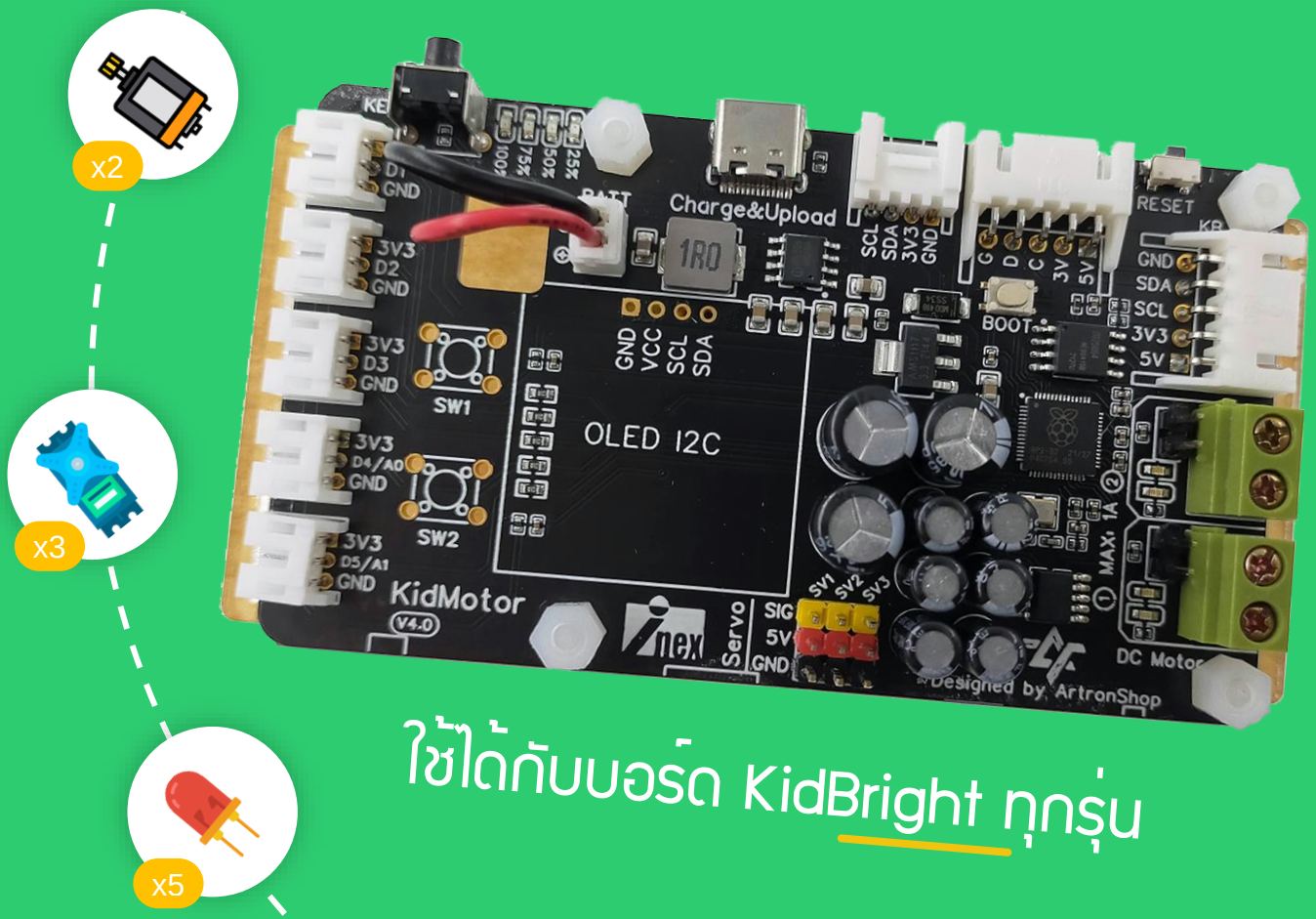
ARTRONSHOP
www.artronshop.co.th

อัพเดทล่าสุด 6 เมษายน 2565

คู่มือการใช้งาน KidMotor V4

เป็นบอร์ดเสริมขับมอเตอร์สำหรับ KidBright



ใช้ได้กับบอร์ด KidBright ทุกรุ่น

เขียนโปรแกรมด้วย

จัดทำโดย บริษัท อาร์ทรอน ชอป จำกัด
ร่วมกับ บริษัท อินโนเวตีฟ เอ็กเพอริเมนต์ จำกัด (INEX)

แจกฟรี ห้ามจำหน่าย

สารบัญ

บทที่ 1 ข้อมูลเกี่ยวกับบอร์ด KidMotor V4 4
1.1 ส่วนประกอบบอร์ด KidMotor V4 5
1.2 วงจรของบอร์ด KidMotor V4 6
1.3 การเปิด-ปิดบอร์ด KidBright32 11
1.4 สถานะแบตเตอรี่ 11
1.5 การชาร์จแบตเตอรี่ 12
1.6 อุปกรณ์อินพุต/เอาต์พุตที่รองรับ 12
1.7 เซอร์โวมอเตอร์ที่รองรับ 12
1.8 การอัพเดทเฟิร์มแวร์ 12
บทที่ 2 การต่อวงจร KidMotor V4 13
2.1 การต่อบอร์ด KidMotor V4 เข้ากับบอร์ดไมโครคอนโทรลเลอร์ 13
2.1.1 การต่อบอร์ด KidMotor V4 เข้ากับบอร์ด KidBright32 13
2.1.2 การต่อบอร์ด KidMotor V4 เข้ากับบอร์ด OpenKB 14
2.1.3 การต่อบอร์ด KidMotor V4 เข้ากับบอร์ด IPST-WiFi 14
2.1.4 การต่อบอร์ด KidMotor V4 เข้ากับบอร์ด ESP32 15
2.1.5 การต่อบอร์ด KidMotor V4 เข้ากับบอร์ด Arduino Uno R3 / Arduino Nano 15
2.2 การต่อแบตเตอรี่ 16
2.3 การต่ออุปกรณ์อินพุต/เอาต์พุต 16
2.4 การต่อมอเตอร์ 17
2.5 การต่อเซอร์โวมอเตอร์ 19
2.6 การต่อเซ็นเซอร์อัลตร้าโซนิค 19
บทที่ 3 การอัพเดท-ปรับแต่งเฟิร์มแวร์ 20
3.1 การเข้าโหมดอัพเดทเฟิร์มแวร์ 20
3.2 การอัพเดทเฟิร์มแวร์จากไฟล์ .uf2 20

3.3 การอัพเดทเฟิร์มแวร์จากโค้ดโปรแกรม 21
บทที่ 4 การเขียนโปรแกรมสั่งงานอุปกรณ์ผ่าน KidMotor 24
4.1 เขียนโปรแกรมสั่งงานด้วย KidBrightIDE 24
4.1.1 ติดตั้งปลั๊กอิน KidMotor V4 24
4.1.2 การเขียนโปรแกรมสั่งงานมอเตอร์ 26
4.1.3 การเขียนโปรแกรมควบคุมเซอร์โวมอเตอร์ 27
4.1.4 การเขียนโปรแกรมควบคุมอินพุต/เอาต์พุต 27
4.1.5 การเขียนโปรแกรมอ่านค่าจากเซ็นเซอร์อัลตร้าโซนิค 28
4.1.6 โครงงานรถวิ่งตามโปรแกรม 28
4.1.6 โครงงานรถวิ่งหนีแสง 30
4.1.7 โครงงานรถบังคับ 31
4.2.9 โครงงานหุ่นยนต์ดูดฝุ่น 35
4.2 เขียนโปรแกรมสั่งงานด้วย KBIDE 36
4.2.1 ติดตั้งปลั๊กอิน KidMotor V4 36
4.2.2 การเขียนโปรแกรมสั่งงานมอเตอร์ 38
4.2.3 การเขียนโปรแกรมควบคุมเซอร์โวมอเตอร์ 39
4.2.4 การเขียนโปรแกรมควบคุมอินพุต/เอาต์พุต 39
4.2.5 การเขียนโปรแกรมอ่านค่าจากเซ็นเซอร์อัลตร้าโซนิค 40
4.2.6 โครงงานรถวิ่งตามโปรแกรม 41
4.2.7 โครงงานรถวิ่งหนีแสง 42
4.2.8 โครงงานรถบังคับ 43
4.2.9 โครงงานหุ่นยนต์ดูดฝุ่น 47
4.3 เขียนโปรแกรมสั่งงานด้วย microBlock IDE 48
4.4.1 ติดตั้งปลั๊กอิน KidMotor V4 48
4.4.2 การเขียนโปรแกรมสั่งงานมอเตอร์ 50
4.4.3 การเขียนโปรแกรมควบคุมเซอร์โวมอเตอร์ 51

4.4.4 การเขียนโปรแกรมควบคุมอินพุต/เอาต์พุต 51
4.4.5 การเขียนโปรแกรมอ่านค่าจากเซ็นเซอร์อัลตร้าโซนิค 51
4.4.6 โครงงานรถวิ่งตามโปรแกรม 52
4.4.7 โครงงานรถวิ่งหนีแสง 53
4.4.8 โครงงานรถบังคับ 54
4.4.9 โครงงานหุ่นยนต์ดูดฝุ่น 58
ภาคผนวก 60
(ก) การแก้ปัญหารถวิ่งผิดทิศทาง 60

บทที่ 1 ข้อมูลเกี่ยวกับบอร์ด KidMotor V4

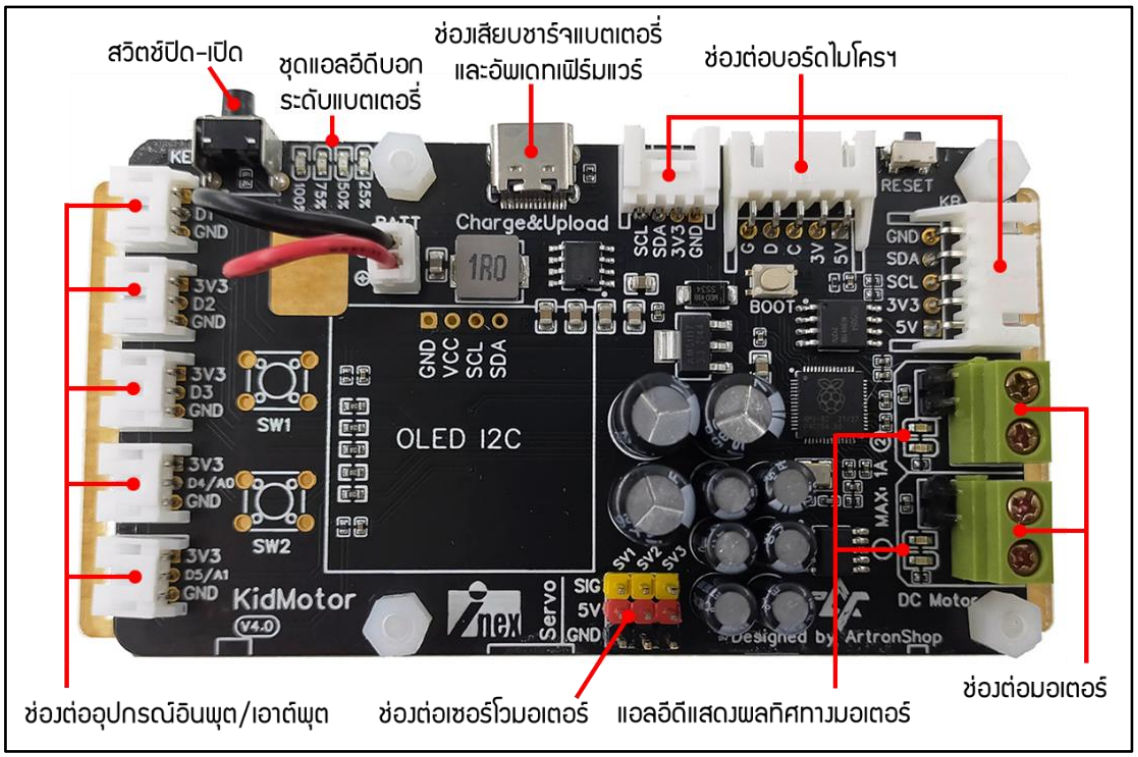
KidMotor V4 บอร์ดเสริมทำหุ่นยนต์สำหรับบอร์ด KidBright เหมาะสำหรับนำไปทำหุ่นยนต์เดินตามเส้น รถบังคับ Self-balancing Car หุ่นยนต์กู้ภัย รองรับการเชื่อมต่อมอเตอร์ DC จำนวน 2 ตัว เซอร์โวมอเตอร์จำนวน 3 ตัว และต่อมีช่องเซ็นเซอร์ สวิตช์ จำนวน 5 ชุด มีแบตเตอรี่ขนาด 2400mAh ในตัว มาพร้อมวงจรชาร์จแบตเตอรี่ในตัว มีหลอดแอลอีดีแสดงผลปริมาณแบตเตอรี่คงเหลือจำนวน 4 ดวง มีหลอดแอลอีดีแสดงสถานะทิศทางการหมุนของมอเตอร์ ใช้ไมโครคอนโทรลเลอร์ RP2040 รองรับการเขียนโปรแกรมสั่งงานด้วย KidBright IDE, KB-IDE, microBlock IDE และ Arduino IDE

รองรับการใช้งานทั้งเป็นบอร์ดเสริมให้กับบอร์ด KidBright และสามารถใช้เป็นบอร์ดหลัก (Standalone) โดยรองรับการเขียนโปรแกรมสั่งงานด้วยโปรแกรม microBlock IDE และ Arduino IDE สำหรับเอกสารฉบับนี้จะแนะนำการใช้งาน KidMotor V4 ในรูปแบบของบอร์ดเสริม ใช้ขับมอเตอร์จะจัดการพลังงาน

ตัวบอร์ดรองรับการอัพเดทเฟิร์มแวร์จากผู้ผลิต โดยผู้ใช้เพียงเชื่อมต่อบอร์ด KidMotor V4 เข้ากับเครื่องคอมพิวเตอร์ผ่านสาย USB-C แล้วคัดลอกไฟล์เฟิร์มแวร์เวอร์ชั่นใหม่ไปวาง เป็นอันเสร็จสิ้นการอัพเดทเฟิร์มแวร์

KidMotor มาพร้อมฐานอะคริลิคหนา 3 มิลลิเมตร ช่วยให้สามารถยึดติดเข้ากับโครงรถได้ง่าย พร้อมแถมเสาไนล่อนสามารถยึดบอร์ด KidBright เข้ากับ KidMotor V4 ได้โดยตรง

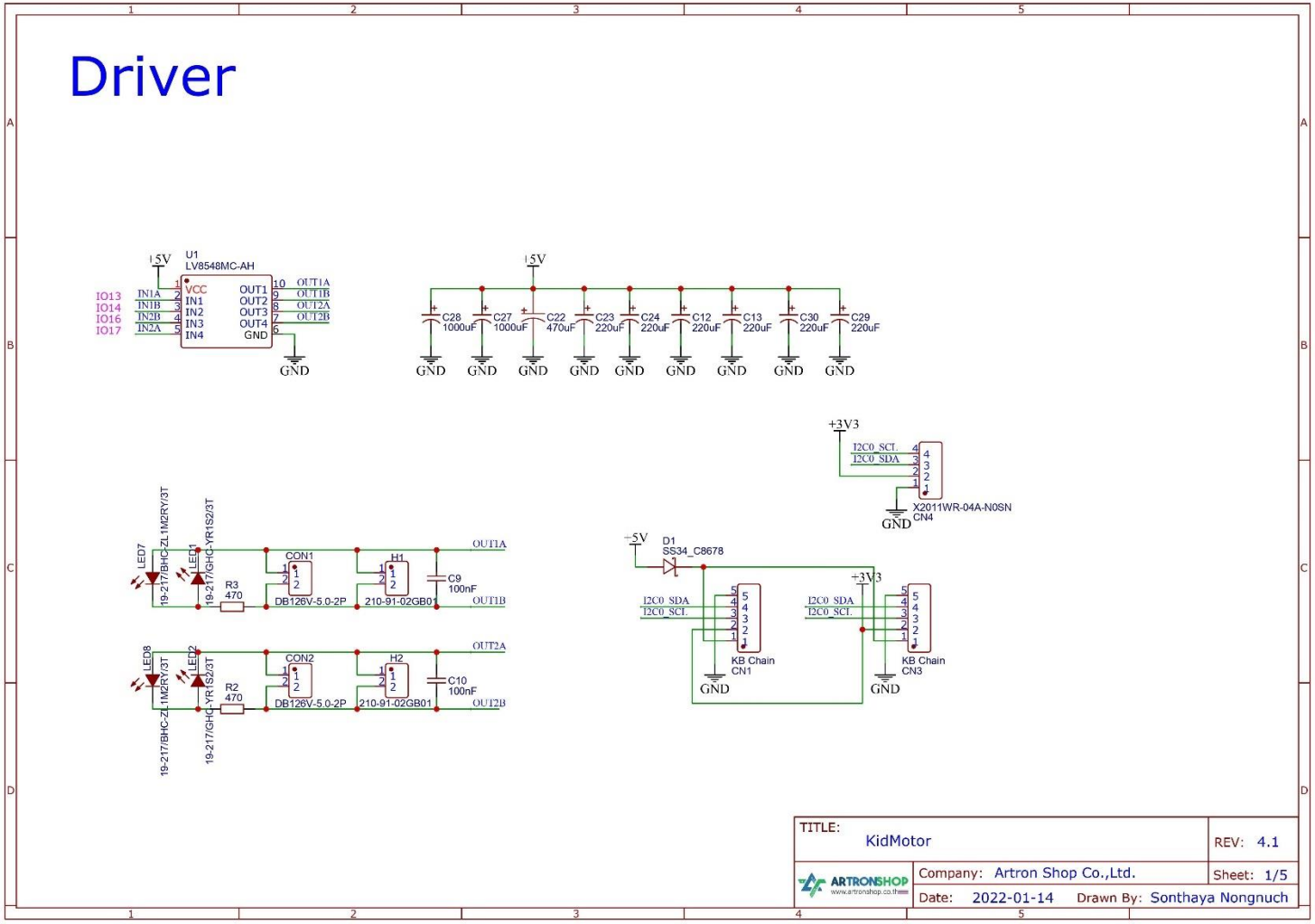
1.1 ส่วนประกอบบอร์ด KidMotor V4



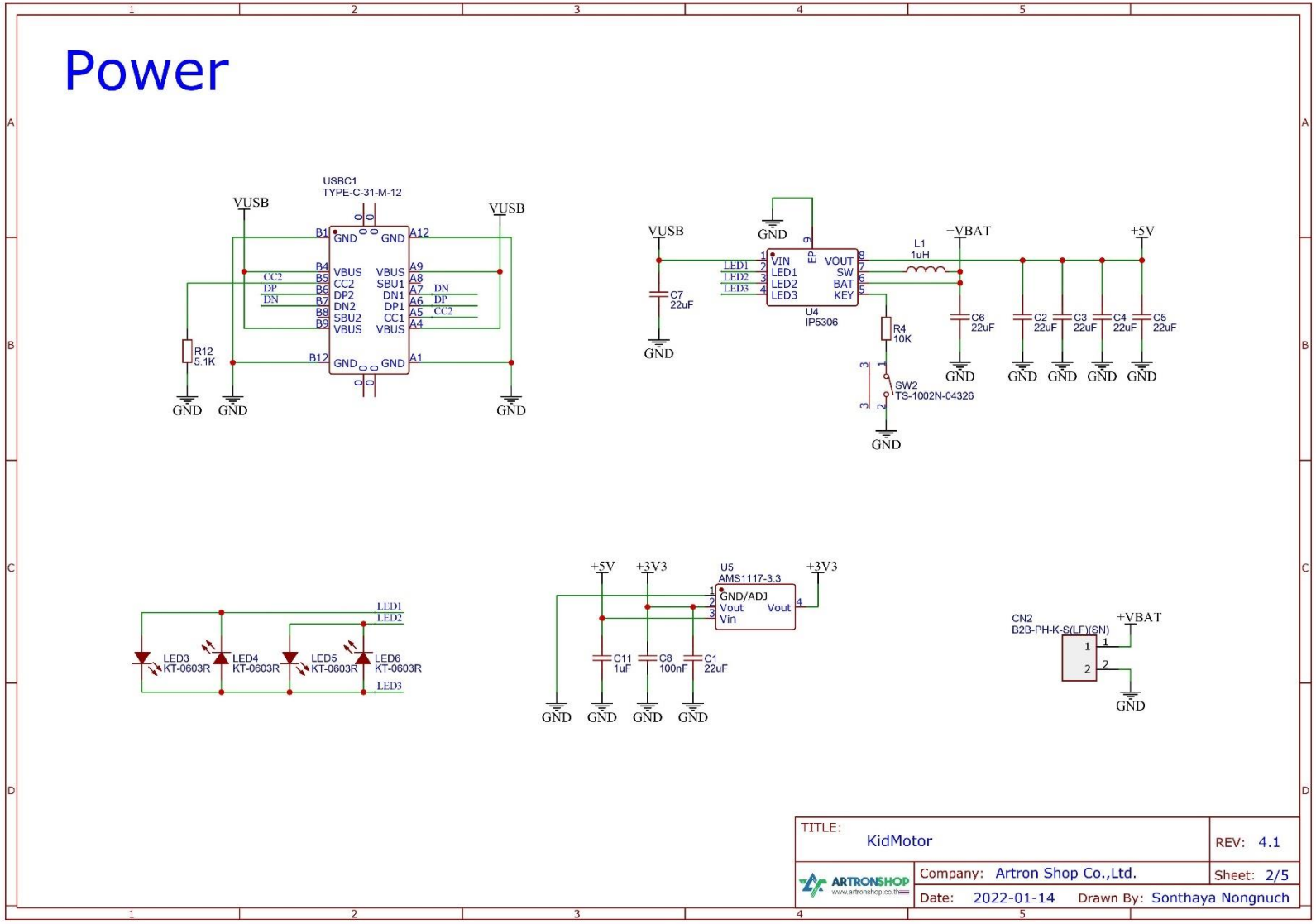
รูปที่ 1.1 ส่วนประกอบของบอร์ด KidMotor V4

- สวิตช์ปิด-เปิด ควบคุมการจ่ายพลังงานมอเตอร์และบอร์ด KidBright32
- ชุดแอลอีดีบอกระดับแบตเตอรี่ ใช้บอกระดับแบตเตอรี่ที่เหลืออยู่
- ช่องเสียบชาร์จแบตเตอรี่และอัพเดทเฟิร์มแวร์ ใช้สาย USB-C สำหรับชาร์จโทรศัพท์มือถือ ชาร์จแบตเตอรี่บนบอร์ด และอัพเดทเฟิร์มแวร์ผ่านเครื่องคอมพิวเตอร์
- ช่องต่อบอร์ดไมโครฯ ใช้ต่อบอร์ด KidMotor V4 เข้ากับบอร์ด KidBright32, micro:bit และบอร์ดอื่น ๆ
- ช่องต่อมอเตอร์ ใช้ต่อมอเตอร์สำหรับทำหุ่นยนต์ (รถ) มี 2 ช่อง สำหรับมอเตอร์ซ้ายและขวา
- แอลอีดีแสดงผลทิศทางมอเตอร์ สีน้ำเงิน และสีแดง แสดงทิศทางการหมุนของมอเตอร์
- ช่องต่อเซอร์โวมอเตอร์ รองรับการต่อเซอร์โวมอเตอร์ได้พร้อมกัน 3 ตัว
- ช่องต่ออุปกรณ์อินพุต/เอาต์พุต ใช้ต่ออุปกรณ์ดิจิทัล และอะแนล็อก
- แบตเตอรี่ 2000mAh ใช้จ่ายพลังงานให้มอเตอร์ บอร์ดไมโครฯ และอุปกรณ์เสริมต่าง ๆ (อยู่ใต้บอร์ด)

