

บทที่ 1

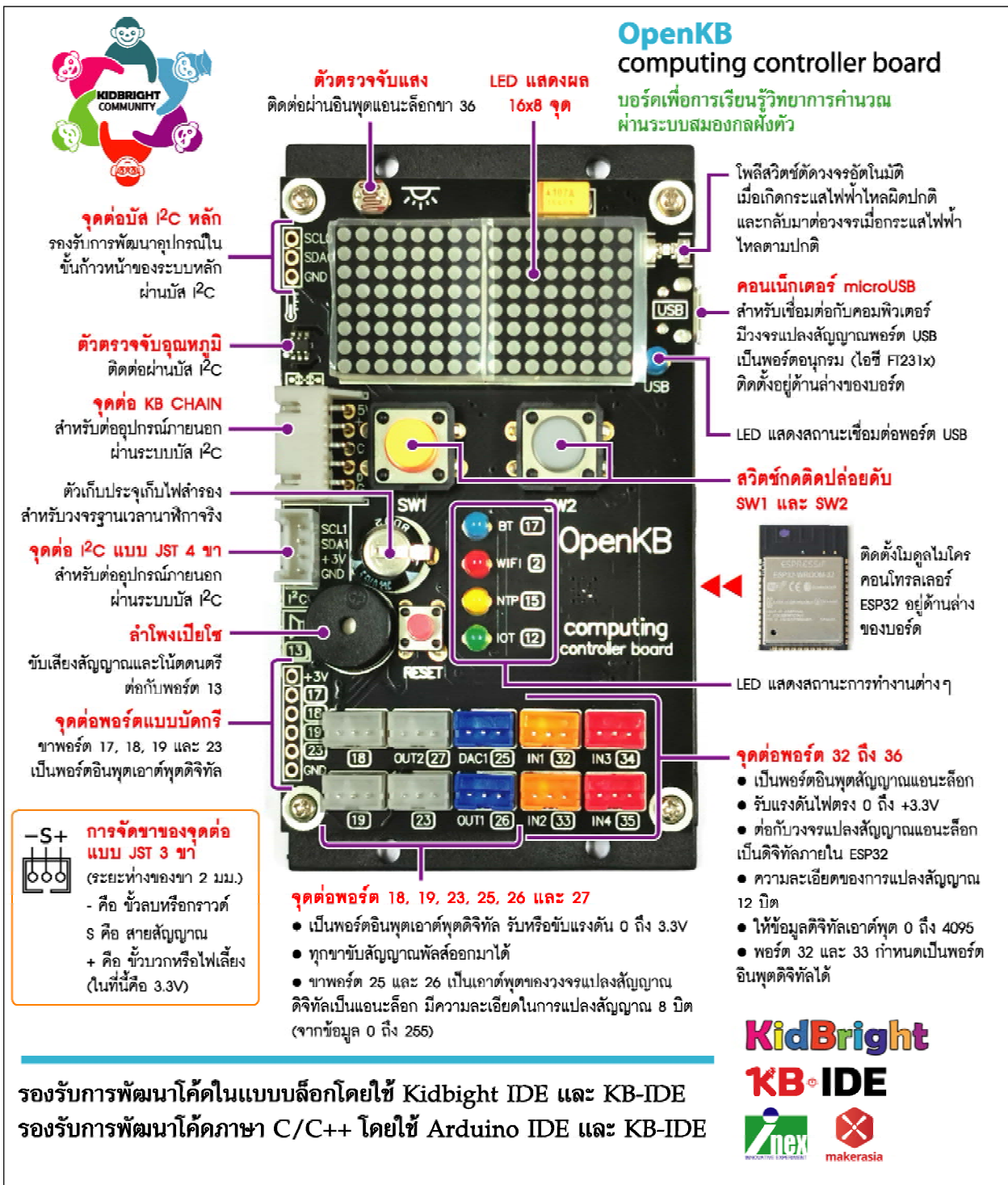
OpenKB แนะนำตัว

OpenKB เป็นบอร์ดไมโครคอนโทรลเลอร์สำหรับการเรียนรู้ด้านวิทยาการคำนวณ ซึ่งได้รับการต่อยอดจากโครงการแพลตฟอร์มเพื่อการเรียนรู้แบบโอเพ่นซอร์สที่ชื่อ KidBright ของสำนักงานวิทยาศาสตร์และเทคโนโลยีแห่งชาติ หรือ สวทช. ประเทศไทย โดย OpenKB สามารถทำงานตามโปรแกรมที่สร้างขึ้นมาจากบล็อกคำสั่งด้วยโปรแกรม KidBright IDE บนคอมพิวเตอร์ และยังรองรับการพัฒนาโค้ดจากเครื่องมือทางซอฟต์แวร์ได้หลากหลายทั้ง KB IDE และ Arduino IDE ที่มีการรองรับไมโครคอนโทรลเลอร์ ESP32

คุณสมบัติทางเทคนิค

- ใช้ไมโครคอนโทรลเลอร์ ESP32 ที่มีวงจร WiFi และบลูทูธกำลังงานต่ำในตัว
- มีส่วนแสดงผล LED คอตเมตริกซ์ขนาด 16 x 8 จุด แบบสีแดง
- มี LED แสดงสถานะการทำงานที่จำเป็นอย่างครบถ้วน ประกอบด้วย
 - สถานะการเชื่อมต่อกับคอมพิวเตอร์ผ่านพอร์ต USB
 - สถานะการเชื่อมต่อ WiFi (ขึ้นกับไลบรารีและบล็อกคำสั่งที่ใช้)
 - สถานะการเชื่อมต่อบลูทูธ (ขึ้นกับไลบรารีและบล็อกคำสั่งที่ใช้)
 - สถานะการตั้งเวลามาตรฐาน NTP (ขึ้นกับไลบรารีและบล็อกคำสั่งที่ใช้)
 - สถานะการเชื่อมต่อกับคลาวด์เซิร์ฟเวอร์ (ขึ้นกับไลบรารีและบล็อกคำสั่งที่ใช้)
- มีลำโพง piezo ขั้วเสียง
