้อยกว่าหรือเท่ากับ	x<=10	x น้อยกว่าหรือเท่ากับ 10
==	เท่ากับ	x==10	x เท่ากับ 10
!=	ไม่เท่ากับ	x!=10	x ไม่เท่ากับ 10

เครื่องหมาย	การกระทำ	ตัวอย่าง	คำอชิบาย
++	เพิ่มค่า 1 ค่า	x++;	เพิ่มค่า x ขึ้น 1 ค่า
	ลดค่า 1 ค่า	x;	ถุดค่า x ถง 1 ค่า
+=	บวก	x+=2;	x ใหม่เท่ากับ x เดิมบวกกับ 2
-=	ດນ	x-=2;	x ใหม่เท่ากับ x เดิมลบด้วย 2
=	คูณ	x=2;	x ใหม่เท่ากับ x เดิมคูณด้วย 2
/=	หาร	x/=2;	x ใหม่เท่ากับ x เดิมหารด้วย 2
%=	หารเอาเศษ	x%=2;	x ใหม่เท่ากับ x เดิมหารด้วย 2 แล้วเอาเศษ
&=	แอนด์	x&=2;	x ใหม่เท่ากับ x เดิมแอนด์ด้วย 2
=	ออร์	x =2;	x ใหม่เท่ากับ x เดิมออร์ด้วย 2

ตัวกระทำประสม

3. ไวยากรณ์ภาษาซี

 ประกาศช่อแทน เป็นการประกาศใช้ชื่ออื่นแทนค่าที่ต้องการเพื่อให้สะควกต่อการเขียนโปรแกรม รูปแบบเป็นดังนี้

#define constantName value constantName: ชื่อที่ต้องการกำหนดตั้ง value: ค่าที่ต้องการกำหนดให้ชื่อนั้นมีค่าเท่ากับ

ตัวอยาง

#define LED 13 หมายถึง กำหนดให้คำว่า LED มีค่าเท่ากับ 13

 การรวมไฟล์อื่นเข้ามาร่วม เป็นการประกาศไฟล์อื่น ๆ เข้ามารวมกับตัวโปรแกรมก่อนการ กอมไพล์ รูปแบบเป็นดังนี้



file: ชื่อไฟล์ที่ต้องการนา เข้ามารวมกับโค้คโปรแกรมที่กา ลังเขียนขึ้น

ตัวอย่าง

#include <SPI.h> หมายถึง ให้โปรแกรมคอมไพลเลอร์ทำการรวมไฟล์ที่ชื่อ SPI.h ก่อนทำการคอมไพล์

3. การใส่หมายเหตุลงในโค้ดโปรแกรม เป็นการใส่ข้อความใด ๆ ลงในโค้คโปรแกรมเพื่อที่จะ อธิบายโปรแกรมหรือเพื่อบันทึกความจำว่าโค้คในตำแหน่งนั้น ๆ เขียนขึ้นเพื่อประสงค์สิ่งใค การใส่หมาย เหตุจะต้องใส่เครื่องหมายเพื่อให้คอมไพล์เลอร์ข้ามการคอมไพล์ในส่วนนี้รูปแบบเป็นดังนี้

//.....ใช้ในกรณีบรรทัดเดียว

/*......*/ ใช้ในกรณีหลายบรรทัด

 การใส่เครื่องหมายท้ายฟังก์ชัน ภาษาซีเป็นภาษาที่มีการใส่เครื่องหมายท้ายฟังก์ชันซึ่ง เป็นสิ่งที่แตกต่างและ โดดเด่นกว่าโปรแกรมในภาษาอื่น ๆ โดยมีหลักคิดดังนี้

; สำหรับฟังก์ชันที่ทำงานเสร็จสิ้นในตัว

{} สำหรับฟังก์ชันที่มีฟังก์ชันอื่นรวมเข้าไปด้วย

หน่วยการเรียนรู้ที่ 5 การลงมือเขียนโปรแกรม

เรื่องย่อยที่ 3 โครงสร้างเงื่อนไข คำสั่งในการติดต่อกับพอร์ต จอแอลซีดี และการสร้างโปรแกรมย่อย

1. ฟังก์ชันการดำเนินการแบบทางเลือก

ในการเขียนโปรแกรมเพื่อให้ไมโครคอนโทรลเลอร์ทำงานในสิ่งที่ต้องการ นอกจากฟังก์ชันที่สั่ง ให้ทำงานเป็นลำดับแล้วจำเป็นต้องใช้ฟังก์ชันที่มีการทำงานแบบให้เลือกเส้นทางการทำงาน โดยการทำ ตามเงื่อนไขหรือการให้ทำซ้ำ แบบมีเงื่อนไขหรือไม่มีเงื่อนไข โดยฟังก์ชันที่มีการทำงานแบบทางเลือก ในภาษาซีมีด้วยกัน 4 ฟังก์ชันคือ

- 1. ฟังก์ชัน if
- 2. ฟังก์ชัน if-else
- 3. ฟังก์ชัน if-else if-else
- 4. ฟังก์ชัน switch

1.1 ฟังก์ชัน if (ทางเลือกเดียว)

ฟังก์ชัน if เป็นฟังก์ชันที่มีการตรวจสอบเงื่อนไข โดยถ้าเงื่อนไขเป็นจริงจะทำงานตามชุดฟังก์ชัน ที่กำหนดไว้ รูปแบบเป็นดังนี้

<u>if (เงื่อนไขที่ตรวจสอบ)</u>



*หากชุดฟังก์ชันที่ให้ทำงานเมื่อเงื่อนไขเป็นจริงมีเพียงฟังก์ชันเดียว ไม่ต้องใส่วงเล็บปีกกา {...} ก็ได้

1.2 ฟังก์ชัน if-else (สองทางเลือก)

การตรวจสอบเงื่อนไขที่มีชุดฟังก์ชันให้ทำงานเมื่อเงื่อนไขเป็นจริงและมีชุดฟังก์ชันให้ทำงาน เมื่อเงื่อนไขเป็นเท็จเราจะใช้ฟังก์ชัน if-else มาใช้งาน มีรูปแบบคังนี้

if (เงื่อนไขที่ตรวจสอบ) else



1.3 ฟังก์ชัน if-else if....else (หลายทางเลือก)

เป็นฟังก์ชันที่มีการตรวจสอบเงื่อนไขหลายเงื่อนไขและมีชุดฟังก์ชันที่เตรียมให้ทำงานในแต่ละ เงื่อนไขหากเงื่อนไขนั้น ๆ ถูกต้อง

<u>if (เง่อนไขที่ตรวจสอบ) else if (เง่อนไขที่ตรวจสอบ)</u>



1.4 ฟังก์ชัน switch...case (หลายทางเลือก)

ฟังก์ชัน switch...case เป็นฟังก์ชันหลายทางเลือกอีกฟังก์ชันหนึ่งที่มีการทำงานคล้าย ๆ ฟังก์ชัน if-else if...else ต่างตรงที่การตรวจสอบเงื่อนไขจะใช้การตรวจสอบการเท่ากันของตัวแปรที่ใช้ตรวจสอบ เท่านั้น โดยเมื่อตรวจสอบก่าแล้วเท่ากับก่าที่กำหนดให้ทำฟังก์ชันที่เตรียมไว้



1.5 ฟังก์ชันการดำเนินการแบบวนซ้ำ

การเขียนโปรแกรมสั่งงานไมโครคอนโทรลเลอร์ต้องมีการทำงานแบบวนซ้ำหรือวนรอบ เพื่อที่จะทำงานในชุดกำสั่งเดิม ลักษณะการทำงานมีทั้งแบบมีเงื่อนไขหรือไม่มีเงื่อนไข ในภาษาซีมี ฟังก์ชันสั่งงานให้ไมโครคอนโทรลเลอร์ทำงานซ้ำ มีดังนี้

- 1. ฟังก์ชัน for
- 2. ฟังก์ชัน while
- 3. ฟังก์ชัน while(1)
- 4. ฟังก์ชัน do-while

1.6 ฟังก์ชัน for

ฟังก์ชัน for เป็นฟังก์ชันที่ใช้ในกรณีที่ทราบจำนวนรอบที่จะทำงานซ้ำ โคยมีรูปแบบคังนี้



ตัวอย่าง

โค้ดโปรแกรม	คำอธิบาย
for(int i=0;i<10;i++)	ประกาศและกำหนดตัวแปรนับรอบเป็นตัวแปร i
{	เป็นตัวแปรชนิด integer โดยให้มีก่าเริ่มต้นเท่ากับ
digitalWrite(13,HIGH);	ศูนย์ทำวนซ้ำไปเรื่อย ๆ หากก่าตัวแปรยังน้อยกว่า
delay(500);	10 โดยในรอบถัดไปให้เพิ่มก่าในตัวแปรนับรอบ
digitalWrite(13,LOW);	ขึ้น 1 ค่า
delay(500);	
}	

1.7 ฟังก์่ชัน while

ฟังก์ชัน while เป็นฟังก์ชันที่ให้ทำงานวนซ้ำหรือวนรอบโดยมีการตรวจสอบเงื่อนไขก่อนถ้าหาก เงื่อนไขเป็นจริงจะทำงานตามชุดฟังก์ชันที่เตรียมไว้ เมื่อทำงานในชุดฟังก์ชันที่เตรียมไว้เสร็จจะมีการวน กลับไปตรวจสอบเงื่อนไขอีก หากเงื่อนไขเป็นจริงจะทำงานในชุดฟังก์ชันที่เตรียมไว้โดยทำแบบนี้ไป เรื่อย ๆ จนกว่าเงื่อนไขจะเป็นเท็จจะออกจากวงรอบการทำซ้ำ โดยมีรูปแบบดังนี้



ตัวอย่าง

โค้ดโปรแกรม	คำอธิบาย
i=0;	ตรวจสอบก่อนว่าเงื่อนไขเป็นจริงอยู่หรือไม่ (i ยัง
while(i<10)	น้อยกว่า 10) หากเงื่อนไขเป็นจริงให้ทำงานที่
{	เตรียมไว้ เมื่อทำงานครบให้กลับมาตรวจสอบ
digitalWrite(13,HIGH);	เงื่อนใขใหม่วนซ้ำไปเรื่อย ๆ จนกว่าเงื่อนใขจะเป็น
delay(500);	เท็จ
digitalWrite(13,LOW);	
delay(500);	
i++;	
}	

1.8 ฟังก์ชัน while(1)

ฟังก์ชัน while เป็นฟังก์ชันที่ให้ทา งานวนซ้ำ หรือวนรอบไม่รู้จบ เนื่องจาก 1 คือเป็นจริงตลอดไป ในโปรแกรม Arduino ก็คือฟังก์ชัน loop() นั่นเอง



1.9 ฟังก์ชัน do-while

ฟังก์ชัน do-while เป็นฟังก์ชันที่ให้ทำงานวนซ้ำหรือวนรอบ โดยมีการตรวจสอบเงื่อนไขการ ทำงานคล้ายกับฟังก์ชัน while ต่างตรงที่ฟังก์ชัน do-while จะทำงานในชุดฟังก์ชันที่เตรียมไว้ทำซ้ำไป ก่อน 1 รอบแล้วจึงตรวจสอบเงื่อนไข



ตัวอย่าง

โค้ดโปรแกรม	คำอธิบาย
i=0;	ทำงานในฟังก์ชันที่เตรียมไว้ แล้วตรวจสอบ
do	เงื่อนไขว่าตัวแปร i ยังมีค่าน้อยกว่า 10 หรือไม่หาก
{	ยังน้อยกว่าให้วนกลับไปทำใหม่ซ้ำๆ จนกว่า
digitalWrite(13,HIGH);	เงื่อนไขละเป็นเท็ล
delay(500);	
digitalWrite(13,LOW);	
delay(500);	
i++;	
} while(i<10)	

ตัวอย่างการทคลอง ต่อสายสัญญาณดังนี้ Vcc -> 5V Gnd -> Gnd SDA -> 4 SCL -> 5 เขียนกำสั่งควบคุมด้วยภาษา c



2. การสร้างฟังก์ชันรองขึ้นใช้เอง

ฟังก์ชันต่าง ๆ ที่ได้กล่าวมาแล้วเป็นพึงก์ชันที่ภาษาซีมิให้ใช้งาน แต่ถ้าหากผู้ใช้งานต้องการ ฟังก์ชันที่มีการทำงานตามลักษณะเฉพาะส่วนอย่างใดอย่างหนึ่งที่มีการเรียกใช้งานซ้ำๆ หรือเพื่อแยกงาน ให้เป็นส่วน ๆ ให้ง่ายต่อการเขียนโปรแกรมสามารถเขียนขึ้นเพื่อใช้งานเองได้ การจัดวางตำแหน่งของ ฟังก์ชันรองสามารถทำได้สองแบบกือ

 วางก่อนพึงก์ชันที่เรียกใช้งาน ตำแหน่งการวางลักษณะในนี้ไม่ต้องประกาศรูปแบบของพึงก์ชัน (Prototype) ซึ่งจะต้องวางพึงก์ชันรองต่อจากหัวโปรแกรมและก่อนพึงก์ชัน setup การวางลักษณะนี้มีข้อดี ตรงที่ไม่ต้องเขียนประกาศรูปแบบ แต่จะทำให้พึงก์ชันที่ใช้งานหลัก (setup,loop) ถูกดันลงไปท้าย ๆ โปรแกรมซึ่งหากมีพึงก์ชันรองหลาย ๆ พึงก์ชันยิ่งถูกดันไปล่าง ๆ ของโปรแกรมมากขึ้น หากโปรแกรมมี พึงก์ชันรองหลายสิบหลายร้อยบรรทัดจะส่งผลให้หาโปรแกรมหลักเพื่อเขียนโปรแกรมยากขึ้นการแก้ไข โปรแกรมก็จะยากขึ้นตาม

 2. วางหลังฟังก์ชันที่เรียกใช้งาน ตำแหน่งการวางลักษณะนี้จะต้องประกาศรูปแบบของฟังก์ชัน (Prototype) ซึ่งจะต้องมีเครื่องหมาย ; ปิดท้ายด้วย โดยประกาศไว้ในท้ายของส่วนหัวโปรแกรมการวาง ลักษณะนี้มีข้อดีตรงที่ฟังก์ชันหลัก (setup, loop) ยังอยู่ตอนบนของโปรแกรมตลอดไม่ว่าจะมีฟังก์ชันรองกี่ ฟังก์ชันก็ตามทำให้การตรวจสอบแก้ไขปรับปรุงในครั้งหลังทำได้สะดวก



(วางก่อนฟังก์ชันที่เรียกใช้งาน)

(วางหลังฟังก์ชันที่เรียกใช้งาน)

โครงสร้างการจัดวางฟังก์ชันรองในภาษาซี

วางก่อนฟังก์ชันที่เรียกใช้งาน	วางหลังฟังก์ชันที่เรียกใช้งาน
#define LED1 2	#define LED1 2
#define LED2 3	#define LED2 3
#define LED3 4	#define LED3 4
void pattern1(void)	<pre>void pattern1(void);</pre>
{	<pre>void pattern2(void);</pre>
<pre>digitalWrite(LED1,HIGH);</pre>	<pre>void pattern3(void);</pre>
<pre>digitalWrite(LED2,LOW);</pre>	void setup()
<pre>digitalWrite(LED3,LOW);</pre>	{
}	pinMode (LED1, OUTPUT) ;
void pattern2(void)	pinMode (LED2,OUTPUT);
{	pinMode (LED3,OUTPUT) ;
<pre>digitalWrite(LED1,LOW);</pre>	}
<pre>digitalWrite(LED2,HIGH);</pre>	void loop()
<pre>digitalWrite(LED3,LOW);</pre>	{
}	<pre>pattern1(); delay(200);</pre>
void pattern3(void)	<pre>pattern2(); delay(200);</pre>
{	<pre>pattern3(); delay(200);</pre>
<pre>digitalWrite(LED1,LOW);</pre>	}
<pre>digitalWrite(LED2,LOW);</pre>	void pattern1(void){
<pre>digitalWrite(LED3,HIGH);</pre>	<pre>digitalWrite(LED1,HIGH);</pre>
}	<pre>digitalWrite(LED2,LOW);</pre>
<pre>void setup()</pre>	<pre>digitalWrite(LED3,LOW);</pre>
{	}
pinMode (LED1, OUTPUT) ;	<pre>void pattern2(void) {</pre>
pinMode (LED2, OUTPUT) ;	digitalWrite(LED1,LOW);
pinMode (LED3, OUTPUT) ;	digitalWrite(LED2,HIGH);
}	<pre>digitalWrite(LED3,LOW);</pre>
void loop()	}
{	<pre>void pattern3(void) {</pre>
<pre>pattern1(); delay(200);</pre>	<pre>digitalWrite(LED1,LOW);</pre>
<pre>pattern2(); delay(200);</pre>	<pre>digitalWrite(LED2,LOW);</pre>
<pre>pattern3(); delay(200);</pre>	<pre>digitalWrite(LED3,HIGH);</pre>
}	}

ฟังก์ชันรองที่เขียนขึ้นใช้เองมีด้วยกัน 4 ประเภทคือ

- ฟังก์ชันที่ไม่รับค่าและไม่ส่งคืนค่า
- ฟังก์ชันที่รับค่าแต่ไม่ส่งคืนค่า
- ฟังก์ชันที่ไม่รับค่าแต่ส่งคืนค่า
- ฟังก์ชันที่รับก่าและส่งคืนก่า

2.1 ฟังก์ชันที่ไม่รับค่าและไม่ส่งคืนค่า

เป็นพึงก์ชันรองที่เขียนขึ้นโดยการรวมชุดพึงก์ชันเพื่อให้ทำงานอย่างหนึ่งอย่างใดเป็นอิสระจาก พึงก์ชันหลักไม่มีการรับค่าใด ๆ จากพึงก์ชันหลักเพื่อนำไปใช้งานในตัวของพึงก์ชันรอง และไม่มีการส่งค่า ใด ๆ กลับมายังพึงก์ชันหลักที่เป็นผู้เรียกใช้งาน ซึ่งข้อกำหนดในการประกาศใช้ของพึงก์ชันประเภทนี้มี ดังนี้

- 1. ข้อกำหนดในการ<u>ไม่รับค่า</u> ทำโดยใส่ (void) หลังชื่อฟังก์ชัน
- 2. ข้อกำหนดในการ<u>ไม่ส่งกืนค่า</u> ทำโดยใส่ void หน้าชื่อฟังก์ชัน

ตัวอย่างโปรแกรมที่มีฟังก์ชันรองที่ไม่รับค่าและไม่ส่งคืนค่า void pattern1(void);

```
sketch sep05a§
   1 #define LED1 2
   2 #define LED2 3
  3 #define LED3 4
   4 void pattern1(void);
   5 void pattern2(void);
   6 void pattern3 (void);
   7 void setup()
  8 {
   9 pinMode(LED1,OUTPUT);
  10 pinMode (LED2, OUTPUT);
  11 pinMode(LED3,OUTPUT);
  12 }
  13 void loop()
 14 {
15 pattern1(); delay(200);
  16 pattern2(); delay(200);
  17 pattern3(); delay(200);
  18 }
di.
 19 void pattern1 (void)
 20 {
di,
  21 digitalWrite (LED1, HIGH);
  22 digitalWrite (LED2, LOW);
  23 digitalWrite(LED3,LOW);
  24 }
 25 void pattern2 (void)
51
 26 {
  27 digitalWrite (LED1, LOW);
  28 digitalWrite (LED2, HIGH);
 29 digitalWrite (LED3, LOW);
  30 }
  31 void pattern3 (void)
 32 {
  33 digitalWrite (LED1, LOW);
  34 digitalWrite (LED2, LOW);
  35 digitalWrite(LED3, HIGH);
  36 1
```

รายละเอียดโค้ดโปรแกรม

- บรรทัคที่ 4, 5, 6 เป็นการประกาศรูปแบบของฟังก์ชันรองที่จะเขียนขึ้นใช้งาน (มี ; ปี คท้าย) และ เป็นฟังก์ชันที่ไม่รับค่าและไม่ส่งคืนค่า (มี void หน้าฟังก์ชันและหลังฟังก์ชัน)
- บรรทัดที่ 14, 15, 16 เป็นการเรียกใช้งานโดยการชื่อฟังก์ชันที่ต้องการตามด้วย () ที่ภายในว่าง
 เปล่าเนื่องจากไม่มีการส่งค่าเข้าในฟังก์ชัน
- บรรทัดที่ 19-24 เป็นฟังก์ชันรองพึงก์ชันแรกที่เขียนขึ้นใช้งานโดยภายในบรรจุฟังก์ชันที่ต้องการ ให้ทำงานเมื่อมีการเรียกใช้
- บรรทัดที่ 25-30, บรรทัดที่ 25-30, บรรทัดที่ 31-36 เป็นฟังก์ชันรองถัด ๆ มาที่เขียนขึ้นใช้งาน

2.2 ฟังก์ชันที่รับค่าแต่ไม่ส่งคืนค่า

เป็นพึงก์ชันที่เขียนขึ้นเพื่อแบ่งย่อยการทำงานของพึงก์ชันหลัก โดยมีการส่งก่าจากพึงก์ชันหลัก ในขั้นตอนการเรียกใช้พึงก์ชันรองเพื่อให้พึงก์ชันรองที่เขียนขึ้นนำไปประมวลผลอย่างใดอย่างหนึ่งตามที่ ผู้สร้างพึงก์ชันต้องการ ซึ่งข้อกำหนดในการประกาศใช้ของพึงก์ชันประเภทนี้มีดังนี้

 1. ข้อกำหนดในการ<u>รับค่า</u>ทำโดยกำหนดชนิดตัวแปร พร้อมตัวแปรที่จะรับค่าท้ายชื่อฟังก์ชันรองที่ สร้างขึ้น เช่น (byte data)

2. ข้อกำหนดในการ**ไม่ส่งกีนก่า** ทำโดยใส่ void หน้าชื่อฟังก์ชัน

ตัวอย่างโปรแกรมที่มีฟังก์ชันรองที่รับค่าและไม่ส่งคืนค่า void send2port(byte data);

```
sketch_sep05a§
 1 #define LED1 2
 2 #define LED2 3
 3 #define LED3 4
 4 #define LED4 5
5 char LED_pin[] = {LED1, LED2, LED3, LED4};
6 void send2port(byte data);
7 void setup()
8 {
9 Serial.begin(9600);
10 for (char i=0;i<4;i++)
11 {
12 pinMode(LED_pin[i],OUTPUT);
13 }
14 }
15 void loop()
16 {
17 send2port(0B1000);delay(200);
18 send2port(0B0100);delay(200);
19 send2port(0B0010);delav(200);
20 send2port(0B0001);delay(200);
21 }
22 void send2port(byte data)
23 {
24 if (data & 1 ) {digitalWrite(LED_pin[0], HIGH);} else {digitalWrite(LED_pin[0], LOW);}
25 if (data & 2 ){digitalWrite(LED_pin[1], HIGH);} else {digitalWrite(LED_pin[1], LOW);}
26 if (data & 4 ) {digitalWrite(LED pin[2], HIGH); } else {digitalWrite(LED pin[2], LOW); }
27 if (data & 8 ) {digitalWrite(LED_pin[3],HIGH);} else {digitalWrite(LED_pin[3],LOW);}
28 }
29
```

รายละเอียดโค้ดโปรแกรม

 บรรทัดที่ 6 เป็นการประกาศรูปแบบของฟังก์ชันรองที่จะเขียนขึ้นใช้งาน (มี ; ปิดท้าย) เป็น ฟังก์ชันที่ไม่มีการส่งกืนค่า (มี void อยู่หน้าฟังก์ชัน) แต่เป็นฟังก์ชันที่รับค่าเพียงอย่างเดียว โดย ในวงเล็บหลังฟังก์ชันจะเป็นตัวแปรที่ใช้สำหรับรับค่าซึ่งจะต้องประกาศชนิดของตัวแปรพร้อม ชื่อตัวแปรสำหรับรับค่าไว้ในวงเล็บท้ายฟังก์ชัน

- บรรทัด 17-20 เป็นการเรียกใช้ฟังก์ชันรองโดยการเรียกชื่อฟังก์ชันรองพร้อมส่งค่าเข้าไปใน ฟังก์ชันโดยค่าที่ส่งไปจะใส่ในวงเล็บท้ายชื่อฟังก์ชันรองที่เรียกใช้ ค่าที่ส่งไปจะเข้าไปเก็บไว้ในตัวแปรที่ ฟังก์ชันรองประกาศไว้ เช่น send2port(0B1000); เป็นการเรียกใช้ฟังก์ชัน send2port โดยส่งค่าตัวเลข 1000 ซึ่งเป็นเลขฐานสองเข้าไปในฟังก์ชันรองนั้นด้วย ค่าตัวเลขดังกล่าวจะถูกเก็บไว้ในตัวแปร data ซึ่งฟังก์ชัน รองได้เตรียมเอาไว้

- บรรทัคที่ 20-28 เป็นฟังก์ชันรองที่เขียนขึ้น ภายในฟังก์ชันมีการนำค่าที่ถูกส่งเข้ามาผ่านตัวแปร data นำมาใช้งาน

2.3 ฟังก์ชันที่ไม่รับก่าแต่ส่งคืนก่า

เป็นฟังก์ชันที่เขียนขึ้นเพื่อแบ่งย่อยการทำงานของฟังก์ชันหลัก เพียงแต่ไม่ได้ส่งข้อมูลเข้า โปรแกรมเพื่อช่วยให้ทำการประมวลผลแต่มีการส่งข้อมูลกลับมายังฟังก์ชันหลัก เช่นฟังก์ชันตรวจสอบ การกดสวิตช์ซึ่งฟังก์ชันนี้ไม่จำเป็นต้องรับข้อมูลใดมาประมวลผลมีเพียงตรวจสอบว่ามีการกดสวิตช์ หรือไม่แล้วส่งค่าการกดสวิตช์กลับไปยังฟังก์ชันหลักเพื่อนำไปใช้งานต่อไป โดยค่าที่ส่งกลับผู้เขียน โปรแกรมจะเป็นผู้กำหนดขึ้นมาเอง เช่น เมื่อสวิตช์ 1 ถูกกดให้ส่งค่า 1 กลับและเมื่อสวิตช์ 2 ถูกกดให้ส่ง ด่า 2 หากไม่มีการกดใด ๆ ให้ส่งกลับค่า 0 เป็นต้น

การเขียนฟังก์ชันรองลักษณะนี้มีข้อกำหนดในการประกาศใช้ของฟังก์ชันดังนี้

1. ข้อกำหนดในการ**ไม่รับก่า** ทำโดยใส่ (void) หลังชื่อฟังก์ชัน

 ข้อกำหนดในการส่งคืนค่า ทำโดยใส่กำหนดชนิดของข้อมูลที่จะส่งคืนหน้าชื่อฟังก์ชันรอง ที่สร้างขึ้น ตัวอย่างการประกาศตัวแปรในขอบเขตที่ต่างกัน

ตัวอย่างโปรแกรมที่มีฟังก์ชันรองที่ไม่รับค่าแต่ส่งคืนค่า byte readSW(void);

	s	ketch_sep05a§
- 1	1	#define LED 2
	2	#define SW1 3
	3	#define SW2 4
	4	byte readSW(void);
	5	void setup()
	6	{
	7	<pre>pinMode(LED,OUTPUT);</pre>
	8	<pre>pinMode (SW1, INPUT_PULLUP);</pre>
	9	<pre>pinMode (SW2, INPUT_PULLUP);</pre>
	10	}
	11	void loop()
	12	{
	13	<pre>byte x=readSW();</pre>
	14	if(x==1)
	15	<pre>digitalWrite(LED,HIGH);</pre>
	16	else if(x==2)
	17	<pre>digitalWrite(LED,LOW);</pre>
	18	}
	19	byte readSW(void)
	20	{
	21	byte Status=0;
	22	if(digitalRead(SW1)==LOW)
	23	Status=1;
	24	<pre>else if(digitalRead(SW2)==LOW)</pre>
	25	Status=2;
	26	return Status;
	27	}
	28	

รายละเอียดโค้ดโปรแกรม

- บรรทัดที่ 4 เป็นการประกาศรูปแบบของฟังก์ชันรอง byte readSW(void); ซึ่งเป็นฟังก์ชันที่มีการ ส่งค่ากลับคืนฟังก์ชันหลัก โดยค่าที่ส่งกลับมีขนาดเป็นค่าของตัวแปร byte

- บรรทัดที่ 13 เป็นการเป็นการเรียกใช้ฟังก์ชันรอง byte x=readSW(); เมื่อฟังก์ชันรองทา งานเสร็จ สิ้นจะส่งค่าเข้ามายังตัวแปร x ซึ่งเป็นตัวแปรได้ประกาศชนิดที่มีขนาดที่เพียงพอที่สามารถรับข้อมูลที่ ส่งกลับมาจากฟังก์ชันรองนี้ได้

- บรรทัด 19-27 เป็นฟังก์ชันรองที่เขียนขึ้นเพื่อใช้งานที่มีการส่งค่าคืนไปยังฟังก์ชันหลัก

- บรรทัดที่ 21 การประกาศตัวแปรที่มีชนิดเดียวกันกับก่าที่ฟังก์ชันรองส่งออก

- บรรทัดที่ 26 ฟึงก์ชัน return เป็นฟึงก์ชันที่ใช้สาหรับส่งค่าออกจากฟึงก์ชันรอง โดยค่าที่ส่งออกจะ อยู่ในตัวแปรท้ายฟึงก์ชัน return ซึ่งมีขนาดเดียวกันกับชนิดของค่าที่ประกาศของฟึงก์ชัน (หน้าชื่อฟึงก์ชัน รอง)

2.4 ฟังก์ชันที่รับก่าและส่งคืนก่า

เป็นพึงก์ชันที่เขียนขึ้นเพื่อแบ่งย่อยการทำงานของพึงก์ชันหลักโดยมีการส่งค่าจากพึงก์ชันหลัก เพื่อให้พึงก์ชันรองที่เขียนขึ้นนำไปประมวลค่าอย่างใดอย่างหนึ่งและส่งค่ากลับมายังพึงก์ชันหลัก ซึ่ง พึงก์ชันรองชนิดนี้มักใช้บ่อยเมื่อต้องการสร้างกลุ่มพึงก์ชันที่ด้องการคำนวณสิ่งใดสิ่งหนึ่ง โดยมีข้อมูลที่ ใช้ในการคำนวณนั้น ๆ ด้วย และเมื่อคำนวณเสร็จสิ้นพึงก์ชันหลักมีความต้องการผลการคำนวณนั้นด้วย เช่น พึงก์ชันแปลงค่าจากการอ่านค่าจากเทอร์มิสเตอร์ NTC และต้องการผลเป็นอุณหภูมิ ซึ่งข้อกำหนดใน การประกาศใช้ของพึงก์ชันประเภทนี้มีดังนี้

 1. ข้อกำหนดในการรับคำทำโดยกำหนดชนิดตัวแปร พร้อมตัวแปรที่จะรับค่าท้ายชื่อฟังก์ชัน รองที่สร้างขึ้น เช่น (int x)

 ข้อกำหนดในการส่งคืนค่า ทำโดยใส่กำหนดชนิดของข้อมูลที่จะส่งคืนหน้าชื่อฟังก์ชันรอง ตัวอย่างโปรแกรมที่มีฟังก์ชันรองที่รับค่าและส่งคืนค่า double Thermistor(int RawADC);

```
sketch_sep06a§
 1 #define NTC A5
 2 double Thermistor(int RawADC);
 3 void setup()
 4 {
 5 Serial.begin(9600);
 6 }
 7 void loop()
 8 {
 9 float Temp=Thermistor(analogRead(NTC));
10 Serial.print("Temperature is : ");
11 Serial.print(Temp);
12 Serial.println(" 'C");
13 delay(1000);
14 }
15 double Thermistor(int RawADC)
16 {
17 double Cal;
18 Cal = log(10000.0/((1024.0/RawADC-1)));
19 Cal = 1 / (0.001129148 + (0.000234125 + (0.0000000876741 * Cal
20 Cal = Cal - 273.15; // Convert Kelvin to Celcius
21 return Cal:
22 }
23
```

รายละเอียดโค้ดโปรแกรม

- บรรทัดที่ 2 เป็นการประกาศรูปแบบของฟังก์ชันรอง double Thermistor(int RawADC); ที่จะ เขียนขึ้นใช้งาน (มี ; ปิดท้าย) เป็นฟังก์ชันที่มีการรับค่าและส่งค่ากลับคืนฟังก์ชันหลัก โดยค่าที่ รับจะถูกเก็บไว้ในตัวแปรชื่อ RawADC ถูกกำหนดให้เป็นชนิด integer และส่งคืนค่ากลับมีขนาด เป็นค่าของตัวแปร double

- บรรทัดที่ 9 เป็นการเรียกใช้ฟังก์ชันรอง float Temp=Thermistor(analogRead(NTC)); โดยส่งค่า เข้าฟังก์ชันรองในวงเล็บหลังชื่อฟังก์ชัน และประกาศตัวแปรไว้รับค่า (Temp) ที่ได้จากการส่งกลับจาก ฟังก์ชันรอง จากตัวอย่างจะเห็นว่าใช้ตัวแปรที่ใช้รับค่ามีขนาดเล็กกว่าค่าที่ส่งกลับดังนั้นจะมีข้อมูลหายไป บางส่วน แต่ถ้าหากผู้เขียนโปรแกรมยอมรับการสูญหายของข้อมูลนั้นได้โดยผลการทา งานของโปรแกรมยัง เท่าเดิมก็สามารถทำได้

- บรรทัดที่ 14-21 เป็นฟังก์ชันรองที่เขียนขึ้นมาเพื่อประมวลผลตามต้องการที่มีการรับค่าเข้ามา กำนวณและส่งค่ากลับคืนยังฟังก์ชันหลัก

3. ขอบเขตของตัวแปร

ขอบเขตของตัวแปรหมายถึงบริเวณหรือตำแหน่งหรือพิกัดของตัวแปรที่พึงก์ชันสามารถเรียกใช้ งานได้ ทั้งนี้ขึ้นอยู่กับตัวแปรตัวนั้น ๆ ว่าประกาศไว้ที่จุดใด โดยขอบเขตของตัวแปรมีด้วยกัน 2 แบบ 1. ตัวแปรประเภทโกลบอล (global) เป็นตัวแปรที่ประกาศนอกพึงก์ชันซึ่งจะประกาศไว้ใน ส่วนของหัวโปรแกรม ตัวแปรเหล่านี้สามารถถูกนำไปใช้งานได้ทุกพึงก์ชัน หรืออาจกล่าว ได้ว่าทุกพึงก์ชันสามารถมองเห็นตัวแปรประเภทนี้