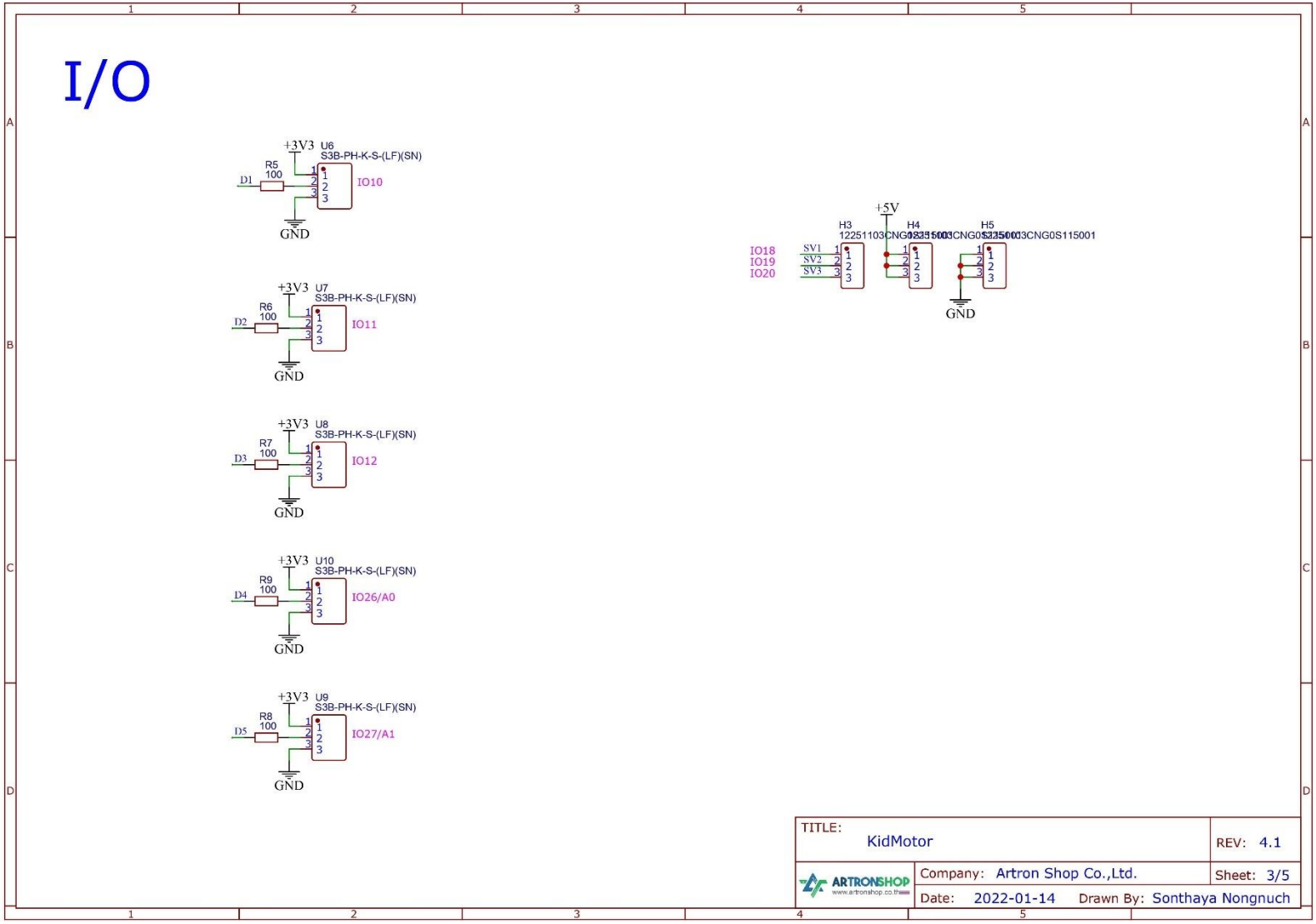
1.2 วงจรของบอร์ด KidMotor V4



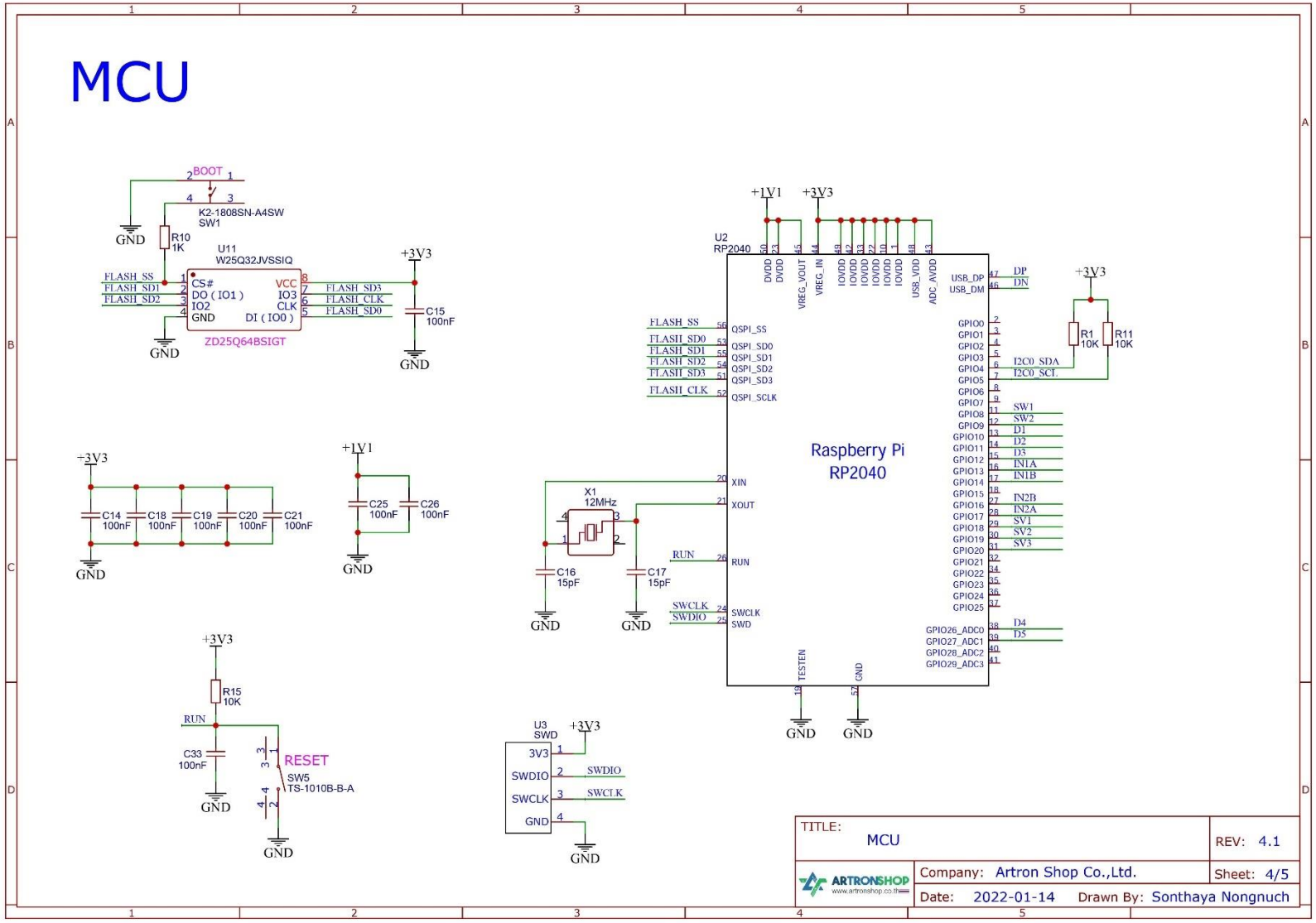
รูปที่ 1.2 วงจรของบอร์ด KidMotor V4 ส่วนวงจรขับมอเตอร์ DC



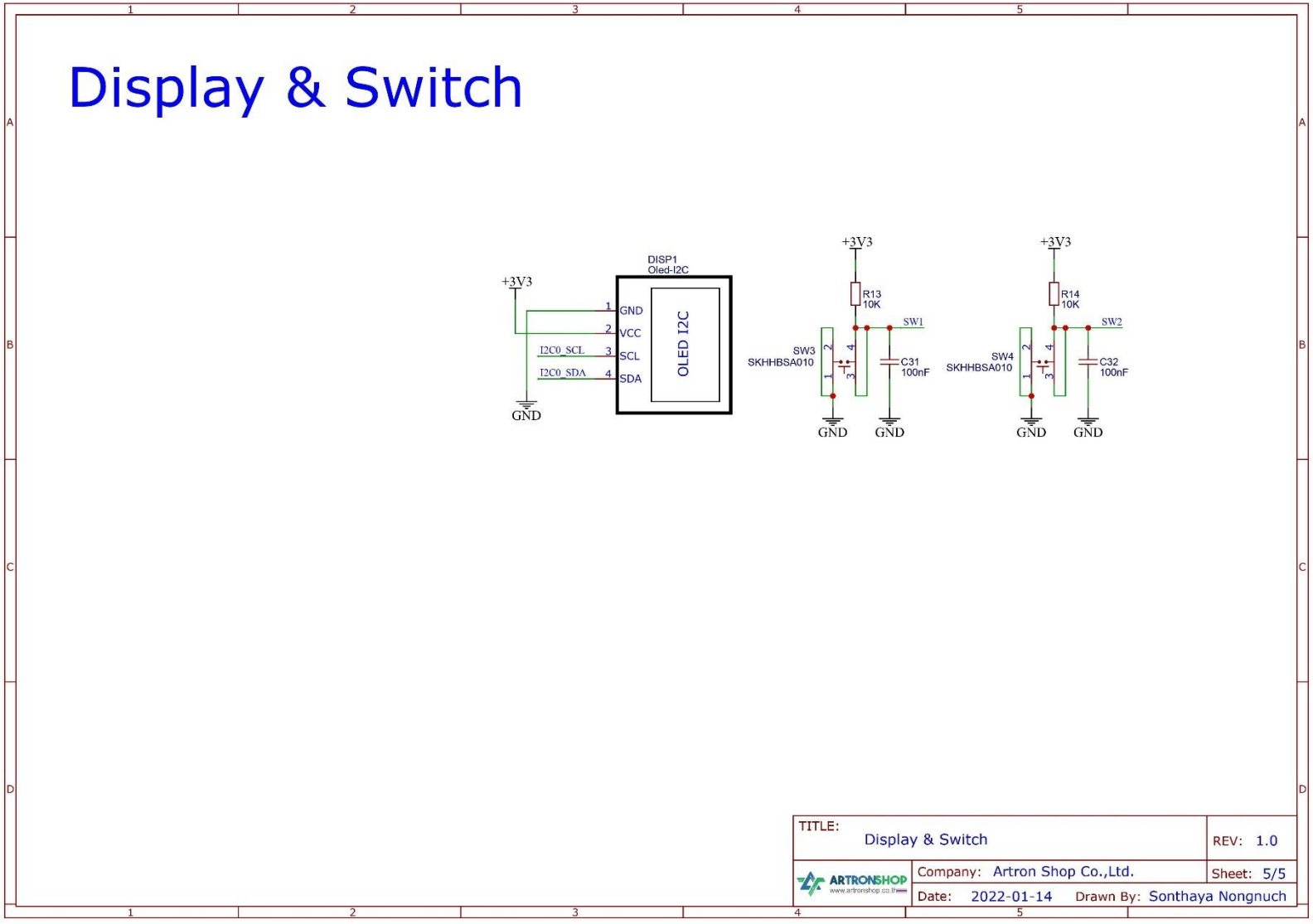
รูปที่ 1.3 วงจรของบอร์ด KidMotor V4 ส่วนวงจรจัดการพลังงาน



รูปที่ 1.4 วงจรของบอร์ด KidMotor V4 ส่วนวงจร I/O และเซอร์โวมอเตอร์



รูปที่ 1.5 วงจรของบอร์ด KidMotor V4 ส่วนไมโครคอนโทรลเลอร์ควบคุมบอร์ด



รูปที่ 1.6 วงจรของบอร์ด KidMotor V4 ส่วนจอแสดงผลและสวิตช์โปรแกรมได้อิสระ (ไม่ได้ใช้งานในโหมดบอร์ดเสริม)

1.3 การเปิด-ปิดบอร์ด KidBright32

การเปิดบอร์ด KidMotor V4 ทำได้โดยกดสวิตช์ KEY เป็นระยะเวลาสั้น ๆ (น้อยกว่า 2 วินาที) จำนวน 1 ครั้ง บอร์ด KidMotor V4 จะถูกเปิดขึ้นมา ไฟแสดงสถานะแบตเตอรี่จะติดขึ้นมา พร้อมทั้งจ่ายไฟเลี้ยงไปยังบอร์ด KidBright32 หรือบอร์ดไมโครคอนโทรลเลอร์หลัก

การปิดบอร์ด KidMotor V4 ทำได้โดยกดสวิตช์ KEY ค้างไว้ประมาณ 5 วินาที หรือกด 2 ครั้งเป็นระยะเวลาสั้น ๆ (น้อยกว่า 2 วินาที) บอร์ด KidMotor V4 จะถูกปิด ไฟแสดงสถานะแบตเตอรี่จะดับทั้งหมด และบอร์ด KidBright32 / บอร์ดไมโครคอนโทรลเลอร์หลัก ถูกตัดไฟเลี้ยง

บอร์ด KidMotor V4 จะปิดตัวเองอัตโนมัติหากมีการใช้พลังงานไฟฟ้าน้อย (ใช้กระแสไฟฟ้าน้อยกว่า 45 mA เป็นเวลา 30 วินาที) เพื่อประหยัดพลังงานแบตเตอรี่

1.4 สถานะแบตเตอรี่

ขณะชาร์จแบตเตอรี่ บอร์ด KidMotor V4 จะทำงานตลอดเวลาที่ชาร์จแบตเตอรี่ ไฟแสดงสถานะแบตเตอรี่จะติดและกระพริบดังนี้

ปริมาณแบตเตอรี่ (%)	แอลอีดี 25%	แอลอีดี 50%	แอลอีดี 75%	แอลอีดี 100%
เต็ม	ติด	ติด	ติด	ติด
มากกว่าหรือเท่ากับ 75%	ติด	ติด	ติด	กระพริบ
มากกว่าหรือเท่ากับ 50% แต่น้อยกว่า 75%	ติด	ติด	กระพริบ	ดับ
มากกว่าหรือเท่ากับ 25% แต่น้อยกว่า 50%	ติด	กระพริบ	ดับ	ดับ
น้อยกว่า 25%	กระพริบ	ดับ	ดับ	ดับ

ขณะใช้งาน ไฟแสดงสถานะแบตเตอรี่บนบอร์ด KidMotor V4 ติดและกระพริบ เพื่อแสดงปริมาณแบตเตอรี่ที่เหลืออยู่ดังนี้

ปริมาณแบตเตอรี่ (%)	แอลอีดี 25%	แอลอีดี 50%	แอลอีดี 75%	แอลอีดี 100%
มากกว่าหรือเท่ากับ 75%	ติด	ติด	ติด	ติด
มากกว่าหรือเท่ากับ 50% แต่น้อยกว่า 75%	ติด	ติด	ติด	ดับ
มากกว่าหรือเท่ากับ 25% แต่น้อยกว่า 50%	ติด	ติด	ดับ	ดับ
มากกว่าหรือเท่ากับ 3% แต่น้อยกว่า 25%	ติด	ดับ	ดับ	ดับ
มากกว่า 0% แต่น้อยกว่า 3%	กระพริบ	ดับ	ดับ	ดับ
0%	ดับ	ดับ	ดับ	ดับ

1.5 การชาร์จแบตเตอรี่

บอร์ด KidMotor V4 ชาร์จแบตเตอรี่ผ่านช่อง USB-C กระแสชาร์จแบตเตอรี่ 1.5A แนะนำให้ใช้อะแดปเตอร์หรือที่ชาร์จโทรศัพท์มือถือที่จ่ายแรงดันไฟฟ้ากระแสตรง 5V กระแสไม่น้อยกว่า 2A ในการชาร์จ

1.6 อุปกรณ์อินพุต/เอาต์พุตที่รองรับ

อุปกรณ์อินพุต/เอาต์พุต ที่ต่อช่อง D1 ถึง D5 ได้ ต้องเป็นอุปกรณ์ที่ใช้แรงดันไฟฟ้า 4.3V ได้เท่านั้น และใช้กระแสไฟฟ้าไม่เกิน 500mA (กรณีใช้กระแสไฟฟ้ามากกว่าที่ระบุ จำเป็นต้องหาแหล่งจ่ายไฟอื่นมาจ่ายให้อุปกรณ์เอง)

กรณีเป็นอุปกรณ์ประเภทอินพุต เช่น เซ็นเซอร์ตรวจจับเส้น เซ็นเซอร์แสง เซ็นเซอร์อุณหภูมิ ต้องให้แรงดันไฟฟ้าออกมาไม่เกิน 4.3V เท่านั้น

กรณีเป็นอุปกรณ์เอาต์พุต อุปกรณ์ต้องใช้กระแสไฟฟ้าไม่เกิน 20 mA กรณีต้องการต่อเข้ากับอุปกรณ์กำลังสูง เช่น มอเตอร์ โซลินอยด์ แอลอีดีกำลังสูง จำเป็นต้องต่อผ่านวงจรขับเท่านั้น

1.7 เซอร์โวมอเตอร์ที่รองรับ

รองรับเซอร์โวมอเตอร์ที่ใช้แรงดันไฟฟ้า 5V - 6V ทุกรุ่น เช่น SG-90, MG996 เป็นต้น โดยรองรับทั้งแบบหมุนได้ 180 องศา และแบบหมุน 360 องศา

1.8 ฟังก์ชั่นพิเศษ

เฟิร์มแวร์ของ KidMotor V4 รองรับการต่อกับอุปกรณ์ต่าง ๆ ซึ่งเป็นฟังก์ชั่นพิเศษ ปัจจุบัน (เฟิร์มแวร์เวอร์ชั่น V1.1) รองรับฟังก์ชั่นพิเศษ ดังนี้

1.8.1 เซ็นเซอร์อัลตร้าโซนิค

รองรับการอ่านค่าระยะห่างระหว่างเซ็นเซอร์อัลตร้าโซนิคกับวัตถุ โดยรองรับเซ็นเซอร์ที่ใช้แรงดันไฟฟ้า 3.3V ได้เท่านั้น เช่น รุ่น HC-SR04p เป็นต้น

1.9 การอัพเดทเฟิร์มแวร์

รองรับการอัพเดทเฟิร์มแวร์ผ่านช่อง USB-C โดยใช้สาย USB-C เสียบเข้ากับเครื่องคอมพิวเตอร์ จากนั้นอัพโหลดไฟล์เฟิร์มแวร์โดยคัดลอกไฟล์ .uf2 ไปวางในไดร์ของ KidMotor V4 หลังคัดลอกเสร็จบอร์ด KidMotor V4 จะรีเซ็ตตัวเองและไปใช้เฟิร์มแวร์เวอร์ชั่นใหม่ทันที

บทที่ 2 การต่อวงจร KidMotor V4

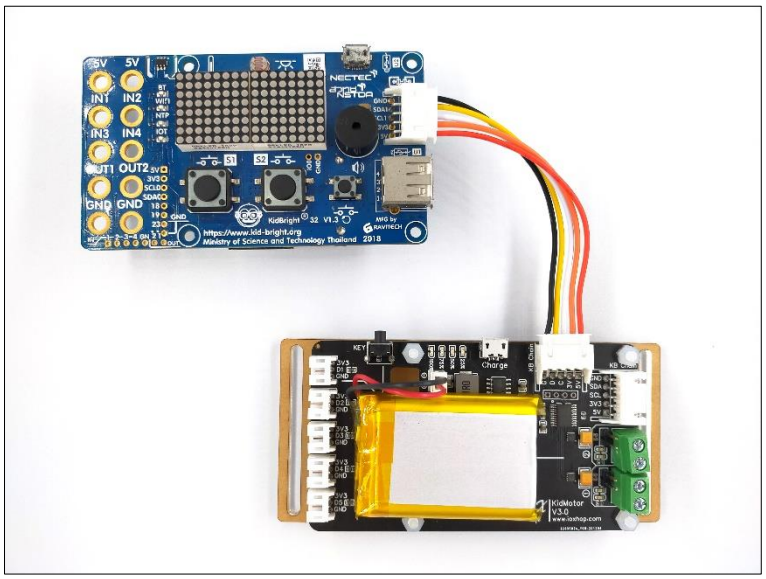
บอร์ด KidMotor V4 มีช่องต่อแบตเตอรี่ ช่องต่อมอเตอร์ ช่องต่ออุปกรณ์อินพุต/เอาต์พุต และช่องต่อ KB Chain เพื่อความปลอดภัยในการใช้งาน จึงจำเป็นต้องทำความเข้าใจการต่อวงจรใช้งานช่องต่าง ๆ บนบอร์ด KidMotor V4 ในหัวข้อนี้

2.1 การต่อบอร์ด KidMotor V4 เข้ากับบอร์ดไมโครคอนโทรลเลอร์

บอร์ด KidMotor V4 รองรับการเชื่อมต่อกับบอร์ดไมโครคอนโทรลเลอร์ที่หลากหลาย แต่ละบอร์ดมีวิธีการต่อวงจรที่แตกต่างกันดังนี้

2.1.1 การต่อบอร์ด KidMotor V4 เข้ากับบอร์ด KidBright32

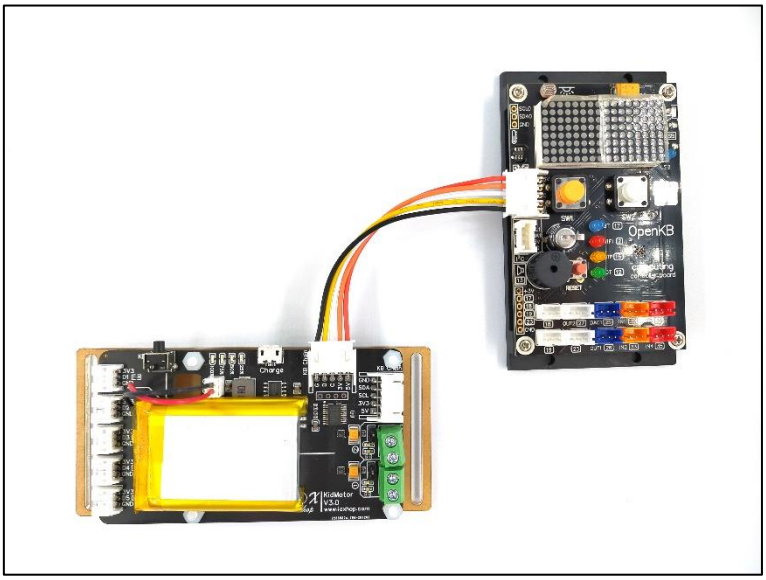
บอร์ด KidMotor V4 รองรับบอร์ด KidBright32 ทุกเวอร์ชั่น โดยการต่อวงจรให้ใช้สาย KB Chain ปลายคู่ JST 5 พิน ต่อด้านหนึ่งเข้ากับบอร์ด KidBright32 ที่ช่อง KB Chain และปลายอีกด้านต่อเข้าบอร์ด KidMotor V4 ที่ช่องใดก็ได้ ดังรูปที่ 2.1



รูปที่ 2.1 การต่อบอร์ด KidMotor V4 เข้ากับบอร์ด KidBright32

2.1.2 การต่อบอร์ด KidMotor V4 เข้ากับบอร์ด OpenKB

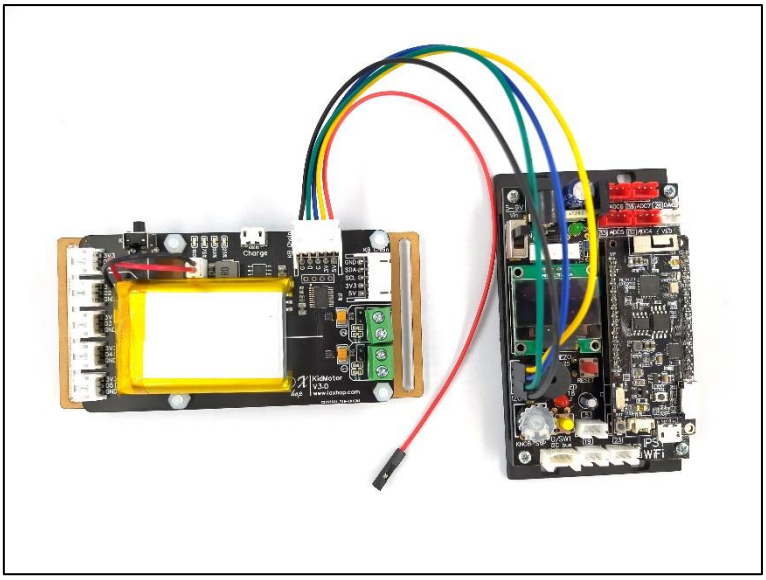
ใช้สาย KB Chain ปลายคู่ JST 5 พิน ต่อด้านหนึ่งเข้ากับบอร์ด OpenKB ที่ช่อง KB Chain และปลายอีกด้านต่อเข้าบอร์ด KidMotor V4 ที่ช่องใดก็ได้ ดังรูปที่ 2.2