- มีวงจรสวิตช์กดติดปล่อยค้ำขนาดใหญ่ 2 ตัว
- มีวงจรฐานเวลาไฟฟ้าจริงพร้อมตัวเก็บประจุความจุสูงพิเศษหรือซูเปอร์คาปาซิเตอร์ เพื่อทำหน้าที่เป็นแหล่งจ่ายไฟสำรองสำหรับรักษาค่าเวลาเมื่อไม่มีไฟเลี้ยง สามารถรักษาค่าเวลาได้สูงสุด 90 ชั่วโมง (ประมาณ 4 วัน)
- มีสวิตช์ RESET การทำงาน

2 • OpenKB บอร์ดเรียนรู้อิเล็กทรอนิกส์คำนวณผ่านระบบสมองกลฝังตัว



รองรับการพัฒนาโค้ดในแบบบล็อกโดยใช้ Kidbright IDE และ KB-IDE
รองรับการพัฒนาโค้ดภาษา C/C++ โดยใช้ Arduino IDE และ KB-IDE



รูปที่ 1-1 แสดงส่วนประกอบของ OpenKB บอร์ดเพื่อการเรียนรู้วิทยาการคำนวณผ่านระบบสมองกลฝังตัว

- เชื่อมต่อกับคอมพิวเตอร์ผ่านพอร์ต USB โดยใช้คอนเน็กเตอร์แบบ microUSB สำหรับการดาวน์โหลดโปรแกรมและสื่อสารข้อมูลอนุกรม (โดยความสามารถในการสื่อสารข้อมูลขึ้นกับ IDE ที่เลือกใช้) และยังใช้ในการรับไฟเลี้ยง +5V ผ่านพอร์ต microUSB ด้วย

- มีจุดต่อพอร์ตที่ใช้คอนเน็กเตอร์ JST 2 มม. 3 ขา (JST : Japan Standard Terminal) รวม 10 ขา ประกอบด้วย

- พอร์ตอินพุตเอาต์พุตดิจิทัล ประกอบด้วย ขา GPIO18, 19, 23 และ 27 หรือ OUT2 (ตามการกำหนดขาของ KidBright) ทั้งยังสามารถขับสัญญาณ PWM และขับเซอร์โวมอเตอร์ได้ทุกขา โดยต่อวงจรและไฟเลี้ยงภายนอกเพื่อเลี้ยงเซอร์โวมอเตอร์