 ตัวแปรประเภท โลคอล (local) เป็นตัวแปรที่ประกาศภายในฟังก์ชันซึ่งสามารถใช้งานได้เฉพาะ ภายในฟังก์ชันที่ประกาศเท่านั้นนอกฟังก์ชันไม่สามารถใช้งานได้ หรืออาจกล่าวได้ว่าตัวแปร Local มองเห็นเฉพาะภายในเท่านั้น

ตัวอย่างการประกาศตัวแปรในขอบเขตที่ต่างกัน



รายละเอียดโค้ดโปรแกรม

- บรรทัดที่ 1,2 เป็นการประกาศตัวแปร LED1, LED2 ซึ่งประกาศอยู่นอกพึงก์ชันเป็นการประกาศ ตัวแปรในตำแหน่งโกลบอลนั่นหมายความว่าทุกพึงก์ชันจะสามารถใช้ (มองเห็น) ตัวแปรนี้ได้ สามารถนำไปใช้งานได้ เช่นในพึงก์ชัน setup ได้นำไปกำหนดโหมดการทำงานในพึงก์ชัน pinMode และในพึงก์ชัน loop ได้นำไปใช้ในพึงก์ชัน digitalWrite

- บรรทัคที่ 10 เป็นการประกาศตัวแปร i ซึ่งจะนำไปใช้นับรอบในคำสั่ง for โดยตัวแปรนี้จะ มองเห็นและใช้งานได้เฉพาะภายในฟังก์ชัน loop เท่านั้น นอกฟังก์ชัน (เช่นฟังก์ชัน setup ใน ตัวอย่าง) ไม่สามารถใช้งานตัวแปรนี้ได้หรือกล่าวได้ว่ามองไม่เห็นตัวแปร i นี้

4. การประกาศตัวแปรแบบสตรัคเจอร์ (Structure) และยูเนียน (Union)

การใช้งานตัวแปรที่มีจำนวนตัวแปรมาก ๆ อาจเกิดการสับสนในการใช้งานได้ ภาษาซีสามารถ จัดรวมกลุ่มตัวแปรเพื่อให้เกิดความสะดวกต่อการเรียกใช้งานได้ ซึ่งตัวแปรที่นำมารวมกลุ่มนั้นสามารถ รวมกลุ่มตัวแปรที่เป็นชนิดแตกต่างกันได้ โดยมีวิธีการประกาศด้วยกัน 2 แบบคือ

1. แบบสตัคเจอร์ (Structure)

2. แบบยูเนียน (Union)

้โดยในแต่ละแบบมีคุณสมบัติในการจัดการที่แตกต่างกัน

4.1 การประกาศตัวแปรแบบสตัคเจอร์ (Structure)

เป็นการประกาศตัวแปรที่สามารถรวมกลุ่มของข้อมูลได้หลายชนิดไม่ว่าจะเป็นตัวเลขจำนวน เต็ม ตัวเลขที่เป็นทศนิยมหรือเป็นตัวอักษรก็ตาม โดยสามารถใช้งานได้ในเวลาเดียวกันเนื่องจากตัวแปร ย่อย ๆ ภายในสตักเจอร์ได้ถูกแยกพื้นที่หน่วยความจำสำหรับการเก็บข้อมูล วิธีการประกาศมีรูปแบบดังนี้

การประกาศ	การใช้งาน
struct ชื่อตัวแปรสตักเจอร์	ชื่อตัวแปรสตักเจอร์ ชื่อตัวแปรใหม่ที่ต้องการตั้งชื่อ=
{ ชนิดตัวแปร ชื่อตัวแปรตัวที่ 1; ชนิดตัวแปร ชื่อตัวแปรตัวที่ 2; 	{ก่าเริ่มต้นของตัวแปรตัวที่ 1, 2,};
};	

ตัวอย่างเช่น

sk	etch_sep06a §
1	struct VALUE
2	{
3	byte a;
4	int b;
5	float c;
6	};
7	VALUE data={123,12345,123.45};
8	void setup()
9	{
10	<pre>Serial.begin(9600);</pre>
11	<pre>Serial.println(data.a);</pre>
12	<pre>Serial.println(data.b);</pre>
13	<pre>Serial.println(data.c);</pre>
14	}
15	void loop()
16	{
17	}
18	

รายละเอียดโค้ดโปรแกรม

- บรรทัคที่ 1-6 เป็นการประกาศตัวแปรแบบสตักเจอร์โดยตั้งชื่อว่า VALUE มีตัวแปรภายใน 3 ตัว ที่เก็บข้อมูลแตกต่างกัน

- บรรทัคที่ 7 เป็นการกำหนดค่าเริ่มต้นโดยสามารถเปลี่ยนชื่อเรียกใหม่ได้ ในที่นี้เปลี่ยนชื่อจาก VALUE เป็นชื่อว่า data และสามารถกำหนดค่าเริ่มต้นเข้าไปในทุกตัวแปรภายในได้ในขั้นตอนนี้ จาก ตัวอย่างกำหนดตัวแปร data.a=123 ตัวแปร data.b=12345 และตัวแปร data.c=123.45 4.2 การประกาศตัวแปรแบบยูเนียน (Union)

เป็นการประกาศตัวแปรที่สามารถรวมกลุ่มของข้อมูลได้หลายชนิดเช่นเดียวกับแบบสตักเจอร์ แต่มีความแตกต่างตรงที่แบบยูเนียนใช้หน่วยความจำเก็บข้อมูลในตำแหน่งเดียวกันในทุกตัวแปร ดังนั้น เวลาใช้งานจึงไม่สามารถใช้พร้อมกันได้ เนื่องจากตัวแปรที่กำหนดค่าตัวหลังสุดจะเป็นตัวที่ได้ใช้งาน หน่วยความจำในตำแหน่งนั้น ๆ ทำให้ข้อมูลในตัวแปรแรก ๆ ถูกแทนที่ไปด้วยค่าในตัวแปรล่าสุดแทน รูปแบบการประกาศดังนี้

การประกาศ	การใช้งาน
union ชื่อตัวแปรยูเนียน	ชื่อตัวแปรยูเนียน ชื่อตัวแปรใหม่ที่ต้องการตั้งชื่อ;
{	
ชนิคตัวแปร ชื่อตัวแปรตัวที่ 1;	
ชนิดตัวแปร ชื่อตัวแปรตัวที่ 2;	
};	

ตัวอย่างเช่น

sk	etch_sep06a§
1	union VALUE
2	{
3	byte a;
4	int b;
5	float c;
6	};
7	VALUE data;
8	void setup()
9	{
10	<pre>Serial.begin(9600);</pre>
11	<pre>data.a=123; Serial.println(data.a);</pre>
12	<pre>data.b=12345; Serial.println(data.b);</pre>
13	<pre>data.c=123.45; Serial.println(data.c);</pre>
14	}
15	void loop()
16	{
17	}
18	

รายละเอียดโค้ดโปรแกรม

- บรรทัดที่ 1-6 เป็นการประกาศตัวแปรแบบยูเนียนโดยตั้งชื่อว่า VALUE มีตัวแปรภายใน 3 ตัวที่ เก็บข้อมูลแตกต่างกัน

- บรรทัคที่ 7 เป็นการกำหนคชื่อเรียกใหม่ในที่นี้เปลี่ยนชื่อจาก VALUE เป็นชื่อว่า data แต่ไม่ สามารถกำหนคค่าเริ่มต้นแบบตัวแปรสตรัคเจอร์ได้

- บรรทัดที่ 11 เป็นการกำหนดค่าให้กับตัวแปรตัวแรก data.a พร้อมนำไปใช้งานทันที โดยตัวอย่าง ดังกล่าวเป็นการนำส่งข้อมูลออกทางพอร์ตอนุกรมเพื่อแสดงผลหน้าจอกอมพิวเตอร์

- บรรทัคที่ 12 เป็นการกำหนดค่าให้กับตัวแปรตัวแรก data.b ค่าของตัวแปรนี้จะ ไปทับข้อมูลของ data.a ซึ่งตอนนี้ไม่ได้ใช้งานแล้ว

- บรรทัดที่ 13 เป็นการกำหนดค่าให้กับตัวแปรตัวแรก data.c ค่าตัวแปรนี้จะไปทับข้อมูลของตัว แปร data.b เนื่องจากใช้ตำแหน่งของหน่วยความจำเดียวกันในการเก็บข้อมูล

หากกำหนดค่าตัวแปรต่อเนื่องกัน ตัวแปรตัวสุดท้ายจะไปทับข้อมูลตัวแปรตัวแรก ๆ ของตัว แปรยูเนียน

ตัวอย่างเช่น

sk	etch_sep06a§
1	union VALUE
2	{
3	byte a;
4	int b;
5	float c;
6	};
7	VALUE data;
8	void setup()
9	{
10	data.a=123;
11	data.b=12345;
12	data.c=123.45;
13	<pre>Serial.begin(9600);</pre>
14	<pre>Serial.println(data.a);</pre>
15	<pre>Serial.println(data.b);</pre>
16	<pre>Serial.println(data.c);</pre>
17	}
18	void loop()
19	{
20	}
21	1

หน่วยการเรียนรู้ที่ 5 การลงมือเขียนโปรแกรม เรื่องย่อยที่ 4 การติดตั้ง library บน Arduino IDE

Libraries คือ โก้คที่ทำให้ sensor และโมคูลต่างๆ ใช้งานได้ง่ายขึ้นสะควกต่อการใช้งานและไม่มี ความซับซ้อนมากนัก ซึ่งการใช้งาน Libraries บางตัวนั้นนักเขียนโปรแกรมใช้งานจะต้องเพิ่มเข้าไปด้วย ตนเอง

้วิธีติดตั้งไลบรารี (Library) ให้กับโปรแกรม Arduino IDE ทำได้ 3 วิธีดังนี้

- 1. ติดตั้งผ่าน Arduino IDE โดยใช้ Library Manager
- 2. ติดตั้งผ่าน Arduino IDE โดยนำเข้าไฟล์ .zip
- 3. ติดตั้งโดยนำไลบรารีไปวางไว้ในโฟลเดอร์ libraries

<mark>หมายเหตุ</mark> : วิธีติดตั้งแบบที่ 2 และแบบที่ 3 ต้องดาวน์โหลดไฟล์ไลบรารีมาไว้ในเครื่องก่อน

1. โดยติดตั้งผ่าน Arduino IDE โดยใช้ Library Manager

<mark>ข้อจำกัด</mark> : ไลบรารีที่จะติดตั้งด้วยวิธีนี้ จะต้องมีข้อมูลอยู่ในระบบ library index ของ arduino เท่านั้น ซึ่งหาก ไม่มีข้อมูลอยู่ในระบบจะทำให้ค้นหาไลบรารีเพื่อติดตั้งไม่เจอ

โดยมีวิธีการติดตั้งมีขั้นตอนดังต่อไปนี้

กลิกที่เมนู Sketch > Include Library > Manage Libraries... เพื่อเปิด Library Manager



รอให้ Library Manager คาวน์โหลด library index จนเสร็จ จึงจะเริ่มใช้งาน Library Manager ได้ (ช้าเร็วขึ้นอยู่กับความเร็วอินเทอร์เน็ต)



พิมพ์ชื่อ ไลบรารีที่ต้องการจะติดตั้ง เช่น พิมพ์กำว่า taskscheduler ที่ช่อง Filter your search... เมื่อ พบ ไลบรารีที่ต้องการจะติดตั้ง ให้กลิกปุ่ม Install จากนั้นรอจนกว่า ไลบรารีนั้นจะถูกคาวน์ โหลด และติดตั้ง กระทั่งสำเร็จ

 Library Manager 	8
Type All 🔹 Topic All 💌	taskscheduler
TaskScheduler by Anatoli Arkhipenko A light weight cooperative multitacking library for arduin execution period in milliseconds or microseconds - frequency of tasks in predefined sequence, dynamic change of task executio entering IDLE sleep mode when tasks are not scheduled to run, error handling and watchdog timer, Local Task Storage pointer functions (esp8266, esp32 only), overall task timeout. <u>More info</u>	no and esp8266 microcontrollers. Supports: periodic task execution (with dynamic f execution), number of iterations (limited or infinite number of iterations), execution of on parameters (frequency, number of iterations, callback methods), power saving via , event-driven task invocation via Status Request object, task IDs and Control Points for (allowing use of same callback code for multiple tasks), layered task prioritization, std::
	Version 2.6.1 💌 Install

ถ้าติดตั้งสำเร็จแล้วจะแสดงคำว่า INSTALLED ต่อท้ายชื่อไลบรารี (การติดตั้งจะช้าเร็วขึ้นอยู่กับ ความเร็วอินเทอร์เน็ต)

 Library Manager 		8
Type All 🔻 Topic All	▼ taskscheduler	
TaskScheduler by Anatoli Arkhipenko Version 2.6. A light-weight cooperative multitask Ultravio execution period in milliseconds or microseconds - frect tasks in predefined sequence, dynamic change of task entering IDLE sleep mode when tasks are not schedule error handling and watchdog timer, Local Task Storage functions (esp8266, esp32 only), overall task timeout. <u>More info</u>	1 INSTALLED esp8266 microcontrollers. Supports: periodic task execution (with dynamic guency of execution), number of iterations (limited or infinite number of iterations), execution of execution parameters (frequency, number of iterations, callback methods), power saving via d to run, event-driven task invocation via Status Request object, task IDs and Control Points for pointer (allowing use of same callback code for multiple tasks), layered task prioritization, std::	

2. ติดตั้งผ่าน Arduino IDE โดยนำเข้าไฟล์ .zip

<mark>ข้อจำกัด</mark> : ไลบรารีที่จะติดตั้งด้วยวิธีนี้จะต้องมีข้อมูลอยู่ในคอมพิวเตอร์ของนกเขียนโปรแกรมขณะนั้น เท่านั้น ซึ่งหากไม่มีข้อมูลอยู่ในคอมพิวเตอร์จะทำให้ค้นหาไฟล์ .zip เพื่อติดตั้งไม่พบ

กลิกที่เมนู Sketch > Include Library > Add .ZIP Library... เพื่อเปิดโฟล์เดอร์ที่จัดเก็บไลบารี่



เมื่อติดตั้งสำเร็จจะมีข้อความขึ้นมาว่า Library added to your libraries. Check "Include library" menu ตามรูปด้านล่าง



3. ติดตั้งโดยนำไลบรารีไปวางไว้ในโฟลเดอร์ libraries มีขั้นตอนดังนี้

- 1. ดาวน์โหลดไลบรารีที่เป็นไฟล์ .zip มาไว้ในเครื่องคอมพิวเตอร์
- 2. แตกไฟล์ .zip จนได้ออกมาเป็นโฟลเดอร์
- 3. ย้ายโฟลเคอร์ที่เพิ่งแตกไฟล์ออกมา ไปไว้ในโฟลเคอร์ libraries
- ที่อยู่ของโฟลเดอร์ libraries สำหรับ Linux : /home/your_user/Arduino/libraries

โดยสามารถตรวจเช็กตำแหน่งของโฟลเดอร์ที่จะนำไปวางได้โดยกลิกที่เมนู File > Preferences



เมื่อหน้าต่าง Preferences แสดงขึ้นมา ให้ดูที่ Sketchbook location นั่นคือตำแหน่งของโฟลเดอร์ที่ เราจะนำไลบรารีไปวาง

Preferences	NAMES AND ADDRESS AND ADDRESS AND ADDRESS ADDRE			
Settings Network				
Sketchbook location:				
C:\Users\COM_PC\Documents\Arduino				
Editor language:	System Default v (requires restart of Arduino)			
Editor font size:	13			
Interface scale:	✓ Automatic 100 → % (requires restart of Arduino)			
Theme:	Default theme 👻 (requires restart of Arduino)			
Show verbose output during: [compilation 🔲 upload			
Compiler warnings:	None 👻			
🔽 Display line numbers				
Enable Code Folding				
Verify code after upload				
Use external editor				
📝 Aggressively cache compile	ed core			
📝 Check for updates on start	up			
Update sketch files to new	extension on save (.pde -> .ino)			
Save when verifying or upl	oading			
Additional Boards Manager URL	Additional Boards Manager URLs: http://arduino.esp8266.com/stable/package_esp8266.com_index.json			
More preferences can be edited	d directly in the file			
C:\Users\COM_PC\AppData\Lo	cal\Arduino 15\preferences. txt			
(edit only when Arduino is not r	unning)			
	OK Cancel			

หน่วยการเรียนรู้ที่ ๖ Arduino Web Editor เขียนโปรแกรมผ่านบราวเซอร์

กลุ่มสาระการเรียนรู้วิทยาศาสตร์ สาระที่ ๔ เทคโนโลยี

มาตรฐาน ว ๔.๒ เข้าใจและใช้แนวกิคเชิงกำนวณในการแก้ปัญหาที่พบในชีวิตจริงอย่างเป็นขั้นตอนและ เป็นระบบ ใช้เทคโนโลยีสารสนเทศและการสื่อสารในการเรียนรู้ การทำงาน และการแก้ปัญหาได้อย่างมี ประสิทธิภาพ รู้เท่าทัน และมีจริยธรรม

ตัวชี้วัด

ງ ໔.ໂຫ ນ.໑/ໂຫ	ออกแบบและเขียนโปรแกรมอย่างง่ายเพื่อแก้ปัญหาทางคณิตศาสตร์หรือวิทยาศาสตร์
ງ ໔.២ ໗.໕/໑	รวบรวม วิเคราะห์ข้อมูล และใช้ความรู้ด้านวิทยาการคอมพิวเตอร์ สื่อดิจิทัล
	เทคโนโลยีสารสนเทศในการแก้ปัญหาหรือเพิ่มมูลค่าให้กับบริการหรือผลิตภัณฑ์ที่ใช้
	ในชีวิตจริงอย่างสร้างสรรค์
ງ ໔.ໄຫ ມ.ຽ/໑	ใช้เทคโนโลยีสารสนเทศในการนำเสนอ และแบ่งปั้นข้อมูลอย่างปลอดภัย มีจริยธรรม
	และวิเคราะห์การเปลี่ยนแปลงเทคโนโลยีสารสนเทศที่มีผลต่อการคำเนินชีวิต อาชีพ
	สังคม และวัฒนธรรม

สาระสำคัญ

ผู้เรียนได้รู้จักและลองเขียนคำสั่งใน Arduino Web Editor ซึ่งเป็นโปรแกรมจำลองการทำงาน เพื่อ นำไฟล์ภาษาเครื่องที่ได้จากการแปลงไปจำลองการทำงานด้วยเว็ปไซต์

สาระการเรียนรู้

<u>ความรู้</u>

นักเรียนได้ทดลองวงจรโดยที่ยังไม่มีวงจรจริง โดยโปรแกรมจะทำการจำลองการทำงานของวงจรที่ ถูกต้องก่อนที่จะต่อวงจรจริงในห้องทดลอง <u>ทักษะ/กระบวนการ</u>

- ผู้เรียนใช้ทักษะการสังเกตทั้งกำสั่ง และการต่อจำลองวงจร
- ๒. ผู้เรียนใช้ทักษะการสังเกตการต่อวงจร เช่น หลอด LED ต้องต่อตัวด้านทาน เป็นต้น
- ๓. ผู้เรียนใช้กระบวนการแก้ปัญหาเมื่อพบว่าการต่อวงจรหรือเขียนกำสั่งเมื่อมีข้อผิดพลาด

<u>คุณลักษณะที่พึงประสงค์</u>

- ผู้เรียนมีใจรักและพร้อมศึกษาในการต่อวงจรแบบจำลอง
- ๒. ผู้เรียนมีใจรักในการเขียนคำสั่ง
- ๓. นักเรียนมีน้ำใจต่อผู้ที่ต้องการให้แนะนำกำสั่งหรือวงจร
- ๔. รักชาติ ศาสน์ กษัตริย์
- ๕. ซื่อสัตย์
- ນີ້ວີນັ້ຍ
- ๗. ใฝ่เรียนรู้
- ๘. อยู่อย่างพอเพียง
- มุ่งมั่นในการทำงาน

หน่วยการเรียนรู้ที่ 6 Arduino Web Editor เขียนโปรแกรมผ่านบราวเซอร์

การใช้งาน Arduino Web Editor โปรแกรมจำลองการทำงาน<u>เ</u>พื่อนำไฟล์ภาษาเครื่องที่ได้จากการ แปลงไปจำลองการทำงานด้วยเว็ปไซต์ <u>https://www.tinkercad.com</u>

สำหรับนักเขียนโปรแกรมที่ต้องการทดลองวงจรโดยที่ยังไม่มีวงจรจริง การจำลองด้วยโปรแกรม จำลองการทำงานของวงจรเป็นทางเลือกที่เหมาะสม ซึ่งในหลักสูตรนี้จะขอแนะนำเว็ปไซต์ https://www.tinkercad.com/ ซึ่งเป็นอีกหนึ่งเว็ปไซต์ที่ได้รับความนิยมในการจำลองเขียนโปรแกรม Arduino IED เบื้องต้น เมื่อเข้าสู่หน้าเว็ปไซต์จะประกอบด้วยขั้นตอนการดำเนินการดังนี้



1. คลิกเลือก Program, simulate, & assemble

2. คลิกเลือก Start Tinkering



3. คลิกเลือก Create personal account

Start Tinkering How will you use Tinkercad?
In School
Educators, start here
Students, join your class
Personal
Create personal account
+
Already have an account? Sign In

4. เลือกประเทศ วัน เคือน ปี เกิด ---> คลิก NEXT จากนั้นจะปรากฏหน้าต่างให้กรอกอีเมล์และ พาสเวิร์ด

Create account	4	
Country, Territory, or Region		
Thailand	•	
Birthday April • 16 • 1989	•	
NEXT		
ALREADY HAVE AN ACCOUNT? SIGN IN		

5. จากนั้นจะปรากฎหน้าต่างสร้างบัญชี (Account created) คลิกเครื่องหมายถูกหน้าข้อความอธิบาย แล้วคลิก DONE



6. จากนั้นจะเข้าสู่หน้าต่างการออกแบบคลิก Circuits --> Create new Circuit




ตัวอย่างการต่อวงจรและการเขียนคำสั่ง

อุปกรณ์ที่ใช้ในการทคลองนี้

- 1. Utrasonic Distance sensor
- 2. หลอด LED จำนวน 2 หลอด
- 3. NP Transister
- 4. Temperature Sensor



คำสั่งควบคุม

```
1 const byte trigger_Pin = A0; //Initialize I/O pins
2 const byte echo_Pin = A1;
4 unsigned long duration; //Since PulseIn return an unsigned Long
unsigned int distance; //To save the distanc
6 int led_1 = A2;
7 int water;
8 int ad;
9 int beep = A3;
10 int tmp = A4;
11 int tmp_1;
12 int led_2 = A5;
13 int c = 0;
14
15 void setup() {
16 Serial.begin(9600); //Initialize Serial communication
17 pinMode(cho_Pin, INPUT); //Echo pin as Input
18 pinMode(trigger_Pin, OUTPUT); //Trigger pin as Output
19 pinMode(led_1,OUTPUT);
20 pinMode(led_2,OUTPUT);
21 digitalWrite(beep,0);
22 pinMode(led_2,OUTPUT);
23 }
24
25 4
25 4
26 4
27 Serial Monitor
20 Distance Serial Communication Serial Communication Serial Monitor
20 Distance Serial Communication Serial Communicat
```

ผลลัพธ์แสดงที่จอ Serial Monitor

Serial Monitor
l Water = 82 ml pump on l Water = 82 ml pump on l Water = 82 ml
Send Clear

หน่วยการเรียนรู้ที่ ๗ จอภาพตอบสนองผู้ชม

กลุ่มสาระการเรียนรู้วิทยาศาสตร์ สาระที่ ๔ เทคโนโลยี

มาตรฐาน ว ๔.๑ เข้าใจแนวคิดหลักของเทกโนโลยีเพื่อการดำรงชีวิตในสังกมที่มีการเปลี่ยนแปลงอย่าง รวดเร็ว ใช้กวามรู้และทักษะทางด้านวิทยาศาสตร์ กณิตศาสตร์ และศาสตร์อื่น ๆ เพื่อแก้ปัญหาหรือพัฒนา งานอย่างมีกวามกิดสร้างสรรค์ด้วยกระบวนการออกแบบเชิงวิศวกรรมเลือกใช้เทคโนโลยีอย่างเหมาะสม โดยคำนึงถึงผลกระทบต่อชีวิต สังกม และสิ่งแวดล้อม

ตัวชี้วัด

೨ ៤.೦ ೫.೦/៤	ออกแบบวิธีการแก้ปัญหา โดยวิเคราะห์ เปรียบเทียบ และตัดสินใจเลือกข้อมูลที่จำเป็น
	นำเสนอแนวทางการแก้ปัญหาให้ผู้อื่นเข้าใจ วางแผนและดำเนินการแก้ปัญหา
ി ര്.െ പ്.െ⁄ര്	ใช้ความรู้และทักษะเกี่ยวกับวัสคุ อุปกรณ์ เครื่องมือ กลไก ไฟฟ้า หรืออิเล็กทรอนิกส์
	เพื่อแก้ปัญหาได้อย่างถูกต้อง เหมาะสมและปลอดภัย
ට ໔.ඉ ໗.๒/໔	ทดสอบ ประเมินผล และอธิบายปัญหาหรือข้อบกพร่องที่เกิดขึ้น ภายใต้กรอบเงื่อนไข
	พร้อมทั้งหาแนวทางการปรับปรุงแก้ไข และนำเสนอผลการแก้ปัญหา
ി ໔.െ ຏ.๒/๕	ใช้ความรู้ และทักษะเกี่ยวกับวัสคุ อุปกรณ์ เครื่องมือ กลไก ไฟฟ้า และอิเล็กทรอนิกส์
	เพื่อแก้ปัญหาหรือพัฒนางานได้อย่างถูกต้อง เหมาะสม และปลอดภัย
ി ഭ്.െ ഃ.ത/ഭ്	ทคสอบ ประเมินผล วิเคราะห์ และให้เหตุผลของปัญหาหรือข้อบกพร่องที่เกิดขึ้น
	ภายใต้ กรอบเงื่อนใข พร้อมทั้งหาแนวทางการปรับปรุงแก้ใข และนำเสนอผลการ
	แก้ปัญหา
ി ໔.െ ຏ.๓/๕	ใช้ความรู้ และทักษะเกี่ยวกับวัสคุ อุปกรณ์ เครื่องมือ กลไก ไฟฟ้าและอิเล็กทรอนิกส์
	ให้ ถูกต้องกับลักษณะของงาน และปลอคภัย เพื่อแก้ปัญหาหรือพัฒนางาน
ી ૡ.૭ મ.ૡ/ૡ	ใช้ความรู้และทักษะเกี่ยวกับวัสคุ อุปกรณ์ เครื่องมือ กลไก ไฟฟ้าและอิเล็กทรอนิกส์
	และเทคโนโลยีที่ซับซ้อนในการแก้ปัญหาหรือพัฒนางาน ได้อย่างถูกต้อง เหมาะสม
	และปลอคภัย

สาระสำคัญ

- ๑. ผู้เรียนได้เรียนรู้เครื่องมือและส่วนประกอบ
- ๒. ผู้เรียนได้เรียนรู้ตัวต้านทาน
- ด. ผู้เรียนได้เรียนรู้ไฟ LED
- ๔. ผู้เรียนได้เรียนรู้การเลื่อนภาพจากจอ LCD 12*2
- ๕. ผู้เรียนได้เรียนรู้การเชื่อมต่อบอร์ค Arduino กับ Processing

สาระการเรียนรู้

<u>ความรู้</u>

ผู้เรียนจะ ได้ศึกษาอุปกรณ์อิเล็กทรอนิกส์ การต่อวงจร ตลอดจนการเขียนกำสั่งควบคุมบอร์ด Arduino Uno R3

<u>ทักษะ/กระบวนการ</u>

 ๑. ใช้กระบวนการสังเกตเพื่อให้ผู้เรียนรับรู้ข้อมูล และศึกษาด้วยวิธีการต่างๆ อันซึ่งนำมาด้วยความรู้ ความเข้าใจ

๒. ใช้กระบวนการกลุ่ม เพื่ออธิบาย รับฟัง และวิจารณ์กิจกรรมที่พิจารณาส่วนประกอบของการ กระทำหรือข้อมูลต่างๆที่เชื่อม โยงเกี่ยวข้องกัน

<u>คุณลักษณะที่พึงประสงค์</u>

- ๑. ผู้เรียนมีใจรักในการเขียนคำสั่งควบคุมบอร์ด Arduino Uno R3
- ๒. ผู้เรียนมีความเข้าใจในอุปกรณ์อิเล็กทรอนิกส์
- ๓. ผู้เรียนรักและมีส่วนร่วมในกระบวนการกลุ่ม

รักชาติ ศาสน์ กษัตริย์

- ๔. ชื่อสัตย์
- ะ. มีวินัย
- ๖. ใฝ่เรียนรู้
- ๗. อยู่อย่างพอเพียง
- ่ ๘. มุ่งมั่นในการทำงาน

หน่วยการเรียนรู้ที่ 7 จอภาพตอบสนองผู้ชม เรื่องย่อยที่ 1 เครื่องมือและส่วนประกอบ

หน่วยการเรียนรู้นี้จะแนะนำเครื่องมือ หรืออุปกรณ์และส่วนประกอบทั้งหมดที่จะใช้ในโครงงานของ หลักสูตรนี้ ซึ่งจะแยกเป็นหมวดหมู่ ดังนี้

<u>หมวดหมู่บอร์ด</u>

1. บอร์ด Arduino



2. Sensor Shield



<u>หมวดหมู่ Motor และการขับเคลื่อน</u>

1. Motor 180 องศา



2. Motor 360 องศา



3. เกียร์ขับมอเตอร์



4. ล้อ



5. ชุดขับมอเตอร์ L298N Dual H-Bridge Motor Controller



6. เฟื่อง



<u>หมวดหมู่ Sensor และ Infrared</u>

1. Ultra Sonic Sensor (เซ็นเซอร์วัดระยะทางด้วยคลื่นเสียง)



2. รี โมทอินฟาเรด Infrared Remote Control



3. Humidity Sensors (เซนเซอร์วัดความชื้นอากาศ)



4. Modules tracking sensor (เซ็นเซอร์ Sensor จับสี ขาวคำ, Sensor หุ่นยนต์, หุ่นยนต์เดินตามเส้น)



5. LDR Photoresistor Sensor Module (เซ็นเซอร์วัดความเข้มแสง)



<u>หมวดหมู่แสดงผล</u>

1. LCD Display Module 16*2 (จอแอลซีดี)



2. Buzzer Module (ถำโพง)



<u>หมวดหมู่อื่นๆ</u>

1. บอร์คทดลอง Bread Board 830 Point



2. หลอคไฟ LED



3. ตัวต้านทานซึ่งมีค่าความต้านทานระหว่าง 220 ถึง 330 โอห์ม



4. สายจัมเปอร์ ผู้-ผู้ , ผู้-เมีย , เมีย-เมีย



5. Relay 5V 2 ช่อง



6. Keypad



7. มือหนีบ



7. สาย USB 2.0



8. สาย USB cable



หน่วยการเรียนรู้ที่ 7 จอภาพตอบสนองผู้ชม เรื่องย่อยที่ 2 ตัวต้านทาน

ตัวต้านทาน (resistor) เป็นอุปกรณ์ไฟฟ้าชนิดหนึ่งที่มีคุณสมบัติในการต้านการไหลผ่านของ กระแสไฟฟ้า ทำด้วยลวดต้านทานหรือถ่านการ์บอน เป็นต้น ถ้าอุปกรณ์นั้นๆ มีความต้านทานมาก กระแสไฟฟ้าที่ไหลผ่านจะน้อยลง และเป็นอุปกรณ์ไฟฟ้าชนิดพาสซีฟสองขั้วที่สร้างความต่างศักย์ไฟฟ้า กร่อมขั้วทั้งสอง (V) โดยมีสัดส่วนมากน้อยตามปริมาณกระแสไฟฟ้าที่ไหลผ่าน (I) อัตราส่วนระหว่างความ ต่างศักย์ และปริมาณกระแสไฟฟ้า คือ ก่าความต้านทานทางไฟฟ้า หรือก่าความต้านทานของตัวนำมีหน่วย เป็นโอห์ม (สัญลักษณ์ : Ω) ซึ่งเขียนเป็นสมการตามกฎของโอห์ม ดังนี้

$$R = \frac{V}{I}$$

ค่าความต้านทานนี้ถูกกำหนดว่าเป็นค่าคงที่สำหรับตัวด้านทานธรรมคาทั่วไปที่ทำงานภายในค่า กำลังงานที่กำหนดของตัวมันเอง

ตัวต้านทานทำหน้าที่ลดการ ใหลของกระแสและในเวลาเดียวกันก็ทำหน้าที่ลดระดับแรงดันไฟฟ้า ภายในวงจรทั่วไป Resistors อาจเป็นแบบค่าความต้านทานคงที่ หรือค่าความต้านทานแปรได้ เช่น ที่พบใน ตัวต้านทานแปรตามอุณหภูมิ (thermistor), ตัวต้านทานแปรตามแรงดัน (varistor), ตัวหรี่ไฟ (trimmer), ตัว ต้านทานแปรตามแสง(photoresistor) และตัวต้านทานปรับด้วยมือ (potentiometer)



รูปความหมายของรหัสสีตัวต้ำนทาน

ทฤษฎีการทำงาน

กฎของโอห์ม

พฤติกรรมของตัวต้านทานในอุคมคติจะถูกกำหนดโดยความสัมพันธ์ที่ระบุไว้ในกฎของโอห์มคังนี้:

$$V = I \cdot R$$
.