รูปที่ 2.2 การต่อบอร์ด KidMotor V4 เข้ากับบอร์ด OpenKB

2.1.3 การต่อบอร์ด KidMotor V4 เข้ากับบอร์ด IPST-WiFi

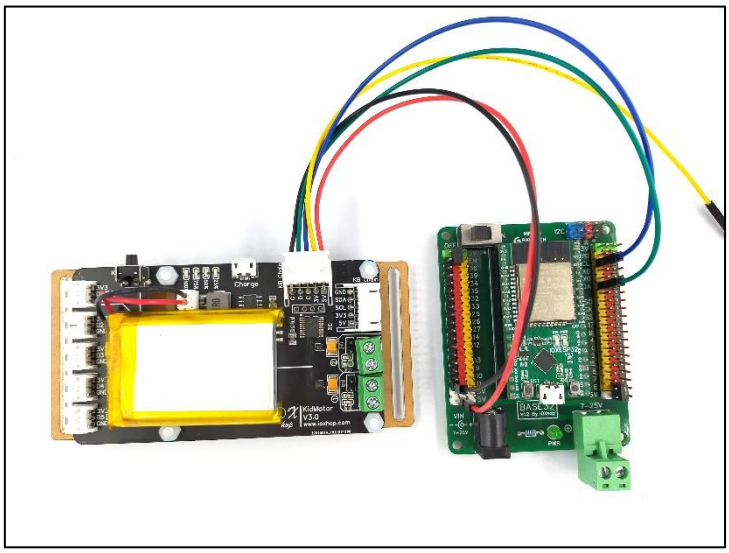
ใช้สาย KB Chain ปลาย IDC ต่อด้าน JST 5 พิน เข้ากับบอร์ด KidMotor V4 ที่ช่องใดก็ได้ ปลายด้าน IDC ต่อสายสีเขียวเข้ากับช่อง I2C ขา SDA, สายสีน้ำเงินต่อ SCL, สายสีแดงต่อ 5V หรือ VIN หรือ VUSB, สายสีดำต่อ GND ตัวอย่างการต่อบอร์ด KidMotor V4 เข้ากับบอร์ด IPST-WiFi แสดงดังรูปที่ 2.3



รูปที่ 2.3 การต่อบอร์ด KidMotor V4 เข้ากับบอร์ด IPST-WiFi

2.1.4 การต่อบอร์ด KidMotor V4 เข้ากับบอร์ด ESP32

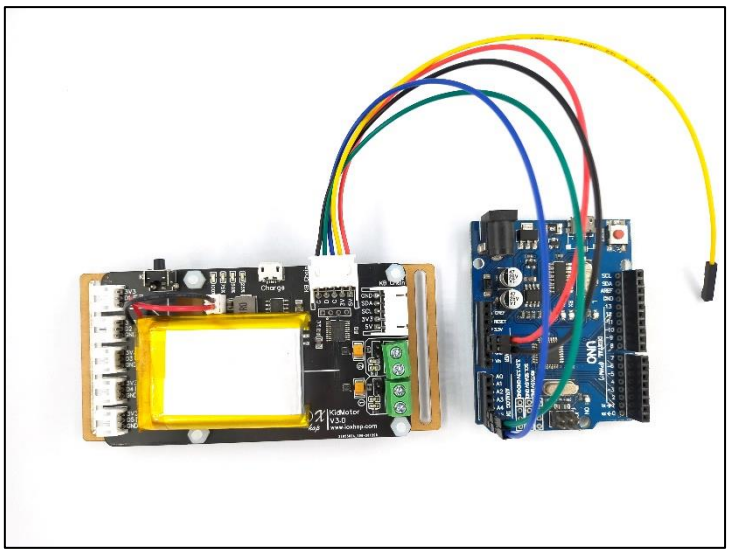
ใช้สาย KB Chain ปลาย IDC ต่อด้าน JST 5 พิน เข้ากับบอร์ด KidMotor V4 ที่ช่องใดก็ได้ ปลายด้าน IDC ต่อสายสีเขียวเข้ากับช่อง I2C ขา SDA, สายสีน้ำเงินต่อ SCL, สายสีแดงต่อ 5V หรือ VIN หรือ VUSB, สายสีดำต่อ GND ตัวอย่างการต่อบอร์ด KidMotor V4 เข้ากับบอร์ด IOXESP32 แสดงดังรูปที่ 2.4



รูปที่ 2.4 การต่อบอร์ด KidMotor V4 เข้ากับบอร์ด ESP32

2.1.5 การต่อบอร์ด KidMotor V4 เข้ากับบอร์ด Arduino Uno R3 / Arduino Nano

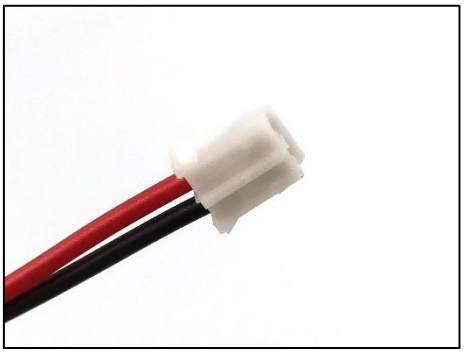
ใช้สาย KB Chain ปลาย IDC ต่อด้าน JST 5 พิน เข้ากับบอร์ด KidMotor V4 ที่ช่องใดก็ได้ ปลายด้าน IDC ต่อสายสีเขียว (SDA) เข้ากับช่อง A4, สายสีน้ำเงิน (SCL) ต่อ A5, สายสีแดงต่อ 5V, สายสีดำต่อ GND ตัวอย่างการต่อบอร์ด KidMotor V4 เข้ากับบอร์ด Arduino Uno R3 แสดงดังรูปที่ 2.5



รูปที่ 2.5 การต่อบอร์ด KidMotor V4 เข้ากับบอร์ด Arduino Uno R3

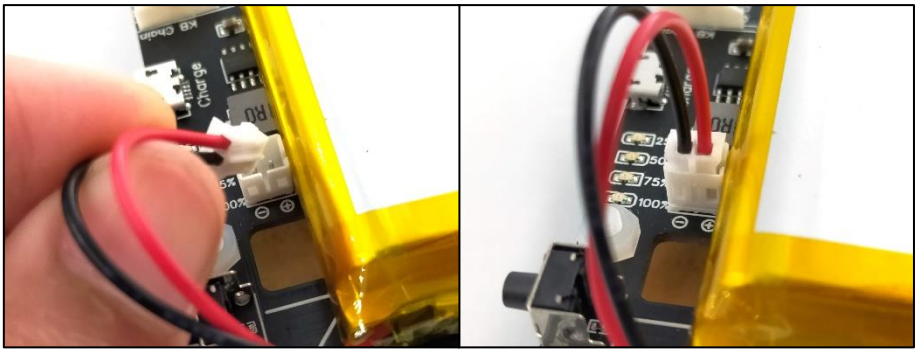
2.2 การต่อแบตเตอรี่

แบตเตอรี่ที่สามารถนำมาต่อเข้ากับบอร์ด KidMotor V4 ได้ จะต้องเป็นแบตลิเรียมไอออน หรือ แบตลีเรียม-โพลิเมอร์ แบบ 1 เซลล์ แรงดันไฟฟ้า 4.7V เท่านั้น โดยต่อผ่านหัว JST ขนาด 2 มิลลิเมตร ที่มีสายบวก (สายสีแดง) อยู่ด้านซ้าย และสายลบ (สายสีดำ) อยู่ด้านขวา ดังรูปที่ 2.6



รูปที่ 2.6 สายแบตเตอรี่ที่ใช้กับบอร์ด KidMotor V4 ได้

การต่อแบตเตอรี่เข้ากับบอร์ด KidMotor V4 ให้เสียบหัว JST ลงในช่องต่อแบตเตอรี่ของบอร์ด KidMotor V4 ดังรูปที่ 2.7



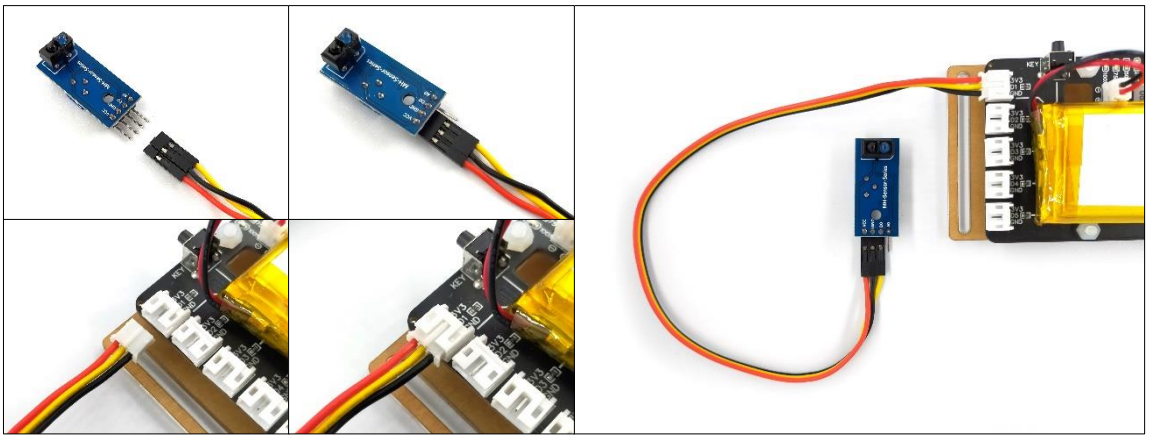
รูปที่ 2.7 การต่อสายแบตเตอรี่เข้ากับบอร์ด KidMotor V4

2.3 การต่ออุปกรณ์อินพุต/เอาต์พุต

ช่อง D1 ถึง D5 ของบอร์ด KidMotor V4 สามารถต่ออุปกรณ์อินพุต/เอาต์พุต เพื่อเขียนโปรแกรมอ่านค่าแบบดิจิทัล อ่านค่าแอนะล็อก หรือเขียนค่าดิจิทัลได้ โดยหากอุปกรณ์ที่นำมาต่อเป็นแบบ JST สามารถใช้สาย JST3AA-8 (ดังรูปที่ 2.7) ต่อได้เลย หรือหากอุปกรณ์ที่นำมาต่อเป็นขาแบบก้างปลา ให้ใช้สายแปลงหัว JST เป็น IDC JST3AA-8 (ดังรูปที่ 2.8) ตัวอย่างการต่อบอร์ด KidMotor V4 เข้ากับเซ็นเซอร์ตรวจจับเส้น แสดงดังรูปที่ 2.9

รูปที่ 2.7 สาย JST3AA-8

รูปที่ 2.8 สายแปลงหัว JST เป็น IDC JST3AA-8



รูปที่ 2.9 การต่อบอร์ด KidMotor V4 เข้ากับเซ็นเซอร์ตรวจจับเส้น

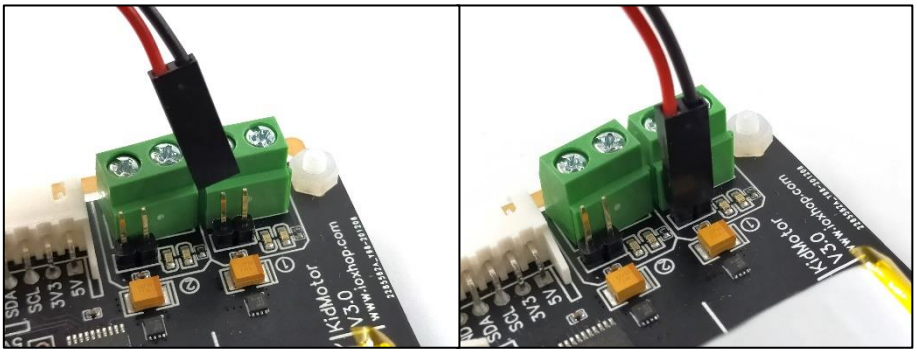
2.4 การต่อมอเตอร์

บอร์ด KidMotor V4 มีช่องต่อมอเตอร์จำนวน 2 ช่อง รองรับมอเตอร์ 3V ถึง 5V ใช้กระแสไฟฟ้าไม่เกิน 1A รองรับมอเตอร์เหลือง (ดังรูปที่ 2.10) และมอเตอร์ N20 (ดังรูปที่ 2.11) และมอเตอร์อื่น ๆ ที่ใช้แรงดันและกระแสไฟฟ้าตามที่กำหนดไว้

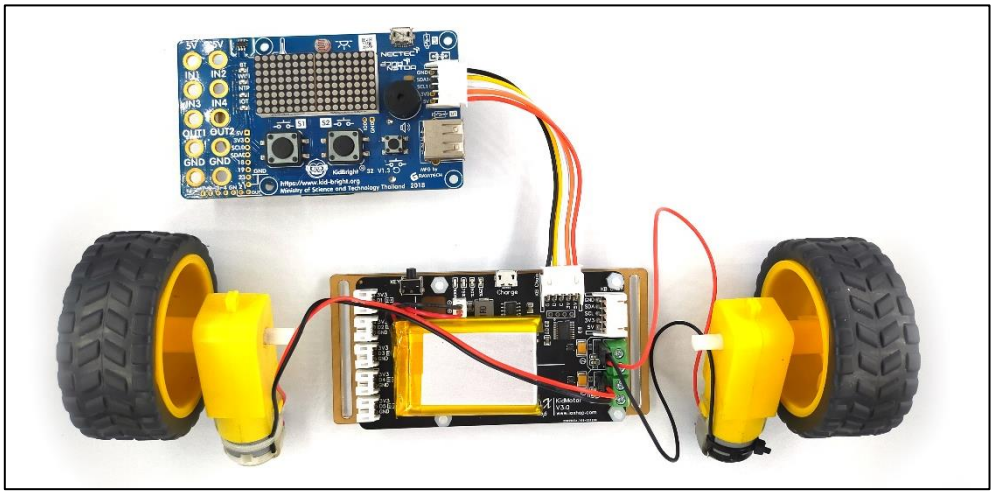
รูปที่ 2.10 มอเตอร์เหลือง

รูปที่ 2.11 มอเตอร์ N20

การต่อมอเตอร์สามารถทำได้ 2 วิธี คือ 1) ต่อผ่านช่อง IDC (แนะนำ) และ 2) ต่อผ่านเทอมินอล ตัวอย่างการต่อมอเตอร์ผ่านช่อง IDC แสดงดังรูปที่ 2.12 และตัวอย่างการต่อ KidMotor V4 เข้ากับบอร์ด KidBright32 และมอเตอร์แสดงดังรูปที่ 2.13



รูปที่ 2.12 การต่อมอเตอร์ผ่านช่อง IDC



รูปที่ 2.13 การต่อ KidMotor V4 เข้ากับบอร์ด KidBright32 และมอเตอร์

2.5 การต่อเซอร์โวมอเตอร์

บอร์ด KidMotor V4 รองรับเซอร์โวมอเตอร์ที่ใช้แรงดันไฟฟ้า 5V – 6V ทั้ง 180 องศา และ 360 องศา จำนวน 3 ตัว โดยต่อเข้าช่อง SV1 ถึง SV3 ตัวอย่างการต่อเซอร์โวมอเตอร์ SG-90 แสดงดังรูปที่ 2.14

2.6 การต่อเซ็นเซอร์อัลตร้าโซนิค

ช่องอินพุต/เอาต์พุต ของบอร์ด KidMotor V4 รองรับการต่อเซ็นเซอร์อัลตร้าโซนิคที่ขาใด ๆ ก็ได้ โดยเลือกขา D1 ถึง D5 ขาใดขาหนึ่งเป็นขา Trig และขาที่เหลือขาใดก็ได้เป็นขา Echo โดยใช้สาย JST3AA-8 ในการต่อ ตัวอย่างการต่อเซ็นเซอร์อัลตร้าโซนิครุ่น HC-SR04p โดยเลือก D1 เป็นขา Trig และ D2 เป็นขา Echo แสดงดังรูปที่ 2.15

บทที่ 3 การอัพเดท-ปรับแต่งเฟิร์มแวร์

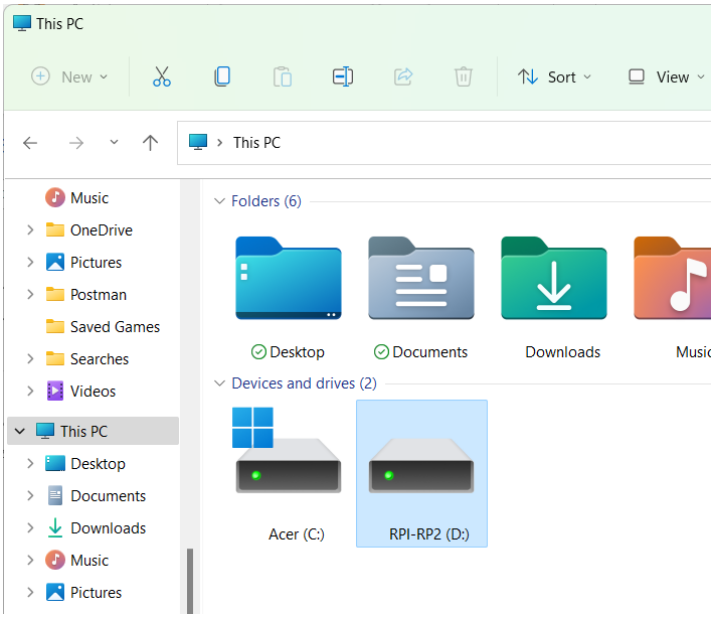
KidMotorV4-I2C-bridge เป็นโค้ดโปรแกรม-เฟิร์มแวร์ที่อยู่ภายในบอร์ด KidMotor V4 ทำหน้าที่รับ-ส่ง-ประมวลผลข้อมูลจากบอร์ดไมโครคอนโทรลเลอร์ที่ส่งเข้ามาที่ KidMotor V4 ผ่าน I^2C (KB Chain / Grove) โดยการอัพเดทเฟิร์มแวร์ดังกล่าวจะช่วยให้บอร์ด KidMotor V4 ได้รับฟีเจอร์ใหม่ และแก้ไขบัคที่อาจจะมีในเวอร์ชั่นเก่า ๆ ด้วย

3.1 การเข้าโหมดอัพเดทเฟิร์มแวร์

การเข้าโหมดอัพเดทเฟิร์มแวร์ทำได้ 2 วิธี

1. ปิดบอร์ด KidMotor V4 จากนั้นกดปุ่ม BOOT ค้างไว้ แล้วเสียบสาย USB-C ไฟสถานะแบตเตอรี่จะต้องติดสว่างขึ้นมา
2. เสียบสาย USB-C จากนั้นกดปุ่ม BOOT ค้างไว้ แล้วกดปุ่ม RESET แล้วจึงปล่อยปุ่ม BOOT

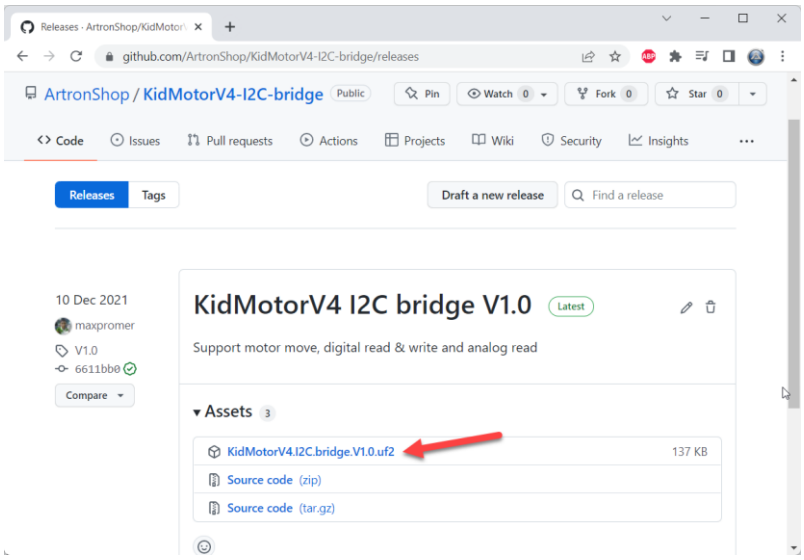
หากเข้าโหมดอัพเดทเฟิร์มแวร์สำเร็จ ใน My Computer จะปรากฏไดรฟ์ใหม่ขึ้นมา



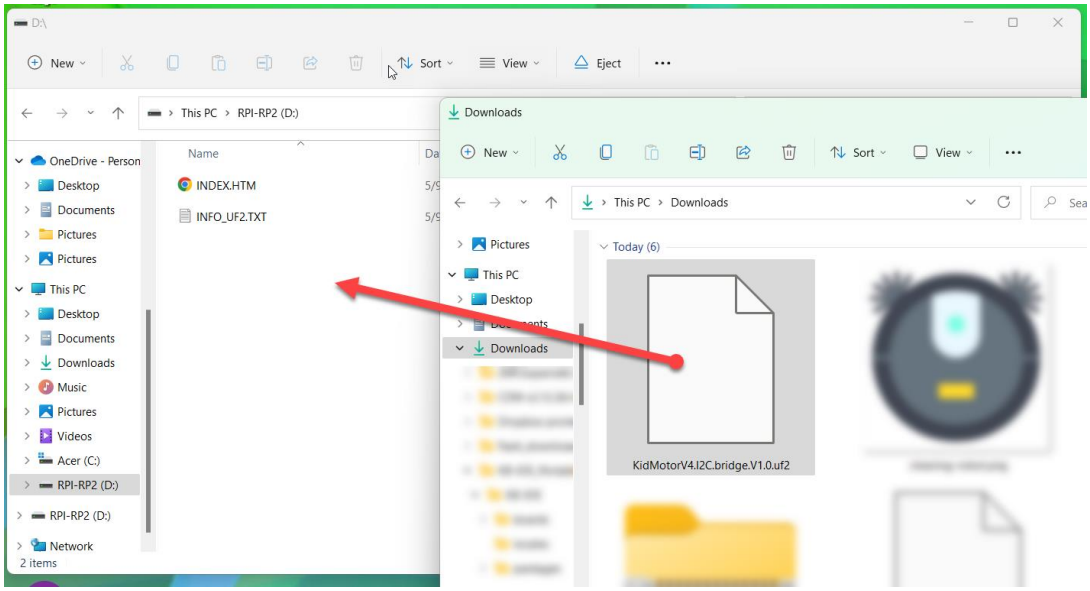
3.2 การอัพเดทเฟิร์มแวร์จากไฟล์ .uf2

ไฟล์ .uf2 เป็นไฟล์ที่ผู้ผลิตคอมไพล์โค้ดโปรแกรม C/C++ ออกมา และมีการทดสอบแล้วว่าใช้งานได้ปกติ สามารถนำไปอัพโหลดลงบอร์ด KidMotor V4 เพื่อใช้งานได้เลย การอัพเดทเฟิร์มแวร์ด้วยไฟล์ .uf2 มีขั้นตอนดังนี้

1) ดาวน์โหลดไฟล์ .uf2 ที่ Releases ∩ ArtronShop/KidMotorV4-I2C-bridge โดยแนะนำให้ดาวน์โหลดเวอร์ชั่นล่าสุด



2) นำไฟล์ .uf2 ที่ได้ คัดลอก แล้ววางลงไปในไดร์ที่เพิ่มเข้ามาใหม่ หรือลากวาง



3) รอคัดลอกไฟล์จนกว่าจะเสร็จ จากนั้นบอร์ด KidMotor V4 จะรีเซ็ตอัตโนมัติ แล้วเฟิร์มแวร์ตัวใหม่จะทำงานทันที