- พอร์ตอินพุตเอาต์พุตดิจิทัลและเป็นเอาต์พุตสัญญาณแอนะล็อกและสร้างสัญญาณ PWM ได้ด้วย ประกอบด้วยขา GPIO25 หรือ DAC1 และ 26 หรือ DAC2 หรือ OUT1 (ตามการกำหนดขาของ KidBright)

- พอร์ตอินพุตดิจิทัลคอลและแอนะล็อก ประกอบด้วย ขา GPI32 ถึง 35 หรือ IN1 ถึง IN4 (ตามการกำหนดขาของ KidBright)

- มีจุดต่อพอร์ตแบบจุกบัดกรีอิสระ 4 ขาคือ ขาพอร์ต 17, 18, 19 และ 23

- มีจุดต่อบัส I²C1 สำหรับต่ออุปกรณ์ภายนอกเพิ่มเติมเพื่อขยายระบบทั้งแบบ KB CHAIN 5 ขา และ แบบ JST 2 มม. 4 ขา (JST : Japan Standard Terminal)

- ติดตั้งตัวตรวจจับแสงแบบ LDR หรือตัวต้านทานแปรค่าตามแสง

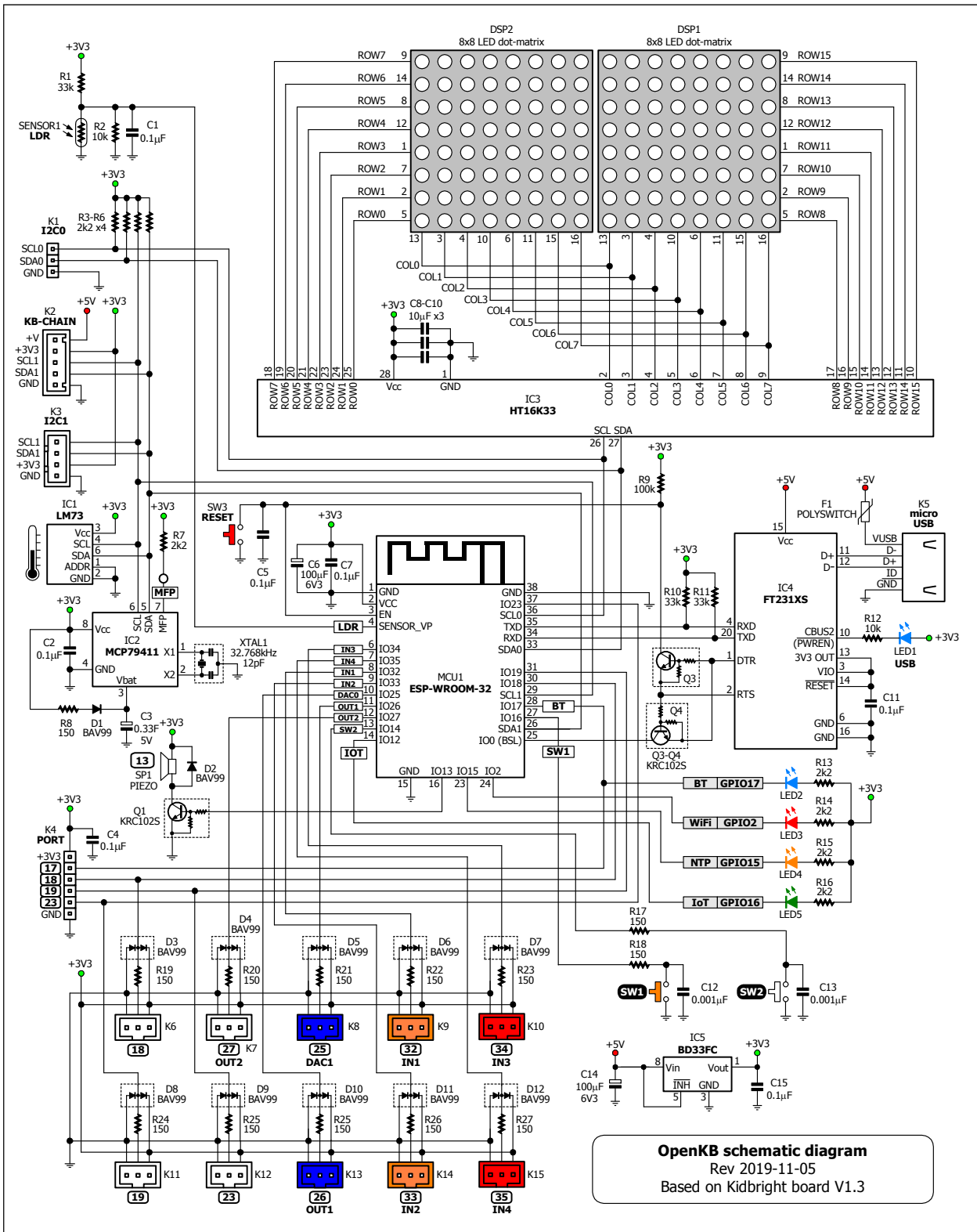
- ติดตั้งตัวตรวจจับอุณหภูมิที่ทำงานผ่านระบบบัส I²C วัดอุณหภูมิได้ -40 ถึง 150 องศาเซลเซียส มีค่าความผิดพลาด ± 1 องศาเซลเซียส