กฎของโอห์ม ระบุว่า แรงคันไฟฟ้า(V) ที่ตกคร่อมความต้านทานจะเป็นสัคส่วนกับกระแส(I) เมื่อ ก่าคงที่ของสัคส่วนเป็นความต้านทาน(R) ในส่วนที่เท่าเทียมกัน กฎของโอห์มสามารถระบุได้ว่า:

$$I = \frac{V}{R}.$$

สมการนี้กำหนดว่ากระแส(I)เป็นสัดส่วนกับแรงคันไฟฟ้า(V)และแปรผกผันกับความต้านทาน(R) สมการนี้จะถูกนำมาใช้โดยตรงในการคำนวณในทางปฏิบัติ ตัวอย่างเช่นถ้าตัวต้านทาน 300 โอห์ม ต่อคร่อม ระหว่างขั้วของแบตเตอรี่ 12 โวลต์ ดังนั้นกระแส 12/300 = 0.04 แอมแปร์ (หรือ 40 milliamperes) จะใหล ผ่านตัวต้านทานตัวนั้น

<u>ตัวด้านทานต่ออนุกรมและต่อขนาน</u>

ในการต่อแบบอนุกรม กระแสที่ไหลผ่านตัวด้านทานทุกตัวมีจำนวนเท่ากัน แต่แรงคันไฟฟ้าในแต่ ละตัวด้านทานจะเป็นสัดส่วนกับความด้านทานของมัน ความต่างศักย์(แรงคัน)ที่เห็นตกคร่อมในเครือข่าย ทั้งหมดคือผลรวมของแรงคันไฟฟ้าเหล่านั้น ความด้านทานรวมสามารถหาได้จากผลรวมของความด้านทาน ของแต่ละตัว



 $R_{\rm eq} = R_1 + R_2 + \dots + R_n.$

ในกรณีที่ตัวด้านทานของจำนวน N ตัวมีความด้านทานเท่ากันเท่ากับ R ต่อกันแบบอนุกรม ความ ด้านทานรวมจะเท่ากับ NxR ดังนั้นหากตัวด้านทานหนึ่งตัวขนาด 100K โอห์ม ต่ออนุกรมกับตัวด้านทาน ขนาด 22K โอห์มหนึ่งตัว ความด้านทานรวมจะเท่ากับ 122K โอห์ม ทั้งสองตัวนี้จะทำงานในวงจรราวกับว่า พวกมันเป็นตัวด้านทานตัวเดียวที่มีค่าความด้านทาน 122K โอห์ม; สาม ตัวต้านทานขนาด 22K โอห์ม (จำนวน = 3, R = 22K) จะสร้างความต้านทานเท่ากับ 3x22K = 66K โอห์ม ตัวด้านทานที่ต่อแบบขนานกัน ความต่างศักย์(แรงดัน)ของแต่ละตัวจะมีค่าเท่ากัน แต่กระแสทั้งหมด จะเท่ากับกระแสที่ไหลผ่านตัวด้านทานแต่ละตัวนำมารวมกัน ค่า conductances ของตัวด้านทานจะถูกนำมา รวมกันเพื่อพิจารณาก่า conductances ของเกรือข่าย ดังนั้นก่าความด้านทานเทียบเท่า (Req) ของเกรือข่ายที่ สามารถกำนวณได้

<u>กระจายพลังงานความร้อน</u>

ณ เวลาใดเวลาหนึ่ง กำลังงาน P ที่ถูกบริโภคโดยตัวต้านทานที่มีค่าความต้านทาน R (โอห์ม)จะถูก กำนวณเป็น :

$$P = I^2 R = IV = \frac{V^2}{R}$$

เมื่อ V (โวลต์) เป็นแรงคันไฟฟ้าตกคร่อมตัวด้านทานและ I (แอมป์) เป็นกระแสไฟฟ้าที่ไหล ผ่าน มัน รูปแบบแรกคือคำสอนที่พูดขึ้นใหม่ของกฎข้อที่หนึ่งของจูล โดยการใช้กฎของโอห์ม รูปแบบอื่นอีกสอง อย่างสามารถถูกนำมาต่อยอดได้ กำลังนี้จะถูกแปลงเป็นความร้อนซึ่งจะต้องกระจายไปในร่างกายของตัว ด้านทาน

พลังงานความร้อนทั้งหมดที่ถูกปล่อยออกมาในช่วงเวลาหนึ่งสามารถกำหนดได้จากผลรวมของกำลัง งานในช่วงเวลานั้น:

$$W = \int_{t_1}^{t_2} v(t)i(t)\,dt.$$

้ดังนั้น เราสามารถเขียนกำลังงานเฉลี่ยที่กระจายไปในช่วงเวลานั้นโดยเฉพาะได้ว่า

$$ar{P} = rac{1}{t_2 - t_1} \int_{t_1}^{t_2} v(t) i(t) \, dt.$$

ถ้าเวลาช่วง $t_1 - t_2$ ถูกเลือกให้เป็นหนึ่งรอบที่สมบูรณ์ของสัญญาณเป็นระยะๆ (หรือเลขจำนวนเต็ม ของรอบ) ดังนั้น ผลนี้จะมีค่าเท่ากับกำลังงานเฉลี่ยระยะยาวที่สร้างขึ้นเป็นความร้อน ซึ่งจะมีการกระจายไป อย่างต่อเนื่อง. ด้วยสัญญาณเป็นระยะ ๆ (เช่น/ แต่ไม่จำกัดแค่เพียงคลื่นไซน์) ดังนั้นค่าเฉลี่ยในช่วงรอบที่ สมบูรณ์ (หรือในระยะยาว) นี้จะหาได้จาก $ar{P} = I_{rms}V_{rms} = I_{rms}^2R = rac{V_{rms}^2}{R}$ เมื่อ I_{rms} และ V_{rms}

เป็นค่า root mean square ของกระแสและแรงคัน ในกรณีใดๆ ความร้อนที่ถูกสร้างขึ้นในตัวด้านทานที่ จะต้องถูกกระจายไปก่อนที่อุณหภูมิของมันเพิ่มขึ้นมากเกินไป

้ตัวต้านทานจะมีอัตราการใช้งานตามการกระจายกำลังงานสูงสุด ตัวต้านทานเดี่ยวๆส่วนใหญ่ใน

ระบบอิเล็กทรอนิกส์โซลิคสเตทสามารถดูดซับกำลังไฟฟ้าได้น้อยกว่าหนึ่งวัตต์มากและไม่ต้องให้ความ สนใจกับระดับกำลังงานเป็นพิเศษ ตัวต้านทานดังกล่าวรวมทั้งส่วนใหญ่ของแพกเกจตามรายละเอียด ด้านล่างนี้ มักจะมีการจัดอัตราเป็น 1/10, 1/8 หรือ 1/4 วัตต์

ตัวด้านทานที่ต้องกระจายกำลังงานจำนวนมาก เช่น ที่ใช้โดยเฉพาะอย่างยิ่งในแหล่งจ่ายไฟ วงจร การแปลงกำลังงาน และตัวขยายสัญญาณกำลัง โดยทั่วไปจะเรียกว่า ตัวด้านทานกำลัง การกำหนดนี้ถูก นำมาใช้อย่างหลวมๆ สำหรับตัวด้านทานที่มีอัตรากำลังที่ 1 วัตต์หรือมากกว่า ตัวด้านทานกำลังมีขนาดใหญ่ กว่าและอาจจะไม่ได้ใช้ก่า, รหัสสี, และแพกเกจภายนอกที่อธิบายไว้ด้านล่าง

หากกำลังงานเฉลี่ยที่กระจายไปโดยตัวต้านทานมีมากกว่าอัตรากำลังงานของมัน ความเสียหายบน ตัวต้านทานอาจเกิดขึ้นหรืออาจเปลี่ยนแปลงความต้านทานอย่างถาวร ซึ่งแตกต่างจากการเปลี่ยนแปลง ย้อนกลับของความต้านทานเนื่องจาก ค่าสัมประสิทธิ์อุณหภูมิของมันเมื่อมันอุ่น กระจายกำลังงานมาก เกินไปอาจเพิ่มอุณหภูมิของตัวต้านทานจนถึงจุดที่มันสามารถเผาไหม้แผงวงจรหรือชิ้นส่วนที่อยู่ใกล้เกียง หรือแม้กระทั่งทำให้เกิดไฟไหม้ มีตัวต้านทานที่ออกแบบมาไม่ติดไฟ ซึ่งจะหยุดทำงาน (เปิดวงจร) ก่อนที่ จะมันจะร้อนมากเกินไปจนเป็นอันตราย

แต่ถ้าการไหลเวียนของอากาศไม่ดี หรืออยู่สูงจากพื้นดิน หรือทำงานในอุณหภูมิที่สูง ตัวต้านทาน อาจต้องใช้อัตราการกระจายกำลังงานที่สูงกว่าปกติ

นอกจากนี้ ตัวต้านทานบางชนิดและบางอัตรายังอาจมีอัตราแรงคันไฟฟ้าสูงสุด ซึ่งอาจจำกัด การ กระจายกำลังที่มีอยู่สำหรับค่าความต้านทานที่สูงขึ้น

<u>ชนิดของตัวต้ำนทาน</u>

ตัวต้านทานที่ผลิตออกมาในปัจจุบันมีมากมายหลายชนิดในกรณีที่แบ่งโดยยึดเอาค่าความต้านทาน เป็นหลักจะแบ่งออกได้เป็น 3 ชนิด คือ

1. ตัวต้านทานแบบค่าคงที่ (Fixed Resistor)

- ตัวต้านทานแบบปรับค่าใด้ (Adjustable Resistor)
- 3. ตัวด้านทานแบบเปลี่ยนค่าได้ (Variable Resistor)

1. ตัวต้านทานแบบค่าคงที่

ตัวต้านทานชนิคค่าคงที่มีหลายประเภท ที่นิยมในการนำมาประกอบใช้ในวงจรทางค้าน อิเล็กทรอนิกส์โคยทั่วไปมีดังนี้

1.1 ตัวต้านทานชนิดการ์บอนผสม (Carbon Composition)

1.2 ตัวต้านทานแบบฟิล์มโลหะ (Metal Film)

1.3 ตัวด้านทานแบบฟิล์มการ์บอน (Carbon Film)

- 1.4 ตัวต้านทานแบบไวร์วาวค์ (Wire Wound)
- 1.5 ตัวด้ำนทานแบบแผ่นฟิล์มหนา (Thick Film Network)
- 1.6 ตัวด้านทานแบบแผ่นฟิล์มบาง (Thin Film Network)

1.1 ตัวต้านทานชนิดการ์บอนผสม (Carbon Composition)

เป็นตัวต้านทานที่นิยมใช้กันแพร่หลายมากและมีราคาถูก โครงสร้างทำมาจากวัสคุที่มีคุณสมบัติเป็น ตัวต้านทานผสมกันระหว่างผงการ์บอนและผงของฉนวน อัตราส่วนผสมของวัสคุทั้งสองชนิดนี้จะทำให้ก่า กวามต้ายทานมีก่ามากน้อยเปลี่ยนแปลงได้ตามต้องการ บริเวณปลายทั้งสองด้านของตัวต้านทานต่อด้วยลวด ตัวนำ บริเวณด้านนอกของตัวต้านทานจะฉาบด้วยฉนวน



รูปตัวต้านทานชนิดการ์บอนผสม

1.2 ตัวต้านทานแบบฟิล์ม โลหะ (Metal Film)

ตัวต้านทานแบบฟิล์มโลหะทำมาจากแผ่นฟิล์มบางของแก้วและโลหะหลอมเข้าด้วยกันแล้วนำไป เคลือบที่เซรามิก ทำเป็นรูปทรงกระบอก แล้วตัดแผ่นฟิล์มที่เคลือบออกให้ได้ก่าความต้านทานตามที่ ต้องการ ขั้นตอนสุดท้ายจะทำการเคลือบด้วยสารอีป๊อกซี (Epoxy) ตัวต้านทานชนิดนี้มีก่าความผิดพลาด บวกลบ 0.1% ถึงประมาณบวกลบ 2% ซึ่งถือว่ามีก่าความผิดพลาดน้อยมาก นอกจากนี้ยังทนต่อการ เปลี่ยนแปลงอุณหภูมิจากภายนอกได้ดี สัญญาณรบกวนน้อยเมื่อเทียบกับตัวต้านทานชนิดอื่น ๆ



รูปแสดงตัวต้ำนทานแบบฟิล์มโลหะ

1.3 ตัวต้ำนทานแบบฟิล์มการ์บอน (Carbon Film)

ตัวต้านทานแบบฟิล์มการ์บอน เป็นตัวต้านทานแบบก่ากงที่โดยการฉาบผงการ์บอนลงบนแท่ง เซรามิกซึ่งเป็นฉนวน หลังจากที่ทำการเกลือบแล้วจะตัดฟิล์มเป็นวงแหวนเหมือนเกลียวน๊อต ในกรณีที่ เกลือบฟิลม์การ์บอนในปริมาณน้อยจะทำให้ได้ก่ากวามต้านทานสูง แต่ถ้าเพิ่มฟิล์มการ์บอนในปริมาณมาก ขึ้นจะทำให้ได้ก่ากวามต้านทานต่ำ ตัวต้านทานแบบฟิลืมโลหะมีก่ากวามผิดพลาด บวกลบ 5% ถึง บวกลบ 20% ทนกำลังวัตต์ตั้งแต่ 1/8 วัตต์ ถึง 2 วัตต์ มีก่ากวามต้านทานตั้งแต่ 1 Ω ถึง 100M Ω



รูปแสดงตัวต้านทานแบบฟิล์มการ์บอน

1.4 ตัวต้านทานแบบไวร์วาวค์ (Wire Wound)

โครงสร้างของตัวต้านทานแบบนี้เกิดจากการใช้ลวดพันลงบนเส้นลวดแกนเซรามิก หลังจากนั้นต่อ ลวดตัวนำด้านหัวและท้ายของเส้นลวดที่พัน ส่วนก่ากวามต้านทานขึ้นอยู่กับวัสดุที่ใช้ทำเป็นลวดตัวนำ ขนาดเส้นผ่าศูนย์กลางของแกนเซรามิกและกวามยาวของลวดตัวนำ ขั้นตอนสุดท้ายจะเกลือบด้วยสาร ประเภทเซรามิกบริเวณรอบนอกอีกกรั้งหนึ่ง ก่ากวามต้านทานของตัวต้านทานแบบนี้จะมีก่าต่ำเพราะ ต้องการให้มีกระแสไหลได้สูง ทนกวามร้อนได้ดีและสามารถระบายกวามร้อนโดยใช้อากาศถ่ายเท

1.5 ตัวด้ำนทานแบบแผ่นฟิล์มหนา (Thick Film Network)

โครงสร้างของตัวต้านทานแบบนี้ทำมาจากแผ่นฟิล์มหนา มีรูปแบบแตกต่างกันขึ้นอยู่กับการใช้งาน ในรูปที่ 6 แสดงตัวต้านทานแบบแผ่นฟิล์มหนาประเภทไร้ขา (Chip Resistor) ตัวต้านทานแบบนี้ต้องใช้ เทกโนโลยี SMT (Surface Mount Technology) ในการผลิต มีอัตราทนกำลังประมาณ 0.063 วัตต์ ถึง 500 วัตต์ ค่าความคลาดเคลื่อนบวกลบ 1% ถึง บวกลบ 5%

1.6 ตัวต้ำนทานแบบแผ่นฟิล์มบาง (Thin Film Network)

โครงสร้างของตัวต้านทานแบบนี้ทำมาจากแผ่นฟิล์มบาง มีลักษณะรูปร่างเหมือนกับตัวไอซี (Integrate Circuit) ใช้เทคโนโลยี SMT (Surface Mount Technology) ในการผลิตเช่นเดียวกับตัวต้านทาน แบบแผ่นฟิล์มหนา โดยส่วนใหญ่จะมีงาทั้งหมด 16 งา การใช้งานต้องบัดกรีเง้ากับแผ่นลายวงจร อัตราทน กำลัง 50 มิลลิวัตต์ มีค่าความคลาดเคลื่อนบวกลบ 0.1% และอัตราทนกำลัง 100 มิลลิวัตต์ จะมีค่าความ คลาดเคลื่อนบวกลบ 5% ที่แรงคันไฟฟ้าสูงสุดไม่เกิน 50 VDC

2. ตัวต้านทานแบบปรับค่าได้

โครงสร้างของตัวต้านทานแบบนี้ทำมาจากแผ่นฟิล์มบาง มีลักษณะรูปร่างเหมือนกับตัวไอซี (Integrate Circuit) ใช้เทคโนโลยี SMT (Surface Mount Technology) ในการผลิตเช่นเดียวกับตัวต้านทาน แบบแผ่นฟิล์มหนา โดยส่วนใหญ่จะมีขาทั้งหมด 16 ขา การใช้งานต้องบัดกรีเข้ากับแผ่นลายวงจร อัตราทน กำลัง 50 มิลลิวัตต์ มีค่าความคลาดเคลื่อนบวกลบ 0.1% และอัตราทนกำลัง 100 มิลลิวัตต์ จะมีค่าความ กลาดเคลื่อนบวกลบ 5% ที่แรงคันไฟฟ้าสูงสุดไม่เกิน 50 VDC



รูปแสดงลักษณะรูปร่างของตัวต้านทานแบบปรับค่าได้

3. ตัวต้านทานแบบเปลี่ยนค่าได้

ตัวต้านทานแบบเปลี่ยนค่าได้ (Variable Resistor) โครงสร้างภายในทำมาจากการ์บอนเซรามิก หรือ พลาสติกตัวนำ ใช้ในงานที่ต้องการเปลี่ยนก่าความต้านทานบ่อย ๆ เช่น ในเครื่องรับวิทยุ, โทรทัศน์ เพื่อปรับ ลดหรือเพิ่มเสียง, ปรับลดหรือเพิ่มแสงในวงจรหรี่ไฟ มีอยู่หลายแบบขึ้นอยู่กับวัตถุประสงค์ของการใช้งาน เช่น โพเทนชิโอมิเตอร์ (Potentiometer) หรือพอต (Pot) สำหรับชนิดที่มีแกนเลื่อนก่าความด้านทานหรือ แบบที่มีแกนหมุนเปลี่ยนก่าความต้านทานคือโวลลุ่ม (Volume) เพิ่มหรือลดเสียงมีหลายแบบให้เลือกคือ 1 ชั้น, 2 ชั้น และ 3 ชั้น เป็นต้น ส่วนอีกแบบหนึ่งเป็นแบบที่ไม่มีแกนปรับโดยทั่วไปจะเรียกว่า โวลลุ่มเกือกม้า หรือทิมพอต (Trimpot)



รูปแสดงลักษณะรูปร่างของตัวต้านทานแบบเปลี่ยนค่าได้

ตัวต้านทานแบบเปลี่ยนค่าได้นี้ สามารถแบ่งออกเป็น 2 ชนิด ด้วยกัน คือ โพเทนชิโอมิเตอร์ (Potentiometer) และเซนเซอร์รีซิสเตอร์ (Sensor Resistor)

<u>การวัดตัวต้านทานด้วยมัลติมิเตอร์</u>

หากไม่แน่ใจเกี่ยวกับค่าที่อ่านได้จากแถบสี นักเขียนโปรแกรมสามารถตรวจสอบโดยใช้มัลติมิเตอร์



รูปการวัดค่าความต้านทานด้วยมัลติมิเตอร์

เมื่อเปิดมัลติมิเตอร์ขึ้นมาแล้วให้ปรับย่านวัดอยู่ในช่วงความต้านทานที่น้อยที่สุด จากนั้นใช้โพรบ ทั้ง 2 ขั้ว ของมัลติมิเตอร์สัมผัสกับขาของตัวต้านทาน เนื่องจากตัวต้านทานไม่มีขั้วหมายความว่า นักเขียน โปรแกรมสามารถใช้โพรบทั้งสายสีดำและสีแดงสัมผัสกับขาใดก็ได้ อีกทั้งยังสามารถใช้คลิปหนึบขั้วโพรบ เพื่อทำการวัดตัวต้านทานก็ได้

หากบนหน้าจอมัลติมิเตอร์ ไม่มีการแสดงตัวเลขหรือแสดงเป็น -1 ให้เปลี่ยนเป็นย่านวัดความ ด้านทานที่สูงขึ้นและปรับย่านวัดขึ้นไปเรื่อยๆ จนกว่าจะเห็นตัวเลขบนหน้าจอมัลติมิเตอร์ซึ่งนักเขียน โปรแกรมต้องสังเกตก่าตัวคูณที่เลือกไว้ เช่น ถ้าอ่านก่าได้ 47 และย่านวัดคือ " 20 กิโลโอห์ม " หรือ " 20k พลลัพธ์ก็คือ 47 กิโลโอห์มหรือ 47,000 โอห์ม



ตัวอย่างการทคลองต่อวงจรและเขียนกำสั่งตัวต้านทาน

คำสั่งควบคุมบอร์ค Arduino



หน่วยการเรียนรู้ที่ 7 จอภาพตอบสนองผู้ชม เรื่องย่อยที่ 3 ไฟ LED

ประวัติของหลอก LED

ปรากฏการณ์ Electroluminescence คือการที่วัตถุปล่อยแสงเมื่อถูกกระตุ้นด้วยไฟฟ้าโดยไม่เกี่ยวกับ ความร้อน ผู้ที่ก้นพบปรากฏการณ์ Electroluminescence เป็นกนแรกในปี ค.ศ. 1907 คือนักวิทยาศาสตร์ชาว อังกฤษชื่อ H. J. Round โดยสารประกอบที่เป็นตัวทำให้เกิดแสงคือซิลิกอนการ์บายด์ (SiC) ในปีค.ศ. 1927 นักวิทยาศาสตร์ชาวรัสเซียชื่อ Oleg Vladimirovich Losev ได้นำปรากฏการณ์นี้มาพัฒนาเป็น LED ได้เป็น ครั้งแรก (แต่ตอนนั้นยังไม่ได้เรียกว่า LED) Losev ได้ตีพิมพ์เผยแพร่ผลงานการวิจัยเกี่ยวกับการเปล่งแสง ของได โอดที่ทำจาก SiC (ในสมัยนั้นเรียกว่า Carborundum) โดยลำพังตัวถนเดียวทั้งในวารสารวิชาการของ รัสเซียและของยุโรปถึง 16 เรื่องในระหว่างปีค.ศ. 1924 - 1930 แต่ผลงานเหล่านี้ไม่ได้รับการสานต่อ อาจ เพราะเกิดขึ้นมาในช่วงรอยต่อของสงกราม โลกครั้งที่ 1 กับ 2 การบูรณะบ้านเมืองกับการสร้างอาวุธข่อม ได้รับความสนใจมากกว่า อีกทั้ง Losev เองก็มีอายุสั้นคือเสียชีวิตเมื่อปีค.ศ. 1942 ด้วยอายุเพียง 39 ปีจากการ ขาดอาหารเพราะกรุงเลนินกราด (หรือกรุงเซนต์ปีเตอร์เบอร์กในปัจจุบัน) ถูกปิดล้อมโดยกองทัพนาซี หลัง สงกรามโลกครั้งที่ 2

ผู้ที่ให้กำเนิด LED ที่ปล่อยแสงที่ตามองเห็นได้เป็นคนแรกในปีค.ศ. 1962 คือ Nick Holonyak โดย เป็น LED ให้แสงสีแดง สารประกอบต้นตอแสงคือ แกลเลี่ยม อาเซนายค์ ฟอสฟายค์ (GaAsP) ด้วยเหตุที่ LED ของ Holonyak ปล่อยแสงที่ตามองเห็นจริงๆ Holonyak จึงได้รับการยกย่องว่าเป็น "บิดาของ LED" Holonyak เป็นลูกศิษย์ปริญญาเอกคนแรกของ John Bardeen นักฟิสิกส์ทฤษฎี เจ้าของรางวัลโนเบลประจำปี ค.ศ. 1956

วงจรและการเขียนโปรแกรมควบคุมหลอด LED

การเขียนโปรแกรมเพื่อควบคุมการ ติค-ดับ ของหลอดไฟ LED ถือเป็นการเขียนโปรแกรมบน Arduino UNO ที่ง่ายที่สุดทำให้ผู้ที่ไม่เคยเขียนโปรแกรมคอมพิวเตอร์มาก่อนก็สามารถทำความเข้าใจได้ไม่ ยาก อีกทั้งยังเป็นการเรียนรู้การต่อวงจรอิเล็กทรอนิกส์เบื้องต้นไปพร้อมกันด้วย ในที่นี้จะมาต่อวงจรให้กับ LED เพื่อทำให้มันสว่าง โดยอุปกรณ์ที่ต้องใช้กีคือ

- 1. บอร์ด Arduino UNO R3
- 2. หลอด LED ขนาด 5MM สีเขียว จำนวน 2 ตัว
- 3. หลอด LED ขนาด 5MM สีแดง จำนวน 1 ตัว
- 4. รีซิสเตอร์ 220 Ohms จำนวน 3 ตัว

- 5. Jumper cable wire 10cm Male to Male (สายเชื่อมต่อ ผู้-ผู้)
- 6. Breadboard Protoboard 400 Tie-points Size 8.5*5.5 cm (โฟโต้บอร์ค)

ฟิสิกส์เบื้องหลัง

ในวงจรอิเล็กทรอนิกส์ ไดโอด (diode) เป็นอุปกรณ์ที่มีบทบาทสำคัญไม่ยิ่งหย่อนไปกว่าตัว ด้านทาน ตัวคาปาซิเตอร์ หรือตัวทรานซิสเตอร์ ไดโอดทำหน้าที่ควบคุมทิศทางการไหลของกระแสไฟฟ้า ให้วิ่งผ่านได้ทางเดียว วิ่งสวนย้อนศรไม่ได้ สิ่งที่มีพฤติกรรมเหมือนไดโอดถูกประดิษฐ์ขึ้นเป็นครั้งแรกเมื่อปี ก.ศ. 1904 โดยนักฟิสิกส์ชาวอังกฤษชื่อ John Ambrose Fleming โดยมีรูปลักษณ์เริ่มต้นเป็นหลอดสุญญากาศ (vacuum tube) ในตอนนั้นเรียกสิ่งประดิษฐ์นี้กันว่า "Fleming valve" แต่ปัจจุบันไดโอดที่ทำจากสารกึ่งตัวนำ ได้เข้ามาแทนที่เพราะมีประโยชน์เหนือกว่าหลายประการเช่นมีราคาถูกกว่า ขนาดเล็กกว่ามาก ตอบสนอง เร็วกว่า ไม่ต้องมีการจุดไส้หลอด ฯลฯ

กล ใกเบื้องหลังการทำงานของ ใด โอคเปล่งแสงต้องอธิบายด้วยทฤษฎีของการเคลื่อนที่ของ ้อิเล็กตรอนที่เรียกกันว่า Band Theory of Solids ซึ่งสร้างอย่บนพื้นฐานคือทฤษฎีควอนตัม Band Theory นี้ ร่วมกันพัฒนาขึ้นโดยนักฟิสิกส์รุ่นลูกศิษย์ของ W. Pauli และ W. Heisenberg หลายคนเช่น F. Bloch R. Peierls และ A. Wilson เป็นต้น ในที่นี้อธิบายพอเป็นสังเขปได้ว่า ไดโอคสารกึ่งตัวนำทำจากการนำสารกึ่ง ตัวนำสองชนิดคือชนิด n (สารกึ่งตัวนำที่โด๊ป (dope) ด้วยมลทินเช่นฟอสฟอรัสเป็นต้น) กับชนิด p (สารกึ่ง ้ตัวนำที่โค๊ปด้วยมลทินเช่นโบรอนเป็นต้น) มาประกบกัน โดยตรงรอยต่อเกิดเป็น p-n junction ซึ่งเป็นเขต ้ปลอดประจุไฟฟ้าที่เป็นพาหะของกระแสไฟฟ้า มีชื่อเป็นทางการว่า depletion region เขตปลอดประจุนี้บาง มาก อยู่ในเรือน ใม โครเมตรหรือต่ำกว่าเท่านั้น อิเล็กตรอน ในแถบ conduction (conduction band หรือ CB) จากค้าน n ไม่สามารถแพร่ข้ามเข้าไปอยู่ในแถบ conduction ของค้าน p ใค้เพราะไม่สามารถข้ามกำแพง ความต่างศักย์ที่สูง V0 คังแสคงในรูปที่ 4(ก) ซึ่งเป็นแผนภาพแถบพลังงานตรงบริเวณ p-n junction ของ ใดโอดที่ยังไม่ได้จ่ายไฟให้ แต่เมื่อมีการจ่ายไฟให้แบบตรงขั้ว (forward biased) สมมติมีค่าความต่างศักย์ V ้ กำแพงคังกล่าวจะเตี้ยลงเหลือเป็นความสูงเพียง V0 – V คังแสคงในรูปที่ 4(ข) ซึ่งเตี้ยพอที่อิเล็กตรอนจะ สามารถข้ามเข้าไปอยู่ในค้าน p ไค้ แต่เนื่องจากสารกึ่งตัวนำชนิค n ถูกโค๊ปเข้มข้นกว่าการโค๊ปของสารกึ่ง ้ ตัวนำชนิด p (แทนด้วยสัญลักษณ์ n+ ในรูป) อิเล็กตรอนจึงแพร่เข้าไปอยู่ด้าน p มากกว่าที่พวก hole ในแถบ valence (valence band หรือ VB) จะแพร่เข้าไปอยู่ในแถบ valence ของด้าน n เมื่อมีกู่ hole อยู่ฝั่งตรงข้ามของ ้ด้านเดียวกัน อิเล็กตรอนจึงสามารกระ โคคลงไปจับคู่ (recombination) แต่เป็นการลคระดับพลังงานของ ้ตัวเองเท่ากับ energy gap (Eg) ระหว่างแถบconduction กับแถบ valence ของสารกึ่งตัวนำชนิดนั้นๆ ทำให้ อิเล็กตรอนต้องปลดปล่อยพลังงานส่วนเกินออกไปตามกฎการคงตัวของพลังงาน (Law of Conservation of Energy) โดยในกรณีนี้ปลดปล่อยออกไปในรูปของแสง ปรากฏการณ์นี้เองที่เรียกว่า Electroluminescence

ความยาวคลื่นของแสง (λ) ที่ถูกปล่อยออกมาตามกลไกในรูปที่ 4 (ข) สามารถคำนวณได้จากสมการง่ายๆ ดังนี้

$$\lambda = hc / Eg$$

เมื่อ h คือค่าคงที่ของพลังค์ และ c คือความเร็วแสงในสุญญากาศ เช่น สารกึ่งตัวนำ GaAs มี Eg = 1.43 eV แสงที่ปล่อยออกมาก็จะมี λ = 867 นาโนเมตร หรือเทียบเท่ากับแสงในย่านอินฟราเรด



รูป แผนภาพแถบพลังงานของ homojunction ระหว่างสารกึ่งตัวนำชนิด p กับชนิด n ที่มี Eg เท่ากัน (ก) ใน กรณีที่ยังไม่จ่ายไฟให้ กับ (ข) กรณีที่จ่ายไฟความต่างศักย์ V ให้แบบตรงขั้ว [1, 2]

ดังนั้นโดยการวิจัยและพัฒนาตัวสารกึ่งตัวนำและเทคนิคการสร้างใดโอด เช่นการพัฒนาและ ดัดแปลงวิธีการปลูกผลีก (crystal growth) เทคนิคการโด๊ปสารมลทิน ชนิดของสารที่ผสมและ / หรือ การ พัฒนาจาก homojunction เป็นแบบ heterojunction (สารกึ่งตัวนำชนิด n กับ p มี Eg ไม่เท่ากัน) เพื่อคัดแปลง ลักษณะของแถบพลังงาน ความบริสุทธิ์ของสารที่เกี่ยวข้องเพราะมลทินหรืความบกพร่องของผลึกมีส่วนทำ ให้การย้ายตำแหน่งของอิเล็กตรอนไม่เกิดผลเป็นแสง เป็นต้น สามารถทำให้เกิดไดโอดที่ให้แสงได้แทบทุก สี และทำให้มีความเข้มแสงหรือความสว่างเพิ่มมากขึ้นได้ด้วย

นอกจากความเข้มแสงจะเพิ่มขึ้นได้ด้วยวิธีการดังกล่าวแล้ว ยังสามารถเพิ่มขึ้นได้ด้วยวิธีการ ดังต่อไปนี้ด้วย

ก) ทำให้เกิดการดูดกลืนภายในลดลง โดยการทำสารกึ่งตัวนำชนิด p เป็นชั้นบางๆ ดังแสดงในรูปที่ 5 เพราะแสงจะปล่อยออกมาจากบริเวณ junction ค่อนไปทางด้าน p



รูป แผนภาพแสดงโครงสร้างแบบหนึ่งของ LED ที่ตั้งใจทำให้สารกึ่งตัวนำชนิด p เป็นชั้นบางๆ

ข) ลดการสะท้อนกลับหมดภายในที่รอยต่อระหว่างผิวหน้าของสารกึ่งตัวนำชนิด p (ค่าดัชนีหักเห ประมาณ 3.3) กับอากาศ (ค่าดัชนีหักเหเท่ากับ 1) (ดูรูปที่ 6 (ก)) โดยการหุ้มด้วยพลาสติกใสที่มีค่าดัชนีหัก เหมากกว่าอากาศ และทำผิวบนของพลาสติกให้เป็นรูปทรงครึ่งทรงกลมเพื่อแสงจะตกตั้งฉากกับผิวนี้เสมอ ดังแสดงในรูปที่ 6 (ข) แสงก็จะออกมาจาก LED ได้มากขึ้น



การเกิดการสะท้อนกลับหมดภายในที่รอยต่อระหว่างผิวหน้าของสารกึ่งตัวนำชนิด p กับอากาศ และ (ข) วิธีการที่มีต้นทุนต่ำที่ช่วยปลดปล่อยแสงจากสารกึ่งตัวนำออกมาทางด้านหน้ามากขึ้นคือหุ้มด้วยพลาสติกใส รูปครึ่งทรงกลม

การที่ LED ได้รับความนิยมเพิ่มขึ้นเรื่อยๆก็เพราะ LED มีข้อดีเหนือหลอดไฟทั้งแบบธรรมดาและ แบบฟลูออเรสเซนต์หลายประการเช่น ในจำนวนวัตต์ที่เท่ากัน LED ผลิตแสงมากกว่าหลอดไฟธรรมดาไม่ ต่ำกว่า 5 เท่า และมากกว่า หลอดตะเกียบ (compact fluorescent lamp หรือ CFL) ไม่ต่ำกว่า 2 เท่า จึงประหยัด พลังงานมากกว่า LED สร้างแสงสีต่างๆ ได้ด้วยตัวเองโดยไม่ต้องอาศัยฟิลเตอร์ช่วย ทำให้มีต้นทุนถูกกว่า สิ้นเปลืองการบำรุงรักษาน้อยกว่าเช่น บริษัท Philips โฆษณาว่า LED ของตนมีอายุการใช้งานนาน 50,000 ชั่วโมง ในขณะที่หลอดไฟธรรมดากับหลอดฟลูออเรสเซนต์มีอายุการใช้งาน 1,000 – 2,000 ชั่วโมง และ 30,000 ชั่วโมง ตามลำดับ มีการตอบสนองเร็ว สว่างเต็มที่ได้ภายในเวลาในเรือนไมโครวินาที มีขนาดเล็ก และน้ำหนักเบามาก ดังจะเห็นได้ว่าสามารถทำเป็นเม็ดแม่สีแสงเล็กๆบนจอโทรทัศน์ (LED TV) หรือจอ แสดงภาพ (LED display) ได้ และไม่ใช้สารปรอทจึงเป็นมิตรกับสิ่งแวดล้อม เป็นต้น ปัจจบันการแข่งขันกันพัฒนา LED ที่มีความเข้มสูงขึ้นๆ แต่มีราคาถูกลงๆกำลังคำเนินไปอย่าง