เป็นอันจบการอัพเดทเฟิร์มแวร์ด้วยไฟล์ .uf2

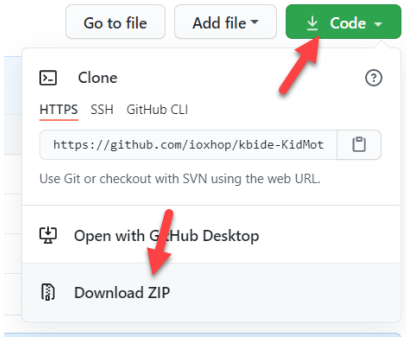
3.3 การอัพเดทเฟิร์มแวร์จากโค้ดโปรแกรม

KidMotorV4-I2C-bridge พัฒนาโดยใช้โปรแกรม Arduino IDE ร่วมกับ Arduino-Pico หากต้องการพัฒนาปรับปรุง หรือเรียนรู้หลักการทำงาน การอัพเดทเฟิร์มแวร์จากโค้ดโปรแกรมมีขั้นตอนดังนี้

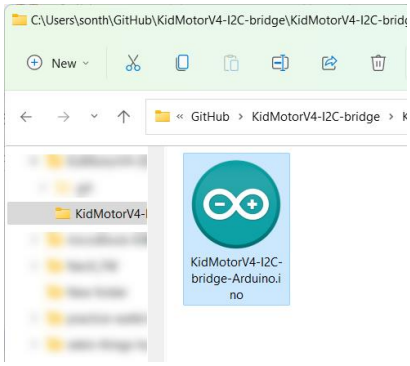
1) ติดตั้งโปรแกรม Arduino IDE

2) ติดตั้ง Arduino-Pico ลงโปรแกรม Arduino IDE

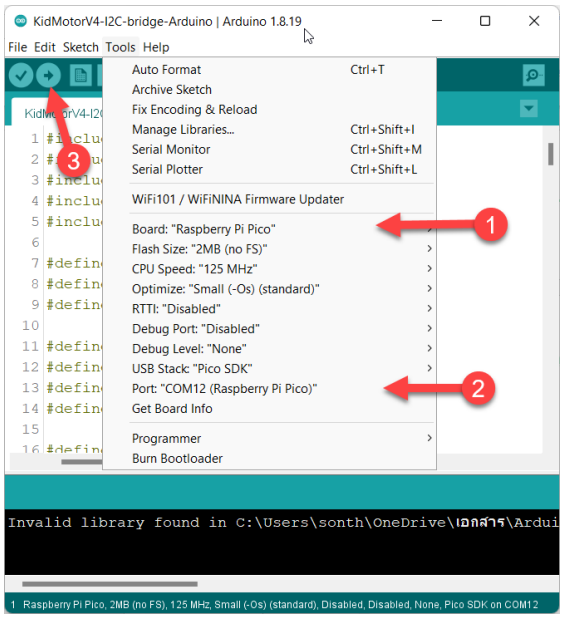
3) ดาวน์โหลด KidMotorV4-I2C-bridge โดยกดปุ่ม Code เลือก Download ZIP



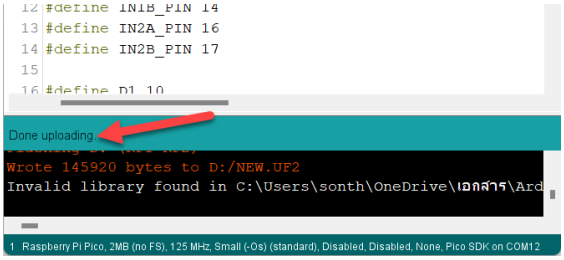
4) คลายไฟล์ .zip ออกมา จะได้ไฟล์ KidMotorV4-I2C-bridge-Arduino.ino มา



5) ที่โปรแกรม Arduino IDE ให้เปิดไฟล์ KidMotorV4-I2C-bridge-Arduino.ino ขึ้นมา จากนั้นเลือกบอร์ดเป็น Raspberry Pi Pico (1) เลือกพอร์ต (2) แล้วกดปุ่มอัพโหลดโค้ดโปรแกรม



6) รอจนกว่าจะขึ้น Done uploading เฟิร์มแวร์ใหม่จะทำงานทันที



เป็นอันจบขั้นตอนการอัพเดทเฟิร์มแวร์จากโค้ดโปรแกรม

บทที่ 4 การเขียนโปรแกรมสั่งงานอุปกรณ์ผ่าน KidMotor

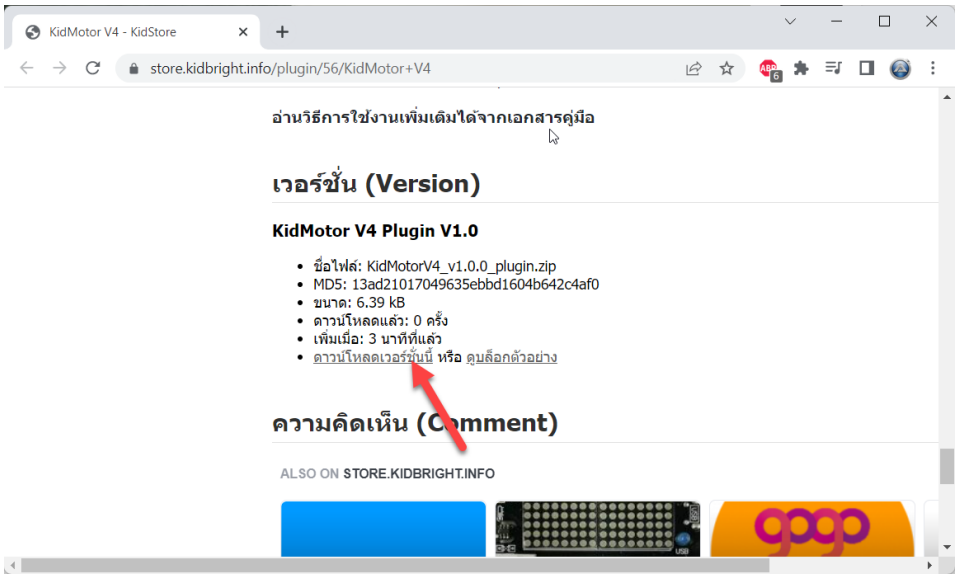
ผู้พัฒนา KidMotor V4 ได้จัดเตรียมปลั๊กอิน / ส่วนเสริม / ไลบารี่ สำหรับใช้ในโปรแกรม KidBrightIDE, KBIDE, microBlock IDE และ ArduinoIDE ไว้ให้แล้ว ผู้ใช้สามารถเลือกใช้โปรแกรมใดก็ได้ในการพัฒนา

4.1 เขียนโปรแกรมสั่งงานด้วย KidBrightIDE

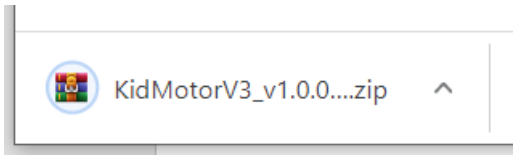
รองรับการเขียนโปรแกรมสั่งงานบอร์ด KidBright32 ทุกรุ่น และบอร์ด OpenKB โดยมีขั้นตอนการเริ่มต้นใช้งานดังนี้

4.1.1 ติดตั้งปลั๊กอิน KidMotor V4

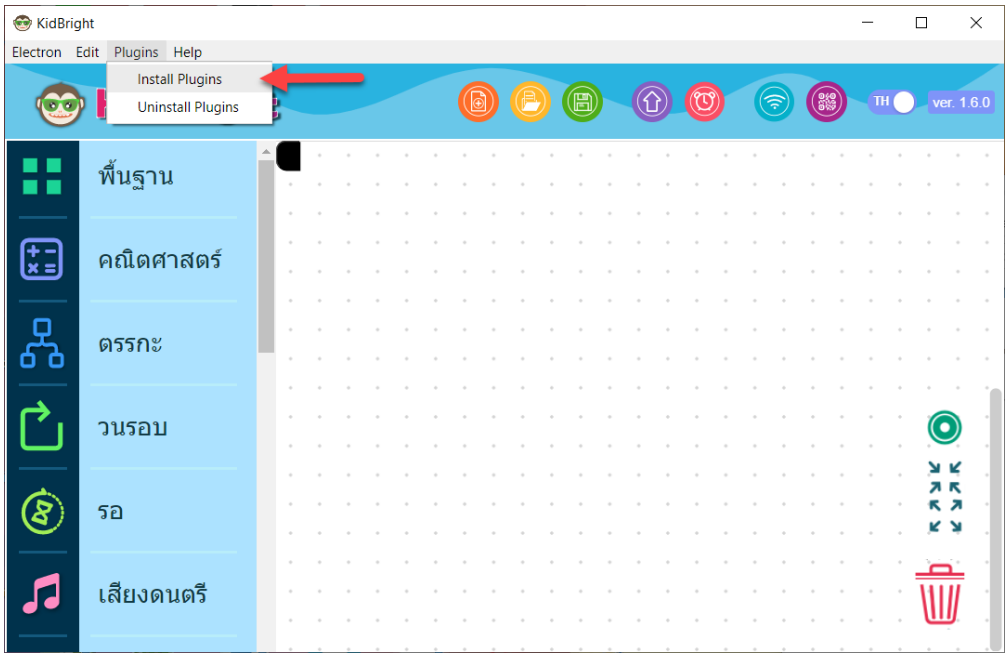
1) ดาวน์โหลดปลั๊กอิน KidMotor V4 ได้ที่ https://store.kidbright.info/plugin/56/KidMotor+V4 โดยเลื่อนมาด้านล่างของหน้า กดข้อความ ดาวน์โหลดเวอร์ชั่นนี้



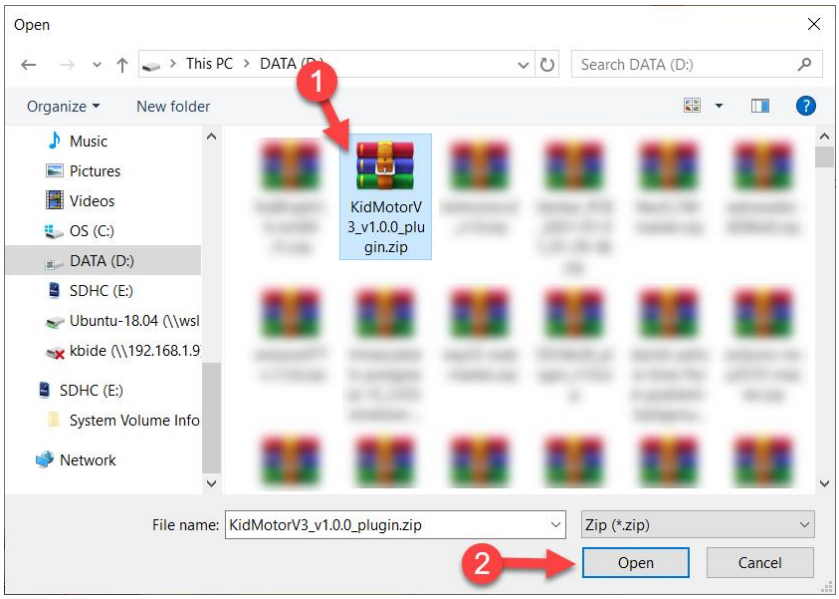
2) จะขึ้นให้ดาวน์โหลดไฟล์ปลั๊กอิน เลือกที่เก็บไฟล์ไว้ที่ใดก็ได้



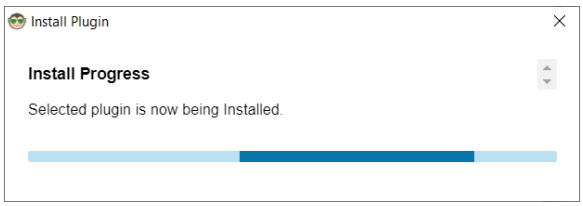
3) เปิดโปรแกรม KidBrightIDE ขึ้นมา กด Plugin > Install Plugin



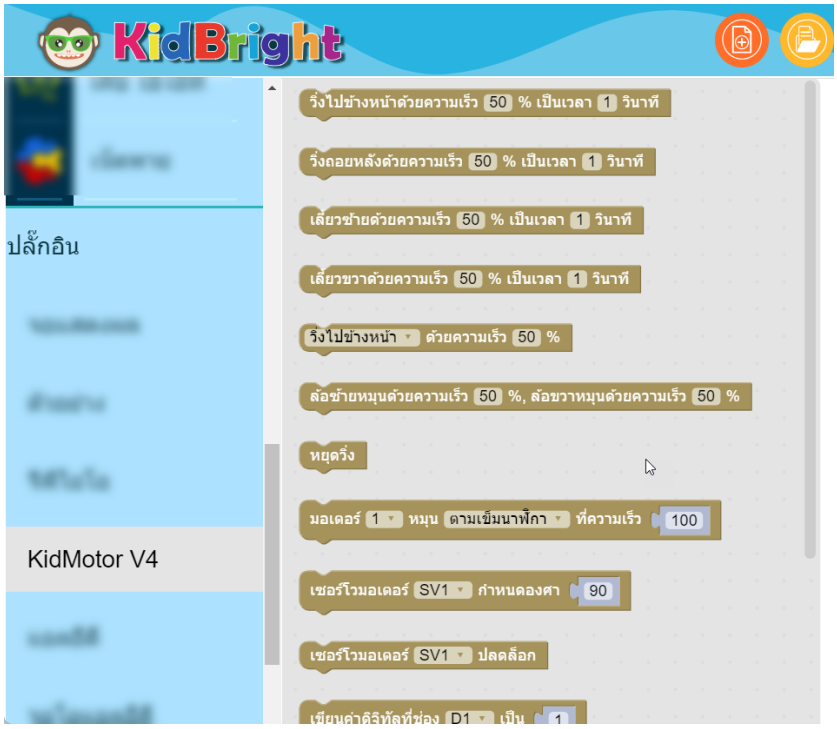
4) กดเลือกไฟล์ปลั๊กอินที่ได้ดาวน์โหลดมา (1) แล้วกดปุ่ม Open (2)



5) รอจนกว่าจะติดตั้งปลั๊กอินเสร็จ



6) หลังจากติดตั้งปลั๊กอิน KidMotor V4 เสร็จแล้ว โปรแกรมจะปิดและเปิดใหม่อัตโนมัติ กดที่เมนู ปลั๊กอิน > KidMotor V4 ก็จะมีบล็อกใหม่เพิ่มเข้ามาแล้ว



4.1.2 การเขียนโปรแกรมสั่งงานมอเตอร์

ปลั๊กอิน KidMotor V4 ได้จัดเตรียมบล็อกคำสั่งสำหรับสั่งงานมอเตอร์โดยแบ่งเป็น 2 กลุ่ม คือกลุ่มบล็อกสำหรับสั่งงานหุ่นยนต์ (รถ) มีบล็อกดังนี้

บล็อกวิ่งไปข้างหน้า ใช้สั่งให้หุ่นยนต์วิ่งไปข้างหน้า ด้วยความเร็ว และเวลาที่กำหนด โดยเมื่อครบเวลาแล้ว หุ่นยนต์จะหยุดวิ่งอัตโนมัติ

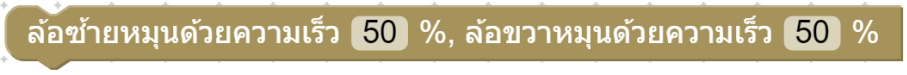
บล็อกวิ่งถอยหลัง ใช้สั่งให้หุ่นยนต์วิ่งถอยหลัง ด้วยความเร็ว และเวลาที่กำหนด โดยเมื่อครบเวลาแล้ว หุ่นยนต์จะหยุดวิ่งอัตโนมัติ

บล็อกเลี้ยวซ้าย ใช้สั่งให้หุ่นยนต์เลี้ยวซ้าย (มอเตอร์ซ้ายหยุดหมุน มอเตอร์ขวาหมุน) ด้วยความเร็ว และเวลาที่กำหนด โดยเมื่อครบเวลาแล้ว หุ่นยนต์จะหยุดวิ่งอัตโนมัติ

บล็อกเลี้ยวขวา ใช้สั่งให้หุ่นยนต์เลี้ยวขวา (มอเตอร์ซ้ายหมุน มอเตอร์ขวาหยุดหมุน) ด้วยความเร็ว และเวลาที่กำหนด โดยเมื่อครบเวลาแล้ว หุ่นยนต์จะหยุดวิ่งอัตโนมัติ

บล็อก...ด้วยความเร็ว ใช้สั่งให้หุ่นยนต์วิ่งไปข้างหน้า/ถอยหลัง/เลี้ยวซ้าย/เลี้ยวขวา ด้วยความเร็วที่กำหนด

บล็อกล็อกซ้าย/ขวาหมุน ใช้สั่งให้หุ่นยนต์วิ่งตามรูปแบบที่กำหนดโดยกำหนดความเร็วของล้อแต่ละข้าง หากกำหนดเป็นค่าบวก 1 ถึง 100 ล้อจะหมุนไปด้านหน้า หากกำหนดค่าเป็นลบ -1 ถึง -100 ล้อจะหมุนไปด้านหลัง หากกำหนดเป็น 0 ล้อจะหยุดหมุน



บล็อกหยุดวิ่ง ใช้สั่งให้หุ่นยนต์หยุดวิ่ง (มอเตอร์ทุกตัวหยุดหมุน)

กลุ่มบล็อกสั่งงานมอเตอร์ มีดังนี้

มอเตอร์...หมุน... ที่ความเร็ว ใช้กำหนดทิศทางและความเร็วของมอเตอร์ที่กำหนด

4.1.3 การเขียนโปรแกรมควบคุมเซอร์โวมอเตอร์

เซอร์โวมอเตอร์...กำหนดองศา... ใช้กำหนดองศาของเซอร์โวมอเตอร์ขาที่กำหนด

เซอร์โวมอเตอร์...ปลดล็อก ใช้ปลดล็อกเซอร์โวมอเตอร์ (หยุดส่งสัญญาณ PWM)

4.1.4 การเขียนโปรแกรมควบคุมอินพุต/เอาต์พุต

เขียนค่าดิจิทัล ใช้สั่งให้ช่อง D1 ถึง D5 ส่งสัญญาณดิจิทัลลอจิก 0 หรือ 1

อ่านค่าดิจิทัล ใช้อ่านค่าดิจิทัลที่ช่อง D1 ถึง D5 ให้ค่าเป็นตัวเลขจำนวนเต็มตามลอจิกที่อ่านได้

อ่านค่าแอนะล็อก ใช้อ่านค่าแอนะล็อกที่ช่อง D1 ถึง D5 ให้ค่าเป็นตัวเลขจำนวนเต็ม 0 ถึง 4095

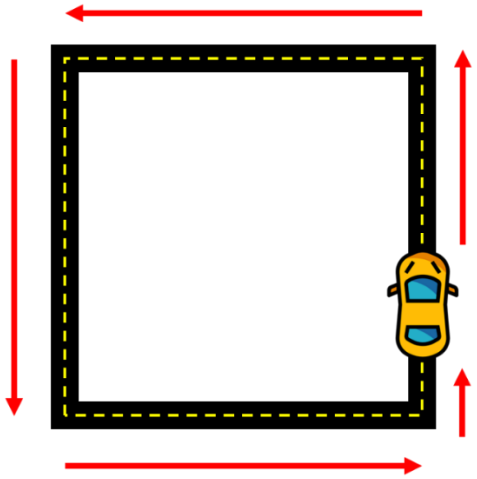
เขียนค่า PWM ใช้สั่งให้ช่อง D1 ถึง D5 ส่งสัญญาณ PWM

4.1.5 การเขียนโปรแกรมอ่านค่าจากเซ็นเซอร์อัลตร้าโซนิค

อัลตร้าโซนิคขา...อ่านระยะ ใช้อ่านค่าระยะจากเซ็นเซอร์อัลตร้าโซนิค หน่วยเซ็นติเมตร

4.1.6 โครงงานรถวิ่งตามโปรแกรม

ตัวอย่างเขียนโปรแกรมให้รถวิ่งตามทิศทางเป็นสี่เหลี่ยมดังนี้



เขียนโปรแกรมได้ดังนี้

หลักการทำงานของโปรแกรม

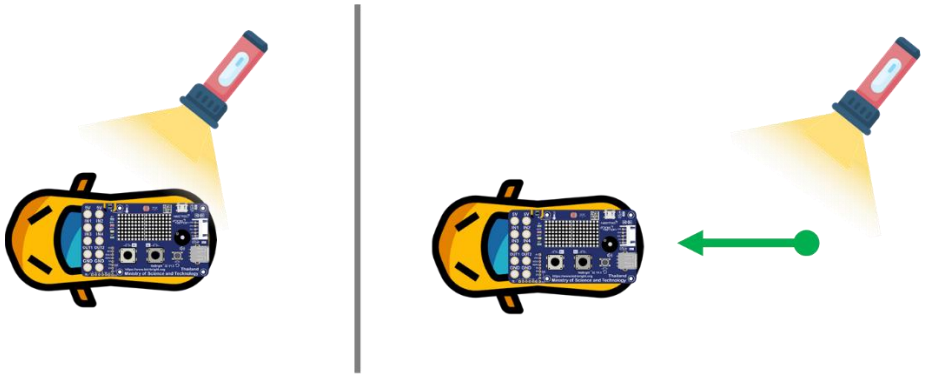
- ใช้บล็อกวนรอบ เพื่อกำหนดให้โค้ดโปรแกรมต่อไปนี้ทำงานตลอดเวลา

- ใช้บล็อกวิ่งไปข้างหน้า เพื่อให้รถวิ่งไปข้างหน้า ตามระยะทางที่กำหนดด้วยเวลา ด้วยความเร็ว 80%
- ใช้บล็อกเลี้ยวซ้าย เพื่อสั่งให้เลี้ยวซ้าย กำหนดระยะเลี้ยวด้วยเวลา ด้วยความเร็วในการเลี้ยว 60%

รถจะวิ่งตรงแล้วเลี่ยวซ้ายไปเรื่อย ๆ ไปเป็นสี่เหลี่ยม

4.1.6 โครงงานรถวิ่งหนีแสง

เมื่อนำแสงเข้าไปใกล้กับตัวรถ รถจะวิ่งไปข้างหน้าเพื่อหนีแสง



โครงงานนี้ใช้เซ็นเซอร์ตรวจจับแสงบนบอร์ด KidBright32 / OpenKB เขียนโปรแกรมได้ดังนี้

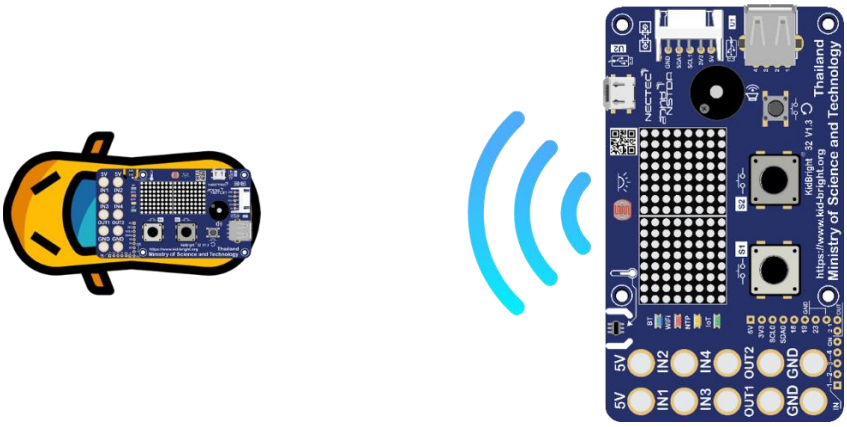
หลักการทำงานของโปรแกรม

- ใช้บล็อกวนรอบ เพื่อกำหนดให้โค้ดโปรแกรมต่อไปนี้ทำงานตลอดเวลา
    - ใช้เงื่อนไข ตรวจสอบว่าเซ็นเซอร์แสงอ่านค่าแสงได้มากกว่า 60% หรือไม่ ถ้าใช่
        - สั่งให้รถวิ่งไปข้างหน้าด้วยความเร็ว 60%
    - ถ้าไม่ใช่ ให้
        - รถหยุดวิ่ง
    - หน่วงเวลาการทำงาน 0.1 วินาที