- รับไฟเลี้ยง +5V จากจุดต่อพอร์ต microUSB ผ่านวงจรควบคุมแรงดันคงที่หรือเรกูเลเตอร์ ให้ได้ไฟเลี้ยงคงที่ที่ +3.3V สำหรับเลี้ยงวงจร

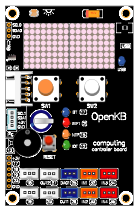
วงจรของบอร์ด OpenKB แสดงดังรูปที่ 1-2 เนื่องจาก OpenKB เป็นบอร์ดที่พัฒนาต่อยอดจาก KidBright ดังนั้นข้อมูลพื้นฐานทั้งทางฮาร์ดแวร์และซอฟต์แวร์จะเข้ากันได้กับบอร์ด KidBright จึงใช้หนังสือหรือคู่มือเบื้องต้นที่ สวทช. และ สสวท. จัดทำขึ้นมาร่วมด้วยได้

ดังนั้นการอธิบายข้อมูลและแนะนำการใช้งานที่มีในเอกสารนี้จะกล่าวถึงเนื้อหาของส่วนที่เพิ่มเติมไปจากบอร์ด KidBright เป็นหลัก

4 • OpenKB บอร์ดเรียนรู้วิทยาการคำนวณผ่านระบบสมองกลฝังตัว



รูปที่ 1-2 วงจรสมบูรณของ OpenKB บอร์ดเพื่อการเรียนรูวิทยาการคำนวณผ่านระบบสมองกลฝังตัว