เข้มข้น ทั้งนี้เพราะมีแรงจูงใจจากข้อมูลที่ว่าตลาด LED มีแนวโน้มที่จะเติบโตขึ้นอย่างต่อเนื่อง เช่นมีผู้ ประเมินว่าในปีนี้ (ค.ศ. 2013) ตลาด LED ทั่วโลกจะมีมูลค่าสูงถึง 7.6 แสนล้านบาท โดยสูงกว่าปีที่แล้วถึง 54 % ตลาดในอนากตของ LED จะยิ่งสดใสกว่านี้

วงจรอิเล็กทรอนิกส์



เริ่มต่อวงจร กับบอร์ด Arduino UNO หลักการในการต่อคือ ต่อให้แรงดันขนาด 5 โวลต์ ให้ไหล ผ่านหลอด LED แล้วไหลลงสู่ GND หลอด LED ก็จะสว่างขึ้นมา แต่เราจำเป็นต้องเพิ่มตัวต้านทานขนาด 220 โอห์ม เข้าไปในเส้นทางของวงจรด้วย เพื่อต้านทานกระแสไฟฟ้าไม่ให้ไหลผ่าน LED มากเกินไป



รูปจำลองการต่อหลอดไฟ LED

การต่อวงจรภายในแผ่นโพโตบอร์ค (Breadboard) จะเป็นคังรูปกล่าวคือ ด้านบนและด้านล่างจะใช้ สำหรับต่อไปเลี้ยงวงจรซึ่งวงจรด้านล่างจะต่อถึงกันทั้งแถว ส่วนการต่อวงจรด้านใน จะต่อถึงกันตามแนวตั้ง จากรูปจำลองการต่อหลอดไฟ LED ด้านบน มีการเขียนโปรแกรม คังนี้

💿 sketch_sep20a Arduino 1.8.9 - 🗆 🗙	
File Edit Sketch Tools Help	
sketch_sep20a§	
<pre>1 char LED_pin[] = {2,3,4,5}; 2 void send2port(byte data); 3 void setup() 4 { 5 Serial.begin(9600); 6 for(char i=0;i<4;i++) 7 { 8 pinMode(LED_pin[i],OUTPUT); 9 } 10 } 11 void loop() 12 { 13 send2port(0B1000); 14 delay(500); 15 send2port(0B0100); 16 delay(500); 17 send2port(0B0010); 18 delay(500); 19 send2port(0B0001); 20 delay(500); 21 }</pre>	^
<pre>22 void send2port(byte data) 23 {</pre>	
<pre>24 if (data & 1){digitalWrite(LED_pin[0],HIGH);} else {digitalWrite(LED_pin[0],LOW);} 25 if (data & 2){digitalWrite(LED_pin[1],HIGH);} else {digitalWrite(LED_pin[1],LOW);} 26 if (data & 4){digitalWrite(LED_pin[2],HIGH);} else {digitalWrite(LED_pin[2],LOW);} 27 if (data & 8){digitalWrite(LED_pin[3],HIGH);} else {digitalWrite(LED_pin[3],LOW);} 28 }</pre>	
29	¥

หน่วยการเรียนรู้ที่ 7 จอภาพตอบสนองผู้ชม เรื่องย่อยที่ 4 การเลื่อนภาพ

ในหน่วยการเรียนรู้นี้จะกล่าวถึงการเลื่อนภาพผ่านจอ LCD (แอลซีดี) ปกติจอ LCD จะใช้สายไฟ หลายเส้นในการต่อใช้งาน ทำให้บางครั้ง Arduino มีขาไม่พอที่จะควบคุมและยังทำให้การเดินสายต่อใช้งาน ไม่สะดวก มีวงจรแปลงสัญญาณสำหรับติดต่อจอ LCD แบบอินเตอร์เฟส I2C ต่อแปลงการใช้สายไฟหลาย ๆ เส้น ให้เหลือสายไฟเพียง 2 เส้น ทำให้การเขียนโปรแกรมและการต่อวงจรทำได้ง่ายขึ้น เพียงแค่ใช้โมดูล LCD I2C ไปต่อกับจอ LCD แบบเดิม ก็สามารถใช้งานได้ทันที โมดูล I2C LCD นี้จะมีตัวด้านทานปรับค่าได้ สำหรับปรับความสว่างหน้าจอมาด้วย



หากสังเกตอย่างละเอียดจะเห็นว่าการแสดงผลของจอ LCD นั้นจะมีข้อความที่เกิดจากช่องสี่เหลี่ยม เล็ก ๆ มีสถานะเปิดกับปิดรวมกันหลาย ๆ ช่องกลายเป็น 1 ตัวอักษร ซึ่งเราสามารถสร้างอักษรของเราขึ้นมา เองได้ อาจจะใช้เว็บนี้ในการช่วยสร้างข้อความ

จอ LCD ที่ใช้งานร่วมกับโมคูล LCD I2C นี้มีอยู่ 2 รุ่นคือ

- 1. รุ่น LCD 16x2 หรือรุ่น LCD 1602 สามารถแสดงข้อความได้ 16 ตัวอักษร 2 บรรทัด
- 2. รุ่น LCD 20x4 หรือรุ่น LCD 2004 สามารถแสดงข้อความได้ 20 ตัวอักษร 4 บรรทัด

ก่อนใช้งาน Arduino จอ LCD แบบ I2C แสคงข้อความ ให้นักเขียนโปรแกรมคาวน์โหลดไลบารี LCD คลิกที่นี่ <u>http://www.allarduino.com/download/lcd_i2c.rar จากนั้นทำการติดตั้ง Library แบบ .zip</u> ดังนี้



การต่อใช้งาน จอ LCD แบบ I2C

Vcc -> 3.3-5V

 $Gnd \rightarrow Gnd$

SDA -> 4

SCL -> 5

คำสั่งควบคุมบอร์ค Arduino Uno R3

00	test12_2 Arduino 1.8.9 —	×
File	Edit Sketch Tools Help	
Ø		Ø
te	st12_2§	
1	<pre>#include <wire.h></wire.h></pre>	
2	<pre>#include <lcd.h></lcd.h></pre>	
3	<pre>#include <liquidcrystal_i2c.h></liquidcrystal_i2c.h></pre>	
4		
5	#define I2C_ADDR 0x27	
6	#define BACKLIGHT_PIN 3	
7		
8	LiquidCrystal_I2C lcd(I2C_ADDR,2,1,0,4,5,6,7);	
9	<pre>void setup() {</pre>	
10	<pre>lcd.begin (20,4);</pre>	
11	<pre>lcd.begin (16,2);</pre>	
12	<pre>lcd.setBacklightPin(BACKLIGHT_PIN, POSITIVE);</pre>	
13	<pre>lcd.setBacklight(HIGH);</pre>	
14	lcd.home (); // ไปที่ตัวอักษรที่ 0 แถวที่ 1	
15		
16	lcd.setCursor(1,0); // ໃປກຕັວວັດษรที่ 1 ແຄວที่ 1	
17	<pre>lcd.print("Welcome ALL");</pre>	
18		
19	lcd.setCursor(3,1); // ใปที่ตัวอักษรที่ 5 แถวที่ 2	
20	<pre>lcd.print("Hi LCD Arduino");</pre>	
21	// lcd.write(1);	
22	// lcd.write(2);	
23		
24	}	
25		
26	}	
27		
28	void loop() {	
29)	

หน่วยการเรียนรู้ที่ 7 จอภาพตอบสนองผู้ชม เรื่องย่อยที่ 5 การเชื่อมต่อ Arduino กับ Processing

การเชื่อมต่อ Arduino กับ Processing

Processing เป็นภาษาโอเพนซอร์สสำหรับการสร้างซอฟต์แวร์ที่มีภาพ ภาพเคลื่อนไหม และการ ตอบโต้ เราสามารถดาวน์โหลด Processing IDE ได้ จากเว็บไซต์ <u>http://processing.org/download สิ่ง</u>แรกที่ อาจสังเกตเห็นก็คือ มันดูเหมือนกับโปรแกรม Arduino IDE นั่นเป็นเพราะ Arduino และ Processing เป็น โปรเจ็กต์พี่น้องกัน มีเป้าหมายเหมือนกันที่ความง่าย เป็นแบบโอเพนซอร์ส และเข้าถึงผู้คนจำนวนมาก

แม้ว่า Processing และ Arduino เป็นคนละภาษากัน แต่ก็คล้ายกันมากพอที่สลับการใช้ได้ไม่ยาก

Processing เรียกโปรแกรมว่า sketch เช่นเดียวกับ Arduino ซึ่งโปรแกรมต่อไปนี้จะอ่ายพอร์ต อนุกรม และแสดงผลในหน้าต่างของ processing เราสั่งให้โปรแกรมทำงานโดยคลิกปุ่ม Run

Import processing.serial.*; (1)

//A serial port that we use to talk to Aruino.

Serial myPort; (2)

//The Processing setup method that's run once
void setup() { (3)

Size(320, 320); // creat a window (4)

//List all the available serial ports: (5)
printIn(Serial.list());

/*On some computer, Arduino will usually be connected to the first serial port. If not, change the 0 to the correct one (examine the output of the previous line of code to figure out which one to use). */

myPort = new Serial(this, Serial.list()[0], 9600);

}

/* The draw() method is called up to 60 times a second unless you change the frame rate of Processing.
Normally, it is use update the graphics onscreen, but we're just polling the serial port here. */
Void draw() $\{ (6) \}$

}

// Put up a black background.
background(0);
// Read the serial port.
if (myPort.available() >0) { (7)
 Char inByte = myPort.readChar();
 print(inByte); // Displays the character that was read
}

```
สำหรับบรรทัดที่สำคัญของโปรแกรมนี้
```

(1) Processing มาพร้อมกับ ใลบารี่สำหรับขยายความสามารถจำนวนหนึ่ง ซึ่งบรรทัคนี้โหลด ไลบารี่ สำหรับการสื่อสารกับพอร์ตอนุกรม

(2) ประกาศออบเจ็กต์ประเภท Serial ในชื่อ myPort

(3) ฟังก์ชัน setup() ทำหน้าที่เหมือนกับ setup() ใน Arduino ซึ่งจะถูกเรียกใช้ครั้งเดียวตอน โปรแกรมเริ่มทำงาน

(4) สร้างหน้าต่างตามความกว้างและความสูงที่ระบุ (เป็นพิเซล)

(5) เราจะเห็นคำสั่งที่คล้ายๆกับบรรทัดที่เป็นตัวหนาในโปรแกรมของ Processing จำนวนมาก สำหรับการสื่อสารกับพอร์ตอนุกรม ในคอมพิวเตอร์ส่วนใหญ่บอร์ค Arduino จะเชื่อมต่อกับพอร์ตอนุกรม แรกของกอมพิวเตอร์ เพราะคอมพิวเตอร์รุ่นใหม่ๆ ไม่มีพอร์ตอนุกรมในตัว ดังนั้นพอร์ตที่ต่ำที่สุดจึงมักจะ เป็นพอร์ตที่ Arduino ใช้อยู่ หากไม่ใช่ ให้ตรวจสอบเอาต์พุตในหน้าต่างการประมวลผล และเปลี่นร 0 ที่ บรรทัดตัวหนาสุดท้ายเป็นเลขพอร์ตอนุกรมที่ถูกต้อง

(6) ฟึงก์ชัน draw() จะเหมือนกับ loop() ของ Arduino ซึ่งมีการเรียกใช้ไปเรื่อยๆ โดยทำงาน 60 ครั้ง ต่อวินาที (60) เว้นแต่ว่าเราจะเปลี่ยนอัตราเฟรมด้วยฟึงก์ชัน frameRate()

(7) คำสั่งบลีอกนี้ตรวจสอบให้แน่ใจว่ามีข้อความจากพอร์ตอนุกรม (serial port) จึงจะอย่านและ พิมพ์ออกมา

หน่วยการเรียนรู้ที่ ๙ ทำความรู้จักเซนเซอร์

กลุ่มสาระการเรียนรู้วิทยาศาสตร์ สาระที่ ๔ เทคโนโลยี

มาตรฐาน ว ๔.๑ เข้าใจแนวคิดหลักของเทคโนโลยีเพื่อการคำรงชีวิตในสังคมที่มีการเปลี่ยนแปลงอย่าง รวดเร็ว ใช้ความรู้และทักษะทางค้านวิทยาศาสตร์ คณิตศาสตร์ และศาสตร์อื่น ๆ เพื่อแก้ปัญหาหรือพัฒนา งานอย่างมีความคิดสร้างสรรค์ค้วยกระบวนการออกแบบเชิงวิศวกรรมเลือกใช้เทคโนโลยีอย่างเหมาะสม โดยคำนึงถึงผลกระทบต่อชีวิต สังคม และสิ่งแวคล้อม

า ๔.๑ ม.๑/๔	ใช้เทคโนโลยีสารสนเทศอย่างปลอคภัย ใช้สื่อและแหล่งข้อมูลตามข้อกำหนดและ
	ข้อตกลง
ી ૡ.૭ ૧૧.૭/૬	ใช้ความรู้และทักษะเกี่ยวกับวัสคุ อุปกรณ์ เครื่องมือ กลไก ไฟฟ้า หรืออิเล็กทรอนิกส์
	เพื่อแก้ปัญหาได้อย่างถูกต้อง เหมาะสมและปลอดภัย

มาตรฐาน ว ๔.๒ เข้าใจและใช้แนวคิคเชิงคำนวณในการแก้ปัญหาที่พบในชีวิตจริงอย่างเป็นขั้นตอนและ เป็นระบบ ใช้เทคโนโลยีสารสนเทศและการสื่อสารในการเรียนรู้ การทำงาน และการแก้ปัญหาได้อย่างมี ประสิทธิภาพ รู้เท่าทัน และมีจริยธรรม

ตัวชี้วัด

ານ
ใน
เม
าง
ររ
ร์พ
ן ייז ו

สาระสำคัญ

- o. Temperature Sensor Module ແລະ Temperature Humidity Sensor
- ه. IR Receiver Module
- ຫ. Tracking Sensor
- د. Buzzer Module
- &. Switch Button Module
- ъ. Light Cup Module
- භ. Light Blocking Sensor
- ສ. Relay ແລະ Motion Sensor Module
- ช. Temperature Humidity Sensor และ LED 4-Digit Display Module

สาระการเรียนรู้

<u>ความรู้</u>

 ๑. ผู้เรียนจะ ได้ศึกษาและทำความเข้าใจรวมถึงการต่อวงจรอุปกรณ์อิเล็กทรอนิกส์เข้ากับบอร์ด Arduinoอย่างละเอียด

๒. ผู้เรียนจะ ได้ศึกษาถึงอุปกรณ์ที่ต้องใช้ชุดคำสั่งเฉพาะ (library) ของตัวอุปกรณ์

ด. ผู้เรียนจะ ได้ศึกษาคำสั่งที่มีเงื่อน ไขเพิ่มขึ้นเพื่อเพิ่มทักษะอัลกอริทึม

<u>ทักษะ/กระบวนการ</u>

๑. ใช้กระบวนการสังเกตเพื่อเก็บข้อมูล วิเคราะห์ถึงสิ่งที่กำลังศึกษาทั้งกำสั่งตลอดจนอุปกรณ์

๒. ใช้กระบวนการปฏิบัติ เพื่อให้ผู้เรียนปฏิบัติจริงจนเกิดทักษะตลอดจนต่อยอดพัฒนาเป็นโกรงการ ที่เป็นประโยชน์

<u>คุณลักษณะที่พึงประสงค์</u>

- ๑. ผู้เรียนมีใจรักและพร้อมพัฒนาโปรแกรมควบคู่กับอุปกรณ์
- ๒. ผู้เรียนมีน้ำใจต่อสมาชิกในกลุ่ม
- ๓. รักชาติ ศาสน์ กษัตริย์
- ๔. ซื่อสัตย์
- ๕. มีวินัย
- ปล่เรียนรู้
- ๗. อยู่อย่างพอเพียง
- ๘. มุ่งมั่นในการทำงาน

หน่วยการเรียนรู้ที่ 8 ทำความรู้จักเซนเซอร์

เรื่องย่อยที่ 1 Temperature Sensor Module และ Temperature Humidity Sensor

Temperature Sensor Module

Temperature Sensor Module เป็นอุปกรณ์เพื่อวัดค่าอุณหภูมิของสิ่งแวดล้อม โดยแสดงผลแบบผ่าน Optamp เปรียบเทียบกับค่าที่ต้องไว้ด้วย Potentiometer สามารถใช้งานกับบอร์ด Arduino และ Raspberry Pi ได้ทุกรุ่น เนื่องจากสามารถทำงานได้หลายระดับแรงดัน ขึ้นกับแรงดัน VCC ที่ป้อน แสดงผลแบบ Digital ที่ Pin Do เช่น หากอุณหภูมิของสิ่งแวดล้อมสูงเกินกว่าค่าที่ตั้งไว้ ขา D0 จะให้ Logic เป็น 1 หากมีอุณหภูมิต่ำ กว่าจะให้แรงดันออกเป็น logic 0 สามารถติดตั้งได้ง่ายเนื่องจากมีช่องเจาะเพื่อเสียบสกูรติดกับบริเวณที่ ต้องการวัดอุณหภูมิ



รายละเอียด

- 1. โมดูลวัดอุณหภูมิดิจิตอล DS18B20 สำหรับ Arduino -55 ° C \sim 125 ° C
- 2. เอาท์พุทสัญญาณคิจิตอล
- 3. ตรวจจับอุณหภูมิอากาศภายนอก
- 4. ใช้ร่วมกับโครงการ Arduino DIY
- 5. ชิปหลัก: เซ็นเซอร์อุณหภูมิ 18B20
- 6. ช่วงการปรับความละเอียด: 9-12
- 7. รูยึดสำหรับติดตั้งง่ายรูรับแสง: 2.5
- 8. ช่วงวัดอุณหภูมิ: -55 ~ +125 🗆
- 9. ความถูกต้องของการวัดอุณหภูมิ: 0.5 🗆
- 10. แรงคันไฟฟ้าที่ใช้งานได้: DC 5V
- 11. ขนาด: 28 มม. x 12 มม. x 10 มม

้ฝึกทักษะการทดลองกิจกรรมที่ 1 เรียนรู้การใช้งาน Temperature Sensor Module

อุปกรณ์ที่ใช้สำหรับทดลอง



การต่อวงจร



รูป จำลองการต่อวงจร

การเขียนโปรแกรมด้วยภาษา C เพื่อทดสอบอุปกรณ์



ผลลัพธ์ที่ได้

อุณหภูมิที่วัดได้หากมีก่าสูงแสดงว่าอุณหภูมิ และหากมีก่าที่วัดได้มีก่าน้อยแสดงว่าอุณหภูมิต่ำ

Temperature Humidity Sensor

การวัดอุณหภูมิและความชื้นสัมพัทธ์ ปัจจุบันมีเซนเซอร์ชนิดหนึ่งที่ถูกออกแบบมาทำงานนี้ โดยเฉพาะ เซนเซอร์ตัวนั้นคือ DHT11 และ DHT22 ซึ่งทั้งสองเป็นโมดูลที่ทำงานแบบเดียวกันทุกประการ เป็นอุปกรณ์ที่ใช้งาสื่อสารเพียงเส้นเดียวเป็นการสื่อสารสองทิศทางและเป็นการสื่อสารที่ให้ข้อมูลแบบ ดิจิทัล ความแตกต่างกันของโมดูลทั้งสองเป็นดังตาราง

DHT11	DHT22		
THE REAL PROPERTY AND			
1. Ultra low cost	1. Low cost		
2.3 to 5V power and I/O	2.3 to 5V power and I/O		
3.2.5mA max current use during conversion	3.2.5mA max current use during conversion		
(while requesting data)	(while requesting data)		
4. Good for 20-80% humidity readings with 5%	4. Good for 0-100% humidity readings with 2-5%		
accuracy	accuracy		
5. Good for 0-50°C temperature readings $\pm 2^{\circ}$ C	5. Good for -40 to 125°C temperature readings		
accuracy	±0.5°C accuracy		
6. No more than 1 Hz sampling rate (once every	6. No more than 0.5 Hz sampling rate (once every		
second)	2 seconds)		
7. Body size 15.5mm x 12mm x 5.5mm	7. Body size 15.1mm x 25mm x 7.7mm		
8.4 pins with 0.1" spacing	8.4 pins with 0.1" spacing		

จากตารางเปรียบเทียบจะเห็นความแตกต่างของโมดูลวัดอุณหภูมิและความชื้นสัมพัทธ์ทั้งสองเบอร์ ได้พอสมควร หน่วยการเรียนรู้นี้เลือกใช้เบอร์ DHT22 (โค้ดโปรแกรมไม่ต่างกัน) รูปร่างหน้าตาและขนาด ของโมดูลตลอดจนหน้าที่ของแต่ละขาเป็นดังรูป



รูปร่างของโมดูลวัดอุณหภูมิและความชื้นสัมพัทธ์ DHT22

การต่อใช้งานโมดูล DHT11, DHT22 จะต้องใช้ตัวด้านทานพูลอัพที่ขาสัญญาณข้อมูลที่มีค่าเท่ากับ 4.7k ดังรูป



การต่อใช้งานโมดูล DHT11, DHT22

การเขียนโค้คโปรแกรมเพื่ออ่านค่าอุณหภูมิและความชื้นสัมพัทธ์ทั้ง DHT11 และ DHT22 ปัจจุบัน มีไลบรารี่ช่วยงานทำให้ผู้เขียนโค้คสามารถเขียนโค้คไค้ง่ายขึ้น ซึ่งไลบรารี่ที่ต้องเอามาใช้งานเพื่ออ่านค่า จากโมดูลคังกล่าวจะต้องใช้ไลบรารี่ 2 ตัว โคยสามารถคาวน์โหลคได้ที่

ไลบรารี่	แหล่งดาวน์โหลด
DHT.h	https://github.com/adafruit/DHT-sensor-library
Adafruit_Sensor.h	https://github.com/adafruit/Adafruit_Sensor

มีขั้นตอนการดำเนินการเพื่อนำไลบรารี่มาใช้งานดังนี้ 1. ดาวน์โหลดไลบรารี่ตัวที่ 1 ซึ่งเป็นไฟล์ Zip ดังรูป



การดาวน์โหลดไลบรารี่ที่นำมาใช้งานตัวที่ 1

2. ดาวน์โหลด ไลบรารี่ตัวที่ 2 ซึ่งเป็นไฟล์ Zip ดังรูป



การดาวน์โหลดไลบรารี่ที่นำมาใช้งานตัวที่ 2

3. ทำการเพิ่มไลบรารี่ทั้ง 2 ลงในโปรแกรม Arduino IDE โดยการเพิ่มจากไฟล์ zip แล้วทำการ ค้นหาไฟล์ zip ที่ได้จากการดาวน์โหลดในข้อ 1 และ 2



การเพิ่มไลบรารี่ที่เป็นไฟล์ zip ลงในโปรแกรม Arduino IDE

ฟังก์ชันที่เกี่ยวกับโมดูล DHT11, DHT22

 ฟังก์ชันหน่วงเวลาหรือฟังก์ชันหยุดการใช้งานสามารถกำหนดตัวเลขของเวลาที่ต้องการหยุดค้าง โดยตัวเลขที่ใส่เป็นตัวเลขของเวลาหน่วยเป็นมิลลิวินาที ตัวเลขของเวลาที่ใส่ได้สูงสุดคือ 4,294,967,295 ซึ่ง เป็นขนาดของตัวแปร unsigned long



speed: ตัวเลขของอัตราเร็วในการสื่อสารผ่านพอร์ตอนุกรม

 พึงก์ชันส่งข้อมูลออกพอร์ต เป็นพึงก์ชันที่ใช้ในการส่งข้อมูลออกทางพอร์ตอนุกรมหรือพิมพ์ ข้อมูลออกทางพอร์ตเพื่อแสดงผลที่จอกอมพิวเตอร์เมื่อพิมพ์เสร็จตัวเกอร์เซอร์จะรออยู่ที่ท้ายสิ่งที่พิมพ์นั้น ๆ



 4. ฟังก์ชันส่งข้อมูลออกพอร์ต คล้ายกับฟังก์ชัน Serial.print ต่างกันตรงที่เมื่อพิมพ์เสร็จตัวเคอร์เซอร์ จะขึ้นมารอยังบรรทัดใหม่ ดังนั้นเมื่อสั่งพิมพ์ครั้งถัดไปข้อมูลที่ปรากฏจะอยู่ที่บรรทัดใหม่ แทนที่จะต่อท้าย เหมือนกับฟังก์ชั่น Serial.print

Serial.println(val);

Serial.println(val, format);

ฟังก์ชันใช้งานของไลบรารี่ DHT

การอ่านค่าจากโมคูล DHT จำเป็นต้องใช้ไลบรารี่ช่วยงาน ซึ่งไลบรารี่ไม่ได้ถูกเพิ่มเข้ามาในตัว โปรแกรม Arduino IDE ตั้งแต่แรกจำเป็นต้องติดตั้งเพิ่มเติม โดยมีฟังก์ชันให้ใช้งานดังนี้

1. ฟังก์ชันกำหนดขาเชื่อมต่อ ใช้ในการระบุขาที่ใช้เชื่อมต่อให้ตัวโปรแกรมรับรู้ รูปแบบเป็น

ดังนี้

DHT dht(DHTPIN, DHTTYPE);

DHTPIN: ตัวเลขระบุขาพอร์ตที่ใช้เชื่อมต่อกับ โมดูล DHTTYPE: ชนิดของ โมดูลที่เชื่อมต่อ ได้แก่ DHT11, DHT21, DHT22

ตัวอย่าง

DHT dht(8, DHT22);

หมายถึง ต่อไปในโปรแกรมจะใช้ชื่อ dht ในการเรียกใช้งานโมคูล โคยมีการเชื่อมต่อสายสัญญาณ เข้าที่ขาพอร์ต D8 และใช้โมคูลชนิด DHT22

 พึงก์ชันอ่านก่ากวามชื้น ก่าที่ได้จากพึงก์ชั่นอยู่ในรูปของตัวแปร float หน่วยเป็นเปอร์เซนต์ของ กวามชื้นในอากาศที่วัดได้ รูปแบบของพึงก์ชันดังนี้

dht.readHumidity();

 พึงก์ชันอ่านค่าความอุณหภูมิ ค่าที่ได้จากพึงก์ชันอยู่ในรูปของตัวแปร float หน่วยเป็นองศาเซลเซียส รูปแบบของพึงก์ชันดังนี้

dht.readTemperature()

้ฝึกทักษะการทดลองกิจกรรมที่ 2 เรียนรู้การใช้งาน Temperature Humidity Sensor

อุปกรณ์ที่ใช้สำหรับทดลอง



การติดตั้ง Libraries

Temperature Humidity Sensor เป็นอีกชุดเซ็นเซอร์หนึ่งที่จะต้องทำการติดตั้ง Libraries เพื่อการ สร้าง Sketch Code คลิกเมนู Sketch --> Include Library --> Add.ZIP Library... --> []] DHT_tem.zip ดังรูป



การต่อวงจร



การเขียนโปรแกรมด้วยภาษา C เพื่อทดสอบอุปกรณ์



ผลลัพธ์ที่ได้

💿 сом8					
					Send
Status	Humidity (%)	Temperature (C)	(F)		=
OK	81.0	32.0	89.6		
OK	81.0	32.0	89.6		
OK	81.0	32.0	89.6		
OK	81.0	32.0	89.6		
OK	81.0	32.0	89.6		
OK	81.0	32.0	89.6		
OK	81.0	32.0	89.6		
OK	80.0	32.0	89.6		
OK	80.0	32.0	89.6		
OK	81.0	32.0	89.6		
OK	80.0	32.0	89.6		
OK	80.0	32.0	89.6		
OK	80.0	32.0	89.6		
OK	80.0	32.0	89.6		
OK	80.0	32.0	89.6		
OK	80.0	32.0	89.6		
OK	80.0	32.0	89.6		
Autoscr	roll 📄 Show timestamp	[Newline 👻	9600 baud 👻	Clear output

เขียนกำสั่งให้ โมดูล DHT11 เซ็นเซอร์วัดความชื้นและอุณหภูมิขึ้นแสดงบนจอ Serial Monitor ซึ่ง ช่อง Humidity คือ ค่าความชื้นมีหน่วยเป็นเปอร์เซ็นต์ และ ช่อง Temperature มีหน่วยเป็นฟาเรนไฮต์

หน่วยการเรียนรู้ที่ 8 ทำความรู้จักเซ็นเซอร์ เรื่องย่อยที่ 2 IR Receive Module

IR Receiver Module

IR Receiver Module คือ เซ็นเซอร์ตรวจจับแสงอินฟราเรค ซึ่งได้มีการนำไปประยุกต์ใช้ในการ รับส่งข้อมูลด้วยแสงอินฟราเรค (Infrared : IR) โดยถือเป็นอีกวิธีหนึ่งในการสื่อสารข้อมูลแบบไร้สาย การ ส่งข้อมูลด้วยอินฟราเรคที่รู้จักกันดีคือ ระบบรีโมตคอนโทรล ซึ่งแสงอินฟราเรคจะถูกปล่อยออกมาจาก LED ของรีโมตคอนโทรล และเซ็นเซอร์จะรับสัญญาณอินฟราเรคเพื่อรับข้อมูลไปประมวลผล เช่น กครีโมต เพื่อให้หุ่นยนต์ของเล่นเดินหน้า ถอยหลัง การกครีโมตเพื่อเปลี่ยนช่องโทรทัศน์

รีโมตคอนโทรล

รีโมตคอนโทรล คือ เครื่องมืออิเล็กทรอนิกส์ชนิดหนึ่ง ใช้สำหรับควบคุมการคำเนินการของ สิ่งประดิษฐ์หรือเครื่องจักรต่างๆ โดยเฉพาะเครื่องใช้ไฟฟ้าภายในบ้านเช่น โทรทัศน์ เครื่องเสียง เครื่องเล่นดี วีดี จากระยะไกล โดยไม่ใช้สายไฟเป็นตัวส่งสัญญาณแต่ใช้อินฟราเรดแทน (หรือใช้สัญญาณวิทยุแต่พบได้ น้อย) ใช้พลังงานจากแบตเตอรึ่งนาดเล็กไม่กี่ก้อนเท่านั้น มีขนาดเหมาะมือและมีปุ่มฟังก์ชันต่างๆ อยู่ครบ กรัน

รี โมตคอนโทรล เป็นการเรียกย่อมาจาก รี โมตคอนโทรลเลอร์ (remote controller) อีกต่อหนึ่ง และ สามารถเรียกย่อลงได้อีกเหลือเพียง รี โมต (อ่านว่า รี-โหมด) ในภาษาไทยสามารถใช้คำไทยแทนได้ว่า เครื่อง ควบคุมระยะไกล หรือ อุปกรณ์ควบคุมระยะไกล

รี โมทคอนโทรล จะสามารถสั่งงานได้ ต้องประกอบด้วย2สิ่งนี้คือ รหัส และ ตัวส่งสัญญาณ 1.รหัส (Code) เป็นระบบสัญญาณจะนำออกได้ต้องมีตัวคลื่นพานำออกไป 2.ตัวส่งสัญญาณ(Carrier) ตัวรับสัญญาณ เพื่อถอดหรือรับรหัสที่ถูกส่งมาใช้ควบคุมอุปกรณ์ต่างๆ

้ฝึกทักษะการทดลองกิจกรรมที่ 1 เรียนรู้การใช้งาน IR Receive Module อุปกรณ์ที่ใช้สำหรับทดลอง



Infrared Receiver Module KY-022



IR Infrared Remote control arduino



สายไฟจัมเปอร์ ผู้-เมีย ยาว 20cm. จำนวน 3 เส้น



สายไฟจัมเปอร์ ผู้-ผู้ ยาว 20cm. จำนวน 2 เส้น



LED ขนาด 5mm สีแดง จำนวน 1 ควง



ตัวต้านทาน 4.7k 1 ตัว

บอร์ดทดลอง Breadboard 830 Point

การติดตั้ง Libraries

IR Receiver Module เป็นอีกชุดเซ็นเซอร์หนึ่งที่จะต้องทำการติดตั้ง Libraries เพื่อการสร้าง Sketch Code ซึ่งนักเขียนโปรแกรมสามารถดาวน์โหลดลิงก์

http://www.mediafire.com/file/e4rco5njo2gloki/IRremote.zip/file จากนั้น คลิกเมนู Sketch --> Include Library --> Add.ZIP Library... ดังรูป



การต่อวงจร

S = งา Signal เสียบที่ช่อง Pin 11, - = ไฟลบ ต่อไปที่ GND (ground), + (งากลาง) = ไฟ
 บวก ต่อไปที่ 5 โวลต์



การเขียนโปรแกรมด้วยภาษา C เพื่อทดสอบอุปกรณ์

1. เปิดโปรแกรม Arduino IDE





ผลลัพธ์ที่ได้

เมื่อกดปุ่มก็จะปรากฏค่าของปุ่มทาง Serial Monitor ซึ่งแต่ละปุ่มก็จะมีค่าแตกต่างกันออกไป



<u>ประยุกต์ใช้รี โมทคอนโทรลกับหลอด LED</u>

การเขียนคำสั่งควบคุมหลอด LED ผ่านการกดปุ่ม 1 และ 2 โดยต้องต่อสายที่ช่อง Pin GND ออกมา เสียบช่อง GND ใน Breadboard จากนั้นเสียบตัวต้านทาน 220 โอห์ม หลอด LED และเสียบสายสัญญาณ จากหลอด LED ที่ช่อง Pin 2 ดังรูป



จากนั้นจะใช้ IR Remote ควบคุมเปิดปิดไฟ LED โดยให้ ปุ่ม 1 เปิดไฟ ปุ่ม 2 ปิดไฟ ให้นำค่าของ ปุ่มที่ 1 (16753245) มาใส่ไว้ใน ตัวแปร code 1 และ นำค่าของปุ่มที่ 2 (16736925) มาใส่ไว้ใน ตัวแปร code 2 ตามรูป แล้วอัพโหลดโค้ด ลงบอร์ด Arduino อีกครั้ง



ผลลัพธ์ที่ได้

เมื่อกดปุ่ม 1 ที่รี โมท ไฟ LED จะติด

เมื่อกคปุ่ม 2 ที่รี โมท ไฟ LED จะคับ





หน่วยการเรียนรู้ที่ 8 ทำความรู้จักเซนเซอร์ เรื่องย่อยที่ 3 Tracking Sensor

tracking sensor TCRT5000 ตรวจจับเส้นขาวคำ ที่นิยมนำมาใช้คู่กับหุ่นยนต์เดินตามเส้น ตรวจจับ เส้นขาวคำหรือนำมาให้ตรวจจับวัตถุข้างหน้า หลักการทำงานของ sensor TCRT5000 ตรวจจับเส้นขาวคำ จะใช้การสะท้อนกลับของแสง ถ้ามีวัตถุ สีขาวจะสะท้อนแสงได้ดี sensor TCRT5000 ถ้าเป็นวัตถุสีคำ sensor TCRT5000 จะไม่สะท้อนแสง ค่าที่ส่งออกมาจะแตกต่างกัน

รายละเอียด

- 1. ใช้เซ็นเซอร์สะท้อนแสงอินฟราเรค TCRT5000
- 2. การตรวจจับระยะการสะท้อน: 1 มิลลิเมตร 25 มิลลิเมตร
- 3. แรงคันไฟฟ้าที่ใช้งาน: 5V
- 4. รูปแบบผลลัพธ์: สัญญาณดิจิตอล (0 และ 1)
- 5. บอร์ด PCB ขนาดเล็ก: 3.5 ซม. x 1 ซม
- 6. น้ำหนักเดียว: 4.5g

อุปกรณ์มีดังนี้



บอร์ดทดลอง Breadboard 830 Point

LED ขนาด 5mm สีแดง จำนวน 5 ควง

สายจัมเปอร์ ผู้-ผู้ จำนวน 7 เส้น





tracking sensor tracking module

ต่อวงจร



กำสั่งควบคุมบอร์ค Arduino Uno R3

38_3 Arduino 1.8.9
File Edit Sketch Tools Help
83§
1 int led = 2;
2 int a_p = A5; //ประกาศตัวแปร ให้ analogPin แทนซา analog ซาที่A5
3 int val = 0;
4 void setup() {
5 pinMode(led, OUTPUT); // sets the pin as output
6 Serial.begin(9600);
7 }
8
9 void loop() {
10 val = analogRead(a_p); //อานคาสัญญาณ analog ขา5 ที่ต่อกับ TCRT5000
11 Serial.print("val = "); // พิมพขอมความส่งเขาคอมพิวเตอร "val = "
12 Serial.println(val); // พิมพคาของตัวแปร val
13 if (val > 500) { // คา 100 สามารถกำหนดปรับไดตามคาแส่งในหองตางๆ
14 digitalWrite(led, HIGH); // สั่งให LED ดิดสวาง
15 }
16 else {
17 digitalWrite(led, LOW); // สังให LED ดับ
18 }
19 delay(100);
20 }
21 1
Done compiling.