เมื่อเซ็นเซอร์แสงได้รับแสงจากไฟฉาย รถจะวิ่งไปข้างหน้าเพื่อหนีแสง ตามที่ได้เขียนโปรแกรมไว้

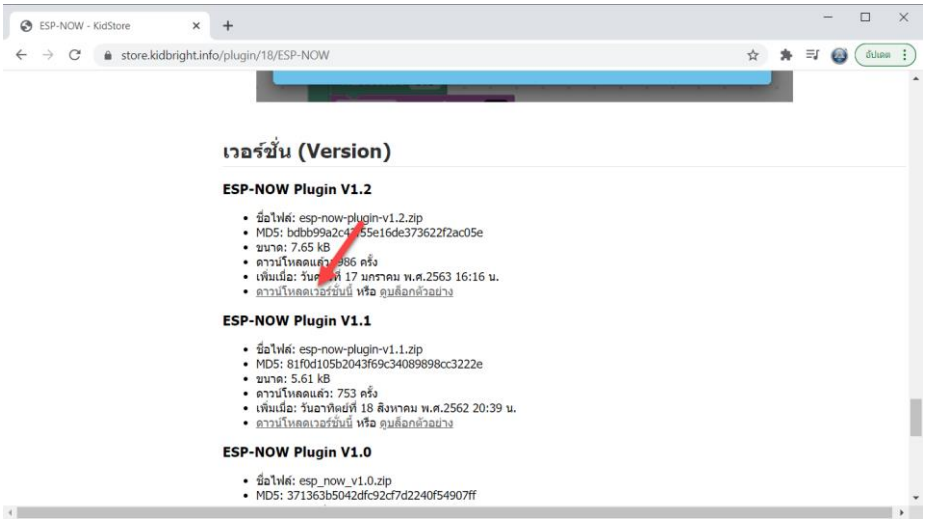
4.1.7 โครงงานรถบังคับ

โครงงานนี้ใช้บอร์ด KidBright32/OpenKB จำนวน 2 ตัว โดยตัวที่ 1 ติดอยู่กับรถและบอร์ด KidMotor V4 และตัวที่ 2 เป็นรีโมทควบคุม

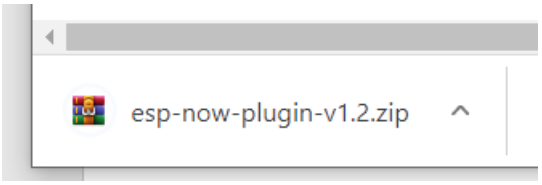


การรับ-ส่งสัญญาณควบคุมรถใช้ ESP-NOW ซึ่งติดตั้งปลั๊กอินเสริมชื่อ ESP-NOW ตามขั้นตอนดังนี้

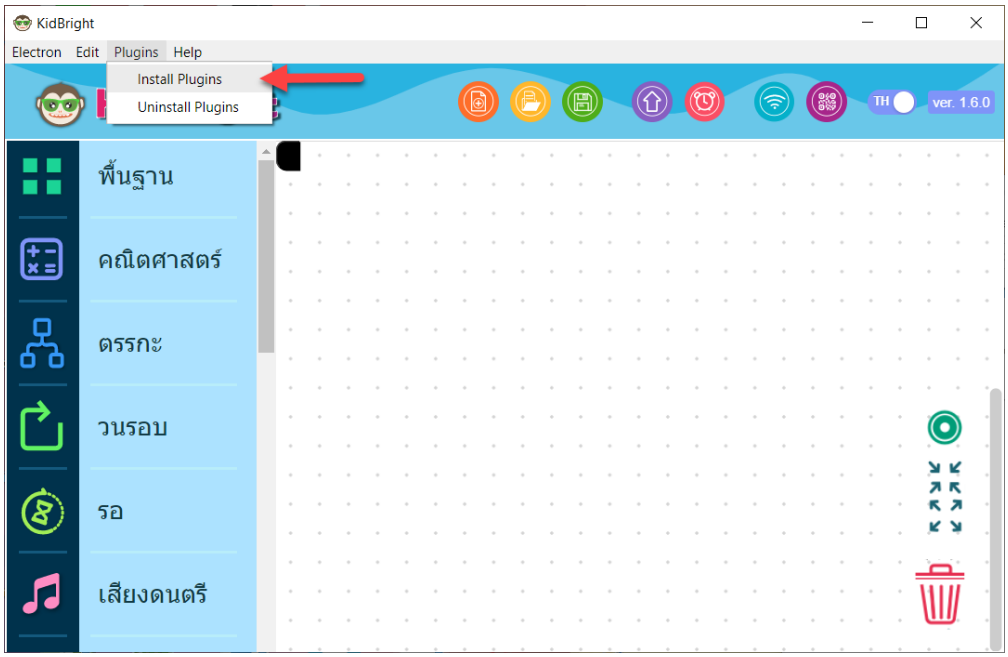
1) เข้าไปดาวน์โหลดปลั๊กอิน ESP-NOW ที่ https://store.kidbright.info/plugin/18/ESP-NOW กด ดาวน์โหลดเวอร์ชั่นนี้ ของเวอร์ชั่นล่าสุด



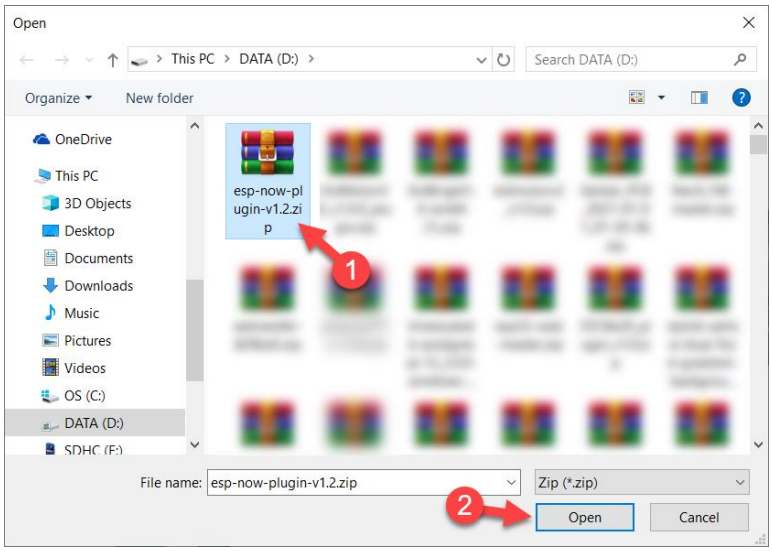
2) จะขึ้นให้ดาวน์โหลดไฟล์ปลั๊กอินแล้ว ให้บันทึกไฟลนี้ไว้ที่ใดก็ได้



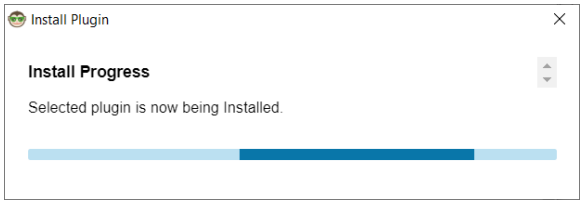
3) ที่โปรแกรม KidBrightIDE กด Plugin > Install Plugin



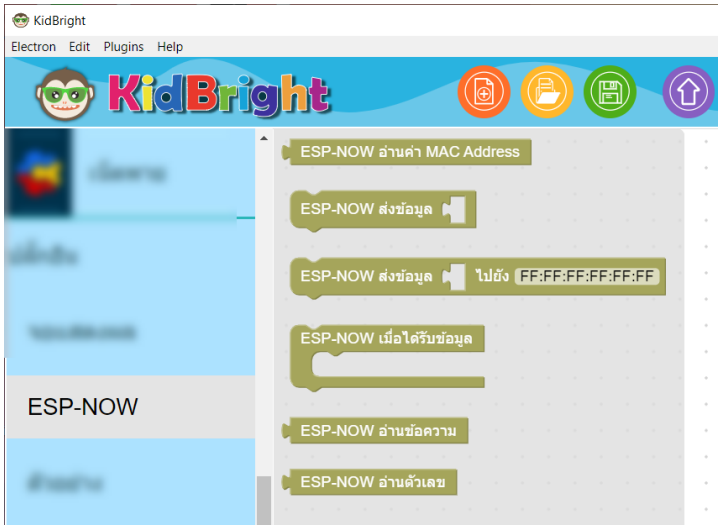
4) กดเลือกไฟล์ปลั๊กอินที่ได้ดาวน์โหลดมา (1) แล้วกดปุ่ม Open (2)



5) รอจนกว่าจะติดตั้งปลั๊กอินเสร็จ



6) หลังจากติดตั้งปลั๊กอิน ESP-NOW เสร็จแล้ว โปรแกรมจะปิดและเปิดใหม่อัตโนมัติ กดที่เมนู ปลั๊กอิน > ESP-NOW ก็จะมีบล็อกใหม่เพิ่มเข้ามาแล้ว



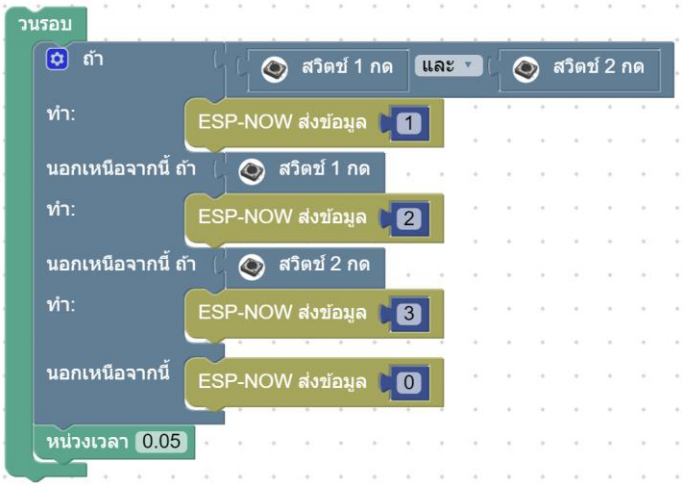
เขียนโปรแกรมที่ฟังรถดังนี้

หลักการทำงานของโปรแกรม

- ใช้บล็อก ESP-NOW เมื่อได้รับข้อมูล กำหนดให้ทำโปรแกรมภายใต้บล็อกนี้เมื่อได้รับข้อมูลจาก ESP-NOW
    - ใช้บล็อก ESP-NOW อ่านตัวเลข อ่านข้อมูลแบบตัวเลขจาก ESP-NOW แล้วเก็บลงตัวแปร x
    - แสดงผลค่าในตัวแปร x บนแอลอีดี 16x8
    - ใช้บล็อกเงื่อนไขตรวจสอบว่าค่าในตัวแปร x คือ 1 ใช่หรือไม่ ถ้าใช่
        - สั่งให้รถวิ่งไปข้างหน้า
    - ถ้าไม่ใช่ ตรวจสอบว่า x คือ 2 ใช่หรือไม่ ถ้าใช่

        - สั่งให้รถเลี้ยวซ้าย
    - ถ้าไม่ใช่ ตรวจสอบว่า x คือ 3 ใช่หรือไม่ ถ้าใช่
        - สั่งให้รถเลี้ยวขวา
    - ถ้าไม่ตรงกับเงื่อนไขใดเลย ให้
        - สั่งให้รถหยุดวิ่ง
- ใช้บล็อกวนรอบ กำหนดโปรแกรมภายใต้บล็อกนี้ทำงานตลอดเวลา
    - หน่วงเวลา 0.5 วินาที

ที่ฝั่งรีโมทเขียนโปรแกรมดังนี้



หลักการทำงานของโปรแกรม

- ใช้บล็อกวนรอบ กำหนดโปรแกรมภายใต้บล็อกนี้ทำงานตลอดเวลา
    - ใช้บล็อกเงื่อนไขตรวจสอบว่าสวิตช์ 1 และสวิตช์ 2 กดอยู่หรือไม่ ถ้าใช่
        - ส่งเลข 1 ผ่าน ESP-NOW
    - ใช้บล็อกเงื่อนไขตรวจสอบว่าสวิตช์ 1 กดอยู่หรือไม่ ถ้าใช่
        - ส่งเลข 2 ผ่าน ESP-NOW
    - ใช้บล็อกเงื่อนไขตรวจสอบว่าสวิตช์ 1 กดอยู่หรือไม่ ถ้าใช่
        - ส่งเลข 3 ผ่าน ESP-NOW
    - ถ้าไม่ตรงกับเงื่อนไขใดเลย
        - ส่งเลข 0 ผ่าน ESP-NOW
    - หน่วงเวลา 0.05 วินาที

หลังจากอัพโหลดโปรแกรมที่ฝั่งรถและฝั่งรีโมทแล้ว ทดสอบการทำงานดังนี้

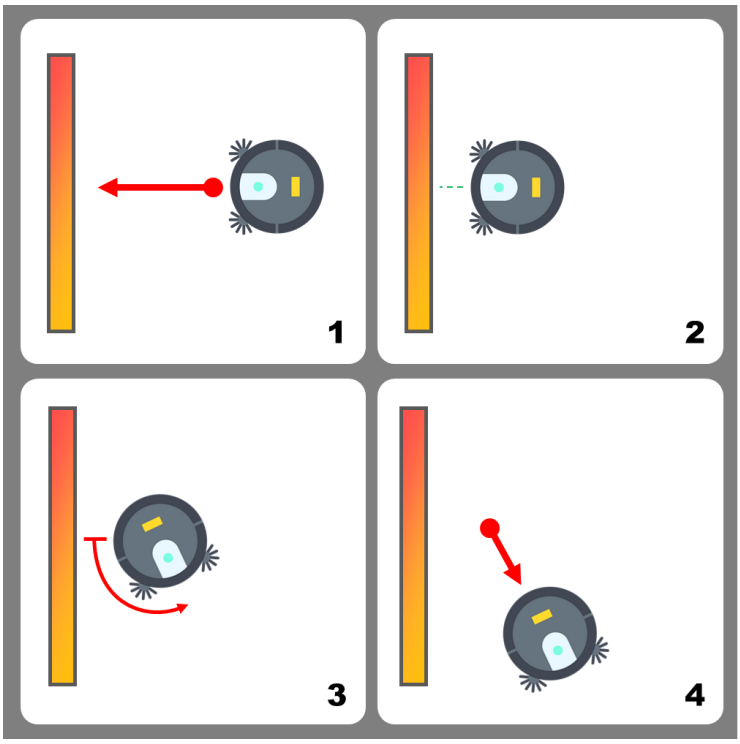
- กดปุ่ม S1 และ S2 ฝั่งรีโมทค้างไว้ ที่ฝั่งรถ แอลอีดี 16x8 จะแสดงเลข 1 และรถวิ่งไปข้างหน้า

- กดปุ่ม S1 ฟั่งรีโมทค้างไว้ ที่ฟั่งรถ แอลอีดี 16x8 จะแสดงเลข 2 และรถเลี้ยวซ้าย
- กดปุ่ม S2 ฟั่งรีโมทค้างไว้ ที่ฟั่งรถ แอลอีดี 16x8 จะแสดงเลข 2 และรถเลี้ยวขวา
- ไม่กดปุ่มใดเลย ที่ฟั่งรถ แอลอีดี 16x8 จะแสดงเลข 0 และรถหยุดวิ่ง

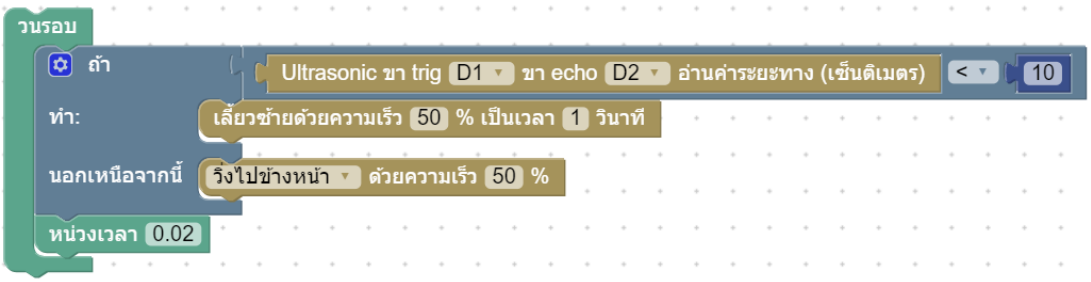
กรณีรถวิ่งทิศทางไม่ถูกต้อง ให้อ่านวิธีแก้ไขได้ในภาคผนวก ก

4.2.9 โครงงานหุ่นยนต์ดูดฝุ่น

หุ่นยนต์ดูดฝุ่นมีหลักการวิ่งที่สำคัญคือวิ่งไม่ชนขอบพนัง / สิ่งกีดขวาง โดยใช้เซ็นเซอร์ตรวจจับวัตถุ วัดระยะห่างระหว่างหุ่นยนต์กับพนัง / สิ่งกีดขวาง หากมีระยะที่ใกล้เกินไป ตัวหุ่นยนต์จะเลี้ยวไปทิศทางอื่น



จากหลักการวิ่งดังกล่าวสามารถจำลองการทำงาน / สร้างหุ่นยนต์ดูดฝุ่นได้โดยใช้เซ็นเซอร์อัลตร้าโซนิค โดยต่อเซ็นเซอร์อัลตร้าโซนิค HC-SR04p ตามหัวข้อ 2.6 การต่อเซ็นเซอร์อัลตร้าโซนิค แล้วเขียนโปรแกรมดังนี้



หลักการทำงานของโปรแกรม

- ใช้บล็อกวนรอบ กำหนดโปรแกรมภายใต้บล็อกนี้ทำงานตลอดเวลา

- ใช้บล็อกเงื่อนไขตรวจสอบว่าระยะห่างเซ็นเซอร์กับสิ่งกีดขวาง น้อยกว่ากว่า 10 เซ็นฯ หรือไม่ ถ้าใช่
  - สั่งให้รถเลี้ยวซ้าย ด้วยความเร็ว 50% เป็นเวลา 1 วินาที
- ถ้าไม่ตรงกับเงื่อนไขใดเลย
  - สั่งให้รถวิ่งตรงต่อไป
- หน่วงเวลา 0.02 วินาที

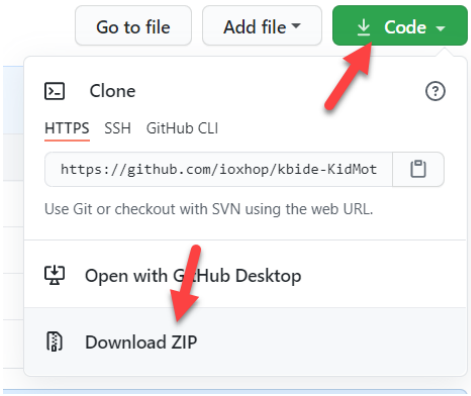
หลังจากอัพโหลดโปรแกรมที่ฟ่งรถและฟ่งรีโมทแล้ว รถจะเริ่มวิ่งทันที เมื่อนำรถวางลงกับพื้นแล้วเซ็นเซอร์เจอสิ่งกีดขวาง รถจะเลี้ยวซ้ายเพื่อหลบสิ่งกีดขวาง แล้ววิ่งตรงต่อไป

4.2 เขียนโปรแกรมสั่งงานด้วย KBIDE

รองรับการเขียนโปรแกรมสั่งงานบอร์ด KidBright32 ทุกรุ่น บอร์ด OpenKB บอร์ด IPST-WiFi บอร์ด Arduino และบอร์ด ESP32 ทุกรุ่น โดยมีขั้นตอนการเริ่มต้นใช้งานดังนี้

4.2.1 ติดตั้งปลั๊กอิน KidMotor V4

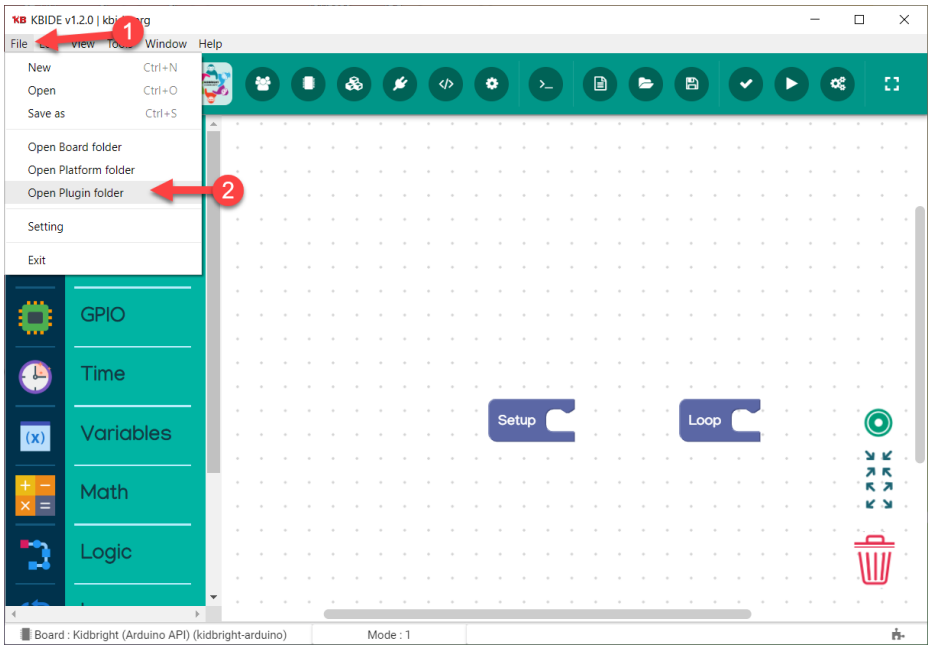
1) ดาวน์โหลดปลั๊กอิน KidMotor V4 ได้ที่ https://github.com/ArtronShop/kbide-KidMotorV4 โดยกดปุ่ม Code เลือก Download ZIP



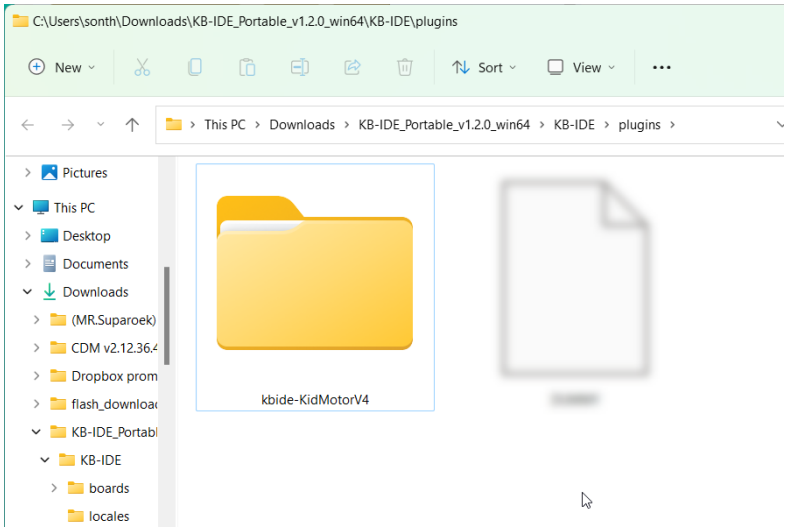
2) จะขึ้นให้ดาวน์โหลดไฟล์ปลั๊กอิน เลือกที่เก็บไฟล์ไว้ที่ใดก็ได้

3) คลายไฟล์ ZIP ด้วยโปรแกรม WinRAR หรือ 7-zip จะได้โฟลเดอร์ kbide-KidMotorV4-main มา