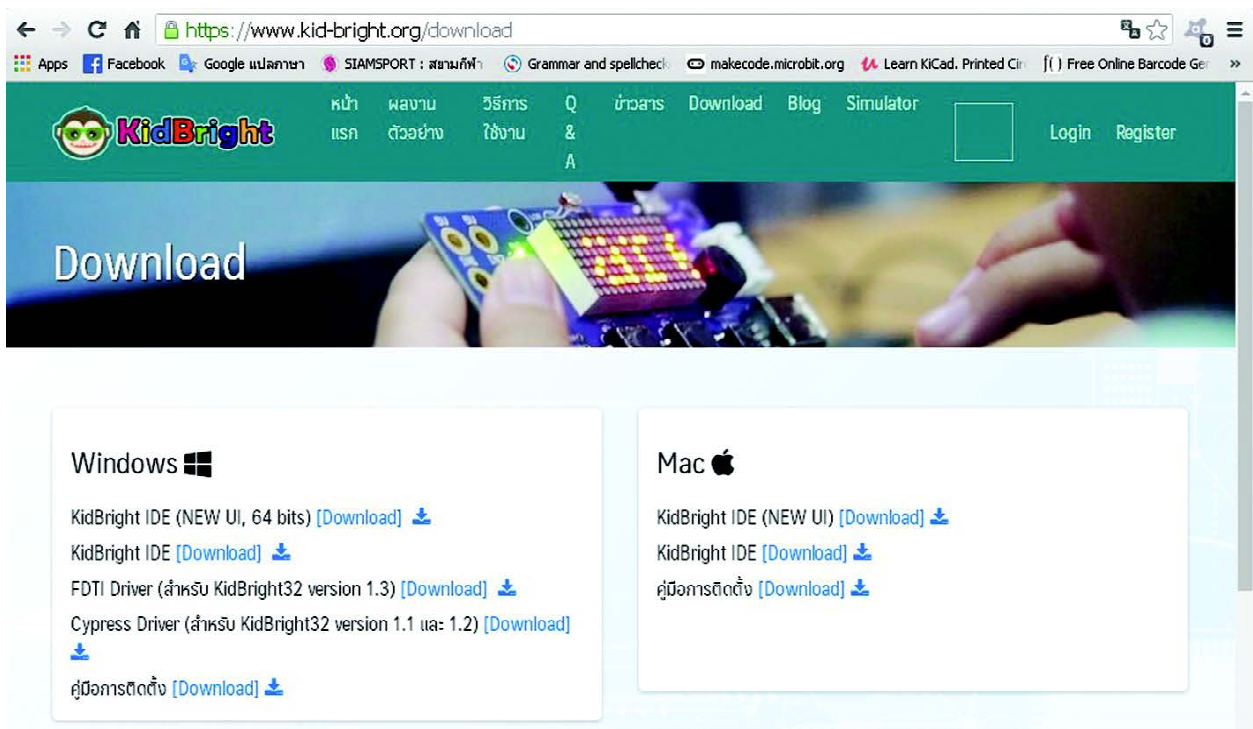


บทที่ 2

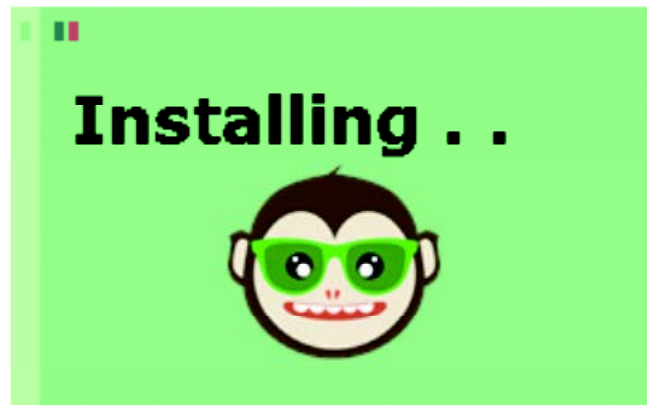
การพัฒนาโค้ดของ OpenKB ด้วย KidBright IDE

2.1 ติดตั้งโปรแกรม KidBright IDE

หัวข้อนี้เป็นการอธิบายขั้นตอนเริ่มต้นของการพัฒนาโค้ดและใช้งานบอร์ด OpenKB กับโปรแกรม KidBright IDE ซึ่งเป็นเครื่องมือหลักที่พัฒนาโดยคณะนักวิจัยของ สวทช. โปรแกรม KidBright IDE ที่จะใช้งานกับบอร์ด OpenKB คือ KidBright IDE เวอร์ชัน New UI (เลือกรุ่น 32 หรือ 64 บิตก็ได้) โดยดาวน์โหลดได้ที่ <https://www.kid-bright.org/download> ตามรูปที่ 2-1 จะได้ไฟล์ KidBright-Installer.zip