จากนั้นเปิด Serial Monitro 🔟 ที่ด้านขวามือบนเพื่อดูตัวเลขที่รับเข้ามาและหลอด LED แสดงผล

6	Arduino/Genuino Uno
oo Arduino_TCRT5000LED Ard	duino 1.6.6
File Edit Sketch Tools Help	
	💿 COM6 (Arduino/Genuino Uno)
Arduino_TCRT5000LED	
int analogrin - 5, //Die	val = 1016
unid setup() /	val = 1016
ninMode (ledPin OUTPUT	val = 1016
Serial.begin (9600):	val = 1016
1	val = 1016
	val = 1017
void loop() {	val = 1016
val = analogRead(analo	val = 1017
<pre>Serial.print("val = ")</pre>	val = 1016
Serial.println(val); /	val = 1016
if (val > 500) { // ค่า	val = 1017
digitalWrite(ledPin,	val = 1017
}	val = 1016
else {	val = 1016
digitalWrite (ledPin,	val = 1016
}	
delay(100);	Autoscroll Both

จะเห็นว่าถ้านำเอา Sensor จ่อกับกระคาษสีขาว Sensor TCRT5000 จะส่งค่า analog มากกว่า 500 ไฟ LED จะคับ ถ้านำเอา Sensor จ่อกับกระคาษที่มีหมึกสีคำ Sensor TCRT5000 จะส่งค่า analog น้อยกว่า 500 ไฟ LED จะติดสว่าง

จะสังเกตเห็นว่า Sensor จะใช้หลักการสะท้อนกลับของแสง กระดาษสีดำจะดูดกลืนแสง กระดาษสี ขาว sensor จะสะท้อนแสงได้ดี ทำให้ค่าที่อ่านได้ต่างกัน บนตัว sensor ที่ค่า สัญญาณ analog A0 และ ค่า สัญญาณ digital D0 และ มีตัวปรับความไวกล่องสีฟ้า ใช้กับขา D0 ให้เราเชื่อมต่อใช้กับ Arduino ได้ง่าย

หน่วยการเรียนรู้ที่ 8 ทำความรู้จักเซ็นเซอร์ เรื่องย่อยที่ 4 Buzzer Module

Passive Buzzer Module



Passive Buzzer Module พาสซีฟบัสเซอร์ เป็นอุปกรณ์ทำหน้าที่คล้ายลำโพง เพียงแต่ออกแบบ มาเพื่อใช้ในการส่งสัญญาณเสียงความถี่สูงๆ บนบอร์คมีทรานซิสเตอร์สำหรับช่วยขับมาให้แล้ว สามารถต่อ ใช้งานกับไมโครคอนโทรลเลอร์ได้โดยตรง การใช้งานเพียงเขียนโค้คสร้างสัญญาณ HIGH LOW สลับกัน ไปมา หรือใน Arduino สามารถใช้ฟังก์ชัน tone() ได้เลย ข้อคีของบัสเซอร์แบบพาสซีฟคือ สามารถกำหนด ความถี่เสียงที่ต้องการได้เอง

บอร์ค Buzzer แบบ Passive มีวงจรขยายสัญญาณทำให้เสียงคังมาก ใช้งานง่าย เพียงเท่านี้ Arduino ก็สามารถร้องเพลงได้ตามที่เราต้องการ ทำงานแบบ Active Low

ข้อมูลเพิ่มเติม Active Buzzer

โมดูล USES 9012 ใครเวอร์ทรานซิสเตอร์ แรงคันไฟฟ้าที่ทำงาน 3.3 - 5 V กระคาน PCB ขนาดเล็กเจ็คชิ้น: 3.3 ซม. * 1.3 ซม **ฝึกทักษะการทดลองกิจกรรมที่ 1 เรียนรู้การใช้งาน Passive Buzzer Module**

อุปกรณ์ที่ใช้สำหรับทดลอง



Passive Buzzer Module



สายจัมเปอร์ ผู้-เมีย 3 เส้น

ขา VCC -> 3.3-5V

ขา Gnd-> Gnd

ขา I/O -> 13





ผลลัพธ์ที่ได้

Buzzer Module ส่งเสียงออกมาและคับ สลับกัน 1 วินาที

บทที่ 8 ทำความรู้จักเซ็นเซอร์ เรื่องย่อยที่ 5 Switch Button Module

ทฤษฎีพื้นฐาน

สวิตช์ที่นำมาใช้งานกับไมโครคอนโทรลเลอร์จะเป็นสวิตช์ชนิดกดติดปล่อยคับเมื่อนำมาใช้งาน จำเป็นจะต้องใช้พอร์ตในการเชื่อมต่อ 1 ช่องต่อสวิตช์ 1 ตัว



การต่อใช้งานสวิตช์กับพอร์ตดิจิทัล

อ่านก่าจากพอร์ตแอนาลอกที่ผ่านมาเป็นการเรียนรู้เกี่ยวกับการใช้งานพอร์ตที่รับสัญญาณเข้าที่เป็น แอนาลอกซึ่งสามารถแยกความแตกต่างได้ 1024 ระดับในช่องรับเพียงช่องเดียว ดังนั้นหากผู้ใช้งานออกแบบ วงจรสวิตช์ให้ต่อวงจรที่สามารถให้แรงดันที่แตกต่างกันได้เมื่อกดสวิตช์แต่ละตัว (ใช้วงจรแบ่งแรงดัน) ก็ สามารถใช้งานสวิตช์หลายตัวในพอร์ตรับสัญญาณเพียงช่องเดียวได้ ดังรูป



การต่อใช้งานสวิตช์กับพอร์ตแอนาลอก

ฟังก์ชัน Arduino ที่ใช้

 ฟังก์ชันกำหนด โหมดการทำงานให้กับงาพอร์ต สามารถกำหนด ได้ทั้งงาดิจิทัล โดยใส่เพียง ตัวเลงงองงา (0, 1, 2,...13) และงาแอนาลอกที่ต้องการให้ทำงานในโหมดดิจิทัลแต่การใส่งาต้องใส่ A นำ หน้าซึ่งใช้ได้เฉพาะ A0, A1,...A5 ส่วนขา A6 และ A7 ไม่สามารถใช้งานในโหมดดิจิทัลได้ รูปแบบของ ฟังก์ชันเป็นดังนี้

pinMode(pin, mode)

pin: หมายเลขงาที่ต้องการเซตโหมด mode: INPUT, OUTPUT, INPUT PULLUP

2. ฟังก์ชันส่งค่าลอจิกดิจิทัลไปยังขาพอร์ต ค่า HIGH เป็นการส่งลอจิก 1 และค่า LOW เป็นการส่ง ลอจิก 0 ออกไปยังขาพอร์ต ซึ่งฟังก์ชันนี้จะทำงานได้ต้องมีการใช้ฟังก์ชัน pinMode ก่อน

digitalWrite(pin, value)

pin: หมายเลขขาที่ต้องการเขียนลอจิกออกพอร์ต

 ฟังก์ชันหน่วงเวลาหรือฟังก์ชันหยุดค้าง การใช้งานสามารถกำหนดตัวเลขของเวลาที่ต้องการหยุด ค้างโดยตัวเลขที่ใส่เป็นตัวเลขของเวลาหน่วยเป็นมิลลิวินาที ตัวเลขของเวลาที่ใส่ได้สูงสุดคือ 4,294,967,295 ซึ่งเป็นขนาดของตัวแปร unsigned long

delay(ms)

ms: ตัวเลขที่หยุดค้างของเวลาหน่วยมิลลิวินาที (unsigned long)

4. ฟังก์ชันกำหนดความเร็วในการสื่อสารทางพอร์ตอนุกรม

delay(ms)

speed: ตัวเลขของอัตราเร็วในการสื่อสารผ่านพอร์ตอนุกรม

5. พึ่งก์ชันส่งข้อมูลออกพอร์ต เป็นพึ่งก์ชันที่ใช้ในการส่งข้อมูลออกทางพอร์ตอนุกรมหรือพิมพ์ ข้อมูลออกทางพอร์ตเพื่อแสดงผลที่จอกอมพิวเตอร์เมื่อพิมพ์เสร็จตัวเกอร์เซอร์จะรออยู่ที่ท้ายสิ่งที่พิมพ์นั้น ๆ

Serial.print(val)

Serial.print(val, format)

6. ฟังก์ชันส่งข้อมูลออกพอร์ต คล้ายกับฟังก์ชัน Serial.print ต่างกันตรงที่เมื่อพิมพ์เสร็จตัวเคอร์เซอร์ จะขึ้นมารอยังบรรทัดใหม่ ดังนั้นเมื่อสั่งพิมพ์ครั้งถัดไปข้อมูลที่ปรากฏจะอยู่ที่บรรทัดใหม่ แทนที่จะต่อท้าย เหมือนกับฟังก์ชัน Serial.print



 พึงก์ชันอ่านสัญญาณแอนาลอก เป็นพึงก์ชันที่อ่านสัญญาณแอนาลอกที่ปรากฏอยู่ที่งาพอร์ตแอ นาลอกที่ต้องการอ่านนั้น ๆ ค่าที่อ่านได้จะอยู่ในช่วง 0-1023 สำหรับแรงคันของสัญญาณแอนาลอกที่ 0-5V ดังนั้น ต้องใช้ตัวแปรที่เป็น int สำหรับเก็บค่าที่อ่านได้

analogRead(pin)

pin: ขาพอร์ตแอนาลอกที่ต้องการอ่านค่าสัญญาณแอนาลอก

วงจรที่ต่อเข้ากับบอร์ด Arduino ดังรูป



รูปวงจรที่ใช้บอร์ค Arduino ในการทคลอง

การต่อวงจร



รูปการต่อลงบอร์คทคลอง

การเขียนโปรแกรมด้วยภาษา C เพื่อทดสอบอุปกรณ์

เขียนโปรแกรมอ่านก่าแอนาลอกจากการกดสวิตช์แต่ละตัวโดยแสดงผลที่จอกอมพิวเตอร์ผ่านทาง พอร์ตอนุกรม



ผลลัพธ์ที่ได้

เขียนคำสั่งการอ่านค่าแอนาลอกจากวงจรสวิตช์โดยนำเข้ามาเก็บไว้ในตัวแปร adc จากนั้นแสดงค่า แอนาลอกจากการกดสวิตช์ผ่านทางพอร์ตอนุกรม
หน่วยการเรียนรู้ที่ 8 ทำความรู้จักเซ็นเซอร์ เรื่องย่อยที่ 6 Light Cup Module

Arduino KY-027 Magic Light Cup Module เป็นชุดบอร์ดที่รวม 2 บอร์ดเข้าด้วยกัน ซึ่งแต่ละโมดูล จะมีสวิตช์ไฟและปรอทเอียง การใช้ PWM เพื่อขับเคลื่อนไฟ LED ในแต่ละโมดูลสามารถเซ็นเซอร์และรับ ก่าแสงได้แม่นยำจากโมดูลหนึ่งไปยังอีกโมดูลหนึ่งขณะเอียงเข้ามากัน ซึ่งตัวบอร์ดประกอบด้วยขาดังรูป

KY-027 เป็นกู่ของโมคูลแต่ละอันประกอบด้วยสวิตช์ปรอทไฟ LED และตัวด้านทาน 10 kΩ แต่ละ บอร์คมีพินเพื่อควบคุมสวิตช์ไฟ LED กำลังไฟและกราวด์

<u>แรงดันไฟฟ้าที่ใช้งาน</u>	3.3V ถึง 5.5V
<u>ขนาดคณะกรรมการ</u>	1.5 ซม. x 3.6 ซม. [0.6in x 1.4in]

รหัสตัวอย่าง KY-027 Arduino

ในร่าง Arduino นี้เราจะใช้ทั้งโมดูล KY-027 เพื่อสร้างเอฟเฟกต์แสงไฟวิเศษ สวิตช์ปรอทในแต่ละ โมดูลให้สัญญาณดิจิตอลที่ใช้ในการควบคุมความสว่างของไฟ LED โดยใช้ PWM การเอียงโมดูลจะลด ความสว่างของโมดูลหนึ่งขณะที่เพิ่มโมดูลอีกอันทำให้เกิดภาพลวงตาของแสงที่ผ่านจากโมดูลหนึ่งไปยังอีก โมดูลหนึ่งอย่างน่าอัศจรรย์



้ฝึกทักษะการทดลองกิจกรรมที่ 1 เรียนรู้การใช้งาน

อุปกรณ์ที่ใช้สำหรับทดลอง



บอร์คทคลอง Breadboard 400 Point



สายไฟจัมเปอร์ ผู้-เมีย ยาว 20cm. จำนวน 4 เส้น



Magic light cup module KY-027

การต่อวงจร



G เสียบช่อง Pin GND

+ เสียบช่อง Pin 5V

S (สวิตช์ปรอท) เสียบช่อง Pin 4

L (ไฟ LED) เสียบช่อง Pin 5

การเขียนโปรแกรมด้วยภาษา C เพื่อทดสอบอุปกรณ์



ผลลัพธ์ที่ได้



ขอบคุณเว็บไซต์ https://www.arduinoall.com/product/338/magic-light-cup-module-ky-027

จะสังเกตเห็นว่าเมื่อสวิตช์ปรอทเอียงมาชนสวิตช์กรบวงจร ไฟ LED จะติด และเมื่อสวิตช์ปรอทไม่ กรบวงจรไฟ LED จะก่อย ๆ จางและดับ เหมือนเทถ้วยแสงกิ้ง กว่ำถ้วยแสงก็จะหาย

หน่วยการเรียนรู้ที่ 8 ทำความรู้จักเซ็นเซอร์ เรื่องย่อยที่ 7 Light Blocking sensor

Light Blocking sensor หรือ ROHM four-way sensor RPI-1031 เป็นเซนเซอร์ตรวจจับความเอียง/ การหมุน ตรวจจับได้ 4 ทิศทาง ด้านหน้า ด้านหลัง ด้านซ้าย ด้านขวา ให้สัญญาณเป็นแบบ analog และใช้ ไฟเลี้ยง 3.3-5V

้ฝึกทักษะการทดลองกิจกรรมที่ 1 เรียนรู้การใช้งาน Light Blocking sensor

อุปกรณ์ที่ใช้สำหรับทดลอง



Light Blocking sensor



สายไฟจัมเปอร์ ผู้-เมีย ยาว 20cm. จำนวน 4 เส้น

การต่อวงจร



รูป จำลองการต่อวงจร Light Blocking sensor

การเขียนโปรแกรมด้วยภาษา C เพื่อทดสอบอุปกรณ์



เปิดดูใน Serial Monitor เมื่อเอียงไปที่มุมต่าง ๆ จะเห็นค่าเซนเซอร์ที่วัดได้



ขอบคุณเว็บไซต์ https://www.arduinoall.com/product/3385/rpi-1031- -เซนเซอร์ตรวจจับความเอียง-4-แกน-

rpi-1031-angle-sensor-four-direction-sensor-4-direction-sens

หน่วยการเรียนรู้ที่ 8 ทำความรู้จักเซนเซอร์ เรื่องย่อยที่ 8 Relay และ Motion Sensor Module

8.1 Relay

รีเลย์ (Relay) เป็นอุปกรณ์ไฟฟ้าชนิดหนึ่ง ซึ่งทำหน้าที่ตัดต่อวงจรแบบเดียวกับสวิตช์ โดยควบคุม การทำงานด้วยไฟฟ้า Relay มีหลายประเภท ตั้งแต่ Relay ขนาดเล็กที่ใช้ในงานอิเล็กทรอนิกส์ทั่วไป จนถึง Relay ขนาดใหญ่ที่ใช้ในงานไฟฟ้าแรงสูง โดยมีรูปร่างหน้าตาแตกต่างกันออกไป แต่มีหลักการทำงานที่ กล้ายกลึงกัน สำหรับการนำ Relay ไปใช้งาน จะใช้ในการตัดต่อวงจร ทั้งนี้ Relay ยังสามารถเลือกใช้งานได้ หลากหลายรูปแบบ



สัญลักษณ์ในวงจรไฟฟ้าของรีเลย์

ภายใน Relay จะประกอบไปด้วยงคลวดและหน้าสัมผัส

หน้าสัมผัส NC (Normally Close) เป็นหน้าสัมผัสปกติปิค โคยในสภาวะปกติหน้าสัมผัสนี้จะต่อเข้า กับขา COM (Common) และจะลอยหรือไม่สัมผัสกันเมื่อมีกระแสไฟฟ้าไหลผ่านขคลวค

หน้าสัมผัส NO (Normally Open) เป็นหน้าสัมผัสปกติเปิด โดยในสภาวะปกติจะลอยอยู่ ไม่ถูกต่อ กับขา COM (Common) แต่จะเชื่อมต่อกันเมื่อมีกระแสไฟไหลผ่านขดลวด

ขา COM (Common) เป็นขาที่ถูกใช้งานร่วมกันระหว่าง NC และ NO ขึ้นอยู่กับว่า ขณะนั้นมี กระแสไฟฟ้าไหลผ่านขดลวดหรือไม่ หน้าสัมผัสใน Relay 1 ตัวอาจมีมากกว่า 1 ชุด ขึ้นอยู่กับผู้ผลิตและ ลักษณะของงานที่ถูกนำไปใช้ จำนวนหน้าสัมผัสถูกแบ่งออกดังนี้

สวิตช์จะถูกแขกประเภทตามจำนวน Pole และจำนวน Throw ซึ่งจำนวน Pole (SP-Single Pole, DP-Double Pole, 3P-Triple Pole, etc.) จะบอกถึงจำนวนวงจรที่ทำการเปิด-ปิด หรือ จำนวนของขา COM นั่นเอง และจำนวน Throw (ST, DT) จะบอกถึงจำนวนของตัวเลือกของ Pole ตัวอย่างเช่น SPST- Single Pole Single Throw สวิตช์จะสามารถเลือกได้เพียงอย่างเดียว โดยจะเป็นปกติเปิด (NO-Normally Open) หรือ ปกติปิค (NC-Normally Close) แต่ถ้าเป็น SPDT- Single Pole Double Throw สวิตช์จะมีหนึ่งคู่เป็นปกติเปิค (NO) และอีกหนึ่งคู่เป็นปกติปิคเสมอ (NC) ดังรูปด้านล่าง



ขอบกุณเว็บไซต์ https://www.thaieasyelec.com/article-wiki/review-product-article/ Example-project-for-control-electrical-device-using-arduino-and-relay-module.html

จากส่วนประกอบข้างต้นที่ได้กล่าวไป ในบทความนี้เราจะใช้งาน Relay แบบ SPDT (Single Pole Double Throw) หลักการทำงานของ Relay นั้น ในส่วนของขคลวค เมื่อมีกระแสไฟฟ้าไหลผ่าน จะทำให้ ขคลวคเกิดการเหนี่ยวนำและทำหน้าที่เสมือนแม่เหล็กไฟฟ้า ส่งผลให้ขา COM ที่เชื่อมต่ออยู่กับหน้าสัมผัส NC (ในสภาวะที่ยังไม่เกิดการเหนี่ยวนำ) ย้ายกลับเชื่อมต่อกับหน้าสัมผัส NO แทน และปล่อยให้ขา NC ลอย เมื่อมองที่ขา NC กับ COM และ NO กับ COM แล้วจะเห็นว่ามีการทำงานติด-ดับลักษณะคล้ายการทำงาน ของสวิชต์ เราสามารถอาศัยกุณสมบัตินี้ไปประยุกต์ใช้งานได้

ในบทความนี้ เราจะกล่าวถึงวิธีการนำ Relay Module ไปประยุกต์ใช้งานจริง แต่ก่อนอื่นเรามาคูวิธี อ่านกุณสมบัติของ Relay ว่าสามารถรองรับการทำงานที่แรงคันและกระแสไฟฟ้าเท่าไร ใช้แรงคันไฟฟ้าใน การทำงานอย่างไรก่อนนะครับ



ในหน่วยการเรียนรู้นี้จะศึกษา relay 2 channel ดังภาพ



บอร์ค Arduino relay 2 channel สำหรับ ควบคุมเปิด/ปิด Relay Module ใค้ 2 ช่อง ใช้ไฟเข้า 5 โวลต์ เหมาะสำหรับ Arduino การควบคุม ส่งสัญญาณควบคุมแบบ Active Low ใช้ Arduino ควบคุม arduino relay

- ถ้าต้องการให้ รีเลย์ติดส่งสัญญาณ 0 ไป

- ถ้าต้องการให้ดับส่งสัญญาณ 1 ไป

วงจร Arduino relay module เป็นแบบ แยกกราวค์ Opto isolated Relay ปลอคภัยต่อวงจร

ใมโครคอนโทรเลอร์ และ Arduino

<u>รายละเอียด Arduino Relay module บอร์ด รีเลย์ 2 ช่อง สำหรับ Arduino</u>

การเชื่อมต่อ Arduino Relay module มาตรฐานที่สามารถใช้ควบคุมได้โดยตรงจาก
 ใมโครคอนโทรนเลอร์ เช่น Arduino, 8051, AVR, PIC, DSP, ARM, ARM, MSP430, TTL logic)

2. ใช้ไฟฟ้าที่ 5 โวลต์

- 3. Relay Module สามารถ 5 โวลต์จากบอร์ค arduino ที่มีขา 5 โวลต์ได้
- 4. ใช้ควบคุมไฟฟ้าแรงสูงได้ที่ DC30V 10A , AC250V 10A

- 5. มีไฟบอกสถานะการทำงานของ Arduino Relay รีเลย์ทุกช่อง
- 6. เชื่อมต่อด้วยขั้วสกรู ทำให้ติดตั้งได้ง่ายและสะดวก
- 7. ใช้กระแสขับ relay แต่ละตัวที่ 15-20 mA
- 8. การส่งสัญญาณควบคุมรีเลย์เป็นแบบ Active low
- 9. วงจรขับรีเลย์เป็นแบบแยกกราวค์ Opto isolated Relay ปลอคภัยต่อวงจรไมโครคอนโทรเลอร์

Arduino Relay

้ฝึกทักษะการทดลองกิจกรรมที่ 1 เรียนรู้การใช้งาน Relay 2

อุปกรณ์ที่ใช้สำหรับทดลอง



การต่อวงจร





- GND --> ใช้สายจัมแปอร์สีเขียว ต่อเข้ากับช่อง Pin GND
- IN1 --> ใช้สายจัมแปอร์สีเหลือง ต่อเข้ากับช่อง Pin 8
- IN2 --> ใช้สายจัมแปอร์สีเหลือง ต่อเข้ากับช่อง Pin 9
- VCC --> ใช้สายจัมแปอร์สีเหลือง ต่อเข้ากับช่อง Pin 5v

💿 sketch_sep20a | Arduino 1.8.9 \times File Edit Sketch Tools Help Ø sketch_sep20a§ 1 int relay1 = 8; 2 int relay2 = 9; 3 void setup() 4 { 5 Serial.begin(115200); pinMode(relay1, OUTPUT); 6 7 pinMode(relay2, OUTPUT); 8 } 9 10 void loop() 11 { 12 digitalWrite(relayl, HIGH); 13 digitalWrite(relay2, LOW); //relay actice LOW Serial.println("^MW relayl ທັນ ແລະ relay2 ຕິດ"); 14 15 delay(2000); digitalWrite(relay2, HIGH); 16 digitalWrite(relayl, LOW); 17 Serial.println("ໃฟ relay2 ດັນ ແລະ relayl ຕິດ"); 18 19 delay(2000); digitalWrite(relay2, LOW); 20 digitalWrite(relayl, LOW); 21 22 Serial.println(""NW relayl ທິດ ແລະ relay2 ທິດ"); 23 delay(3000); 24 digitalWrite(relay2, HIGH); digitalWrite(relay1, HIGH); 25 26 Serial.println("ใฟ relayl ดับ และ relay2 ดับ"); 27 delay(3000); 28 } 29

การเขียนโปรแกรมด้วยภาษา C เพื่อทดสอบอุปกรณ์

ผลลัพธ์ที่ได้

Relay เงื่อนไขเป็นจริงที่ LOW หรือ Active LOW การทคลองวงจรนี้คือ ให้วงจรทำงานวนซ้ำโดย เริ่มคำสั่งคือ

relay1 (IN1) คับ และ relay2 (IN2) ติด เป็นเวลา 2 วินาที

relay1 (IN2) ดับ และ relay1 (IN2) ติด เป็นเวลา 2 วินาที

relay1 (IN1) ติด และ relay2 (IN2) ติด เป็นเวลา 3 วินาที

relay1 (IN1) คับ และ relay2 (IN2) คับ เป็นเวลา 3 วินาที

8.2 Motion Sensor Module

PIR Motion Sensor ใช้สำหรับตรวจจับความเคลื่อนไหวจากความร้อน เช่น สิ่งมีชีวิต เมื่อมีคนเดิน ผ่าน motion sensor switch ก็จะจับค่าความร้อนที่เปลี่ยนแปลง แล้วส่งค่าสัญญาณมีไฟ ออกมา ในกรณีที่ไม่ มีการเปลี่ยนแปลงส่งค่าไฟ สัญญาณ ov ออกมา เราสามารถนำค่านี้ไปสั่งควบคุม Arduino ได้ สามารถปรับ เวลาหน่วงเวลาในการตรวจจับครั้งต่อไปได้ ปรับระยะทางการตรวจจับได้ 3-7 เมตร มีช่องให้ต่อ LDR เพิ่ม เพื่อให้ทำงานตรวจจับแค่ตอนกลางคืน

งา OUT จะให้เอาต์พุตออกมาในรูปของดิจิตอล สามารถปรับลักษณะของสัญญาณขาออกได้จาก การเปลี่ยนจั้มเปอร์ มีรูปแบบสัญญาณเอาต์พุต 2 แบบ คือ

สัญญาณแบบคลื่นพัลส์ต่อเนื่อง

สัญญาณลอจิก 1 ค้างไว้ จนกว่าจะไม่สามารถจับความเคลื่อนไหวได้ จึงจะกลับมาเป็นลอจิก 0 ปุ่ม ปรับ Delay Time Adjust ใช้ปรับการหน่วงเวลาตรวจจับ หากปรับไวมาก จะทำให้ค้างสถานะ 1 ไว้นาน เนื่องจากการตรวจจับไม่ให้ผลแบบทันทีทันใด เมื่อตรวจจับพบ จะให้สถานะทางเอาต์พุตเป็น 1 ค้างไว้ เวลา ที่ค้างไว้ขึ้นอยู่กับการปรับ Delay Time Adjust เมื่อหมดเวลาจึงตรวจจับอีกครั้ง หากตรวจจับไม่เจอก็จะให้ สถานะเป็น 0 แต่หากจับเจอ จะค้างสถานะ 1 ไว้

รายละเอียด PIR เซนเซอร์ตรวจจับความเคลื่อนใหว Motion Sensor Module

Motion sensor switch Dimension: 3.2cm x 2.4cm x 1.8cm (approx) Infrared sensor switch with control circuit board The sensitivity and holding time can be adjusted Working Voltage Range: DC 3V- 5V Current drain:<60uA Voltage Output: High/Low level signal:3.3V TTL output Detection distance: 3--7M(can be adjusted) Detection range: <140° Delay time: 5-200S(can be adjusted, default 5s +-3%) Blockade time: 2.5 S (default) Trigger: L: Non-repeatable trigger H: Repeat Trigger (default) Work temperature:-20-+80°C Trigger Method: L unrepeatable trigger / H repeatable trigger

้ฝึกทักษะการทดลองกิจกรรมที่ 1 เรียนรู้การใช้งาน Motion Sensor Module อุปกรณ์ที่ใช้สำหรับทดลอง



Motion Sensor Module



สายไฟจัมเปอร์ ผู้-เมีย ยาว 20cm. จำนวน 4 เส้น

หลอดไฟ LED 1 หลอด

199

การต่อวงจร



รูป จำลองการต่อวงจร Motion Sensor Module

การเขียนโปรแกรมด้วยภาษา C เพื่อทดสอบอุปกรณ์

🥯 Motion Arduino 1.8.9	-	×
File Edit Sketch Tools Help		
		ø
Motion		
<pre>1 int ledPin = 3;</pre>		^
<pre>2 int inputPin = 2;</pre>		
3 void setup()		
4 {		
<pre>5 Serial.begin(9600);</pre>		
<pre>6 pinMode(ledPin, OUTPUT);</pre>		
7 }		
8		
9 void loop()		
10 []		
12 Serial printlp(Value);		
13		
14 if (Value == 1)		
15 {		
<pre>16 digitalWrite(ledPin, HIGH);</pre>		
17 delay(2000);		
18 }		
19 else		
20 {		
<pre>21 digitalWrite(ledPin,LOW);</pre>		
22 }		
23 }		~
Done compiling.		
Skotah wasa 2220 butoa (68) of program ator		 A North

ผลลัพธ์ที่ได้

ค่า Motion Sensor Module วัคได้คือ ค่าดิจิตอล 0 และ 1 ถ้า Motion Sensor Module จับความ เคลื่อนไหวจากความร้อนได้หรือค่าที่เซนเซอร์ได้เท่ากับ 1 ให้หลอดไฟ LED ติด เป็นเวลา 2 วินาที

หน่วยการเรียนรู้ที่ 8 ทำความรู้จักเซ็นเซอร์

เรื่องย่อยที่ 9 Temperature Humidity Sensor และ LED 4 Digit Display Module

การทดลองระหว่าง Temperature Humidity Sensor และ LED 4 Digit Display Module

1. Temperature Humidity Sensor

DHT11 Module (DHT11 Humitdity and Temperature Sensor) การวัดอุณหภูมิและความชื่นใน อากาศ



+ คือ ขา ไฟ (VCC) ต่อเข้ากับ VCC บนบอร์ค Arduino out คือ ขา DATA ต่อเข้ากับ I/O บนบอร์ค Arduino - คือ ขากราวค์ ต่อเข้ากับ GND บนบอร์ค Arduino

คุณสมบัติ

1. ย่านวัดกวามชื้น 20-90% RH (Relative humidity) โดยมีก่ากวามแม่นยำ +- 5% RH กวามละเอียด ในการวัด 1 % แสดงผลแบบ 8 บิต

 2. ย่านวัดอุณหภูมิ 0 -50 องศาเซลเซียส โดยมีค่าความแม่นย่า +- 2 องศาเซลเซียส ความละเอียดใน การวัด 1 องศาเซลเซียส แสดงผลแบบ 8 บิต

- 3. กินกระแส 0.5 2.5 mA (ขณะทำการวัดค่า) ที่ระดับแรงคัน 3 5.5 VDC
- 4. อ่านค่าสัญญาณ (Sample Rate) ทุก 1 วินาที
- 5. ในการต่อวัดแบบปกติ คือ ระยะห่างระหว่าง Sensor กับตัว Arduino ห่างกันไม่เกิน 20 เมตร
- 6. โดยใช้แหล่งจ่ายแรงคันขนาด 3-5.5 VDC

การส่งข้อมูลของ DHT11 ใช้การสื่อสารกับ MCU ของเราด้วยวิธี Single-wire Two-way Serial interface คือ การสื่อสารอนุกรมสองทางโดยใช้สายเส้นเดียว นั่นหมายความว่า การสื่อสารแบบนี้จะใช้สาย สื่อสารเพียงเส้นเดียวและส่งข้อมูลได้ทั้งจาก MCU ไปที่ตัว DHT11 และในทางกลับกันก็ได้ ในการสื่อสารโดยใช้สายเส้นเดียวนั้นจำเป็นต้องใช้โปรโตกอลที่ตกลงกันไว้ระหว่างตัว MCU และ อุปกรณ์ที่ต้องการสื่อสารด้วย ซึ่งในขั้นแรก บอร์ด Arduino จะส่ง Start signal ที่เป็นแรงคันไฟฟ้าระดับต่ำ อย่างน้อย 18 ไมโกรวินาที ไปที่ DHT11 จากนั้น บอร์ด Arduino จะรอไป 20-40 ไมโกรวินาทีเพื่อรอ DHT11 ตอบกลับ