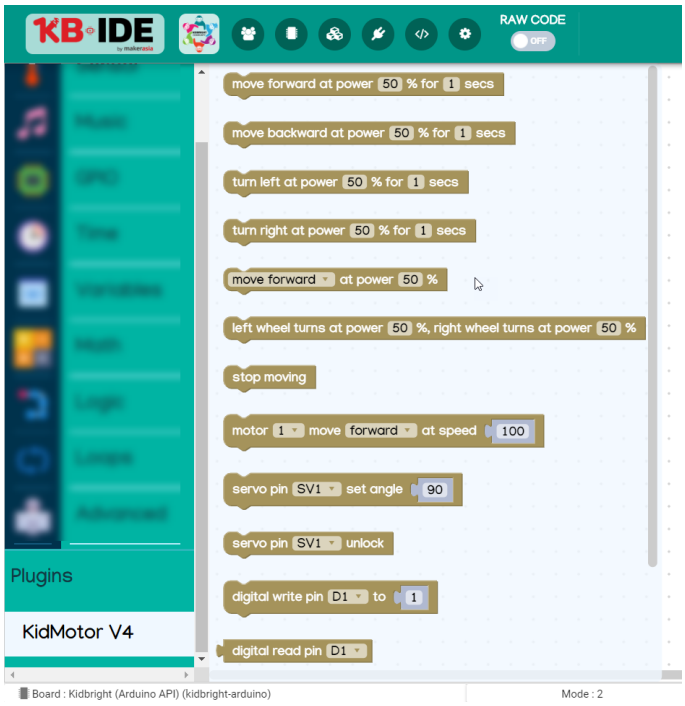
4) เปิดโปรแกรม KBIDE ขึ้นมา กด File > Open Plugin folder



5) โฟลเดอร์ plugin จะเปิดขึ้นมา ให้คัดลอกโฟลเดอร์ kbide-KidMotorV4-main มาใส่ในโฟลเดอร์นี้



6) ปิดและเปิดโปรแกรม KBIDE ใหม่อีกครั้ง กดที่เมนู Plugin > KidMotor V4 ก็จะมีบล็อกใหม่เพิ่มเข้ามาแล้ว



4.2.2 การเขียนโปรแกรมสั่งงานมอเตอร์

ปลั๊กอิน KidMotor V4 ได้จัดเตรียมบล็อกคำสั่งสำหรับสั่งงานมอเตอร์โดยแบ่งเป็น 2 กลุ่ม คือกลุ่มบล็อกสำหรับสั่งงานหุ่นยนต์ (รถ) มีบล็อกดังนี้

บล็อกวิ่งไปข้างหน้า ใช้สั่งให้หุ่นยนต์วิ่งไปข้างหน้า ด้วยความเร็ว และเวลาที่กำหนด โดยเมื่อครบเวลาแล้ว หุ่นยนต์จะหยุดวิ่งอัตโนมัติ

บล็อกวิ่งถอยหลัง ใช้สั่งให้หุ่นยนต์วิ่งถอยหลัง ด้วยความเร็ว และเวลาที่กำหนด โดยเมื่อครบเวลาแล้ว หุ่นยนต์จะหยุดวิ่งอัตโนมัติ

บล็อกเลี้ยวซ้าย ใช้สั่งให้หุ่นยนต์เลี้ยวซ้าย (มอเตอร์ซ้ายหยุดหมุน มอเตอร์ขวาหมุน) ด้วยความเร็ว และเวลาที่กำหนด โดยเมื่อครบเวลาแล้ว หุ่นยนต์จะหยุดวิ่งอัตโนมัติ

บล็อกเลี้ยวขวา ใช้สั่งให้หุ่นยนต์เลี้ยวขวา (มอเตอร์ซ้ายหมุน มอเตอร์ขวาหยุดหมุน) ด้วยความเร็ว และเวลาที่กำหนด โดยเมื่อครบเวลาแล้ว หุ่นยนต์จะหยุดวิ่งอัตโนมัติ

บล็อก...ด้วยความเร็ว ใช้สั่งให้หุ่นยนต์วิ่งไปข้างหน้า/ถอยหลัง/เลี้ยวซ้าย/เลี้ยวขวา ด้วยความเร็วที่กำหนด

บล็อกล็อกซ้าย/ขวาหมุน ใช้สั่งให้หุ่นยนต์วิ่งตามรูปแบบที่กำหนดโดยกำหนดความเร็วของล้อแต่ละข้าง หากกำหนดเป็นค่าบวก 1 ถึง 100 ล้อจะหมุนไปด้านหน้า หากกำหนดค่าเป็นลบ -1 ถึง -100 ล้อจะหมุนไปด้านหลัง หากกำหนดเป็น 0 ล้อจะหยุดหมุน

บล็อกหยุดวิ่ง ใช้สั่งให้หุ่นยนต์หยุดวิ่ง (มอเตอร์ทุกตัวหยุดหมุน)

กลุ่มบล็อกสั่งงานมอเตอร์ มีดังนี้

มอเตอร์...หมุน... ที่ความเร็ว ใช้กำหนดทิศทางและความเร็วของมอเตอร์ที่กำหนด

4.2.3 การเขียนโปรแกรมควบคุมเซอร์โวมอเตอร์

เซอร์โวมอเตอร์...กำหนดองศา... ใช้กำหนดองศาของเซอร์โวมอเตอร์ขาที่กำหนด

เซอร์โวมอเตอร์...ปลดล็อก ใช้ปลดล็อกเซอร์โวมอเตอร์ (หยุดส่งสัญญาณ PWM)

4.2.4 การเขียนโปรแกรมควบคุมอินพุต/เอาต์พุต

เขียนค่าดิจิทัล ใช้สั่งให้ช่อง D1 ถึง D5 ส่งสัญญาณดิจิทัลลอจิก 0 หรือ 1

อ่านค่าดิจิทัล ใช้อ่านค่าดิจิทัลที่ช่อง D1 ถึง D5 ให้ค่าเป็นตัวเลขจำนวนเต็มตามลอจิกที่อ่านได้

อ่านค่าแอนะล็อก ใช้อ่านค่าแอนะล็อกที่ช่อง D1 ถึง D5 ให้ค่าเป็นตัวเลขจำนวนเต็ม 0 ถึง 4095

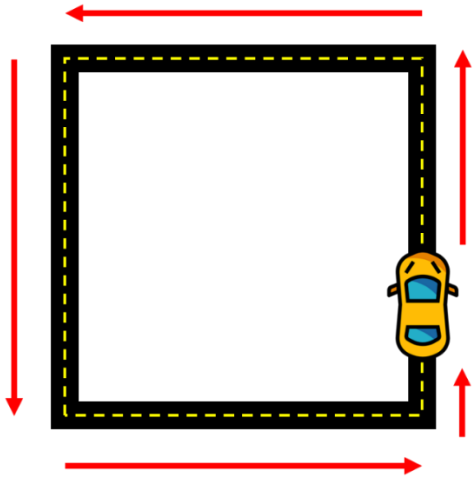
เขียนค่า PWM ใช้สั่งให้ช่อง D1 ถึง D5 ส่งสัญญาณ PWM

4.2.5 การเขียนโปรแกรมอ่านค่าจากเซ็นเซอร์อัลตร้าโซนิค

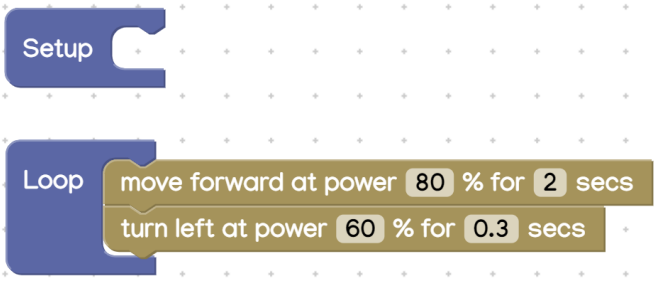
อัลตร้าโซนิคขา...อ่านระยะ ใช้อ่านค่าระยะจากเซ็นเซอร์อัลตร้าโซนิค หน่วยเซ็นติเมตร

4.2.6 โครงงานรถวิ่งตามโปรแกรม

ตัวอย่างเขียนโปรแกรมให้รถวิ่งตามทิศทางเป็นสี่เหลี่ยมดังนี้



เขียนโปรแกรมได้ดังนี้



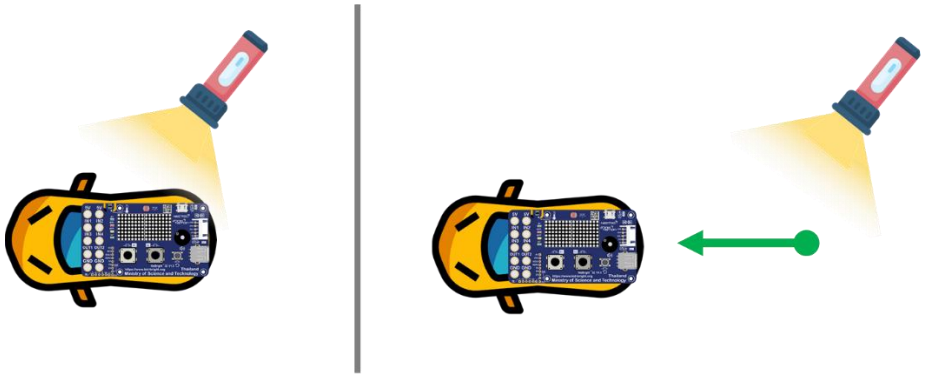
หลักการทำงานของโปรแกรม

- ใช้บล็อกวนรอบ เพื่อกำหนดให้โค้ดโปรแกรมต่อไปนี้ทำงานตลอดเวลา
    - ใช้บล็อกวิ่งไปข้างหน้า เพื่อให้รถวิ่งไปข้างหน้า ตามระยะทางที่กำหนดด้วยเวลา ด้วยความเร็ว 80%
    - ใช้บล็อกเลี้ยวซ้าย เพื่อสั่งให้เลี้ยวซ้าย กำหนดระยะเลี้ยวด้วยเวลา ด้วยความเร็วในการเลี้ยว 60%

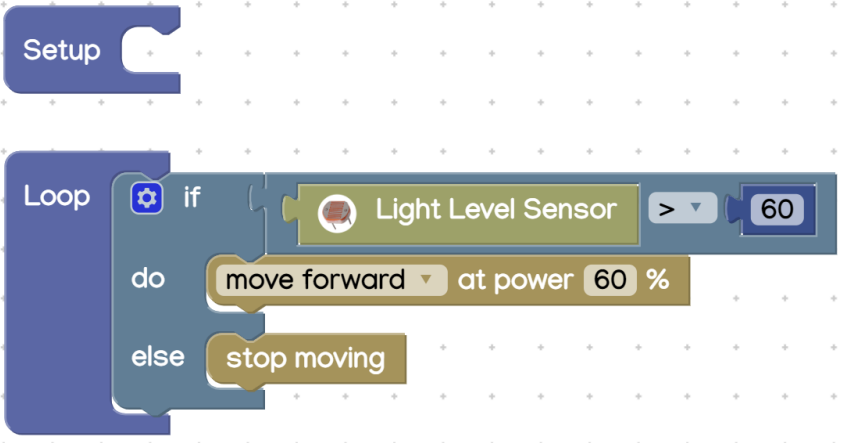
รถจะวิ่งตรงแล้วเลี่ยวซ้ายไปเรื่อย ๆ ไปเป็นสี่เหลี่ยม

4.2.7 โครงงานรถวิ่งหนีแสง

เมื่อนำแสงเข้าไปใกล้กับตัวรถ รถจะวิ่งไปข้างหน้าเพื่อหนีแสง



โครงงานนี้ใช้เซ็นเซอร์ตรวจจับแสงบนบอร์ด KidBright32 / OpenKB เขียนโปรแกรมได้ดังนี้



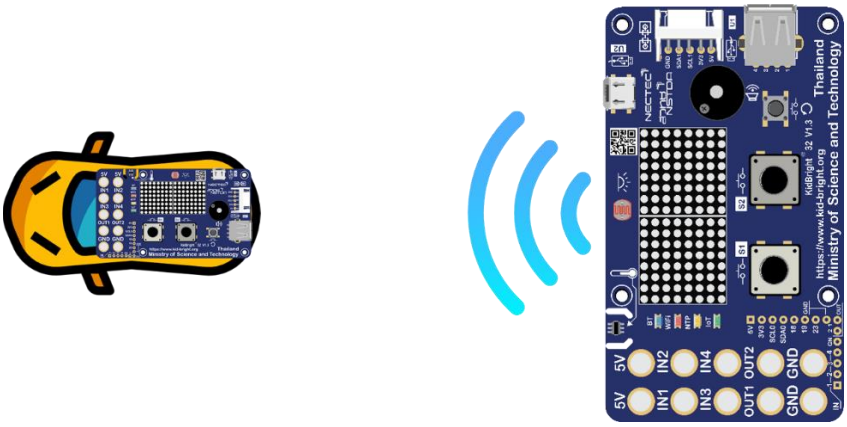
หลักการทำงานของโปรแกรม

- ใช้บล็อกวนรอบ เพื่อกำหนดให้โค้ดโปรแกรมต่อไปนี้ทำงานตลอดเวลา
    - ใช้เงื่อนไข ตรวจสอบว่าเซ็นเซอร์แสงอ่านค่าแสงได้มากกว่า 60% หรือไม่ ถ้าใช่
        - สั่งให้รถวิ่งไปข้างหน้าด้วยความเร็ว 60%
    - ถ้าไม่ใช่ ให้
        - รถหยุดวิ่ง
    - หน่วงเวลาการทำงาน 0.1 วินาที

เมื่อเซ็นเซอร์แสงได้รับแสงจากไฟฉาย รถจะวิ่งไปข้างหน้าเพื่อหนีแสง ตามที่ได้เขียนโปรแกรมไว้

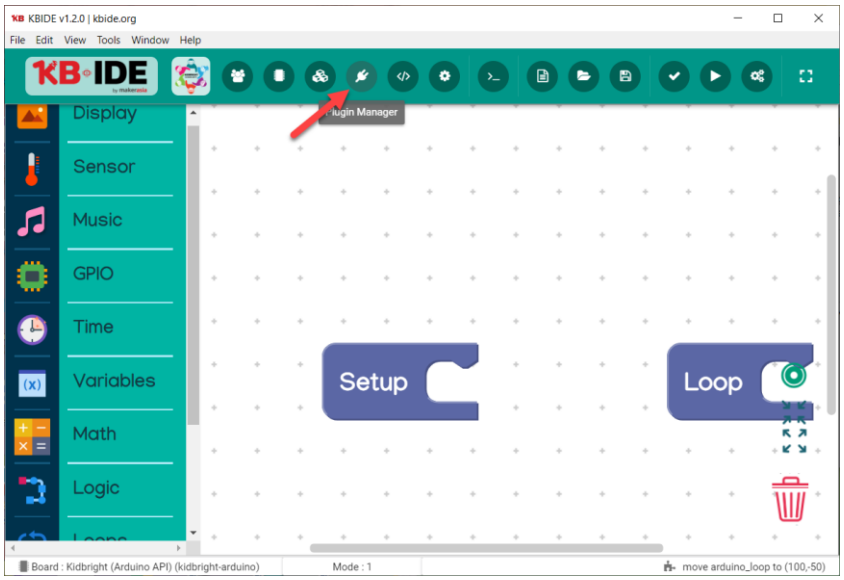
4.2.8 โครงงานรถบังคับ

โครงงานนี้ใช้บอร์ด KidBright32/OpenKB จำนวน 2 ตัว โดยตัวที่ 1 ติดอยู่กับรถและบอร์ด KidMotor V4 และตัวที่ 2 เป็นรีโมทควบคุม

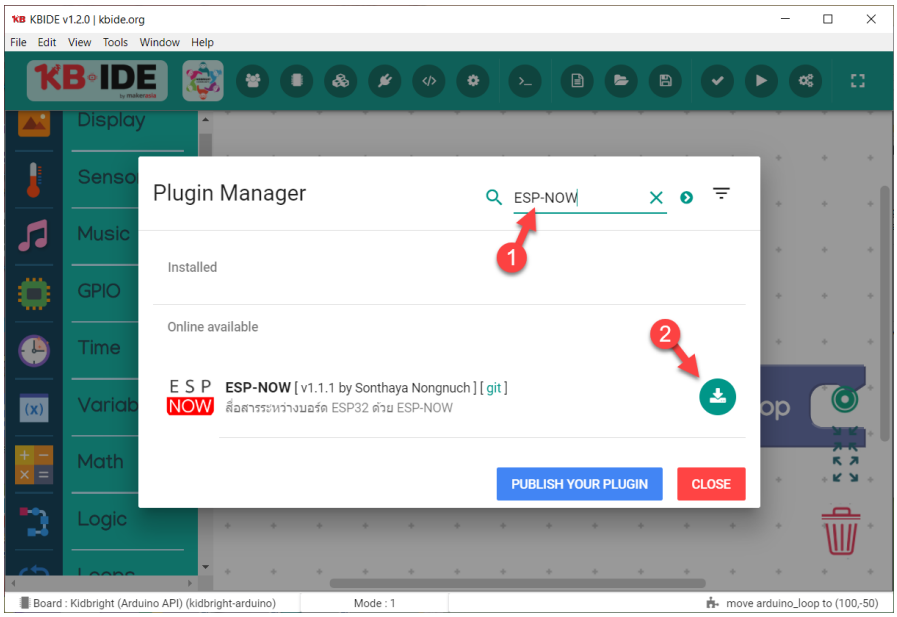


การรับ-ส่งสัญญาณควบคุมรถใช้ ESP-NOW ซึ่งติดตั้งปลั๊กอินเสริมชื่อ ESP-NOW ตามขั้นตอนดังนี้

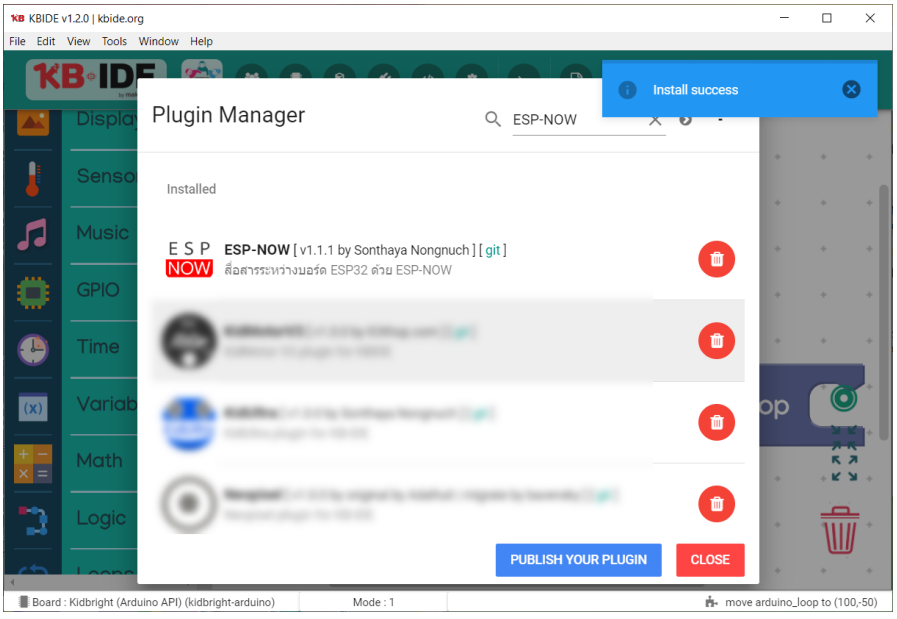
1) ติดตั้งปลั๊กอิน ESP-NOW โดยกดที่เมนู Plugin



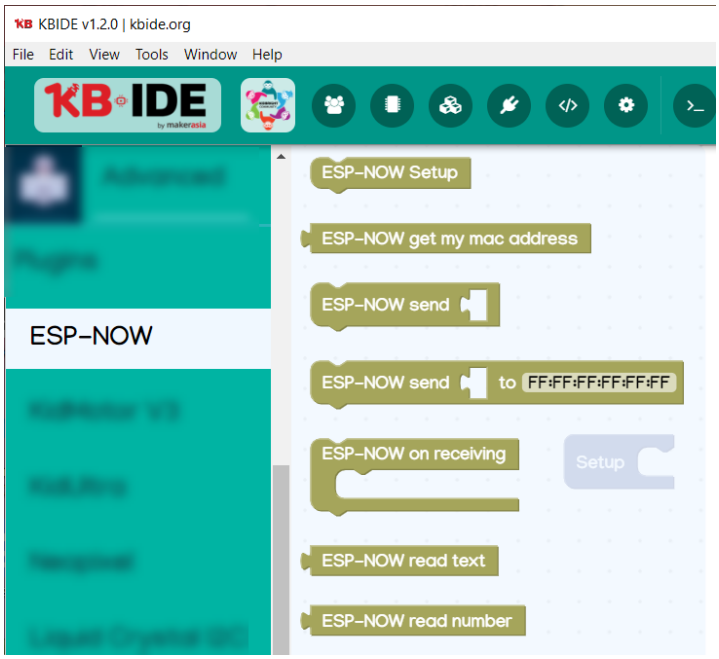
2) ช่องค้นหา พิมพ์ ESP-NOW (1) จากนั้นกดปุ่มติดตั้ง (2)



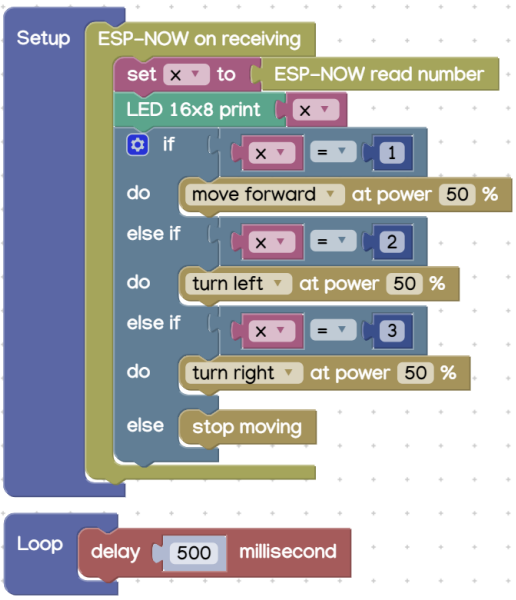
3) เมื่อติดตั้งเสร็จแล้ว ชื่อ ESP-NOW จะขึ้นมาอยู่ในรายการ Installed (ดังรูป) ให้กดปิดหน้าต่าง Plugin Manager ไปได้เลย



4) หลังจากติดตั้งปลั๊กอิน ESP-NOW เสร็จแล้ว กดที่เมนู Plugin > ESP-NOW ก็จะมีบล็อกใหม่เพิ่มเข้ามาแล้ว



เขียนโปรแกรมที่ฝั่งรถดังนี้



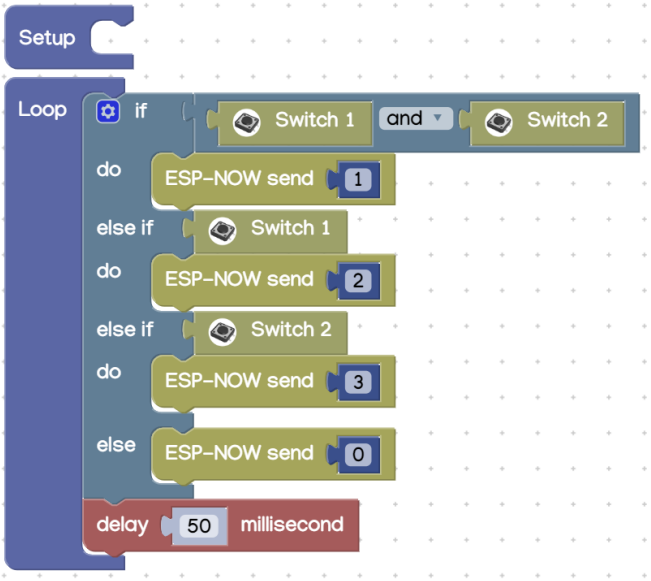
หลักการทำงานของโปรแกรม

- ใช้บล็อก ESP-NOW เมื่อได้รับข้อมูล กำหนดให้ทำโปรแกรมภายใต้บล็อกนี้เมื่อได้รับข้อมูลจาก ESP-NOW
    - ใช้บล็อก ESP-NOW อ่านตัวเลข อ่านข้อมูลแบบตัวเลขจาก ESP-NOW แล้วเก็บลงตัวแปร x
    - แสดงผลค่าในตัวแปร x บนแอลอีดี 16x8
    - ใช้บล็อกเงื่อนไขตรวจสอบว่าค่าในตัวแปร x คือ 1 ใช่หรือไม่ ถ้าใช่
        - สั่งให้รถวิ่งไปข้างหน้า