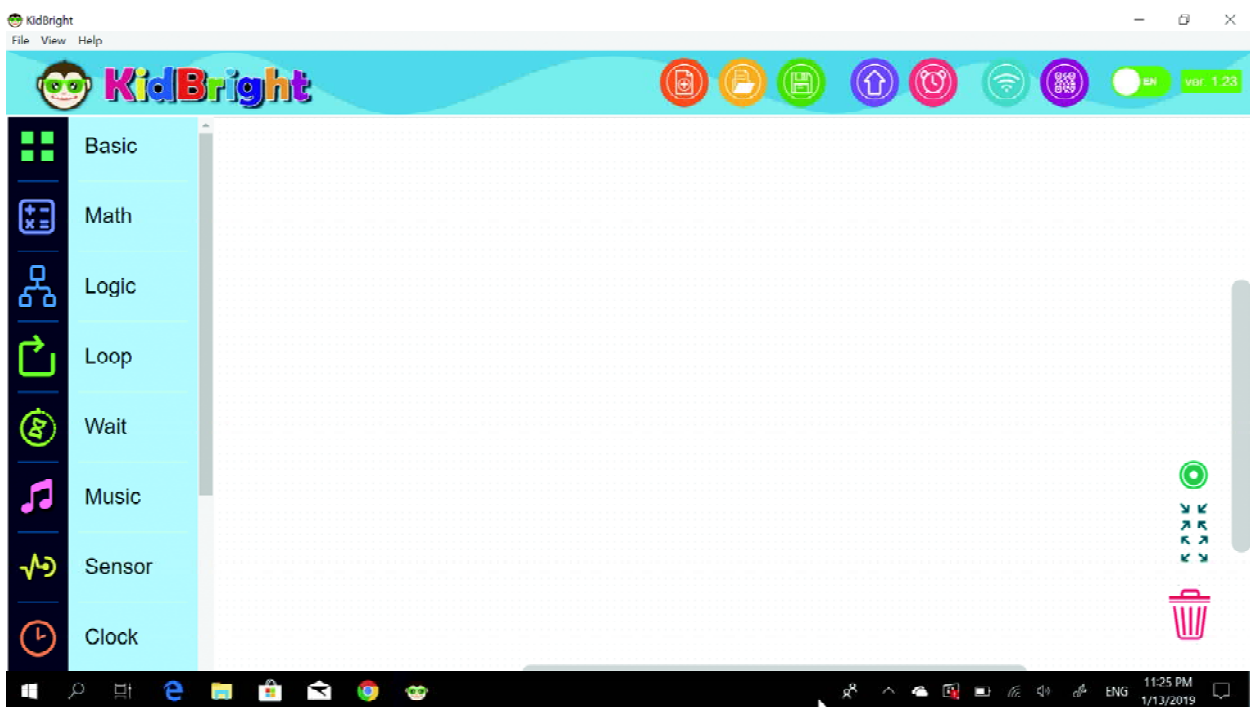
รูปที่ 2-1 แสดงเว็บเพจสำหรับดาวน์โหลดโปรแกรม KidBright IDE



รูปที่ 2-2 หน้าต่างแสดงการติดตั้ง KidBright IDE

จากนั้นทำการแตกไฟล์ จะได้ไฟล์ KidBright-Installer.exe ดับเบิลคลิกไฟล์นี้เพื่อเริ่มต้นการติดตั้งโปรแกรม จะปรากฏหน้าต่าง Installing... ตามรูปที่ 2-2 รอจนกระทั่งการติดตั้งเสร็จสมบูรณ์ ในขั้นตอนนี้อาจใช้เวลาานพอสมควร (มากกว่า 2 นาที)

เมื่อติดตั้งโปรแกรมเสร็จ หน้าต่างหลักของโปรแกรม KidBright IDE จะปรากฏขึ้นมา ดังรูปที่ 2-3 เป็นการยืนยันว่า ขณะนี้การติดตั้งโปรแกรม KidBright IDE เรียบร้อย และพร้อมสำหรับการพัฒนาโค้ด