DHT11 จะส่งแรงคันระคับต่ำกลับซึ่งใช้เวลาส่งกลับนาน 80 ไมโครวินาที เพื่อให้ บอร์ค Arduino รู้ ว่า DHT11 พร้อมแล้ว จากนั้นจะรออีก 80 ไมโครวินาที ก่อนที่จะส่งข้อมูลบิตแรก



<u>การส่งบิต "0" กับ บิต "1"</u>

การส่งบิตเป็น "0" ตัว DHT11 จะดึงระดับแรงดันลงต่ำนาน 50 ไมโครวินาที และปล่อยเป็นระดับ "สูง" นาน 26-28 ไมโครวินาที ดังรูปด้านบน (สังเกตช่วง Sending 0)

การส่งข้อมูลเป็น "1" ตัวส่งจะคึงสายสัญญาณลงระดับต่ำ 50 ไมโครวินาที และปล่อยให้เป็น ระดับสูงนาน 70 ไมโครวินาที (สังเกตช่วง Sending 1)

ข้อควรรู้

ส่งข้อมูลแบบนี้จนครบ 1 ชุค ซึ่งในแต่ละชุดของข้อมูลที่ส่งมาจาก DH11 ตัว MCU รับข้อมูลแล้ว จะต้องเอามาแปลงต่อนะครับ ว่าข้อมูลที่ส่งมามันแปลว่าอะไร แต่ละชุดข้อมูลจะยาว 40 บิต และใช้เวลาส่ง ประมาณ 40 มิลลิวินาที

ใน 40 บิตที่ส่งมา ประกอบด้วย " 8bit integral RH data + 8bit decimal RH data + 8bit integral T data + 8bit decimal T data + 8bit check sum" ซึ่งการตีความหมายของการรับ-ส่ง ข้อมูลนี้จะอยู่ในชุดคำสั่ง library เรียบร้อยแล้ว

ข้อควรรู้

DHT มีหลายรุ่น library ก็เขียนมาให้ใช้ได้หลายรุ่น แต่ก่าพารามิเตอร์ในการวัดก็ต่างกัน ฉะนั้นจึง ควรเลือกรุ่นให้ถูกใน sketch หรือเมื่อทำการคอมไพล์โปรแกรม ซึ่ง library ในตอนนี้ประกอบด้วย DHT.cpp DHT.h DHT_U.cpp และ DHT_U.h

2. LED 4 Digit Display Module

โมดูลแสดงผลตัวเลข 4 หลัก แบบมีจุดนาฬิกาคั่น หน้าจอ 0.36" ใช้ไฟ 3.3-5 โวลต์ ใช้ IC TM1637 เป็นตัวขับ การเชื่อมต่อแบบอินเตอร์เฟส I2C ใช้สายสัญญาณเพียง 2 เส้น มี library มาให้พร้อมใช้งาน สำหรับการแสดงผลตัวเลขที่มีเวลาเช่น นาฬิกา , แสดงผลการจับเวลาและอื่น ๆ (ตัวเลขที่แสดงบนหน้าจอ เป็นสีแดง)

library ของ LED 4 Digit Display Module คือ DigitalTube For Arduino



การทดลองระหว่าง Temperature Humidity Sensor และ LED 4 Digit Display Module มีอุปกรณ์ดังต่อไปน





ต่อขา Module เข้าบอร์ค Arduino คังนี้

DHT11 ไป บา 2 CLK ไป บา 3 DIO ไป บา 4 VCc ไป 5V Gnd ไป Gnd

<u>การต่อวงจร</u>



<u>คำสั่งควบคุมบอร์ค Arduino และ Module</u>

Ø	
s	ketch_oct15a§
1	<pre>#include "dht.h"</pre>
2	#include "TM1637.h"
3	
4	#define dht_pin 2
5	#define CLK 3
6	#define DIO 4
7	
8	TM1637 tm1637(CLK,DIO);
9	dht DHT;
10	
11	<pre>void setup(){</pre>
12	
13	<pre>tml637.init();</pre>
14	<pre>tml637.set(BRIGHT_TYPICAL);</pre>
15	<pre>delay(1500);//Delay to let system boot</pre>
16	}
17	
18	<pre>void loop(){</pre>
19	DHT.readll(dht_pin);
20	<pre>int temp = DHT.temperature;</pre>
21	<pre>int humidity = DHT.humidity;</pre>
22	
23	int digitoneT = temp / 10;
24	int digittwoT = temp % 10;
25	
26	int digitoneH = humidity / 10;
27	int digittwon - humidity * 10;
20	tml627 digmlaw(l digitanam).
30	<pre>tml637.display(2_digittyoT);</pre>
31	$\pm m1637$ display(2, digite wor),
32	cartor. display (0,12), // pac a c ac che ena
33	delay (3000):
34	
35	tml637.display(1,23);
36	tml637.display(2,digitoneH):
37	tml637.display(3, digittwoH);
38	}
39	

<u>อธิบายโค้คคำสั่ง</u>

ค่าที่ Temperature Humidity Sensor ตรวจรับได้จะแสดงบน LED 4 Digit Display Module

หน่วยการเรียนรู้ที่ 8 ทำความรู้จักเซนเซอร์ เรื่องย่อยที่ 10 เครื่องวัดอากาศ PM 2.5

PM 2.5 คือ ฝุ่นละอองขนาดเล็กไม่เกิน 2.5 ไมครอน เทียบได้ว่ามีขนาดประมาณ 1 ใน 25 ส่วนของ เส้นผ่านศูนย์กลางเส้นผมมนุษย์ เล็กจนขนจมูกของมนุษย์ที่ทำหน้าที่กรองฝุ่นนั้นไม่สามารถกรองได้ จึง แพร่กระจายเข้าสู่ทางเดินหายใจ กระแสเลือด และเข้าสู่อวัยวะอื่นๆ ในร่างกายได้ ตัวฝุ่นเป็นพาหะนำสารอื่น เข้ามาด้วย เช่น แคดเมียม ปรอท โลหะหนัก และสารก่อมะเร็งอื่นๆ

สาเหตุที่ทำให้เกิด PM 2.5

ฝุ่นละอองขนาคไม่เกิน 2.5 ไมครอน (PM 2.5) มาจากสองแหล่งกำเนิคใหญ่ๆ คือ

1. แหล่งกำเนิดโดยตรง ได้แก่ การเผาในที่โล่ง การคมนาคมขนส่ง การผลิตไฟฟ้า อุตสาหกรรมการ ผลิต

 2. การรวมตัวของก๊าซอื่นๆ ในบรรยากาศ โดยเฉพาะซัลเฟอร์ ไดออกไซด์ (SO₂) และออกไซด์ของ ในโตรเจน (NO_x) รวมทั้งสารพิษอื่นๆ ที่ล้วนเป็นอันตรายต่อร่างกายมนุษย์ เช่น สารปรอท (Hg), แคดเมียม (Cd), อาร์เซนิก (As) หรือโพลีไซกลิกอะโรมาติกไฮโดรการ์บอน (PAHs)

ระดับความรุนแรงของ PM 2.5

ดัชนีคุณภาพอากาศ (Air Quality Index : AQI) เป็นการรายงานข้อมูลคุณภาพอากาศในรูปแบบที่ ง่ายต่อความเข้าใจของประชาชนทั่วไป เพื่อเผยแพร่ประชาสัมพันธ์ให้สาธารณชนได้รับทราบถึงสถานการณ์ มลพิษทางอากาศในแต่ละพื้นที่ว่าอยู่ในระดับใด มีผลกระทบต่อสุขภาพอนามัยหรือไม่ ดัชนีคุณภาพอากาศ 1 ค่า ใช้เป็นตัวแทนก่าความเข้มข้นของสารมลพิษทางอากาศ 6 ชนิด ได้แก่

 1. ฝุ่นละอองขนาดไม่เกิน 2.5 ไมครอน (PM2.5) เป็นฝุ่นที่มีเส้นผ่านศูนย์กลางไม่เกิน 2.5 ไมครอน เกิดจากการเผาไหม้ทั้งจากยานพาหนะ การเผาวัสดุการเกษตร ไฟป่า และกระบวนการอุตสาหกรรม สามารถ เข้าไปถึงถุงลมในปอดได้ เป็นผลทำให้เกิดโรคในระบบทางเดินหายใจ และโรคปอดต่างๆ หากได้รับใน ปริมาณมากหรือเป็นเวลานานจะสะสมในเนื้อเยื่อปอด ทำให้การทำงานของปอดเสื่อมประสิทธิภาพลง ทำ ให้หลอดลมอักเสบ มีอาการหอบหืด 2. ฝุ่นละอองขนาดไม่เกิน 10 ไมครอน (PM10) เป็นฝุ่นที่มีขนาดเส้นผ่านศูนย์กลางไม่เกิน 10 ไมครอน เกิดจากการเผาไหม้เชื้อเพลิง การเผาในที่โล่ง กระบวนการอุตสาหกรรม การบด การ โม่ หรือการ ทำให้เป็นผงจากการก่อสร้าง ส่งผลกระทบต่อสุขภาพเนื่องจากเมื่อหายใจเข้าไปสามารถเข้าไปสะสมใน ระบบทางเดินหายใจ

3. ก๊าซโอโซน (O₃) เป็นก๊าซที่ไม่มีสีหรือมีสีฟ้าอ่อน มีกลิ่นฉุน ละลายน้ำได้เล็กน้อย เกิดขึ้นได้ทั้ง ในระดับบรรยากาศชั้นที่สูงจากผิวโลก และระดับชั้นบรรยากาศผิวโลกที่ใกล้พื้นดิน ก๊าซโอโซนที่เป็นสาร มลพิษทางอากาศคือก๊าซโอโซนในชั้นบรรยากาศผิวโลก เกิดจากปฏิกิริยาระหว่างก๊าซออกไซด์ของ ในโตรเจน และสารประกอบอินทรีย์ระเหยง่าย โดยมีแสงแดดเป็นตัวเร่งปฏิกิริยา มีผลกระทบต่อสุขภาพ โดยก่อให้เกิดการระคายเคืองตาและระคายเคืองต่อระบบทางเดินหายใจและเยื่อบุต่างๆ ความสามารถในการ ทำงานของปอดลดลง เหนื่อยเร็ว โดยเฉพาะในเด็ก คนชรา และคนที่เป็นโรกปอดเรื้อรัง

4. ก๊าซการ์บอนมอนอกไซด์ (CO) เป็นก๊าซที่ไม่มีสี กลิ่น และรส เกิดจากการเผาไหม้ที่ไม่สมบูรณ์ ของเชื้อเพลิงที่มีการ์บอนเป็นองก์ประกอบ ก๊าซนี้สามารถสะสมอยู่ในร่างกายได้โดยจะไปรวมตัวกับ ฮีโมโกลบินในเม็คเลือดแดงได้ดีกว่าออกซิเจนประมาณ 200-250 เท่า เมื่อหายใจเข้าไปทำให้ก๊าซชนิดนี้จะ ไปแย่งจับกับฮีโมโกลบินในเลือด เกิดเป็นการ์บอกซีฮีโมโกลบิน (CoHb) ทำให้การลำเลียงออกซิเจนไปสู่ เซลล์ต่างๆ ของร่างกายลดน้อยลง ส่งผลให้ร่างกายเกิดอาการอ่อนเพลีย และหัวใจทำงานหนักขึ้น

5. ก๊าซในโตรเจนใดออกไซด์ (NO₂) เป็นก๊าซที่ไม่มีสีและกลิ่น ละลายน้ำได้เล็กน้อย มีอยู่ทั่วไปใน ธรรมชาติ หรือเกิดจากการกระทำของมนุษย์ เช่น การเผาไหม้เชื้อเพลิงต่างๆ อุตสาหกรรมบางชนิด เป็นต้น ก๊าซนี้มีผลต่อระบบการมองเห็นและผู้ที่มีอาการหอบหืดหรือ โรคเกี่ยวกับทางเดินหายใจ

6. ก๊าซซัลเฟอร์ ใดออกไซด์ (SO₂) เป็นก๊าซที่ไม่มีสี หรืออาจมีสีเหลืองอ่อนๆ มีรสและกลิ่นที่ระดับ กวามเข้มข้นสูง เกิดจากธรรมชาติและการเผาไหม้เชื้อเพลิงที่มีกำมะถัน (ซัลเฟอร์) เป็นส่วนประกอบ สามารถละลายน้ำได้ดี สามารถรวมตัวกับสารมลพิษอื่นแล้วก่อตัวเป็นอนุภากฝุ่นขนาดเล็กได้ ก๊าซนี้มี ผลกระทบโดยตรงต่อสุขภาพ ทำให้เกิดการระกายเคืองต่อเยื่อบุตา ผิวหนัง และระบบทางเดินหายใจ หาก ได้รับเป็นเวลานาน ๆ จะทำให้เป็นโรคหลอดลมอักเสบเรื้อรังได้้

ดัชนีคุณภาพอากาศของประเทศไทยแบ่งเป็น 5 ระดับ คือ ตั้งแต่ 0 ถึง 201 ขึ้นไป ซึ่งแต่ละระดับจะ ใช้สีเป็นสัญลักษณ์เปรียบเทียบระดับของผลกระทบต่อสุขภาพอนามัย (ตารางที่ 1) โดยดัชนีคุณภาพอากาศ 100 จะมีค่าเทียบเท่ามาตรฐานคุณภาพอากาศในบรรยากาศโดยทั่วไป หากดัชนีคุณภาพอากาศมีค่าสูงเกิน กว่า 100 แสดงว่าค่าความเข้มข้นของมลพิษทางอากาศมีค่าเกินมาตรฐานและคุณภาพอากาศในวันนั้นจะเริ่ม มีผลกระทบต่อสุขภาพอนามัยของประชาชน

AQI	คุณภาพอากาศ	สีที่ใช้	ข้อความแจ้งเตือน
0 - 25	ดีมาก	ฟ้า	เหมาะสำหรับกิจกรรมกลางแจ้งและการท่องเที่ยว
26 - 50	ดี	เขียว	สามารถทำกิจกรรมกลางแจ้งและการท่องเที่ยวได้ตามปกติ
51 - 100	ปานกลาง	เหลือง	สามารถทำกิจกรรมกลางแจ้งได้ตามปกติ แต่ถ้าเป็นผู้ที่ต้องดูแลสุขภาพเป็นพิเศษ หาก มีอาการเกี่ยวกับระบบทางเดินหายใจ และระคายเคืองตา ไม่ควรทำกิจกรรมกลางแจ้ง นาน
101 - 200	เริ่มมีผลกระ ทบ ต่อสุขภาพ	ส้ม	ควรเฝ้าระวังสุขภาพ ถ้ามีอาการเกี่ยวกับระบบทางเดินหายใจ ระคายเคืองตา ไม่ควรทำ กิจกรรมกลางแจ้งนาน หรือใช้อุปกรณ์ป้องกัน ส่วนผู้ที่ต้องดูแลสุขภาพเป็นพิเศษ แล้ว มีอาการทางสุขภาพ เช่น ไอ หายใจลำบาก ตาอักเสบ แน่นหน้าอก ปวดศีรษะ หัวใจ เด้นไม่เป็นปกติ คลื่นไส้ อ่อนเพลีย ควรปรึกษาแพทย์
201 ขึ้นไป	มีผลกระทบต่อ สุขภาพ	แคง	ทุกคนควรหลีกเลี่ยงกิจกรรมกลางแจ้งทุกอย่างหลีกเลี่ยงพื้นที่ที่มีมลพิษทางอากาศสูง หรือใช้อุปกรณ์ป้องกันตนเองหากมีความจำเป็น หากมีอาการทางสุขภาพควรปรึกษา แพทย์

	0 - 25	26 - 50	51 - 100	101 - 200	201 ขึ้นไป
สรุปความหมายของสี	ดีมาก	ดี	ป่านกลาง	เริ่มมีผลกระทบ ต่อ สุขภาพ	มีผบกระทบ ต่อ สุขภาพ

แนวทางการป้องกัน

1. สวมหน้ากากป้องกันฝุ่น โดยหน้ากากที่สามารถป้องกันฝุ่น PM 2.5 ได้คือ หน้ากาก N95 หรือจะ
 ใช้หน้ากากอนามัยธรรมดาแต่สวมทับ 2 ชั้น หรือซ้อนผ้าเช็ดหน้าหรือทิชชูไว้ด้านในก็ได้เช่นกัน

2. หลีกเลี่ยงกิจกรรมกลางแจ้งทุกชนิด หากจำเป็นต้องใส่หน้ากาก N95

 3. ใช้เครื่องฟอกอากาศ เนื่องจากภายในอาการอาจไม่ปลอดภัยจาก PM 2.5 เสมอไป โดยเฉพาะ อาการที่มีการเปิดปิดประตูบ่อยครั้งจากการที่มีผู้คนเข้าออกจำนวนมาก ดังนั้นเครื่องฟอกอากาศจึงเป็น สิ่งจำเป็นเพื่อให้หายใจในอาการอย่างสบายใจ

เครื่องวัดอากาศ PM 2.5

คุณสมบัติหลัก

1. อัตราการเตือนที่ผิดพลาดเป็นศูนย์

2. การตอบสนองตามเวลาจริง

3. เส้นผ่านศูนย์กลางของอนุภากที่แตกต่างขั้นต่ำ: 0.3 ไมโครเมตร

- 4. มีกรอบที่มีความแข็งแรงคงทนที่ป้องกันการกระทบรอบด้าน
- 5. เพิ่มช่องอากาศเข้าและทางออกเพื่อปรับให้เข้ากับทิศทางที่แตกต่างกัน

ภาพรวม

PMS5003 เป็นเซ็นเซอร์ความเข้มข้นของอนุภาคคิจิตอลและวัคค่าระดับสากล ซึ่งสามารถใช้เพื่อให้ ใด้จำนวนอนุภาคแขวนลอยในอากาศ เช่น ความเข้มข้นของอนุภาคและส่งออกในรูปแบบคิจิตอล อินเตอร์ เฟซ เซ็นเซอร์นี้สามารถแทรกลงในเครื่องมือที่เกี่ยวข้องกับตัวแปร ความเข้มข้นของอนุภาคแขวนลอยใน อากาศหรือสิ่งแวคล้อมอื่น ๆ ได้

หลักการทำงาน

หลักการกระเจิงเลเซอร์ใช้สำหรับเซ็นเซอร์เช่น ผลิตโดยการกระเจิง ใช้เลเซอร์เพื่อฉายแสงอนุภาค แขวนลอยในอากาศแล้วรวบรวมการกระเจิง แสงในระดับหนึ่งและในที่สุดก็ได้เส้นโค้งของการกระเจิงแสง ที่เปลี่ยนไปกับเวลา อนุภาคที่เท่ากันและจำนวนของอนุภาค ที่มีเส้นผ่าศูนย์กลางที่แตกต่างกันต่อหน่วย ปริมาตรสามารถกำนวณได้โดยไมโครโปรเซสเซอร์ ตามทฤษฎีมิเอะ (MIE theory) หลัการทำงานแสดงได้ ดังนี้



รูป บล็อกใดอะแกรมการทำงานของเซ็นเซอร์

ข้อมูลขา

PIN1	VCC	กำลังไฟ 5V (+)
PIN2	GND	พลังงานเชิงลบ (-)
PIN3	SET	ตั้งระดับพิน / TTL ที่ 3.3V สูงกว่านั้นหรือ การระงับเป็นสถานะการทำงาน ปกติในขณะที่อยู่ในโหมดสลีป
PIN4	RX	พอร์ตอนุกรมที่รับกำลังไฟ 3.3V
PIN5	TX	พอร์ตอนุกรมที่ส่งกำลังไฟ 3.3V
PIN6	RESET	โมดูลรีเซ็ตสัญญาณ / ระดับกำลังไฟ 3.3V หรือระดับต่ำให้ รีเซ็ต
PIN7/8	NC	

ผลลัพธ์ที่ได้

ส่วนใหญ่ส่งออกเป็นกุณภาพและจำนวนของแต่ละอนุภากที่มีขนาดแตกต่างกัน ต่อหน่วยปริมาตร ปริมาตรหน่วยของหมายเลขอนุภากคือ 0.1L และหน่วยของ ความเข้มข้นของมวลคือ µ g/m³ มีสอง ตัวเลือกสำหรับเอาต์พุตดิจิตอล : พาสซีฟและแอกทีฟ โหมดเริ่มต้น ทำงานหลังจากเปิดเครื่อง ในโหมด เซ็นเซอร์นี้จะส่งข้อมูลแบบอนุกรมไปที่ โฮสต์โดยอัตโนมัติโหมดการใช้งานจะถูกแบ่งออกเป็นสองโหมด ย่อย คือ 1. เสถียร โหมด

2. โหมดเร็ว

หากการเปลี่ยนแปลงความเข้มข้นมีขนาดเล็กเซ็นเซอร์จะทำงานในโหมดเสถียรโดยมีช่วงเวลาจริง 2.3 และถ้าความเข้มข้นมีการเปลี่ยนแปลงเซ็นเซอร์จะเปลี่ยนเป็นโหมดเร็วโดยอัตโนมัติด้วย ช่วงเวลา 200 ~ 800ms

วงจรทั่วไป



รูป วงจรทั่วไป

วัดอากาศ PM 2.5 ควบคุมด้วยบอร์ด Arduino

1. อุปกรณ์ที่ใช้ในการทดลองเครื่องวัดอากาศ PM 2.5

บอร์ด Arduino Uno



จอแสดงผล LCD Display 16x2



กล่องพลาสติก



เซ็นเซอร์วัดอากาศ PM 2.5



สาย Jumper แบบ ผู้ – เมีย



2. การต่อวงจร

2.1 การต่อวงจร เซ็นเซอร์วัคอากาศ PM2.5 เข้ากับบอร์ค Arduino

การต่อวงจรจะทำการจั้มสายไฟทั้งหมด 4 เส้นดังนี้

- ขา VCC ต่อเข้ากับ ขาไฟ 5V บนบอร์ค Arduino
- ขา GND ต่อเข้ากับ ขา GND บนบอร์ค Arduino
- ขา TX ต่อเข้ากับขา RX (pin digital 0) บนบอร์ค Arduino
- ขา RX ต่อเข้ากับขา TX (pin digital 1) บนบอร์ค Arduino



2.2 ต่อวงจร LCD Display เข้ากับบอร์ด Arduino

- การต่อวงจรจะทำการจั้มสายไฟทั้งหมด 4 เส้นดังนี้
- ขา VCC ต่อเข้ากับ ขาไฟ 5V บนบอร์ค Arduino
- ขา GND ต่อเข้ากับ ขา GND บนบอร์ค Arduino
- ขา SDA ต่อเข้ากับขา A4 (pin analog 4) บนบอร์ค Arduino
- ขา SCL ต่อเข้ากับขา A5 (pin analog 5) บนบอร์ค Arduino



2.3 นำอุปกรณ์ทั้งหมดมาเชื่อมต่อกันทั้งหมดจะได้ตามรูป



3. การเขียนโปรแกรมเพื่อควบคุมการทำงาน

3.1 การนำไลบรารี่ต่างเข้าโปรแกรมเพื่อให้สามารถเรียกใช้งานได้

ให้ทำการแตกไฟล์ ที่ชื่อ pm_library.rar จะได้ Folder จำนวน 2 Folder แล้วให้นำไปวางไว้ใน Folder libraries ของโปรแกรม Arduino IDE ซึ่งตำแหน่งของไลบรารี่จะอยู่ที่ C:\Users\COM_NAME\Documents\Arduino\libraries

C\Users\CON	A_PC\Documents\Arduino\libraries	- 49 Sean	ch libraries	
Organize 👻 🎲 Open	Share with E-mail Burn New	folder	(E) (· 🖬 🕯
🖈 Favorites 📰 Desktop	Documents library		Arrange by:	Folder *
Downloads	Name	Date modified	Туре	Size
Recent Places	Adafruit-MCP23017-Arduino-Library-ma_	09/08/2562 11:10	File folder	
1 Sharahara	👗 Blynk	25/05/2562 14:11	File folder	
Documents	DHT	02/02/2557 00:10	File folder	
Music	DHT_sensor_library_for_ESPx	26/07/2562 11:02	File folder	
Pictures	ESP8266_Microgear	26/07/2562 14:07	File folder	
Videos	FaBo_212_LCD_PCF8574	28/10/2562 10:51	File folder	
La macor	LiquidCrystal_PCF8574	28/10/2562 10:55	File folder	
Computer	LiquidCrystal-12C2	06/11/2562 17:06	File folder	
WINT (C-)	3 LM35	15/08/2562 08:46	File folder	
a data (D:)	J MCP3008	10/09/2562 15:40	File folder	
CD Drive (L:)	👪 NewPing	03/08/2562 10:09	File folder	
	🕌 PMS-master	06/11/2562.16:25	File folder	
Network	🎍 PubSubClient	25/05/2562 13:25	File folder	
N/2	TridentTD_LineNotify-master	25/05/2562 14:03	File folder	
	i readme	25/05/2562 10:47	Text Document	1

3.2 ให้ทำการคัดลอกโค้คนี้ไปใส่ในโปรแกรม ArduinoIDE แล้วโหลดลงบอร์คได้เลย

///**************************กรุณาอ่าน สำคัญมาก************************///

/// ก่อนทำการอัพโหลดโปรแกรมเข้าสู่บอร์ด ให้ทำการถอดสาย TX,RX ที่ต่อเข้ากับ บอร์ดออกก่อน ///

/// เมื่อทำการอัดโหลดโปรแกรมเข้าบอร์ดเรียบร้อยก็ทำการเสียบสาย TX,RX เข้ากับ บอร์ดได้เลย ///

#include <Arduino.h>

#include "PMS.h" // นำเข้าไลบรารี เพื่อใช้ในการอ่านค่า PM2.5 #include <Wire.h> // นำเข้าไลบรารี เพื่อใช้ในการแสดงข้อความบน LCD Display #include <LiquidCrystal_I2C.h> // นำเข้าไลบรารี เพื่อใช้ในการแสดงข้อความบน LCD

Display

////การกำหนดตัวแปลในส่วนของการอ่านค่า PM2.5 ////

PMS pms(Serial);

PMS::DATA data;

void setup()

{

Serial.begin(9600); //เริ่มต้นการเชื่อมต่อ Serial Port ด้วย buadrate 9600 lcd.begin(); // เปิดการใช้งาน LCD Display

```
lcd.backlight(); // เปิดไฟ backlight
                 lcd.setCursor(0, 0); // ไปที่ตัวอักษรที่ 0 บรรทัคที่ 0
                 lcd.print("Test School"); // พิมพ์ข้อความบรรทัคที่ 0 เป็นชื่อสถานที่
                 }
                  void loop()
                  {
                 if (pms.read(data)) //เมื่อเซ็นเซอร์อ่านค่า PM2.5
                  {
                 pm25=data.PM AE UG 2 5; //เก็บก่าที่อ่านได้ไว้ในตัวแปล
                 Serial.print("PM 2.5 (ug/m3): "); //พิมพ์ข้อความไปยัง Serial Port
                 Serial.println(pm25); // พิมพ์ค่า PM2.5 ที่อ่านได้ไปยัง Serial Port พร้อมกับขึ้นบรรทัด
ใหม่
                 lcd.setCursor(0, 1): // ไปที่ตัวอักษรที่ 0 บรรทัดที่ 1
                 lcd.print("PM2.5= " + String(pm25) + "(ug/m3)"); //แสดงค่าที่อ่านได้ไปยัง LCD
Display ในบรรทัดที่ 1
                 }
                 }
```

217
หน้าต่างโค้ดในโปรแกรม ArduinoIDE

```
💿 pm_uno | Arduino 1.8.9
File Edit Sketch Tools Help
                +
             +
  pm uno
#include <Arduino.h>
∳include "PMS.h" // นำเข้า"ໄຂມຽາรี ເพื่อใช้ในการอ่านค่า PM2.5
∲include <Wire.h> // นำเข้าไลมราร์ เพื่อใช้ในการแสดงข้อความมน LCD Display
≢include <LiquidCrystal I2C.h> // นำเข้าไลบราร์ เฟื้อใช้ในการแสดงข้อความมน LCD Display
////การกำหนดตวัแปลในส่วนของการอ่านค่า PM2.5 ////
PMS pms(Serial);
PMS::DATA data;
LiquidCrystal_I2C lcd(0x27, 16, 2); // ด่อสาย
int pm25; //ประกาศตัวแปลเพื่อเก็บค่า PM2.5 ที่อ่านมาใต้
void setup()
{
  Serial.begin(9600); //เริ่มต้นการเชื่อมต่อ Serial Port ด้วย buadrate 9600
  lcd.begin(); // ເປິດກາຮໃຫ້ນາພ LCD Display
  lcd.backlight(); // ເນິດໃW backlight
  lcd.setCursor(0, 0); // "ໄປທີ່ດ້ວວັກນະທີ່ 0 ມະະທັດທີ່ 0
  lcd.print("Test School"); // พิมพ์ข้อความบรรทัดที่ 0 เป็นชื่อสถานที่
3
void loop()
ł
  if (pms.read(data)) //เมื่อเป็นเซอร์อ่านค่า PM2.5
  ł
    pm25=data.PM_AE_UG_2_5; //เก็มค่าที่อ่านได้ไว้ในตวัแปล
    Serial.print("PM 2.5 (ug/m3): "); //พิมพ์ข้อความ"ໃນຢນັ Serial Port
    Serial.println(pm25); // พิมพ์ค่า PM2.5 ที่อ่านได้ไปยัง Serial Port พร้อมกับขึ้นบรรทัดใหม่
    lcd.setCursor(0, 1); // ໃນກໍ່ດ້າວັກນຣກທີ່ 0 ນຣຣກດັກທີ່ 1
    lcd.print("PM2.5= " + String(pm25) + "(ug/m3)"); //แสดงค่าที่อ่านได้ไปยัง LCD Display ในบรรหัดที่ 1
  }
}
```

หมายเหตุ ก่อนทำการ โหลด โปรแกรมให้ทำการถอดสายระหว่างเซ็นเซอร์วัดอากาศ PM2.5 กับ บอร์ด ออกก่อน ถ้าไม่ถอดตอนอัพโหลดจะ error และอัพโหลดไม่ผ่าน

เมื่อทำการ โหลด โปรแกรมลงบอร์ดเรียบร้อย และ ได้ผลลัพธ์เป็นดังที่ต้องการแล้ว จากนั้นจึงทำการ นำอุปกรณ์ทั้งหมดจัดเก็บลงกล่องจัดเก็บอุปกรณ์

4. การจัดเก็บอุปกรณ์ลงกล่อง



4.1 นำบอร์คยึดเข้ากับแผ่นอะคริลิกแล้วนำไปประกอบยึดตอดกับฐานของกล่อง

4.2 นำเซ็นเซอร์วัดอากาศ PM2.5 และ LCD Display ไปยึดติดกับฝาของกล่องตามรูป



4.3 ทำการเชื่อมต่อสายและนำเก็บเข้ากล่องให้เรียบร้อย และเมื่อทำการเสียบสาย USB ที่มีไฟเลี้ยง เข้าไปก็จะแสดงผลการวัดค่า PM20.5 ขึ้นที่ LCD Display



หน่วยการเรียนรู้ที่ ธ การประยุกต์ใช้โปรแกรมแลบวิว

กลุ่มสาระการเรียนรู้วิทยาศาสตร์ สาระที่ ๔ เทคโนโลยี

มาตรฐาน ว ๔.๑ เข้าใจแนวคิดหลักของเทกโนโลยีเพื่อการคำรงชีวิตในสังกมที่มีการเปลี่ยนแปลงอย่าง รวดเร็ว ใช้ความรู้และทักษะทางค้านวิทยาศาสตร์ กณิตศาสตร์ และศาสตร์อื่น ๆ เพื่อแก้ปัญหาหรือพัฒนา งานอย่างมีความคิดสร้างสรรค์ด้วยกระบวนการออกแบบเชิงวิศวกรรมเลือกใช้เทคโนโลยีอย่างเหมาะสม โดยคำนึงถึงผลกระทบต่อชีวิต สังคม และสิ่งแวคล้อม

ตัวชี้วัด

ງ໔.໑ ນ.໑/໔	ออกแบบวิธีการแก้ปัญหา โดยวิเคราะห์ เปรียบเทียบ และตัดสินใจเลือกข้อมูลที่จำเป็น นำเสนอแนวทางการแก้ปัญหาให้ผู้อื่นเข้าใจ วางแผนและคำเนินการแก้ปัญหา
า ๔.๑ ม.๑/๕	ใช้ความรู้และทักษะเกี่ยวกับวัสคุ อุปกรณ์ เครื่องมือ กลไก ไฟฟ้า หรืออิเล็กทรอนิกส์เพื่อ แก้ปัญหาได้อย่างถูกต้อง เหมาะสมและปลอดภัย
า ๔.๑ ม.๒/๔	ทดสอบ ประเมินผล และอธิบายปัญหาหรือข้อบกพร่องที่เกิดขึ้น ภายใต้กรอบเงื่อนไข พร้อมทั้งหาแนวทางการปรับปรุงแก้ไข และนำเสนอผลการแก้ปัญหา
ງ ໔.໑ ມ.๒/໕	ใช้ความรู้ และทักษะเกี่ยวกับวัสคุ อุปกรณ์ เครื่องมือ กลไก ไฟฟ้า และอิเล็กทรอนิกส์ เพื่อแก้ปัญหาหรือพัฒนางานได้อย่างถูกค้อง เหมาะสม และปลอดภัย
ງ໔.໑ ມ.ຓ/໔	ทดสอบ ประเมินผล วิเคราะห์ และให้เหตุผลของปัญหาหรือข้อบกพร่องที่เกิดขึ้นภายใต้ กรอบเงื่อนไข พร้อมทั้งหาแนวทางการปรับปรุงแก้ไข และนำเสนอผลการแก้ปัญหา
ງ໔.໑ຆ.ຓ/໕	ใช้ความรู้ และทักษะเกี่ยวกับวัสคุ อุปกรณ์ เครื่องมือ กลไก ไฟฟ้าและอิเล็กทรอนิกส์ให้ ถูกต้องกับลักษณะของงาน และปลอคภัย เพื่อแก้ปัญหาหรือพัฒนางาน
ી ૡૼ.૭ પ્ર.ૡૼ/ૡૼ	ใช้ความรู้และทักษะเกี่ยวกับวัสคุ อุปกรณ์ เครื่องมือ กลไก ไฟฟ้าและอิเล็กทรอนิกส์และ เทคโนโลยีที่ซับซ้อนในการแก้ปัญหาหรือพัฒนางาน ได้อย่างถูกต้อง เหมาะสมและ ปลอดภัย