- ถ้าไม่ใช่ ตรวจสอบว่า x คือ 2 ใช่หรือไม่ ถ้าใช่
  - สั่งให้รถเลี้ยวซ้าย
- ถ้าไม่ใช่ ตรวจสอบว่า x คือ 3 ใช่หรือไม่ ถ้าใช่
  - สั่งให้รถเลี้ยวขวา
- ถ้าไม่ตรงกับเงื่อนไขใดเลย ให้
  - สั่งให้รถหยุดวิ่ง

- ใช้บล็อกวนรอบ กำหนดโปรแกรมภายใต้บล็อกนี้ทำงานตลอดเวลา
  - หน่วงเวลา 0.5 วินาที

ที่ฟังรีโมทเขียนโปรแกรมดังนี้



หลักการทำงานของโปรแกรม

- ใช้บล็อกวนรอบ กำหนดโปรแกรมภายใต้บล็อกนี้ทำงานตลอดเวลา
  - ใช้บล็อกเงื่อนไขตรวจสอบว่าสวิตช์ 1 และสวิตช์ 2 กดอยู่หรือไม่ ถ้าใช่
    - ส่งเลข 1 ผ่าน ESP-NOW
  - ใช้บล็อกเงื่อนไขตรวจสอบว่าสวิตช์ 1 กดอยู่หรือไม่ ถ้าใช่
    - ส่งเลข 2 ผ่าน ESP-NOW
  - ใช้บล็อกเงื่อนไขตรวจสอบว่าสวิตช์ 1 กดอยู่หรือไม่ ถ้าใช่
    - ส่งเลข 3 ผ่าน ESP-NOW
  - ถ้าไม่ตรงกับเงื่อนไขใดเลย
    - ส่งเลข 0 ผ่าน ESP-NOW
  - หน่วงเวลา 0.05 วินาที

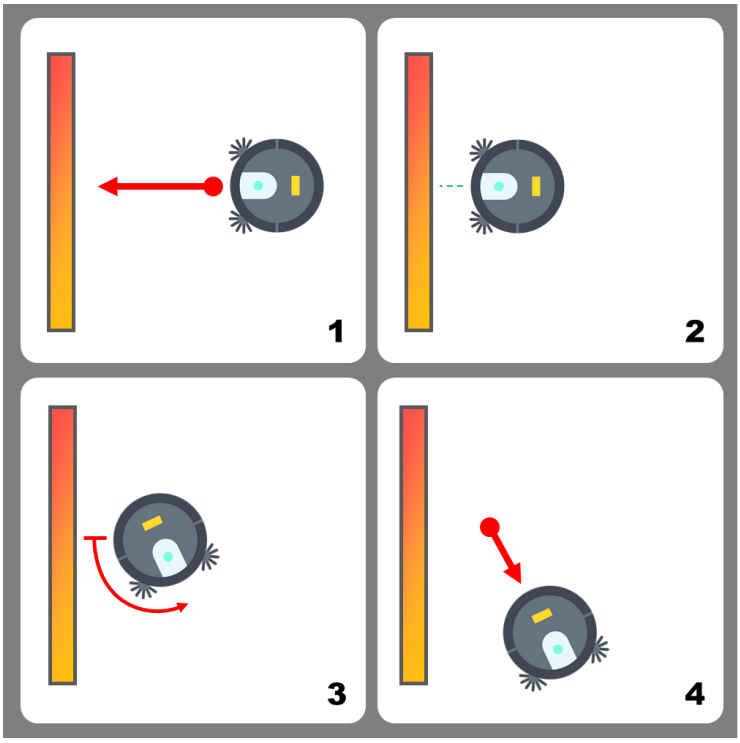
หลังจากอัพโหลดโปรแกรมที่ฟั่งรถและฟั่งรีโมทแล้ว ทดสอบการทำงานดังนี้

- กดปุ่ม S1 และ S2 ฟั่งรีโมทค้างไว้ ที่ฟั่งรถ แอลอีดี 16x8 จะแสดงเลข 1 และรถวิ่งไปข้างหน้า
- กดปุ่ม S1 ฟั่งรีโมทค้างไว้ ที่ฟั่งรถ แอลอีดี 16x8 จะแสดงเลข 2 และรถเลี้ยวซ้าย
- กดปุ่ม S2 ฟั่งรีโมทค้างไว้ ที่ฟั่งรถ แอลอีดี 16x8 จะแสดงเลข 2 และรถเลี้ยวขวา
- ไม่กดปุ่มใดเลย ที่ฟั่งรถ แอลอีดี 16x8 จะแสดงเลข 0 และรถหยุดวิ่ง

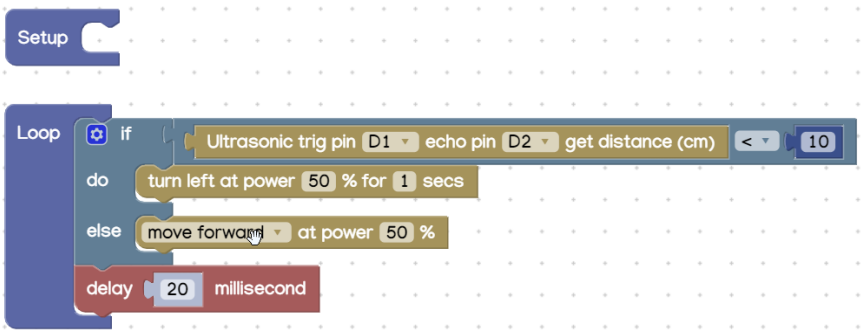
กรณีรถวิ่งทิศทางไม่ถูกต้อง ให้อ่านวิธีแก้ไขได้ในภาคผนวก

4.2.9 โครงงานหุ่นยนต์ดูดฝุ่น

หุ่นยนต์ดูดฝุ่นมีหลักการวิ่งที่สำคัญคือวิ่งไม่ชนขอบผนัง / สิ่งกีดขวาง โดยใช้เซ็นเซอร์ตรวจจับวัตถุ วัดระยะห่างระหว่างหุ่นยนต์กับผนัง / สิ่งกีดขวาง หากมีระยะที่ใกล้เกินไป ตัวหุ่นยนต์จะเลี้ยวไปทิศทางอื่น



จากหลักการวิ่งดังกล่าวสามารถจำลองการทำงาน / สร้างหุ่นยนต์ดูดฝุ่นได้โดยใช้เซ็นเซอร์อัลตร้าโซนิค โดยต่อเซ็นเซอร์อัลตร้าโซนิค HC-SR04p ตามหัวข้อ 2.6 การต่อเซ็นเซอร์อัลตร้าโซนิค แล้วเขียนโปรแกรมดังนี้



หลักการทำงานของโปรแกรม

- ใช้บล็อกวนรอบ กำหนดโปรแกรมภายใต้บล็อกนี้ทำงานตลอดเวลา
    - ใช้บล็อกเงื่อนไขตรวจสอบว่าระยะห่างเซ็นเซอร์กับสิ่งกีดขวาง น้อยกว่ากว่า 10 เซ็นฯ หรือไม่ ถ้าใช่
        - สั่งให้รถเลี้ยวซ้าย ด้วยความเร็ว 50% เป็นเวลา 1 วินาที
    - ถ้าไม่ตรงกับเงื่อนไขใดเลย
        - สั่งให้รถวิ่งตรงต่อไป
    - หน่วงเวลา 0.02 วินาที

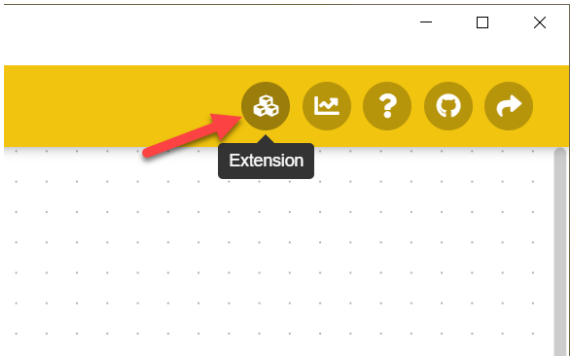
หลังจากอัพโหลดโปรแกรมที่ฟั่งรถและฟั่งรีโมทแล้ว รถจะเริ่มวิ่งทันที เมื่อนำรถวางลงกับพื้นแล้วเซ็นเซอร์เจอสิ่งกีดขวาง รถจะเลี้ยวซ้ายเพื่อหลบสิ่งกีดขวาง แล้ววิ่งตรงต่อไป

4.3 เขียนโปรแกรมสั่งงานด้วย microBlock IDE

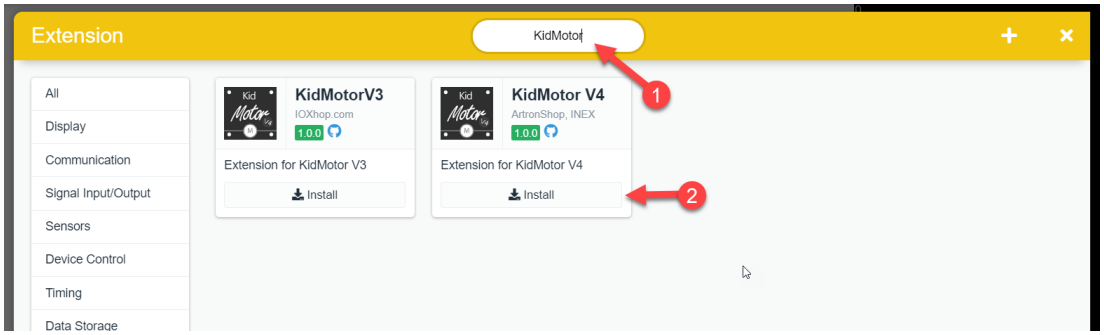
รองรับการเขียนโปรแกรมสั่งงานบอร์ด KidBright32 ทุกรุ่น บอร์ด OpenKB บอร์ด IPST-WiFi โดยมีขั้นตอนการเริ่มต้นใช้งานดังนี้

4.4.1 ติดตั้งปลั๊กอิน KidMotor V4

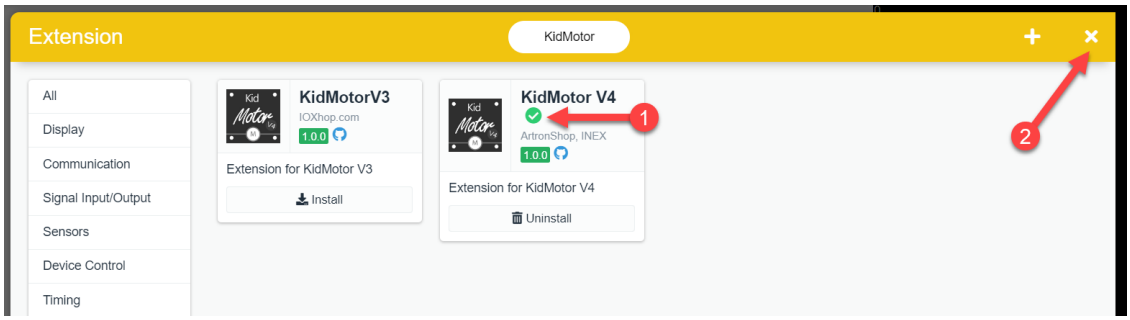
1) เปิดโปรแกรม microBlock IDE ขึ้นมา กดปุ่ม Extension



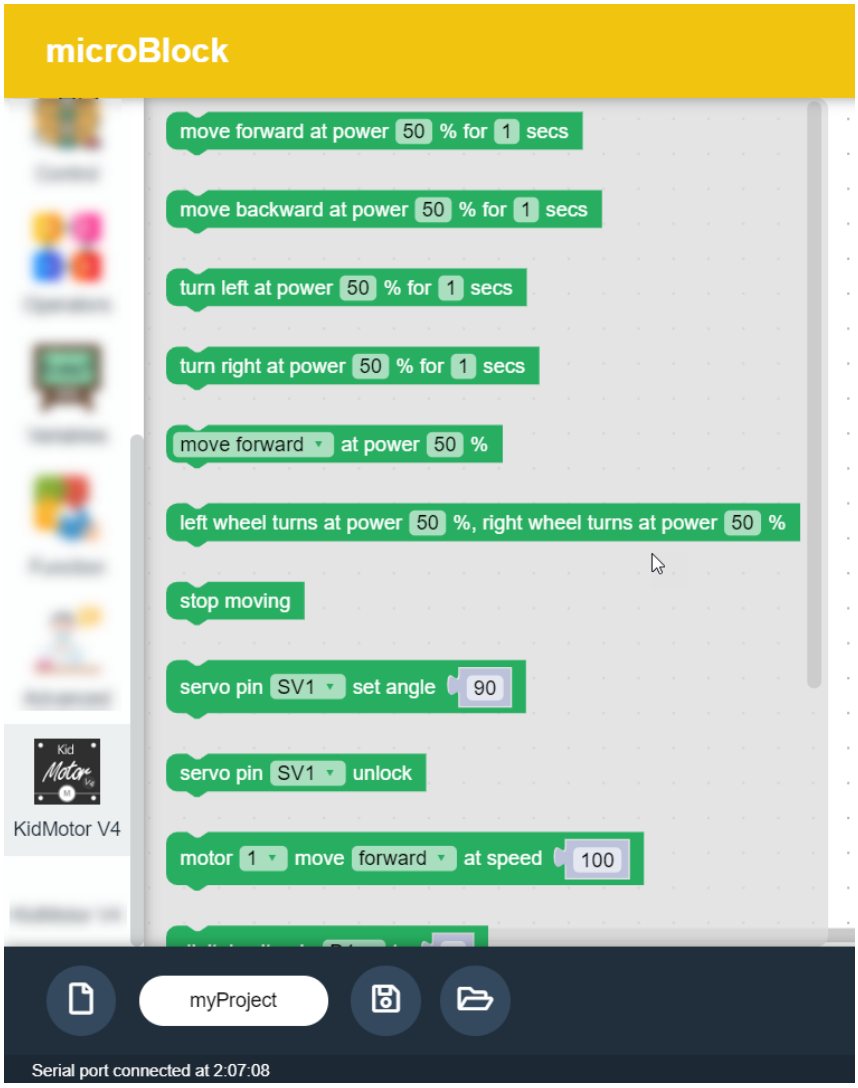
2) หน้าต่าง Extension จะเปิดขึ้นมา ให้ค้นหา KidMotor (1) แล้วกดปุ่ม Install (2)



3) รอจนกว่าจะติดตั้งเสร็จ เมื่อติดตั้งเสร็จแล้วจะมีเครื่องหมายถูกแสดง (1) ให้กดปิดหน้าต่างไปได้เลย (2)



4) ที่หมวด KidMotorV3 จะมีบล็อกเพิ่มขึ้นมาแล้ว

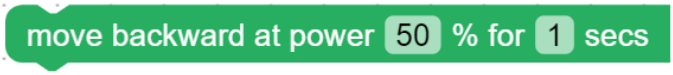


4.4.2 การเขียนโปรแกรมสั่งงานมอเตอร์

ปลั๊กอิน KidMotor V4 ได้จัดเตรียมบล็อกคำสั่งสำหรับสั่งงานมอเตอร์โดยแบ่งเป็น 2 กลุ่ม คือกลุ่มบล็อกสำหรับสั่งงานหุ่นยนต์ (รถ) มีบล็อกดังนี้

บล็อกวิ่งไปข้างหน้า ใช้สั่งให้หุ่นยนต์วิ่งไปข้างหน้า ด้วยความเร็ว และเวลาที่กำหนด โดยเมื่อครบเวลาแล้ว หุ่นยนต์จะหยุดวิ่งอัตโนมัติ

บล็อกวิ่งถอยหลัง ใช้สั่งให้หุ่นยนต์วิ่งถอยหลัง ด้วยความเร็ว และเวลาที่กำหนด โดยเมื่อครบเวลาแล้ว หุ่นยนต์จะหยุดวิ่งอัตโนมัติ



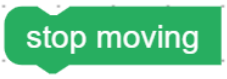
บล็อกเลี้ยวซ้าย ใช้สั่งให้หุ่นยนต์เลี้ยวซ้าย (มอเตอร์ซ้ายหยุดหมุน มอเตอร์ขวาหมุน) ด้วยความเร็ว และเวลาที่กำหนด โดยเมื่อครบเวลาแล้ว หุ่นยนต์จะหยุดวิ่งอัตโนมัติ

บล็อกเลี้ยวขวา ใช้สั่งให้หุ่นยนต์เลี้ยวขวา (มอเตอร์ซ้ายหมุน มอเตอร์ขวาหยุดหมุน) ด้วยความเร็ว และเวลาที่กำหนด โดยเมื่อครบเวลาแล้ว หุ่นยนต์จะหยุดวิ่งอัตโนมัติ

บล็อก...ด้วยความเร็ว ใช้สั่งให้หุ่นยนต์วิ่งไปข้างหน้า/ถอยหลัง/เลี้ยวซ้าย/เลี้ยวขวา ด้วยความเร็วที่กำหนด

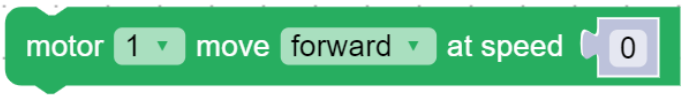
บล็อกล็อกซ้าย/ขวาหมุน ใช้สั่งให้หุ่นยนต์วิ่งตามรูปแบบที่กำหนดโดยกำหนดความเร็วของล้อแต่ละข้าง หากกำหนดเป็นค่าบวก 1 ถึง 100 ล้อจะหมุนไปด้านหน้า หากกำหนดค่าเป็นลบ -1 ถึง -100 ล้อจะหมุนไปด้านหลัง หากกำหนดเป็น 0 ล้อจะหยุดหมุน

บล็อกหยุดวิ่ง ใช้สั่งให้หุ่นยนต์หยุดวิ่ง (มอเตอร์ทุกตัวหยุดหมุน)



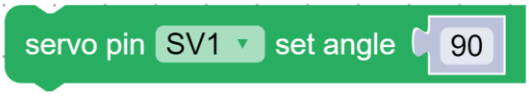
กลุ่มบล็อกสั่งงานมอเตอร์ มีดังนี้

มอเตอร์...หมุน... ที่ความเร็ว ใช้กำหนดทิศทางและความเร็วของมอเตอร์ที่กำหนด



4.4.3 การเขียนโปรแกรมควบคุมเซอร์โวมอเตอร์

เซอร์โวมอเตอร์...กำหนดองศา... ใช้กำหนดองศาของเซอร์โวมอเตอร์ขาที่กำหนด



เซอร์โวมอเตอร์...ปลดล็อก ใช้ปลดล็อกเซอร์โวมอเตอร์ (หยุดส่งสัญญาณ PWM)

4.4.4 การเขียนโปรแกรมควบคุมอินพุต/เอาต์พุต

เขียนค่าดิจิทัล ใช้สั่งให้ช่อง D1 ถึง D5 ส่งสัญญาณดิจิทัลลอจิก 0 หรือ 1

อ่านค่าดิจิทัล ใช้อ่านค่าดิจิทัลที่ช่อง D1 ถึง D5 ให้ค่าเป็นตัวเลขจำนวนเต็มตามลอจิกที่อ่านได้

อ่านค่าแอนะล็อก ใช้อ่านค่าแอนะล็อกที่ช่อง D1 ถึง D5 ให้ค่าเป็นตัวเลขจำนวนเต็ม 0 ถึง 4095

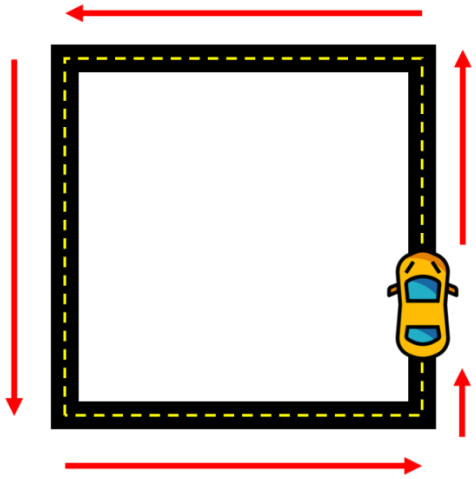
เขียนค่า PWM ใช้สั่งให้ช่อง D1 ถึง D5 ส่งสัญญาณ PWM

4.4.5 การเขียนโปรแกรมอ่านค่าจากเซ็นเซอร์อัลตร้าโซนิค

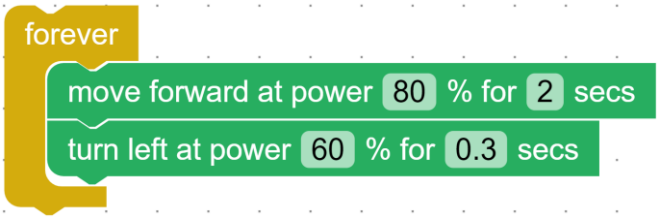
อัลตร้าโซนิคขา...อ่านระยะ ใช้อ่านค่าระยะจากเซ็นเซอร์อัลตร้าโซนิค หน่วยเซ็นติเมตร

4.4.6 โครงงานรถวิ่งตามโปรแกรม

ตัวอย่างเขียนโปรแกรมให้รถวิ่งตามทิศทางเป็นสี่เหลี่ยมดังนี้



เขียนโปรแกรมได้ดังนี้



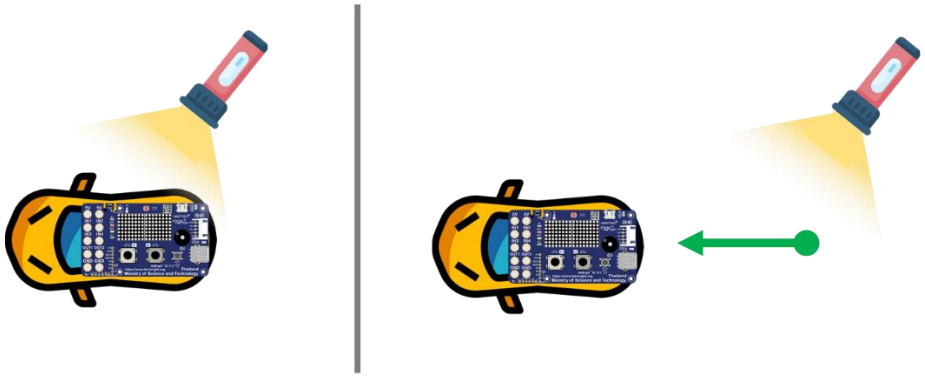
หลักการทำงานของโปรแกรม

- ใช้บล็อกวนรอบ เพื่อกำหนดให้โค้ดโปรแกรมต่อไปนี้ทำงานตลอดเวลา
    - ใช้บล็อกวิ่งไปข้างหน้า เพื่อให้รถวิ่งไปข้างหน้า ตามระยะทางที่กำหนดด้วยเวลา ด้วยความเร็ว 80%
    - ใช้บล็อกเลี้ยวซ้าย เพื่อสั่งให้เลี้ยวซ้าย กำหนดระยะเลี้ยวด้วยเวลา ด้วยความเร็วในการเลี้ยว 60%

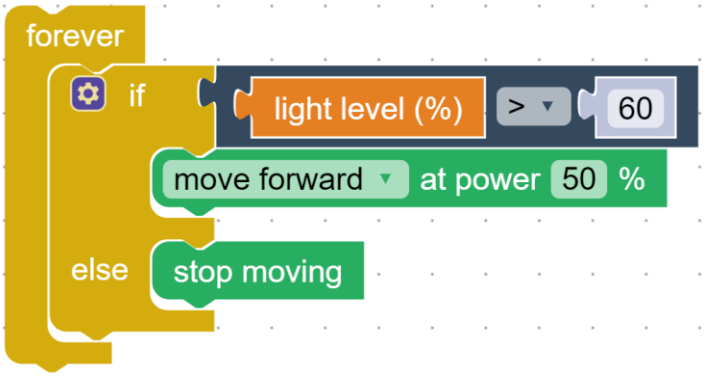
รถจะวิ่งตรงแล้วเลี่ยวซ้ายไปเรื่อย ๆ ไปเป็นสี่เหลี่ยม