รูปที่ 2-3 หน้าต่างหลักของโปรแกรม KidBright IDE

2.2 ติดตั้งไดรเวอร์ USB ของบอร์ด OpenKB

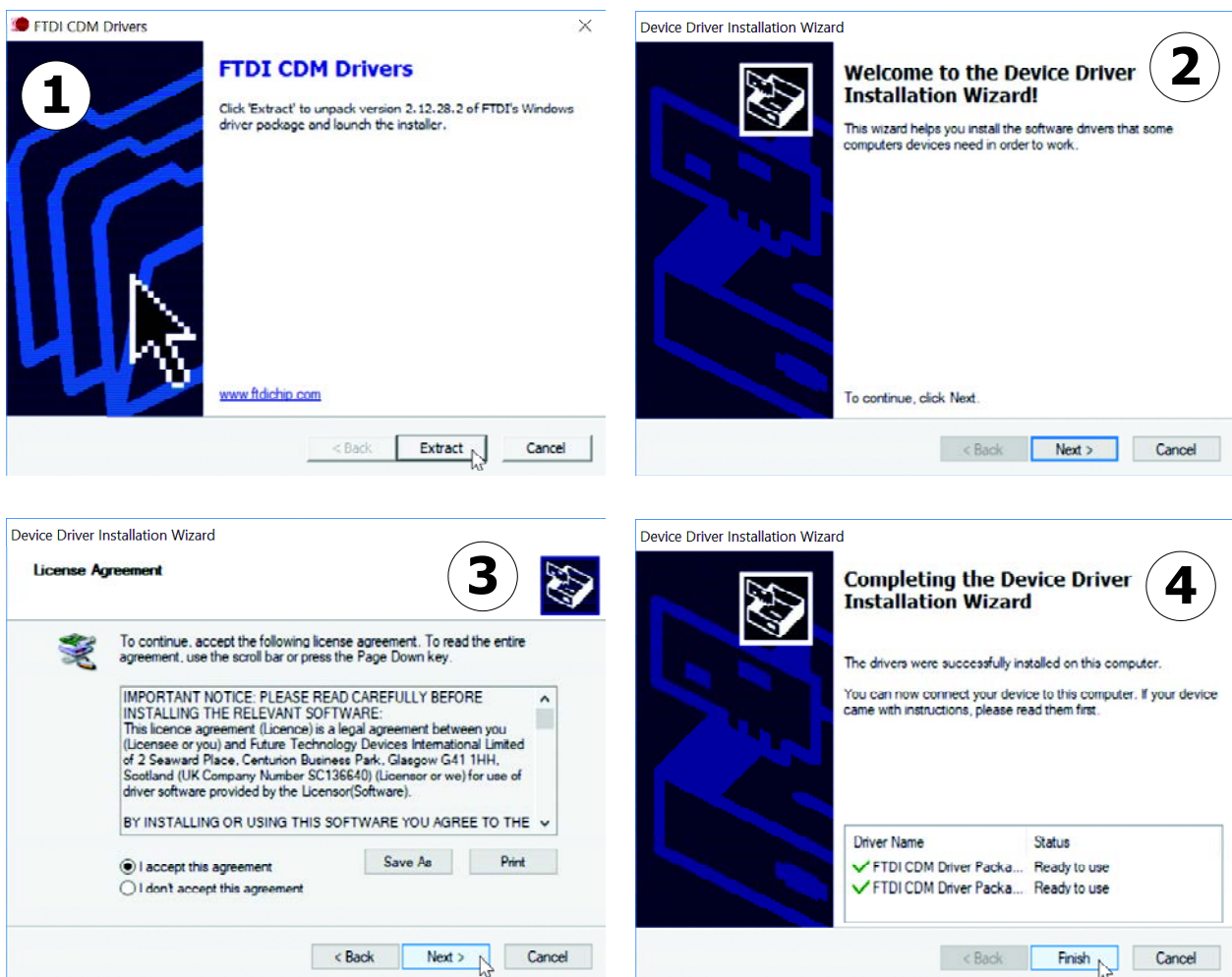
บอร์ด OpenKB ต้องการไดรเวอร์ USB แบบ FTDI มีขั้นตอนดังนี้

(1) ดาวน์โหลด FTDI Driver จาก

https://www.kid-bright.org/file/download/CDM 21228_Setup.zip

(2) ทำการแตกไฟล์ .zip จะได้ไฟล์ CDM21228_Setup.exe

(3) ดับเบิลคลิกเพื่อเริ่มการติดตั้งไดรเวอร์ แล้วคลิกตอบรับจนกระทั่งการติดตั้งเสร็จสิ้น
ตามรูปที่ 2-4