สาระสำคัญ

- ๑. ผู้เรียนได้เรียนรู้เครื่องมือและส่วนประกอบ
- ๒. ผู้เรียนได้เรียนรู้ตัวต้านทาน
- ๓. ผู้เรียนได้เรียนรู้ไฟ LED
- ๔. ผู้เรียนได้เรียนรู้การเลื่อนภาพจากจอ LCD 12*2
- ๕. ผู้เรียน ได้เรียนรู้การเชื่อมต่อบอร์ด Arduino กับ Processing

สาระการเรียนรู้

<u>ความรู้</u>

- ผู้เรียนจะได้ศึกษาอุปกรณ์อิเล็กทรอนิกส์ การต่อวงจร ตลอดจนการเงียนคำสั่งควบคุมบอร์ด

<u>ทักษะ / กระบวนการ</u>

 ๑. ใช้กระบวนการสังเกตเพื่อให้ผู้เรียนรับรู้ข้อมูล และศึกษาด้วยวิธีการต่างๆ อันซึ่งนำมาด้วย กวามรู้ ความเข้าใจ

๒. ใช้กระบวนการกลุ่ม เพื่ออธิบาย รับฟัง และวิจารณ์กิจกรรมที่พิจารณาส่วนประกอบของการ กระทำหรือข้อมูลต่างๆที่เชื่อม โยงเกี่ยวข้องกัน

<u>คุณลักษณะที่พึงประสงค์</u>

- ๑. ผู้เรียนมีใจรักในการเขียนคำสั่งควบคุมบอร์ค Arduino Uno R3
- ๒. ผู้เรียนมีความเข้าใจในอุปกรณ์อิเล็กทรอนิกส์
- ผู้เรียนรักและมีส่วนร่วมในกระบวนการกลุ่ม
- ๕. รักชาติ ศาสน์ กษัตริย์
- ๕. ซื่อสัตย์
- b. ນີວິນັບ
- ๗. ใฝ่เรียนรู้
- ๘. อยู่อย่างพอเพียง
- ย. มุ่งมั่นในการทำงาน

หน่วยการเรียนรู้ที่ 9 การประยุกต์ใช้แลบวิว

LabVIEW เป็นโปรแกรมคอมพิวเตอร์ที่สร้างเพื่อนำมาใช้ในด้านการวัดและเครื่องมือวัดสำหรับ งานทางวิศวกรรม LabVIEW ย่อมาจาก Laboratory Virtual Instrument Engineering Workbench ซึ่ง หมายความว่าเป็นโปรแกรมที่สร้าง เครื่องมือวัดเสมือนจริงในห้องปฏิบัติการทางวิศวกรรม ดังนั้น จุดประสงค์หลักของการทำงานของโปรแกรมนี้ก็คือการจัดการในด้านการวัดและเครื่องมือวัด อย่างมี ประสิทธิภาพ และในด้วของโปรแกรมจะประกอบไปด้วยฟังก์ชันที่ใช้ช่วยในการวัดมากมายและแน่นอน ที่สุด โปรแกรมนี้จะมีประโยชน์อย่างสูงเมื่อใช้ร่วมกับเครื่องมือวัดทางวิศวกรรมต่างๆ โปรแกรมแลบวิว สามารถติดต่อสื่อสารกับเครื่องมือต่างๆ ที่อยู่ภายนอกผ่านอุปกรณ์ที่ใช้สื่อสารในการวัดมากมายและแน่นอน การกวบคุมอุปกรณ์ต่างๆ เช่น อุปกรณ์ไฟฟ้า เป็นต้น และยังสามารถเชื่อมต่อกับอุปกรณ์ในการวัดทาง กายภาพ อุณหภูมิ ความชื้น หรือสัญญาณอื่นๆ โดนมีตัวเซนเซอร์รับสัญญาณเข้ามาและเอาต์พุตที่ได้จาก เซนเซอร์เหล่านี้จะมีก่าเป็นแรงคันหรือกระแส ซึ่งโปรแกรมแลบวิวสามารถอ่านก่าที่ผ่านเข้ามาและนำมา ประมวลผลแล้วบันทึกก่าเป็นไฟล์ข้อมูลได้ ดังนั้นการนำโปรแกรมแลบวิวไปใช้จะพิจารณาถึงวัตอุประสงค์ ในการนำไปประยุกต์ใช้ และต้องกำนึงถึงอินพุตที่ต้องการ และเอาต์พุตที่ต้องการ จากนั้นจึงทำการเลือก ฮาร์ตแวร์ให้ตรงตามวัดกุประสงค์ที่ด้องการ

สิ่งที่ LabVIEW แตกต่างจากโปรแกรมอื่นอย่างเห็นได้ชัดที่สุดก็คือ LabVIEW นี้เป็นโปรแกรม ประเภท GUI (Graphic User Interface) โดยสมบูรณ์ นั่นคือเราไม่จำเป็นต้องเขียน code หรือคำสั่งใดๆ ทั้งสิ้น และที่สำคัญลักษณะภาษาที่ใช้ในโปรแกรมนี้เราจะเรียกว่าเป็น ภาษารูปภาพ หรือเรียกอีกอย่างว่า ภาษา G (Graphical Language) ซึ่งจะแทนการเขียนโปรแกรมเป็นบรรทัดอย่างที่เรากุ้นเกยกับภาษาพื้นฐาน เช่น C, BASIC หรือ FORTRAN ด้วยรูปภาพหรือสัญลักษณ์ทั้งหมด ซึ่งแม้ว่าในเบื้องต้นเราอาจจะสับสน กับการจัดเรียบหรือเขียนโปรแกรมบ้าง แต่เมื่อเรากุ้นเกยกับการใช้โปรแกรมนี้แล้วเราจะพบว่า LabVIEW นี้ มีความสะดวกและสามารถลดเวลาในการเขียนโปรแกรมลงไปได้มาก โดยเฉพาะในงานเขียนโปรแกรม กอ ม พิ ว เต อ ร์ เพื่ อ เชื่ อ ม ต่ อ กั บ อุ ป ก ร ณ์ อื่ น ๆ เพื่ อ ใ ช้ ใ น ก าร วั ด แ ล ะ ก าร ค ว บ คุ ม การประยุกต์ใช้งานโปรแกรมแลบวิวนั้นด้องติดตั้งโปรแกรมเพิ่มเติมเพื่อใช้ในการเชื่อมต่อบอร์ด ในโกรกอนโทรลเลอร์อาร์ดุยโน่ ซึ่งโปรแกรมที่ดิดตั้งเพิ่มคือ โปรแกรม NI-VISA ซึ่งสามารถดาวน์โหลด ได้ที่ http://www.ni.com/downlond/ni-visa-5.0.3/2251/en/ และอีกหนึ่งโปรแกรม Pipackage จะต้องทำการดิดตั้ง แบบออนไลน์

1. ติดตั้งโปรแกรม NI-VISA

1. เข้าเว็บไซต์ <u>http://www.ni.com/downlond/ni-visa-5.0.3/2251/en/</u>

Page Error

The page you requested may have moved, been deleted, or is inaccessible. Search below for other content, or contact NI for assistance.

NI-VISA	٩
NI-VISA in Entire Site	
NI-VISA in Manuals	
NI-VISA in White Papers	
NI-VISA in Drivers	
VISA: List all VISA Devices - NI Community in Examples	
NI-VISA .NET Class Reference Page in Knowledgebase	

2. ก้นหากำว่า NI-VISA กลิกเลือกลิ้งก์เพื่อดาวน์โหลดโปรแกรม



3. คลิกปุ่ม DOWNLOARD เพื่อคาวน์ โหลดโปรแกรม



4. ดับเบิ้ลคลิก 🕅 ni-visa_19.0_online.exe --> คลิกปุ่ม Run เพื่อลงโปรแกรม

🔒 🛃 📮 I	Program Files (x86)	Manage		_		<
File Home	Share View	Application Tools			^	?
Pin to Quick Copy access	Paste	ו New folder	Properties	Select		
Спрова		nize New	Open			
$\leftarrow \rightarrow \checkmark \uparrow$	w WIN1 > Prog	gram Files (x86) >	✓ [™] Searce	h Progra	m Fil 🔎	
A Quick ac	Name	^	Date modi	fied	Туре	^
	Windows M	ultimedia Platform	19/03/2562	13:23	File fold	d
📃 Desl 🖈	Windows N	Т	19/03/2562	12:02	File fold	d
👆 Dow 🖈	Windows Ph	noto Viewer	19/03/2562	11:11	File fold	b
🔮 Doc 🖈	Windows Po	ortable Devices	19/03/2562	13:23	File fold	d
Pict 🖈	WindowsPo	werShell	19/03/2562	11:52	File fold	d
	WinSCP		22/02/2561	14:22	File fold	d
- ·	WMIHook		11/08/2560	22:51	File fold	d
image	🔜 🗐 i-BOX III.rar		18/02/2562	11:15	WinZip	
image	☑ mi-visa_19.0	_online.exe	15/10/2562	10:45	Applica	1.
🔜 หลักสูตร 🗸	<				>	
55 items 1 item	selected 3.63 MB					

5. ในหน้าข้อตกลงให้กลิก I accept the above license agreement.

NI Package Manager				×
Select	Agree	Review	Perform	
You must accept t	he license agreem	ents below to pro	oceed.	
NI				
NATIONA	L INSTRUMENTS SO	OFTWARE LICENSE	AGREEMENT	
CAREFULLY READ THIS THE SOFTWARE AND/OR PROCESS, YOU AGREE WISH TO BECOME A I CONDITIONS, DO NOT IN ACCOMPANYING WRITTI RECEIPT. ALL RETURNS are accepting these terms these terms.	SOFTWARE LICENSE AG CLICKING THE APPLICAT TO BE BOUND BY THE PARTY TO THIS AGREE STALL OR USE THE SOFT EN MATERIALS AND THEI TO NI WILL BE SUBJECT on behalf of an entity, you	REEMENT ("AGREEMEN" BLE BUTTON TO COMPLE FERMS OF THIS AGREEF MENT AND BE BOUND WARE, AND RETURN THE R CONTAINERS) WITHIN TO NI'S THEN-CURRENT agree that you have auth	T"). BY DOWNLOADING ETE THE INSTALLATION MENT. IF YOU DO NOT BY ITS TERMS AND SOFTWARE (WITH ALL THIRTY (30) DAYS OF RETURN POLICY. If you ority to bind the entity to	
The terms of this Agreeme	nt apply to the computer so	oftware provided with this A	greement, all updates or	Ŧ
This license agreement appl	ies to the following package	s: NI Package Manager		
		○ I accept	the above license agreeme	nt.
		● I do not	accept the license agreeme	ent.
			Next	

6. ในหน้าต่างตั้งก่า Windows ให้กลิกเกรื่องหมายถูกหน้าข้อกวาม Disable Windows fast startup ออก จากนั้นกลิก Next



7. คลิก Next

NI Package Manager			×
Select	Agree	Review	Perform
Review the follow	ing summary bef	ore continuing.	
▼ Install			
NI Package Manager			19.5.1
Back			Next

8. ถัดมาเป็นหน้าต่างสถานะการลงโปรแกรม ในหน้าต่างนี้ให้รอจนกว่าระบบจะลงโปรแกรมเสร็จ สิ้น เมื่อลงเสร็จสิ้นให้คลิก Next

NI P	Package Manager			×
	Select	Agree	Review	Perform
In	stalling NI Package Mana	iger Deployment Support		
				Next

Installing NI-VISA Select Review Finish Agree Additional items you may wish to install: . Installs the local assembly for development of NI-VISA applications using the standard .NET API specified by the IVI Foundation. ✓ NI-VISA .NET Runtime Enables you to run applications that use the NI-VISA .NET API. ✓ NI-VISA C Examples Provides examples for programming using the C API. ✓ NI-VISA Configuration Support Provides support for viewing and/or configuring NI-VISA resources in NI MAX. ▼ NI-VISA Driver Development Wizard Provides the NI-VISA Driver Development Wizard for creating INF files for PXI and USB. ✓ NI-VISA Interactive Control Provides the NI-VISA Interactive Control utility to interact and communicate with NI-VISA resources. ▼ NI-VISA support for LabVIEW NXG 3.0 Run-Time Engine Provides support for deploying applications using NI-VISA within LabVIEW NXG 3.0 Run-Time Engine. NI-VISA LabWindows/CVI Examples Select All Deselect All Next

9. ในหน้าต่างการเลือกคือ รายการเพิ่มเติมที่ผู้ใช้งานต้องการติดตั้งเพิ่ม

2. ติดตั้งโปรแกรม LabVIEW (เฉพาะโปรแกรมแลบวิวที่ขณะติดตั้งให้ปิดอินเตอร์ก่อน)

1. ดาวน์โหลดได้จาก

https://lumen.ni.com/nicif/US/EVALLVUSER/content.xhtml?du=https://www.ni.com/en-

<u>th/support/downloads/software-products/download/packaged.labview.305508.html</u> โดยทำการถงทะเบียน เพื่อสามารถดาวน์โหถดโปรแกรมโดยไม่เสียค่าใช้จ่าย

3. ติดตั้งโปรแกรม VI package

1. ดาวน์โหลดได้จาก <u>http://jki.net/vipm/download</u> เมื่อเข้าสู่หน้าเว็บไซต์แล้ว ให้คลิกเลือก FREE DOWNLOAD

Create, disco	VIPM FREE
One VIPM FREE lic	tense
Limited technical	support
F ree updates	
	FREE
	DOWNLOAD

2. คลิกเลือก Windows VIPM 2019 จากนั้นกรอกอีเมล์ในช่องสีขาวค้านล่าง เมื่อกรอกอีเมล์ เรียบร้อยแล้ว คลิก Download

Windows VIPM 2019	Mac OS X VIPM 2017	Linux VIPM 2017
john.smith@jki.net	ENTER YOUR E-MAIL	
	Download	

3. ดับเบิ้ลคลิกไฟล์ VI Package Manager.exe เพื่อทำการลงโปรแกรม

PC > DATA (D:) > lesson9 > J	KI 🔹 VI Package Manager
Name	Date modified
📙 help	16/10/2562 21:04
icons	16/10/2562 21:03
support	16/10/2562 21:06
🔁 License_Agreement	13/03/2557 08:40
VI Package Manager	02/03/2562 06:07
🚋 VI Package Manager	15/10/2562 12:42



4. รองนกระทั่ง VI Package โหลดชุดคำสั่งหรือโปรแกรมเสริมเสร็จสิ้น

5. พิมพ์ Arduino ที่ช่องค้นหาจากนั้นให้ดับเบิ้ลคลิก LabVIEW Interface for Arduino จากนั้นคลิก

ที่ 🎦

Install

เพื่อทำการถงโปรแกรมเพื่อติดต่อบอร์ด Arduino

VI JKI VI Package Manager			_	
File Edit View Package Tools Wind	low Help			
1 1 2 2 2	1	999 🔽 📷 All	V Arduino	×
Name /\	Version	Repository	Company	
Arduino Compatible Compiler for LabV	1.0.0.21	NI LabVIEW Tools Network	Aledyne-TSXperts	
Arduino Compatible Compiler for LabV	1.0.0.21	NI LabVIEW Tools Network	Aledyne-TSXperts	
Digilent LINX (Control Arduino, Raspbe	erry Pi, BeagleBone and more)	NI LabVIEW Tools Network	Digilent	
LabVIEW Interface for Arduino	2.2.0.79	NI LabVIEW Tools Network	National Instruments	

VI JKI VIPM - LabVIEW Inte	erface for Arduino
File Help	
Package Information Select an action	to perform on the package.
12 2018 V	LabVIEW Interface for Arduino Product Homepage
Show in Palettes	LabVIEW Interface for Arduino v2.2.0.79 by National Instruments Released On: Thu, 19 May 2016 16:22:42 -0500 Author: Sam Kristoff Copyright: Copyright (c) 2012, National Instruments Compatible LabVIEW Versions: >= 2009. Compatible Copyright Copyrights (c) 2012, National Instruments Compatible Copyright (c) 2012, National Instruments
Show Examples	Compatible US Versions: ALL. Repository Name: NI LabVIEW Tools Network Description: The LabVIEW Interface for Arduino toolkit provides an easy to use interface for the Arduino microcontroller platform. The LabVIEW Interface for Arduino allows you to quickly and easily create graphical user interfaces for virtually any component that is compatible with the Arduino microcontroller. This open source toolkit is made for customization allowing users to create custom drivers for their sensors. Please visit www.ni.com/ arduino for more information.
	Release Notes: LabVIEW Interface for Arduino National Instruments

จากนั้นจะปรากฏหน้าต่างการดาวน์โหลด software ให้กลิก [Agree (Don't Prompt me Again)]

VI VIPM - Download Acknowledgement	× i
You are about to download software external to VI Package Manager. You acknowledge that the software you are downloading contains a software license that is different from VI Package Manager and, by downloading such software, you agree that you are responsible for compliance with any license terms.	24
I Agree (Don't Prompt me Again) I Agree 🄀 Cancel	

จากนั้นรอการเชื่อมต่อ LabVIEW จนเสร็จสมบูรณ์

	<u>ې</u>						
	Connecting to LabVIEW						
VI JKI VI Pack	age Manager					_	
File Edit View	Package Tools Window Help VI JKI VIPM - Last Action Results Edit				— C	1 ×	
Name /\	Her	e are the resu	Its of the last action	on.			
Arduino Con Arduino Con	Product	Action	Status \/				1
Digilent LIN)	✓ LabVIEW Interface for Arduino v2.2.0.79	installed	No Errors				
	Include Dependencies				•		
	Select / Deselect All						
		🕜 Help]		1	Finish	-
<	·						>

VI JKI VI Package Manager - 🗆 🗙				
File Edit View Package Tools Wine	dow Help			
100 Internet	1	D 1 2 🕞 All	V Arduino X	
Name /\	Version	Repository	Company	
Arduino Compatible Compiler for LabV	1.0.0.21	NI LabVIEW Tools Network	Aledyne-TSXperts	
Arduino Compatible Compiler for LabV	1.0.0.21	NI LabVIEW Tools Network	Aledyne-TSXperts	
Digilent LINX (Control Arduino, Raspb	3.0.1.192	NI LabVIEW Tools Network	Digilent	
LabVIEW Interface for Arduino	2.2.0.79	NI LabVIEW Tools Network	National Instruments	
A				
-				
<			>	
Ready				

การต่อวงจรในโปรแกรม LabVIEW 2018

การทคลองนี้คือ การต่อวงจรในโปรแกรม LabVIEW 2018 ซึ่งการต่อวงจรนี้จะอ้างอิงถึง TX และ L (ไฟ LED) บนบอร์ค Arduino เมื่อต่อวงจรเรียบร้อยแล้ว จะเขียนคำสั่งที่โปรแกรม Arduino IDE และ ควบคุมการเปิค-ปิคผ่านโปรแกรม LabVIEW ซึ่งการต่อวงจรและการเขียนคำสั่งจะเรียบเรียงเป็นขั้นตอน ดังต่อไปนี้

1. เปิดโปรแกรม LabVIEW 2018 --> Create Project --> Blank Project --> File --> New VI ดังรูป





เมื่อ New VI เข้ามาแล้วจะพบกับ 2 หน้าต่าง คือ หน้าต่าง Front Panel และหน้าต่าง Block Diagram 1.1 หน้าต่าง Block Diagram

Block Diagram นี้ ถือว่าเป็นภาษา C และเป็นภาษา Executable Program คือสามารถที่จะทำงานได้ ทันที และข้อดีอีกประการหนึ่งก็คือ LabVIEW จะมีการตรวจสอบความผิดพลาดของโปรแกรมตลอดเวลา ทำให้โปรแกรมจะทำงานได้ก็ต่อเมื่อไม่มีข้อผิดพลาดในโปรแกรมเท่านั้น

ส่วนประกอบภายใน Block Diagram นี้จะประกอบด้วย ฟังก์ชัน ค่าคงที่ โปรแกรมควบคุมการทำงาน หรือ โครงสร้าง จากนั้นในแต่ละส่วนเหล่านี้ ซึ่งจะปรากฏในรูปของ Block เราจะได้รับการต่อสาย (wire) สำหรับ Block ที่เหมาะสมเข้าด้วยกัน เพื่อกำหนดลักษณะการไหลของข้อมูลระหว่าง block เหล่านั้น ทำให้ ข้อมูลได้รับการประมวลผลตามที่ต้องการ และแสดงผลออกมาให้แก่ผู้ใช้ต่อไป



1.2 หน้าต่าง Front Panel หรือหน้าปัทม์

จะเป็นส่วนที่ใช้สื่อความกันระหว่างผู้ใช้กับโปรแกรม (หรือที่นิยมเรียก user interface) โดยทั่วไปจะ มีลักษณะเหมือนกับหน้าป้ทม์ของของเครื่องมือหรืออุปกรณ์ที่ใช้งานค้านการวัคทั่วๆ ไป โดยทั่วไปจะ ประกอบค้วย สวิตซ์ปิคเปิค, ปุ่มบิค, ปุ่มกค จอแสดงผลหรือแม้แต่ค่าที่ผู้ใช้สามารถกำหนด คังนั้นสำหรับ นักเขียนโปรแกรมที่คุ้นเคยกับการเขียนโปรแกรมประเภท Visual ทั้งหลายคงจะเข้าใจถ้าหากบอกว่า Front Panel นี้จะเปรียบเสมือนเป็น GUI ของโปรแกรมหรือ VI นั่นเอง

⋗ Untitled 1 Front Pane	-		
File Edit View Proje	ct Operate Tools Window Help		HTHM
수 & 🔵 II	15pt Application Font 🔻 🚛 🖬 🖓	🗳 • 🔍	? HIH 🗺
			~



2. ในหน้าต่าง Front Panel พิมพ์ก้นหากำว่า Modern ที่ช่องก้นหา

กลิกเลือก Boolean --> กลิกลาก Vertical Toggle Switch มาวางบนหน้าต่าง Front Panel ซึ่งขณะเดียวกันใน หน้าต่าง Block Diagram ก็จะถูกวางไปด้วย



ก้นหา Programming --> Instrument I/O --> Serial --> กลิกลาก Configure Port Write และ Close มาวาง ระนาบเดียวกันบนหน้าต่าง Block Diagram จากนั้นเชื่อมต่อสายสัญญาณ ที่ตัว Configure Port เข้าสู่ Write และ Close ดังรูป



คลิกขวาที่ VISA SERIAL คลิกเลือก All Control จากนั้นทำการลบ Port ทั้งหมดเหลือเพียง VISA I/O ไว้



กลิกขวาที่ VISA SERIAL กลิกเลือก All Constants จากนั้นทำการลบ Port ทั้งหมดเหลือเพียง 9600 ไว้ เท่านั้น







สร้าง Loop ให้กับ LED โดยคลิกชุดคำสัง Programming --> Structures --> While Loop คลิกลากคลุม LED และ Write ดังรูป



จากนั้นคลิกขวาที่ Loop Condition 極 --> Create Constant



รูป เมื่อต่อวงจรเสร็จสิ้น



จากนั้นเปิดโปรแกรม Arduino IDE เขียนโค้ดคำสั่ง และอัปโหลดลงไปยังบอร์ค Arduino ต่อไปนี้

00	test_led_conVIEW Arduino 1.8.9 —	\times
File	Edit Sketch Tools Help	
Ø		ø
te	st_led_conVIEW§	
1	int led = 13;	^
2	<pre>int ledl;</pre>	
3	<pre>void setup() {</pre>	
4	<pre>Serial.begin(9600);</pre>	
5	<pre>pinMode(led,OUTPUT);</pre>	
6		
7	}	
8		
9	<pre>void loop() {</pre>	
10	<pre>if (Serial.available())</pre>	
11	{	
12	<pre>ledl = Serial.read();</pre>	
13	<pre>if(ledl == 'a')</pre>	
14	{	
15	<pre>digitalWrite(led,HIGH);</pre>	
16	}	
17	else	
18	{	
19	<pre>digitalWrite(led,LOW);</pre>	
20	}	
21	}	
22		
23	}	\sim

จากนั้นไปคลิกที่ Port COM เลือก Port และคลิกอัปโหลด <u>หมายเหต</u>ุ ถ้าหากต่อวงจรไม่ถูกต้องลูกศรอัปโหลดจะเป็นรูปแตกหัก



รูปการแสดงผล เมื่ออัปโหลดคำสั่งลงบอร์คเรียบร้อยแล้ว



อธิบายโค้ดคำสั่ง





หน่วยการเรียนรู้ที่ 10 การประยุกต์ใช้งานบอร์ด Arduino

เรื่องย่อยที่ 1 ตัวอย่างการใช้งาน

หน่วยการเรียนรู้นี้เป็นการเขียนคำสั่งควบคุม Keypad membrane ซึ่งหลักการทำงานของ Keypad membrane แบบ 4 x 4 ชนิดนี้ประกอบไปด้วยปุ่ม 16 ปุ่ม ที่เรียงต่อกันเป็นเมตริกซ์แบบ 4 Row และ 4 Column ซึ่งการกดนั้นเพื่อให้หน้าสัมผัสที่เป็นชั้นสีแดงในรูปด้านล่างไปแตะกันทำให้เป็นการเชื่อมต่อกัน ทางไฟฟ้าไปอีกด้านหนึ่งของสวิตช์

หลักการในการตรวจสอบว่าผู้ใช้กำลังกดปุ่มอะไรอยู่นั้นก็ใช้วิธีการ scan ไปทีละ Column จนครบทุก Column แล้วนำมาตีความว่ามีการตอบสนองออกมาเป็นแบบใดบ้าง เช่น ถ้ามีการกดเลข 1 อยู่ ในขณะที่ จ่ายแรงดัน 5 โวลต์ ไปที่ Column ที่ 1 จะมีเพียง Row แรกเท่านั้นที่จะอ่านค่า แรงดันได้ High นอกนั้นจะ เป็น Low หรือ ถ้ามีการกดปุ่ม # อยู่ ขณะที่ Scan ไปแต่ละ Column นั้นจะไม่เจอแรงดัน High ที่ Row ใด จนกว่าจะ Scan ไปถึง Column ที่ 3 ซึ่งจะพบว่ามีการตอบสนองกลับมาจาก Row ที่ 4 นั้นเอง ดังนั้นเมื่อ พบว่าเป็นการ Scan Column ที่ 3 และมี Row 4 ตอบสนอง ก็คือปุ่ม '#' นั่นเอง

ซึ่งจังหวะการกดของคนแต่ละคน กดด้วยช่วงเวลาสั้นยาวไม่เท่ากัน ทำให้ต้องมีการตรวจสอบโดยมี ระยะเวลาเข้ามาเกี่ยวข้อง ไม่อย่างนั้นจะสับสนระหว่างการกดแช่ หรือ การกดแบบซ้ำปุ่ม นอกจากนั้นยัง ต้องมี debouncing ด้วยเพื่อป้องกันการรับค่าผิดจากการ bounce ของสวิตช์ อีกทั้งยังต้องมีการจดจำปุ่มที่กด ไปแล้วอีกว่ากดอะไรไปบ้าง กี่ครั้งแล้ว ก่อนที่จะมาประมวลผลตรวจสอบค่าที่ใส่เปรียบเทียบกับค่าที่ตั้งไว้ เป็นรหัส ซึ่งการตรวจสอบทั้งหมดนี้จะอยู่ในชุด Library (#include <Keypad.h>)

ทฤษฎี อาร์คุยโน่กับโมดูถกีร์แพค (Keypad) อปกรณ์ มีดังนี้

1. บอร์คไมโครคอนโทรลเลอร์อาร์คุยโน่ รุ่น UNO R3



2. คีย์แพด 4*4 1 ตัว



3. สายต่อวงจรระหว่างบอร์ด Arduino และ โปแกรม



การกำหนดขาในการต่อใช้งาน มีดังนี้

ขาที่ 1 ของคีย์แพค ต่อกับขาที่ 2 ของบอร์คไมโครคอนโทรลเลอร์ ขาที่ 2 ของคีย์แพค ต่อกับขาที่ 3 ของบอร์คไมโครคอนโทรลเลอร์ ขาที่ 3 ของคีย์แพค ต่อกับขาที่ 4 ของบอร์คไมโครคอนโทรลเลอร์ ขาที่ 4 ของคีย์แพค ต่อกับขาที่ 5 ของบอร์คไมโครคอนโทรลเลอร์ ขาที่ 5 ของคีย์แพค ต่อกับขาที่ 6 ของบอร์คไมโครคอนโทรลเลอร์ ขาที่ 6 ของคีย์แพค ต่อกับขาที่ 7 ของบอร์คไมโครคอนโทรลเลอร์ ขาที่ 7 ของคีย์แพค ต่อกับขาที่ 8 ของบอร์คไมโครคอนโทรลเลอร์ ขาที่ 8 ของคีย์แพค ต่อกับขาที่ 9 ของบอร์คไมโครคอนโทรลเลอร์



โค้คคำสั่ง มีคังนี้

```
File Edit Sketch Tools Help
```

```
Ø
   Ð
                                                                                  -
 key §
 1 #include <Keypad.h>
 2
 3 const byte numRows= 4;
 4 const byte numCols= 4;
 5
 6 char keymap[numRows][numCols]=
 7 {
8 {'1', '2', '3', 'A'},
9 {'4', '5', '6', 'B'},
10 {'7', '8', '9', 'C'},
11 {'*', '0', '#', 'D'}
12 };
13
14 byte rowPins[numRows] = {9,8,7,6};
15 byte colPins[numCols] = {5,4,3,2};
16
17 Keypad myKeypad= Keypad (makeKeymap (keymap), rowPins, colPins, numRows, numCols);
18
19 void setup()
20 {
21 Serial.begin(9600);
22 }
23
24 void loop()
25 {
26 char keypressed = myKeypad.getKey();
27 if (keypressed != NO KEY)
28 {
29
   if (keypressed == 'B')
30
   {
     Serial.println("เก่งมากจ้า");
31
32
   ł
33
    else
34
   {
35
     Serial.println("ปรับปรุงนะจ้า");
36 }
37 }
38 }
39
```

อธิบายโก้คกำสั่ง ดังนี้

- บรรทัดที่ 1 //การเรียกใช้ไลบรารีคีย์แพด
- บรรทัดที่ 3 //การกำหนดจำนวนคอลัมน์ จำนวน 4 คอลัมน์
- บรรทัดที่ 4 //การกำหนดจำนวนแถว จำนวน 4 แถว
- บรรทัดที่ 6-12 //การกำหนดปุ่มทั้งหมดบนคีย์แพด
- บรรทัดที่ 14 //การกำหนดการใช้งานร่วมกับคีย์แพดของแถว
- บรรทัดที่ 15 //การกำหนดขาในการใช้งานร่วมกับคีย์แพดของคอลัมน์
- บรรทัดที่ 17 //กำหนดค่าตัวแปรในการรับค่าของคีย์แพด
- บรรทัดที่ 21 //การกำหนดค่าเริ่มต้นการสื่อสารที่ 9600
- บรรทัดที่ 26 //ประกาศให้ตัวแปร key ตรวจสอบการกดคีย์แพด
- บรรทัดที่ 29 //ถ้ากดปุ่ม B บนคีย์แพด ให้แสดงข้อความบน Serial Monitor ว่า เก่งมากจ๊า
- บรรทัดที่ 35 //ถ้านอกเงื่อนไขพด ให้แสดงข้อความบน Serial Monitor ว่า ปรับปรุงนะจ๊า

หน่วยการเรียนรู้ที่ 10 การประยุกต์ใช้บอร์ด Arduino

เรื่องย่อยที่ 2 ตัวอย่าง โครงงาน

หน่วยการเรียนรู้นี้เป็นการเขียนตัวอย่างโครงงานที่มีชื่อว่า "รถมือหนีบ" ซึ่งเป็นการเขียนคำสั่ง ควบคุมมอเตอร์ ตัวขับมอเตอร์ และรีโมท คอนโทรล ซึ่งหลักการของอุปกรณ์มีคังนี้

<u>1. มอเตอร์</u>

มอเตอร์ ไฟฟ้าถูกนำไปใช้งานที่หลากหลายเช่น พัคลมอุตสาหกรรม เครื่องเป่า ปั้ม เกรื่องมือ เครื่องใช้ในครัวเรือน และคิสก์ไครฟ์ มอเตอร์ไฟฟ้าสามารถขับเคลื่อนโดยแหล่งจ่ายไฟกระแสตรง (DC) เช่น จากแบตเตอรี่, ยานยนต์หรือวงจรเรียงกระแส หรือจากแหล่งจ่ายไฟกระแสสลับ (AC) เช่น จากไฟบ้าน อินเวอร์เตอร์ หรือ เครื่องปั่นไฟ มอเตอร์ขนาดเล็กอาจจะพบในนาฬิกาไฟฟ้า มอเตอร์ทั่วไปที่มีขนาดและ คุณลักษณะมาตรฐานสูงจะให้พลังงานกลที่สะดวกสำหรับใช้ในอุตสาหกรรม มอเตอร์ไฟฟ้าที่ใหญ่ที่สุดใช้ สำหรับการใช้งานลากจูงเรือ และ การบีบอัดท่อส่งน้ำมันและปั้มป์สูบจัดเก็บน้ำมันซึ่งมีกำลังถึง 100 เมกะ วัตด์ มอเตอร์ไฟฟ้าอาจจำแนกตามประเภทของแหล่งที่มาของพลังงานไฟฟ้าหรือตามโครงสร้างภายในหรือ ตามการใช้งานหรือตามการเคลื่อนไหวของเอาต์พุต และอื่น ๆ อุปกรณ์เช่นขดลวดแม่เหล็กไฟฟ้าและสำโพง ที่แปลงกระแสไฟฟ้าให้เป็นการเคลื่อนไหว แต่ไม่ได้สร้างพลังงานกลที่ใช้งานได้ จะเรียกถูกว่า actuator และ transducer ตามลำคับ คำว่ามอเตอร์ไฟฟ้านั้น ต้องใช้สร้างแรงเชิงเส้น(linear force) หรือ แรงบิด (torque) หรือเรียกอีกอย่างว่า หมุน (rotary) เท่านั้น