4.4.7 โครงงานรถวิ่งหนีแสง

เมื่อนำแสงเข้าไปใกล้กับตัวรถ รถจะวิ่งไปข้างหน้าเพื่อหนีแสง



โครงงานนี้ใช้เซ็นเซอร์ตรวจจับแสงบนบอร์ด KidBright32 / OpenKB เขียนโปรแกรมได้ดังนี้



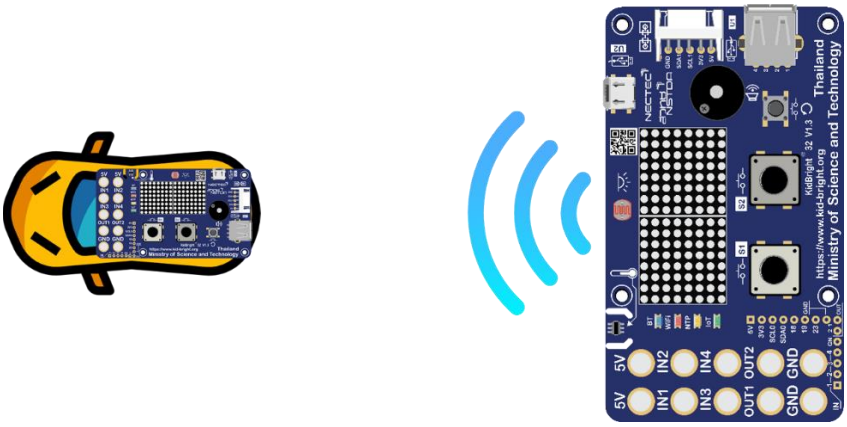
หลักการทำงานของโปรแกรม

- ใช้บล็อกวนรอบ เพื่อกำหนดให้โค้ดโปรแกรมต่อไปนี้ทำงานตลอดเวลา
    - ใช้เงื่อนไข ตรวจสอบว่าเซ็นเซอร์แสงอ่านค่าแสงได้มากกว่า 60% หรือไม่ ถ้าใช่
        - สั่งให้รถวิ่งไปข้างหน้าด้วยความเร็ว 60%
    - ถ้าไม่ใช่ ให้
        - รถหยุดวิ่ง
    - หน่วงเวลาการทำงาน 0.1 วินาที

เมื่อเซ็นเซอร์แสงได้รับแสงจากไฟฉาย รถจะวิ่งไปข้างหน้าเพื่อหนีแสง ตามที่ได้เขียนโปรแกรมไว้

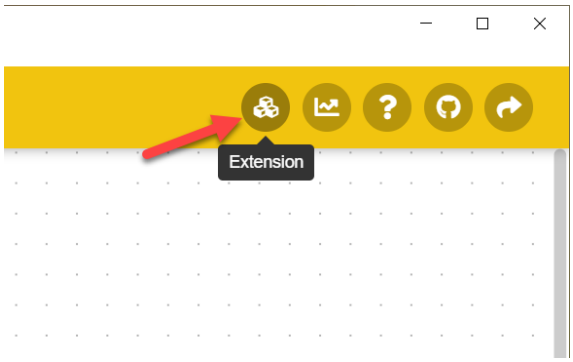
4.4.8 โครงงานรถบังคับ

โครงงานนี้ใช้บอร์ด KidBright32/OpenKB จำนวน 2 ตัว โดยตัวที่ 1 ติดอยู่กับรถและบอร์ด KidMotor V4 และตัวที่ 2 เป็นรีโมทควบคุม

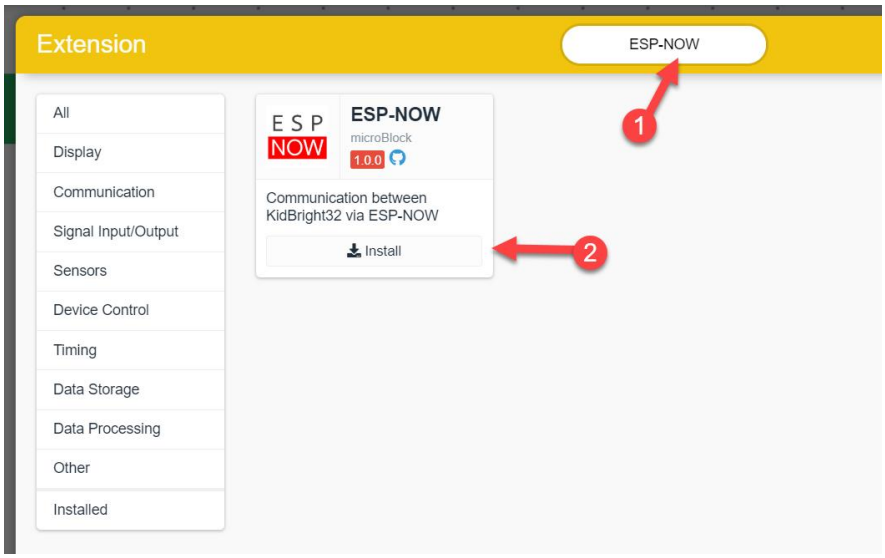


การรับ-ส่งสัญญาณควบคุมรถใช้ ESP-NOW ซึ่งติดตั้งปลั๊กอินเสริมชื่อ ESP-NOW ตามขั้นตอนดังนี้

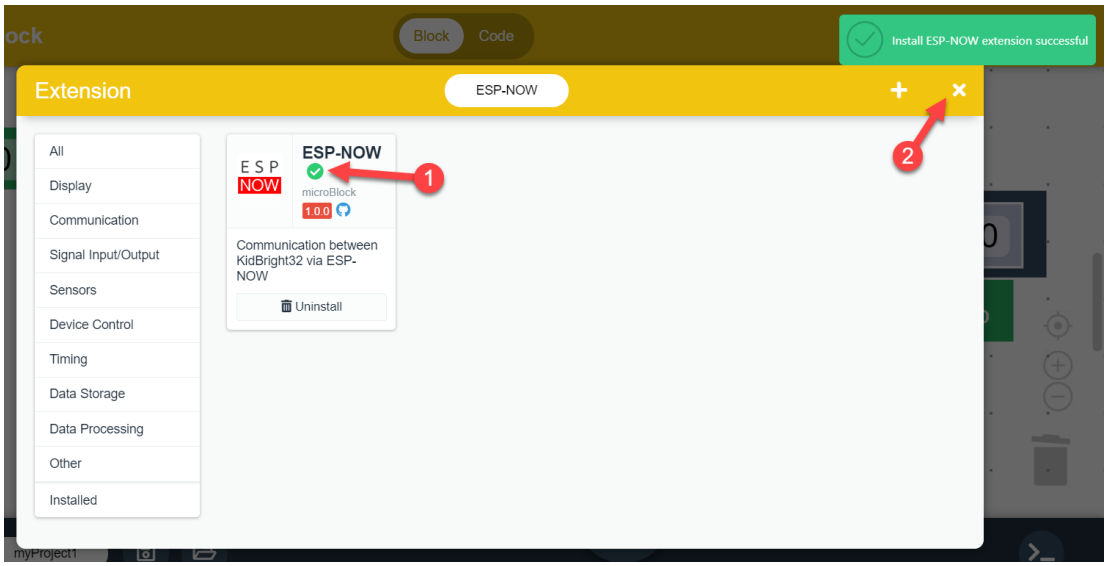
1) ติดตั้งปลั๊กอิน ESP-NOW โดยกดที่เมนู Plugin



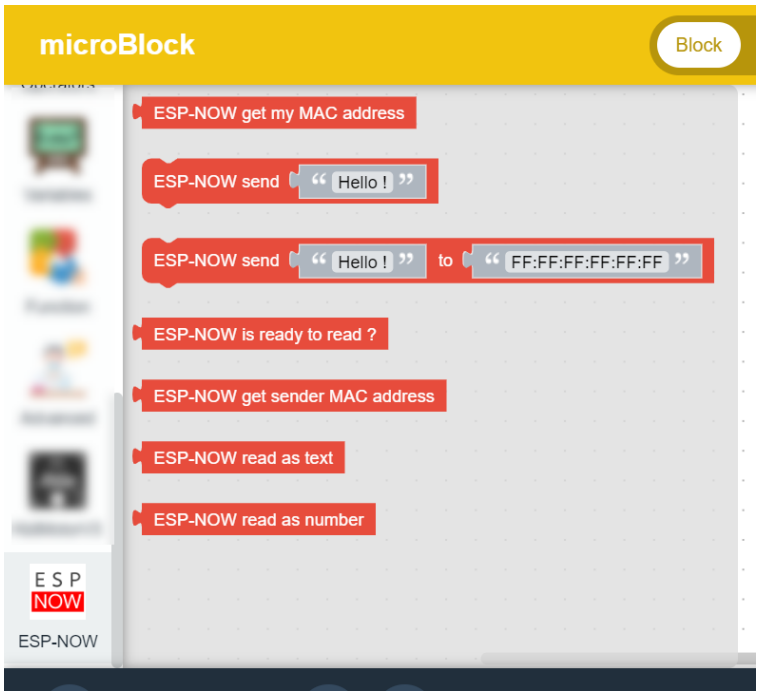
2) ช่องค้นหา พิมพ์ ESP-NOW (1) จากนั้นกดปุ่ม Install (2)



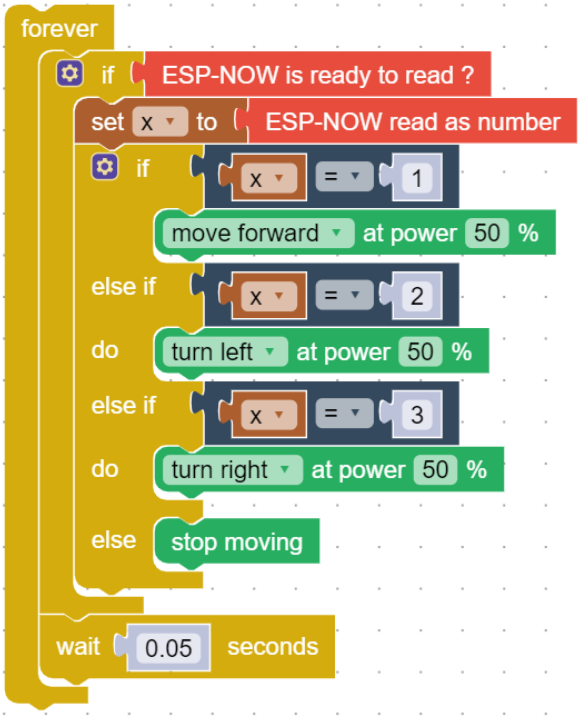
3) รอจนกว่าจะติดตั้งเสร็จ เมื่อติดตั้งเสร็จแล้วจะมีเครื่องหมายถูกแสดง (1) ให้กดปิดหน้าต่างไปได้เลย (2)



4) ที่หมวด KidMotorV3 จะมีบล็อกเพิ่มขึ้นมาแล้ว



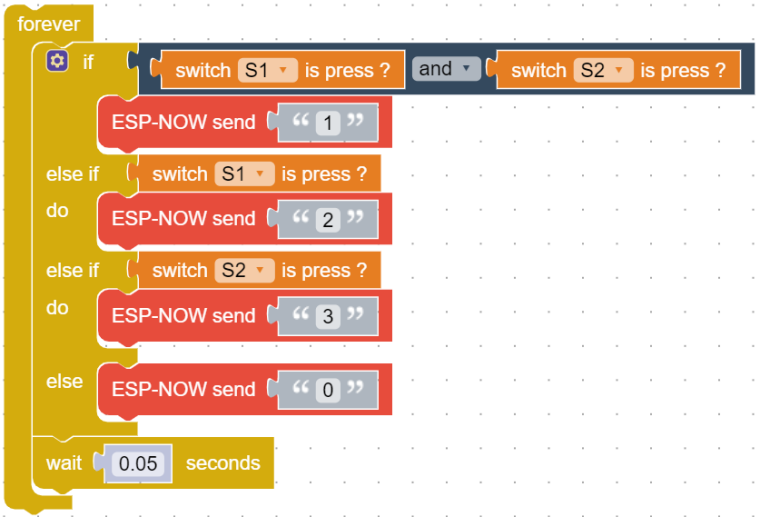
เขียนโปรแกรมที่ฝั่งรถดังนี้



หลักการทำงานของโปรแกรม

- ใช้วนรอบ สั่งให้โปรแกรมต่อไปนี้ทำงานตลอดเวลา
    - ใช้บล็อกเงื่อนไข ตรวจสอบว่าขณะนี้อ่านข้อมูลจาก ESP-NOW ได้หรือไม่ ถ้าได้ ให้ทำโค้ดต่อไปนี้
        - ใช้บล็อก ESP-NOW อ่านตัวเลข อ่านข้อมูลแบบตัวเลขจาก ESP-NOW แล้วเก็บลงตัวแปร x
        - แสดงผลค่าในตัวแปร x บนแอลอีดี 16x8
        - ใช้บล็อกเงื่อนไขตรวจสอบว่าค่าในตัวแปร x คือ 1 ใช่หรือไม่ ถ้าใช่
            - สั่งให้รถวิ่งไปข้างหน้า
        - ถ้าไม่ใช่ ตรวจสอบว่า x คือ 2 ใช่หรือไม่ ถ้าใช่
            - สั่งให้รถเลี้ยวซ้าย
        - ถ้าไม่ใช่ ตรวจสอบว่า x คือ 3 ใช่หรือไม่ ถ้าใช่
            - สั่งให้รถเลี้ยวขวา
        - ถ้าไม่ตรงกับเงื่อนไขใดเลย ให้
            - สั่งให้รถหยุดวิ่ง
    - หน่วงเวลา 0.05 วินาที

ที่ฝั่งรีโมทเขียนโปรแกรมดังนี้



หลักการทำงานของโปรแกรม

- ใช้บล็อกวนรอบ กำหนดโปรแกรมภายใต้บล็อกนี้ทำงานตลอดเวลา
    - ใช้บล็อกเงื่อนไขตรวจสอบว่าสวิตช์ 1 และสวิตช์ 2 กดอยู่หรือไม่ ถ้าใช่
        - ส่งเลข 1 ผ่าน ESP-NOW
    - ใช้บล็อกเงื่อนไขตรวจสอบว่าสวิตช์ 1 กดอยู่หรือไม่ ถ้าใช่
        - ส่งเลข 2 ผ่าน ESP-NOW
    - ใช้บล็อกเงื่อนไขตรวจสอบว่าสวิตช์ 1 กดอยู่หรือไม่ ถ้าใช่
        - ส่งเลข 3 ผ่าน ESP-NOW
    - ถ้าไม่ตรงกับเงื่อนไขใดเลย
        - ส่งเลข 0 ผ่าน ESP-NOW
    - หน่วงเวลา 0.05 วินาที

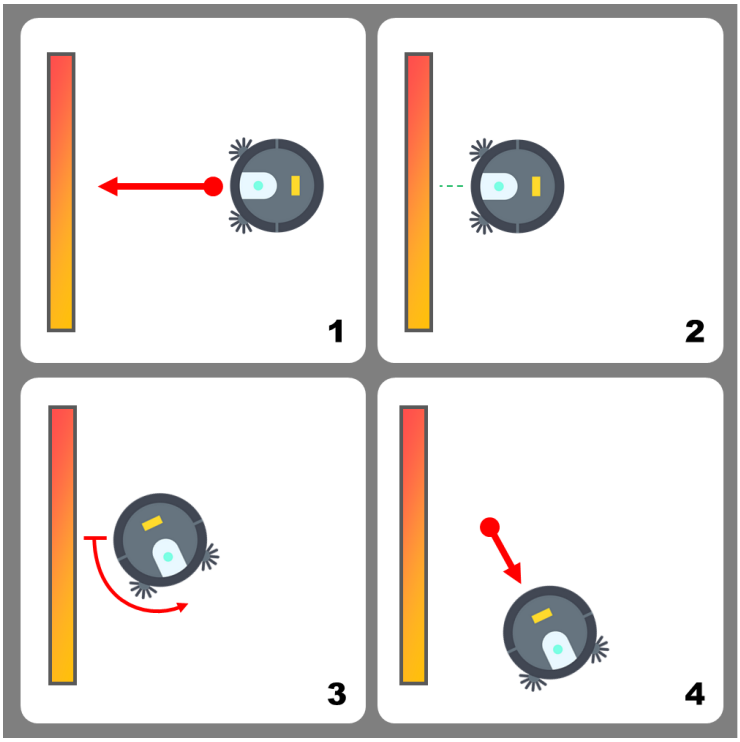
หลังจากอัพโหลดโปรแกรมที่ฝั่งรถและฝั่งรีโมทแล้ว ทดสอบการทำงานดังนี้

- กดปุ่ม S1 และ S2 ฝั่งรีโมทค้างไว้ ที่ฝั่งรถ แอลอีดี 16x8 จะแสดงเลข 1 และรถวิ่งไปข้างหน้า
- กดปุ่ม S1 ฝั่งรีโมทค้างไว้ ที่ฝั่งรถ แอลอีดี 16x8 จะแสดงเลข 2 และรถเลี้ยวซ้าย
- กดปุ่ม S2 ฝั่งรีโมทค้างไว้ ที่ฝั่งรถ แอลอีดี 16x8 จะแสดงเลข 2 และรถเลี้ยวขวา
- ไม่กดปุ่มใดเลย ที่ฝั่งรถ แอลอีดี 16x8 จะแสดงเลข 0 และรถหยุดวิ่ง

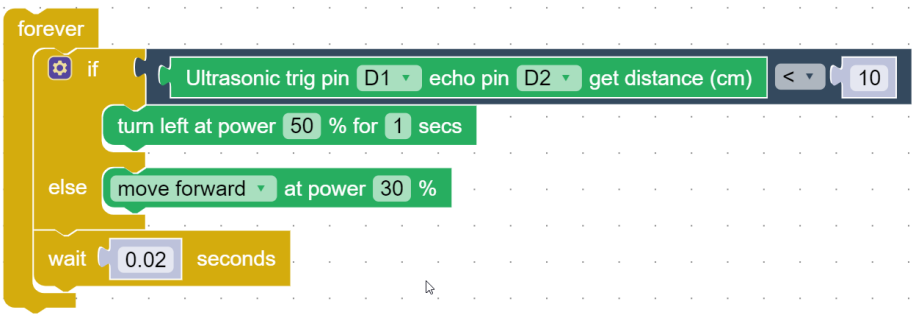
กรณีรถวิ่งทิศทางไม่ถูกต้อง ให้อ่านวิธีแก้ไขได้ในภาคผนวก

4.4.9 โครงงานหุ่นยนต์ดูดฝุ่น

หุ่นยนต์ดูดฝุ่นมีหลักการวิ่งที่สำคัญคือวิ่งไม่ชนขอบพนัง / สิ่งกีดขวาง โดยใช้เซ็นเซอร์ตรวจจับวัตถุ วัดระยะห่างระหว่างหุ่นยนต์กับพนัง / สิ่งกีดขวาง หากมีระยะที่ใกล้เกินไป ตัวหุ่นยนต์จะเลี้ยวไปทิศทางอื่น



จากหลักการวิ่งดังกล่าวสามารถจำลองการทำงาน / สร้างหุ่นยนต์ดูดฝุ่นได้โดยใช้เซ็นเซอร์อัลตร้าโซนิค โดยต่อเซ็นเซอร์อัลตร้าโซนิค HC-SR04p ตามหัวข้อ 2.6 การต่อเซ็นเซอร์อัลตร้าโซนิค แล้วเขียนโปรแกรมดังนี้



หลักการทำงานของโปรแกรม

- ใช้บล็อกวนรอบ กำหนดโปรแกรมภายใต้บล็อกนี้ทำงานตลอดเวลา
  - ใช้บล็อกเมื่อนไขตรวจสอบว่าระยะห่างเซ็นเซอร์กับสิ่งกีดขวาง น้อยกว่ากว่า 10 เซ็นฯ หรือไม่ ถ้าใช่
    - สั่งให้รถเลี้ยวซ้าย ด้วยความเร็ว 50% เป็นเวลา 1 วินาที
  - ถ้าไม่ตรงกับเงื่อนไขใดเลย
    - สั่งให้รถวิ่งตรงต่อไป

- หน่วงเวลา 0.02 วินาที

หลังจากอัพโหลดโปรแกรมที่ฟั่งรถและฟั่งรีโมทแล้ว รถจะเริ่มวิ่งทันที เมื่อนำรถวางลงกับพื้นแล้วเซ็นเซอร์เจอสิ่งกีดขวาง รถจะเลี้ยวซ้ายเพื่อหลบสิ่งกีดขวาง แล้ววิ่งตรงต่อไป

ภาคผนวก

(ก) การแก้ปัญหารถวิ่งผิดทิศทาง

เกิดจากต่อมอเตอร์ไม่ถูกต้อง หรือผิดขั้ว แนะนำให้ดำเนินการตรวจสอบตามขั้นตอนดังนี้ (หากขั้นตอนใดไม่พบปัญหา ให้ข้ามขั้นตอนนั้นไปได้เลย

1. ช่องต่อมอเตอร์ 1 ต้องต่อเข้ากับมอเตอร์ซ้าย
2. ช่องต่อมอเตอร์ 2 ต้องต่อเข้ากับมอเตอร์ขวา
3. ใช้บล็อกเดินหน้า มอเตอร์ซ้าย และขวา ต้องหมุนในทิศทางที่พารถวิ่งไปด้านหน้า หากมอเตอร์ข้างใดหมุนผิดทิศทาง ให้สลับขั้วต่อมอเตอร์ช่องนั้น ๆ
4. ใช้บล็อกเลี้ยวซ้าย ล้อซ้ายจะต้องไม่หมุน ส่วนล้อขวาจะหมุนไปในทิศทางที่พารถวิ่งไปด้านหน้า
   a. ถ้าล้อซ้ายหมุน แต่ล้อขวาไม่หมุน แสดงว่าต่อมอเตอร์สลับช่อง ให้ย้ายสายต่อมอเตอร์ช่อง 1 ไปช่อง 2 และย้ายช่อง 2 ไปช่อง 1 (สลับช่องต่อมอเตอร์)
   b. ถ้าล้อขวาหมุนผิดทิศทาง ให้สลับขั้วต่อมอเตอร์ขวา
5. ใช้บล็อกเลี้ยวขวา ล้อซ้ายจะหมุนไปในทิศทางที่พารถไปข้างหน้า ส่วนล้อขวาจะไม่หมุน
   a. ถ้าล้อซ้ายไม่หมุน แต่ล้อขวาหมุน แสดงว่าต่อมอเตอร์สลับช่อง ให้ย้ายสายต่อมอเตอร์ช่อง 1 ไปช่อง 2 และย้ายช่อง 2 ไปช่อง 1 (สลับช่องต่อมอเตอร์)
   b. ถ้าล้อซ้ายหมุนผิดทิศทาง ให้สลับขั้วต่อมอเตอร์ซ้าย
6. ใช้บล็อกถอยหลัง มอเตอร์ซ้าย และขวา ต้องหมุนในทิศทางที่พารถวิ่งถอยหลัง หากมอเตอร์ข้างใดหมุนผิดทิศทาง ให้สลับขั้วต่อมอเตอร์ช่องนั้น ๆ

บริษัท อาร์ทรอน ชอป จำกัด รับพัฒนาสินค้าอิเล็กทรอนิกส์ บอร์ดอิเล็กทรอนิกส์ ด้านระบบ IoT พัฒนาเว็บไซต์ระบบ IoT ด้วย ReactJS / Next.js รับทำระบบหลังบ้านให้อุปกรณ์ IoT รับเขียนเฟิร์มแวร์อุปกรณ์ IoT ด้วย Arduino IDE / PlatformIO / Atmel Studio / MPLAB IDE / STM32CubeIDE / IAR / ESP-IDF / Arm Keil รับจัดหาชิ้นส่วนอิเล็กทรอนิกส์ ออกใบเสนอราคา และใบกำกับภาษีได้

37/146 ซ.โรงเรียนสวนกุหลาบนนทบุรี ถ.ติวานนท์ ตำบลปากเกร็ด อำเภอปากเกร็ด จังหวัดนนทบุรี 11120

ติดต่อสอบถามข้อมูลเพิ่มเติม

www.artronshop.co.th

02 003 3688 (สำนักงาน)

contact@atronshop.co.th

@artronshop

ArtronShop

ArtronShop