โครงสร้างมอเตอร์

- โรเตอร์

ในมอเตอร์ไฟฟ้า ส่วนที่เกลื่อนที่คือโรเตอร์ ซึ่งจะหมุนเพลาเพื่อจ่ายพลังงานกล โรเตอร์มักจะมี ขคลวคตัวนำพันอยู่โคยรอบ ซึ่งเมื่อมีกระแสไหลผ่าน จะเกิดอำนาจแม่เหล็กที่จะไปทำปฏิกิริยากับ สนามแม่เหล็กถาวรของสเตเตอร์ ขับเพลาให้หมุนได้ อย่างไรก็ตามโรเตอร์บางตัวจะเป็นแม่เหล็กถาวร และสเตเตอร์จะมีขคลวคตัวนำสลับที่กัน

- สเตเตอร์

้จะเป็นส่วนที่อยู่กับที่ซึ่งจะประกอบด้วยโครงของมอเตอร์ แกนเหล็กสเตเตอร์ และขดลวด

- ช่องว่างอากาศ

ระหว่างโรเตอร์และสเตเตอร์จะเป็นช่องว่างอากาศ ซึ่งจะต้องมีขนาดเล็กที่สุดเท่าที่จะเป็นไปได้ ช่องว่างขนาดใหญ่จะมีผลกระทบทางลบอย่างมากต่อประสิทธิภาพการทำงานของมอเตอร์ไฟฟ้า

- ขคลวค

งคลวคจะพันโดยรอบเป็นคอยล์ ปกติจะพันรอบแกนแม่เหล็กอ่อนที่เกลือบฉนวน เพื่อให้เป็น ขั้วแม่เหล็กเมื่อมีกระแสไฟฟ้าไหลผ่าน

- ตัวสับเปลี่ยน

ด้วสับเปลี่ยนเป็นกลไกที่ใช้ในการสลับอินพุทของมอเดอร์ AC และ DC เพื่อให้กระแสที่ไหลใน ขดลวดในโรเตอร์ไหลทางเดียวตลอดเวลาในระหว่างการหมุน ประกอบด้วยวงแหวนลื่น(slip ring) ขึ้นเล็กๆ แขกจากกันด้วยฉนวนวงแหวนนี้ยังแขกจากเพลาของมอเตอร์ด้วยฉนวนอีกด้วย วงแหวนแต่ละคู่ที่อยู่ตรง ข้ามกันจะเป็นขดลวดหนึ่งชุด กระแสที่จ่ายให้มัดข้าวต้มหรือที่เรียกว่า armature ของมอเตอร์จะถูกส่งผ่าน แปรงถ่าน(brush) สองตัวที่แตะอยู่กับตัวสับเปลี่ยนแต่ละด้านที่กำลังหมุนอยู่ ซึ่งจะทำให้กระแสจาก แหล่งจ่ายไฟ AC ที่ไหลกลับทางไหลในขดลวดทิศทางเดียวในขณะที่โรเตอร์หมุนจากขั้วหนึ่งไปอีกขั้วหนึ่ง ในกรณีที่ไม่มีกระแสแหล่งจ่ายไม่กลับทางมอเตอร์จะเบรกหยุดอยู่กับที่ ในแง่ของความก้าวหน้าที่สำคัญ ในช่วงไม่กี่ทศวรรษที่ผ่านมา อันเนื่องมาจากเทคโนโลยีที่ดีขึ้นในการควบคุมอิเล็กทรอนิกส์ มอเตอร์ เหนี่ยวนำที่กวบคุมโดยไม่ใช้เซ็นเซอร์และมอเตอร์ที่มีสนามแม่เหลีกถาวร มอเตอร์ที่มีดัวสับเปลี่ยนแบบ กลไกไฟฟ้า กำลังถูกแทนที่เพิ่มขึ้นด้วยมอเตอร์เหนี่ยวนำที่ใช้ตัวสับเปลี่ยนภายนอกและมอเตอร์แบบ แม่เหล็กถาวร

<u>2. ตัวขับมอเตอร์ (L298N Dual H-Bridge Motor Controller)</u>

การใช้งาน :

L298N เป็นชุดขับมอเตอร์ชนิด H-Bridge ซึ่งส่วนใหญ่จะถูกนำไปใช้ในการควบคุมทิศทางและ ความเร็วของมอเตอร์ซึ่งสามารถควบคุมมอเตอร์ได้ทั้งหมด 2 Channel

หลักการทำงาน:

วงจร H-Bridge ของ L298N จะขับกระแสเข้ามอเตอร์ ตามขั้วที่กำหนดด้วยลอจิกเพื่อควบคุม ทิศทาง ส่วนความเร็วของมอเตอร์นั้นจะถูกควบคุมด้วยสัญญาณ (PWM Pulse Width Modulation) ซึ่งต้องมี การปรับความถี่ให้เหมาะสมกับมอเตอร์ที่ผู้เขียนโปรแกรมใช้



ขาต่างๆ

Out 1: ช่องต่อขั้วไฟของมอเตอร์ A	Out 3: ช่องต่อขั้วไฟของมอเตอร์ B
Out 2: ช่องต่อขั้วไฟของมอเตอร์ A	Out 4: ช่องต่อขั้วไฟของมอเตอร์ B

12V: ช่องจ่ายไฟเลี้ยงมอเตอร์ 12V (ต่อได้ตั้งแต่ 5V ถึง 35V)

GND: ช่องต่อไฟลบ (Ground)

5V: ช่องจ่ายไฟเลี้ยงมอเตอร์ 5V (หากมีการต่อไฟเลี้ยงที่ช่อง 12V แล้วช่องนี้จะทำหน้าที่จ่ายไฟ ออก เป็น 5V Output

สามารถต่อไฟจากช่องนี้ไปเลี้ยงบอร์ค Arduino ได้

ENA: ช่องต่อสัญญาณ PWM สำหรับมอเตอร์ A IN1: ช่องต่อสัญญาณลอจิคเพื่อควบคุมทิศทางของมอเตอร์ A IN2: ช่องต่อสัญญาณลอจิคเพื่อควบคุมทิศทางของมอเตอร์ A IN3: ช่องต่อสัญญาณลอจิคเพื่อควบคุมทิศทางของมอเตอร์ B IN4: ช่องต่อสัญญาณลอจิคเพื่อควบคุมทิศทางของมอเตอร์ B ENB: ช่องต่อสัญญาณ PWM สำหรับมอเตอร์ B

สเปกทางเทคนิค:

Dual H bridge Drive Chip : L298N แรงดันสัญญาณลอจิก : 5V Drive voltage: 5V-35V กระแสของสัญญาณลอจิก : 0-36mA กระแสขับมอเตอร์ : สูงสุดที่ 2A (เมื่อใช้มอเตอร์เดียว) กำลังไฟฟ้าสูงสุด : 25W ขนาด : 43 x 43 x 26 มิลลิเมตร น้ำหนัก : 26 กรัม

*มี Power Supply 5V ในตัว สามารถจ่ายไฟออกจากช่อง 5V (เพื่อจ่ายให้บอร์ค Arduino) ได้เมื่อต่อไฟเลี้ยง เข้าที่ช่อง 12V

ตารางการทำงาน มอเตอร์ A

ENA	IN1	IN2	คำอธิบาย
0	N/A	N/A	มอเตอร์ A ไม่ทำงาน
0	LOW	LOW	มอเตอร์ A หยุดการทำงานแบบหมุนอิสระ (แบบไม่เบรค)
0-255	LOW	HIGH	มอเตอร์ A ทำงานและหมุนกลับหลังด้วยความเร็วที่ ENA
0-255	HIGH	LOW	มอเตอร์ A ทำงานและหมุนเดินหน้าด้วยกวามเร็วที่ ENA
0	HIGH	HIGH	มอเตอร์ A หยุดการทำงานแบบทันที (แบบเบรก)

ตารางการทำงาน มอเตอร์ B

ENA	IN1	IN2	คำอธิบาย
0	N/A	N/A	มอเตอร์ B ไม่ทำงาน
0	LOW	LOW	มอเตอร์ B หยุคการทำงานแบบหมุนอิสระ (แบบไม่เบรค)
0-255	LOW	HIGH	มอเตอร์ B ทำงานและหมุนกลับหลังด้วยกวามเร็วที่ ENB
0-255	HIGH	LOW	มอเตอร์ B ทำงานและหมุนเดินหน้าด้วยความเร็วที่ ENB
0	HIGH	HIGH	มอเตอร์ B หยุดการทำงานแบบทันที (แบบเบรก)

<u>3. รีโมตคอนโทรล</u>

รีโมตคอนโทรล (อังกฤษ: remote control) คือเครื่องมืออิเล็กทรอนิกส์ชนิดหนึ่ง ใช้สำหรับควบคุม การดำเนินการของสิ่งประดิษฐ์หรือเครื่องจักรต่างๆ โดยเฉพาะเครื่องใช้ไฟฟ้าภายในบ้านเช่น โทรทัศน์ เครื่องเสียง เครื่องเล่นดีวีดี จากระยะไกล โดยไม่ใช้สายไฟเป็นตัวส่งสัญญาณ แต่ใช้อินฟราเรคแทน (หรือใช้ สัญญาณวิทยุแต่พบได้น้อย) ใช้พลังงานจากแบตเตอรี่ขนาดเล็กไม่กี่ก้อนเท่านั้น มีขนาดเหมาะมือ และมีปุ่ม ฟังก์ชันต่างๆ อยู่ครบครัน

รี โมตคอนโทรล เป็นการเรียกย่อมาจาก รี โมตคอนโทรลเลอร์ (remote controller) อีกต่อหนึ่ง และ สามารถเรียกย่อลงได้อีกเหลือเพียง รี โมต (อ่านว่า รี-โหมด) ในภาษาไทยสามารถใช้คำไทยแทนได้ว่า เครื่อง ควบคุมระยะไกล หรือ อุปกรณ์ควบคุมระยะไกล
รี โมทคอนโทรล จะสามารถสั่งงานได้ ต้องประกอบด้วย2สิ่งนี้คือ รหัส และ ตัวส่งสัญญาณ 1.รหัส (Code) เป็นระบบสัญญาณจะนำออกได้ต้องมีตัวคลื่นพานำออกไป 2.ตัวส่งสัญญาณ(Carrier) ตัวรับสัญญาณ เพื่อถอดหรือรับรหัสที่ถูกส่งมาใช้ควบคุมอุปกรณ์ต่างๆ



ทฤษฎี อาร์คุยโน่กับมอเตอร์ ตัวขับมอเตอร์ และรี โมตคอนโทรล อุปกรณ์ มีดังนี้







การต่อวงจร

1. การต่อตัวขับมอเตอร์ A

ต่อสาย VCC (สายไฟสีแดงจากมอเตอร์ล้อ) ไปที่ช่อง Out 1 ในบอร์ดขับมอเตอร์ L298N ต่อสาย GND (สายไฟสีดำจากมอเตอร์ล้อ) ไปที่ช่อง Out 2 ในบอร์ดขับมอเตอร์ L298N 2. การต่อตัวขับมอเตอร์ B

ต่อสาย GND (สายไฟสีคำจากมอเตอร์ล้อ) ไปที่ช่อง Out 3 ในบอร์คขับมอเตอร์ L298N ต่อสาย VCC (สายไฟสีแคงจากมอเตอร์ล้อ) ไปที่ช่อง Out 4 ในบอร์คขับมอเตอร์ L298N

3. การต่อทิศทางของล้อ (ใช้สายจัมเปอร์ เมีย-เมีย)

มอเตอร์ A

D9 ที่บอร์ค Shield ไปที่ช่อง IN 1 ในบอร์คขับมอเตอร์ L298N D6 ที่บอร์ค Shield ไปที่ช่อง IN 2 ในบอร์คขับมอเตอร์ L298N มอเตอร์ B D11 ที่บอร์ค Shield ไปที่ช่อง IN 3 ในบอร์คขับมอเตอร์ L298N D10 ที่บอร์ค Shield ไปที่ช่อง IN 4 ในบอร์คขับมอเตอร์ L298N

4. การต่อไฟเข้ายังบอร์ค Shield (ใช้สายจัมเปอร์ ผู้-ผู้)

เส้นที่ 1 จาก ไฟ 12 โวลต์ ในบอร์คขับมอเตอร์ L298N ไปที่ช่อง VCC ของบอร์ค Shield เส้นที่ 2 จาก ไฟ GND ในบอร์คขับมอเตอร์ L298N ไปที่ช่อง GND ของบอร์ค Shield

5. การต่อมือหนีบ

ดึงสายจัมเปอร์จาก Servo 180 องศา โดยสายจัมเปอร์สีส้มอ่อนเสียมที่ D4 จากนั้นสายไฟ และ GND ก็เสียบเรียงตามช่อง

6. การต่อชุครี โมทคอนโทรล

IR receiver (ใช้สายจัมเปอร์เมีย-เมีย 3 เส้น) ที่ช่อง D5 จากนั้นสายไฟ และ GND ก็เสียบเรียงตามช่อง

โค้ดคำสั่ง

อันดับแรกให้ทำการเขียนโค้ดเพื่อรับก่าปุ่มต่างๆ ก่อน ซึ่งโค้คมีดังต่อไปนี้





หน้าต่าง Serial Monitor

้<mark>ข้อควรจ</mark>ำ ให้ใช้รหัสที่รับได้จากรี โมตขณะที่ทำการทดลองกดปุ่มสร้างเงื่อนไขการทดลองรถมือหนีบ

โค้ดคำสั่งโครงงาน เรื่อง รถมือหนีบ

```
File Edit Sketch Tools Help
```

```
car_remot§
 1 #include <IRremote.h>
 2 #include <Servo.h>
 3 #define pinServo 4
 4
 5 Servo myServo;
 6
 7 int RECV PIN = 5;
 8 int led = 13;
 9 String codeLeft = "8c22657b"; //ปุ่มลูกศรช้าย
10 String codeRight = "449e79f"; //ปุ่มลูกศรชวา
11 String codeForword = "3d9ae3f7"; //ปุ่มลูกศรหน้า
12 String codeTurn = "lbc0157b"; //ปุ่มลูกศรหลัง
13 String codeStop = "488f3cbb"; //1)A ok
14 String codeIn ="cl0le57b"; //ปุ่ม * มือกลหนึบ
15 String codeOut ="f0c41643"; //ปุ่ม # มือกลอา
16
17 IRrecv irrecv(RECV_PIN);
18 decode_results results;
19
20 String vl="";
21 String vstatus = " ";
22
23 int motor left[] = {9, 6};
                                 // ----> motor left[0]= 9
                                  // ----> motor left[1]= 6
24
25
26 int motor right[] = {11, 10}; // ----> motor right[0]= 11
27
                                  // ----> motor_right[1]= 10
28
29 void setup() {
30 Serial.begin(9600);
31
     myServo.attach(pinServo); // ----> Servo ซาที่ 4
32
33
 34 //Setup Motors
    for (int i = 0; i < 2; i++)
 35
 36
     - {
 37
      pinMode (motor_left[i], OUTPUT); // ----> motor_left[0],OUTPUT ###### บาที่ 9
 38
                                       // ----> motor_left[1],OUTPUT ###### บาที่ 6
 39
      pinMode(motor_right[i], OUTPUT); // ----> motor_right[0],OUTPUT ###### ยาที่ 11
 40
                                         // ----> motor right[1],OUTPUT  ####### บาที่ 10
41
 42
     1
     irrecv.enableIRIn(); // Start the receiver
43
44 }
45
46 void stopnow()
47 {
48 digitalWrite(motor left[0], LOW); // ----> motor left บาที่ 9 หยุดหมุน
49 digitalWrite(motor_left[1], LOW); // ----> motor_left บาที่ 6 หยุดหมุน
50 digitalWrite (motor_right[0], LOW); // ----> motor_left บาที่ 11 หยุดหมุน
51 digitalWrite (motor_right[1], LOW); // ----> motor_left บาที่ 10 หยุดหมุน
52 }
 53
```

```
54 void drive forward()
 55 {
 56 digitalWrite(motor left[0], LOW);
 57
     digitalWrite(motor_left[1], HIGH);
 58 digitalWrite(motor_right[0], LOW);
 59 digitalWrite(motor_right[1], HIGH);
 60 }
 61
 62 void turn_around()
 63
     {
 64
       digitalWrite(motor_left[0], HIGH);
 65
      digitalWrite(motor left[1], LOW);
      digitalWrite(motor right[0], HIGH);
 66
       digitalWrite(motor right[1], LOW);
 67
 68
 69
     }
 70
 71 void turn right()
 72
      {
 73
       digitalWrite(motor left[0], LOW);
 74
       digitalWrite(motor_left[1], HIGH);
       digitalWrite(motor_right[0], LOW);
 75
 76
       digitalWrite(motor_right[1], LOW);
 77
       delay(200);
 78
       stopnow();
 79
      }
 80
81
     void turn_left()
82
     {
83
      digitalWrite(motor_left[0], LOW);
84
     digitalWrite(motor_left[1], LOW);
85
     digitalWrite(motor right[0], LOW);
86
      digitalWrite(motor right[1], HIGH);
87
      delay(200);
88
      stopnow();
89
    }
90
 91
     void camp_out()
92
    {
93
     myServo.write(80);
94
      delay(1000);
95
     }
96
97
      void camp_in()
98
     {
     myServo.write(30);
99
100
      delay(1000);
101
     }
102
```

```
103 void loop()
104 {
105
106
     if (irrecv.decode(&results)) {
107
     vl = String(results.value, HEX);
108
     Serial.println(vl);
109
       if (vl == codeForword) { //
110
            vstatus = "F";
111
             drive forward();
112
       }
       if (vl == codeTurn) { //
113
            vstatus = "T";
114
115
            turn around();
116
       }
       if (vl == codeLeft) { //
117
           vstatus = "L";
118
119
            turn left();
120
       }
       if (vl == codeRight) { //
121
122
            vstatus = "R";
123
            turn_right();
124
       }
125
       if (vl == codeStop) { //
126
            vstatus = "S";
127
            stopnow();
128
        }
       if (vl == codeIn) { //
129
130
            vstatus = "I";
131
            camp in();
132
        1
       if (vl == codeOut) { //
133
            vstatus = "O";
134
135
            camp_out();
136
       }
137
     irrecv.resume(); // Receive the next value
138
     }
139
140
     Serial.println(vstatus);
141 }
142
```

้ลักษณะการทำงาน โดยรี โมตจะควบคุมการเกลื่อนที่ของรถและการหนีบสิ่งของๆ มือกล

- 1. เมื่อกคปุ่มลูกศรซ้าย รถจะเคลื่อนที่ไปทางซ้าย
- 2. เมื่อกดปุ่มลูกศรขวา รถจะเคลื่อนที่ไปทางขวา
- 3. เมื่อกดปุ่มลูกศรขึ้น รถจะเคลื่อนที่ไปด้านหน้า
- 4. เมื่อกดปุ่มลูกศรลง รถจะเคลื่อนที่ไปด้านหลัง
- 5. เมื่อกดปุ่ม ok รถจะหยุคเคลื่อนที่
- 6. เมื่อกดปุ่ม * มือกลจะหนีบ
- 7. เมื่อกดปุ่ม # มือกลจะอ้า

หน่วยการเรียนรู้ที่ 10 การประยุกต์ใช้บอร์ด Arduino

เรื่องย่อยที่ 3 ตัวอย่างโครงงาน IOT

ปัจจุบันได้เริ่มมีการนำเกษตรกรรมบทบาทมากขึ้นในด้านการพัฒนาประเทศอย่างยั่งยืน เพื่อช่วย ขับเคลื่อนเศรษฐกิจใน Thailand 4.0 ซึ่งเรื่องที่เป็นจุดเน้นมากที่สุดคือ กลุ่มอาหาร เกษตร และ เทคโนโลยีชีวภาพ จุดเริ่มต้นหรือที่มาส่วนหนึ่งของ Smart Farmer คือ การไม่ทำร้ายธรรมชาติ ใช้ทรัพยากร เท่าที่จำเป็น ทำแล้วต้องสบายขึ้นเรื่อย ๆ ไม่ใช่ยิ่งทำยิ่งเหนื่อย เช่น การมีพื้นที่เล็ก ๆ แต่สามารถออกแบบให้ ปลูกแบบผสมผสานและเกื้อกูลกันได้ ต้องใช้เทคโนโลยีเป็น ซึ่งก็ถูกต้อง เพราะคนที่จะเป็น Smart Farmer ต้องเชื่อมโลกได้เอง Smart Farmer ต้องเข้าใจตั้งแต่กระบวนการผลิต การบริหารจัดการ เข้าใจธรรมชาติ และเข้าใจเทคโนโลยี

แนวคิดของ Internet of Things (IoT)

Internet of Things (IoT) คือ "อินเตอร์เน็ตในทุกสิ่ง" หมายถึง การที่อุปกรณ์ต่างๆ สิ่งต่างๆ ได้ถูก เชื่อมโยงทุกสิ่งทุกอย่างสู่โลกอินเตอร์เน็ต ทำให้มนุษย์สามารถสั่งการควบคุมการใช้งานอุปกรณ์ต่างๆ ผ่าน ทางเครือข่ายอินเตอร์เน็ต เช่น การเปิด-ปิด อุปกรณ์เครื่องใช้ไฟฟ้า (การสั่งการเปิดไฟฟ้าภายในบ้านด้วยการ เชื่อมต่ออุปกรณ์ควบคุม เช่น มือถือ ผ่านทางอินเตอร์เน็ต) รถยนต์ โทรศัพท์มือถือ เครื่องมือสื่อสาร เครื่องมือทางการเกษตร อาการ บ้านเรือน เครื่องใช้ในชีวิตประจำวันต่างๆ ผ่านเครือข่ายอินเตอร์เน็ต เป็นต้น



<u>หลักการทำงานของ IoT</u>

การทำงานของ IoT สามารถแบ่งได้เป็น 3 ขั้นตอนตามลำดับ ได้แก่ การเก็บข้อมูล (data collection) การตรวจทานและส่งต่อข้อมูล (collate and transfer) การวิเคราะห์ข้อมูลและสั่งการทำงาน (analyze data and take action) ดังนั้น ข้อมูล (data) จึงเปรียบเสมียนปุ๋ยหรือธาตุอาหารที่ต้องใส่เข้าไปในคอมพิวเตอร์หรือ ระบบการประมวลผลนั้นเอง

การเก็บข้อมูล (data collection) ข้อมูลที่ถูกเก็บจากอุปกรณ์ต่างๆ ผ่านเครื่องมืออีเล็ก โทรนิกส์ สำหรับวัดค่าหรือข้อมูลที่เราสนใจ ซึ่ง โดยทั่วไปมี 3 ชนิด ได้แก่ เซ็นเซอร์ (sensor) ซึ่งออกแบบมาให้วัดค่า ต่างๆ เช่น อุณหภูมิ ความชื้น ปริมาณแสง เป็นต้น เสาอากาศ (antenna) สำหรับรับสัญญาณจากคลื่นต่างๆ อุปกรณ์ IoT สุดท้าย คือ ไมโครคอนโทรลเลอร์ (microcontroller) ซึ่งเปรียบเสมือนคอมพิวเตอร์ขนาดเล็กที่ สามารถเก็บข้อมูล และสั่งการให้อุปกรณ์อีเล็กโทรนิกส์ต่างให้ทำงานตามคำสั่งถูกเขียนจาก ภาษาคอมพิวเตอร์ เช่น ภาษา C ภาษา JAVA หรือ ภาษา python และภาษาอื่นๆ เป็นต้น

การตรวจทานและส่งต่อข้อมูล (collate and transfer) ข้อมูลถูกเก็บจะถูกส่งไปที่พื้นที่เก็บข้อมูลหรือ ศูนย์เก็บข้อมูล (hub หรือ gateway) เช่น พื้นที่เก็บข้อมูลบนอินเตอร์เน็ต หรือ ส่งข้อมูลให้ไปเก็บใน กอมพิวเตอร์ เป็นต้น หรืออาจส่งข้อมูลไปตรวจสอบความถูกต้องในคอมพิวเตอร์ส่วนตัวและเก็บใน อินเตอร์เน็ต

การวิเคราะห์ข้อมูลและสั่งการทำงาน (analyze data and take action) หลังจากตรวจสอบความ ถูกต้องของข้อมูลแล้วก็จะทำการประมวลผล หรือวิเคราะห์ข้อมูลในระบบอินเตอร์เน็ตหรือคอมพิวเตอร์ ส่วนตัว ข้อมูลที่ได้จากการวิเคราะห์เสร็จแล้ว ระบบอินเตอร์เน็ตหรือคอมพิวเตอร์ก็จะทำการสั่งอุปกรณ์ให้ ทำตามกำสั่งที่กำหนดไว้ เช่น ข้อมูลอุณหภูมิอากาศที่ถูกวัดจากเซ็นเซอร์วัดก่าอุณหภูมิ ได้ถูกส่งไปที่สมาร์ท โฟนเพื่อทดสอบความถูกต้อง จากนั้นสมาร์ทโฟนได้ทำการวิเคราะห์ข้อมูลอุณหภูมิแล้วปรากฏว่าอากาศ ร้อนเกินไป สมาร์ทโฟนก็จะสั่งการให้สปริงเกอร์ฉีดน้ำเพื่อให้ลดอุณหภูมิให้ลดลง เป็นต้น

ตัวอย่างโครงงาน Internet of Things (IoT) ระบบตรวจวัดอุณหภูมิและความชื้น

การติดตั้งและใช้งานแอพลิเคชัน Blynk

Blynk เป็นแอปพลิเคชันบนโทรศัพท์มือถือ (Mobile Application) สำหรับระบบ Internet of Things (IoT) สามารถติดตั้งได้ทั้งระบบ Android และ ios จึงทำให้สามารถควบคุมการทำงานของบอร์ค ESP8266, Arduino, และRaspberry Pi ได้ผ่านระบบอินเทอร์เน็ต โดยแอปพลิเคชันผู้ใช้สามารถเลือก Widget ที่ ต้องการใช้งานมาวางบนหน้าจอโดยที่ทำงานแบบกราฟิก (Graphic Interface) ทำให้สามารถสร้าง dashboard เพื่อควบคุมอุปกรณ์ IoT ได้ง่ายมากขึ้น 1. ติดตั้งแอพลิเคชัน Blynk บนมือถือ



2. เปิดแอพลิเคชันขึ้นมา หลังจากนั้นให้สร้างบัญชีในการใช้งานแอพลิเคชันขึ้นมาใหม่ โดยแตะที่ Create

New Account



3. สมัคร Account ทำการกรอกข้อมูลในช่อง Email และ Password หลังจากนั้นให้แตะที่ SignUp



4. สร้างโปรเจ็คต์ใหม่ โดยแตะที่ New Project



5. กำหนดค่าในโปรเจ็กต์ที่สร้างขึ้นใหม่ ดังนี้

- ตั้งชื่อ Project Name เรื่อง IoT Farm

- ช่อง CHOOSE DEVICE เป็นการเลือกอุปกรณ์ บอร์ค Wi-Fi ที่ต้องการเชื่อมต่อ ให้แตะลูกศรซิ้ลง แล้วเลือก ESP8266

- ช่อง CONNECTION TYPE เป็นการเลือกชนิดของการเชื่อมต่อ ให้แตะลูกศรชี้ลงแล้วเลือก Wi-Fi

第1212年9日、中一〇日月2日第125日	5 1714 n. 9	500 (0 m 🚥	51124 a. •	0007 m
← Create New Project	← Create New Project		← Create New Project	
Project Name	Project Name		Project Name	
CHOOSE DEVICE	Denter (Devel)		OPPOSE SEVER	
ESP8266 ↓	ESP8266	$\overline{\Psi}$	ESP826	
THEME	condition more		NAME.	
DARK LIGHT	Wi-Fi	\downarrow	1540x	
	Here			
	Select connection type	OK	Select your hardware	OK
		chipKIT-Uno32 Dinistrumo Qak		Uno32 10 Oak
	Ethernet		ESP32 Dev Board	
	Wi-Fi		ESP8266	
	USB		Generic Board	
Create	GSM		Intel Er	dison
	Bluetooth		Intel Galilieo	
	≡ ∽	5	= 0	-

- ช่อง THEME ให้เลือกโหมดการแสดงผลหน้าจอ

DARK จอแบบคำ LIGHT จอแบบขาว เมื่อกำหนดค่าโปรเจ็กเสร็จแล้ว แตะที่ Create

SI-17:15 L. Φ	n. 🗢 🛛 🙃 🖉 🖛 🗰	
← Crea	ate New Projec	t
loT Farr	n	
CHOOSE DEVICE		
	ESP8266	Ŷ
CONNECTION TYPE		
	Wi-Fi	\downarrow
THEME		
DAF	ak 👘	LIGHT
	Create	
=	ŝ	5

6. จะเข้าสู่หน้าจอการแจ้งเตือนว่า ได้ส่งรหัส Token ของแอพลิเคชัน Blynk ไปให้ในอีเมล์แล้ว แตะที่ OK



7. ตรวจสอบอีเมล์ที่ระบุไว้ตอนสร้าง Account จะมีข้อความแจ้งรหัส Token มาให้ ตรง Auth Token



8. จะเข้าสู่หน้าจอในการออกแบบ ให้แตะที่เครื่องหมาย + 1 ครั้ง จะปรากฎหน้าต่าง Widget Box ขึ้นมาให้ เลือกใช้ไปออกแบบ



9. แตะที่ Button 1 ครั้ง จะปรากฏหน้าจอ Button Settings ให้กำหนดค่าดังนี้

- Title : กำหนดข้อกวามบนหน้าจอ "Auto/Manual"
- PIN : เลือกเป็น V8
- MODE : เถือกเป็น SWITCH
- กำหนดช่วงข้อมูลเป็น 0 1
- ช่อง OFF LABEL : กำหนดหน่วยให้เป็น "Manual"
- ช่อง ON LABEL : กำหนดหน่วยให้เป็น "Auto"

2117314.0 BB(3.9		P 111.000	S (2204.00-	0.01 (0.00000)
	Widget Box		← Styled Button	Settings i
		+ Add	(arts	*
	Button		📾 Auto/Manual	
	Styled Button		puteur:	
			V8 0	1
	Vertical Slider		PUSH) SWITCH
	C Timer		ouser statis	
	Joystick		Manual 🦲	Auto
	zeRGBa			
	Step H	I	T T	
	≡ ≎	5	≡ 0	5

10. แตะที่หน้าจอออกแบบ 1 ครั้ง หลังจากนั้นแตะที่ Gauge 1 ครั้ง จะปรากฎหน้าจอ Gauge Settings ให้ กำหนดค่าดังนี้

- Gauge : กำหนดข้อความบนหน้าจอ Gauge เป็น "ความชิ้นดิน"
- PIN : เถือกเป็น V12
- กำหนดช่วงข้อมูลเป็น 0 100
- ช่อง LABEL : กำหนดหน่วยให้เป็น "%RH"



11. แตะที่หน้าจอออกแบบ 1 ครั้ง หลังจากนั้นแตะที่ Button 1 ครั้ง จะปรากฎหน้าจอ Button Settings ให้ กำหนดค่าดังนี้

- Title : กำหนดข้อความบนหน้าจอ "Springle"
- PIN : เลือกเป็น V4
- MODE : เลือกเป็น SWITCH
- กำหนดช่วงข้อมูลเป็น 0 1



12. แตะที่หน้าจอออกแบบ 1 ครั้ง หลังจากนั้นแตะที่ Slider 1 ครั้ง จะปรากฏหน้าจอ Slider Settings ให้ กำหนดค่าดังนี้

- Title : กำหนดข้อความบนหน้าจอ "Slider"
- PIN : เลือกเป็น V5
- MODE : เถือกเป็น SWITCH
- กำหนดช่วงข้อมูลเป็น 0 100





<u>อุปกรณ์ที่ใช้ในการทคลอง IoT</u>



relay 2 channel

บอร์ด NodeMCU ESP8266 (บอร์ด Wi-Fi)



สายจั้มเปอร์ ผู้-ผู้



สาย<mark>จ</mark>ั้มเปอร์ ผู้-เมีย





breadboard

Soil Moisture Sensor



<u>การต่อวงจรอุปกรณ์ระบบตรวจวัดอุณหภูมิและความชื้น</u>

1. เสียบ NodeMCU ESP8266 บน Breadboard



 ใช้สาย Jumper แบบผู้-ผู้ เสียบบน Breadboard แถวเดียวกับงา GND ของ NodeMCU ส่วนปลาย สายเสียบเข้ากับช่อง - บน Breadboard

 ใช้สาย Jumper แบบผู้-ผู้ เสียบบน Breadboard แถวเคียวกับขา Vin ของ NodeMCU ส่วนปลาย สายเสียบเข้ากับช่อง + บน Breadboard



4. Soil Moisture Sensor ประกอบตัว MS Sensor หรือผังบอร์ค ตัววัคอุณหภูมิความชื้นคิน และสาย

Jumper

- เสียบสาย Jumper เข้ากับผังบอร์ค MS Sensor โคยเสียบสายเส้นหนึ่งเข้าขั้วบวก สายเส้นหนึ่ง

เข้าขั้วลบ





- ผังบอร์ค MS Sensor อีกฝั่ง ประกอบด้วยขา A0(ขา Analog) ,ขา D0 (ขา Digital), ขา GND, ขา

VCC



5. เสียบสาย Jumper (ผู้-เมีย) เข้ากับขา A0 ของ MS Sensor ส่วนปลายสายเสียบบน Breadboard แถวเคียวกับขา A0 ของ NodeMCU

6. เสียบสาย Jumper (ผู้-เมีย) เข้ากับขา GND ของ MS Sensor ส่วนปลายสายเสียบเข้ากับแถวช่อง -บน Breadboard

7. เสียบสาย Jumper (ผู้-เมีย) เข้ากับขา VCC ของ MS Sensor ส่วนปลายสายเสียบเข้ากับแถวช่อง + บน Breadboard



8. relay 2 channel

- เสียบสาย Jumper แบบผู้-เมีย เข้ากับขา In1 ของตัว Relay ส่วนปลายสายเสียบบน Breadboard แถว เดียวกับขา D2 ของ NodeMCU

- เสียบสาย Jumper แบบผู้-เมีย เข้ากับขา GND ของตัว Relay ส่วนปลายสายเสียบเข้ากับแถวช่อง - บน Breadboard

เสียบสาย Jumper แบบผู้-เมีย เข้ากับขา VCC ของตัว Relay ส่วนปลายสายเสียบเข้ากับแถวช่อง + บน
 Breadboard