รูปที่ 2-4 ขั้นตอนการติดตั้งไดรเวอร์ USB ของบอร์ด OpenKB

2.3 ติดตั้งบล็อกส่วนขยายสำหรับใช้งานกับบอร์ด OpenKB

บอร์ด OpenKB เป็นบอร์ดที่ทำงานได้ตรงกับ KidBright32 ดังนั้นบล็อกคำสั่งมาตรฐาน จะใช้งานได้กับบอร์ด OpenKB ได้ด้วย (ยกเว้นบล็อกคำสั่งในหมวด IoT ซึ่งจะใช้งานได้หลังจากทำการลงทะเบียนกับ NETPIE ซึ่งทางผู้ผลิตจะดำเนินการให้เรียบร้อยแล้วในปี พ.ศ. 2562) เนื่องจากบอร์ด OpenKB มีความสามารถทางฮาร์ดแวร์ที่เพิ่มเติมจาก KidBright32 จึงมีบล็อกคำสั่งเพิ่ม จะต้องทำการติดตั้งบล็อกคำสั่งส่วนขยายหรือปลั๊กอิน (plug-in) สำหรับ OpenKB เพิ่ม โดยต้องทำการเตรียมการในขั้นต้นดังนี้

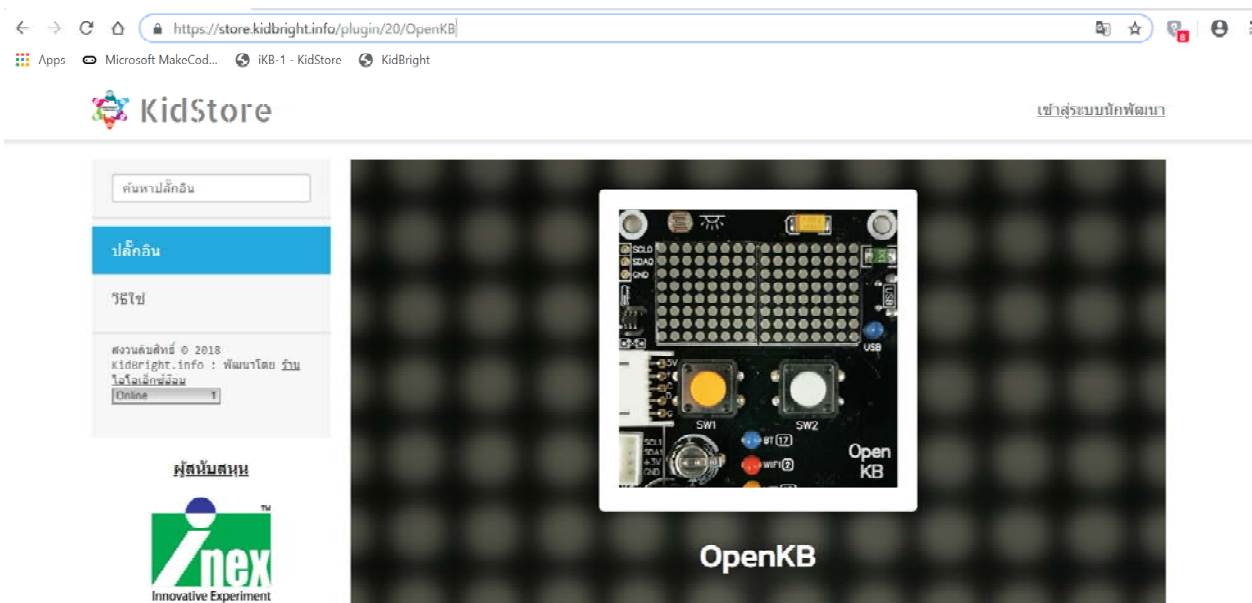
(1) ดาวน์โหลดไฟล์ติดตั้งบล็อกส่วนขยาย OpenKB จากเว็บไซต์ที่รวบรวมบล็อกส่วนขยายของ KidBright ที่ชื่อ **KidStore** จาก URL ต่อไปนี้ <https://store.kidbright.info> จากนั้นเลือก **OpenKB** ดังรูปที่ 2-5

(2) เลื่อนหน้าเว็บจนพบบรรทัดรายการดาวน์โหลด หรือดาวน์โหลดไฟล์จาก

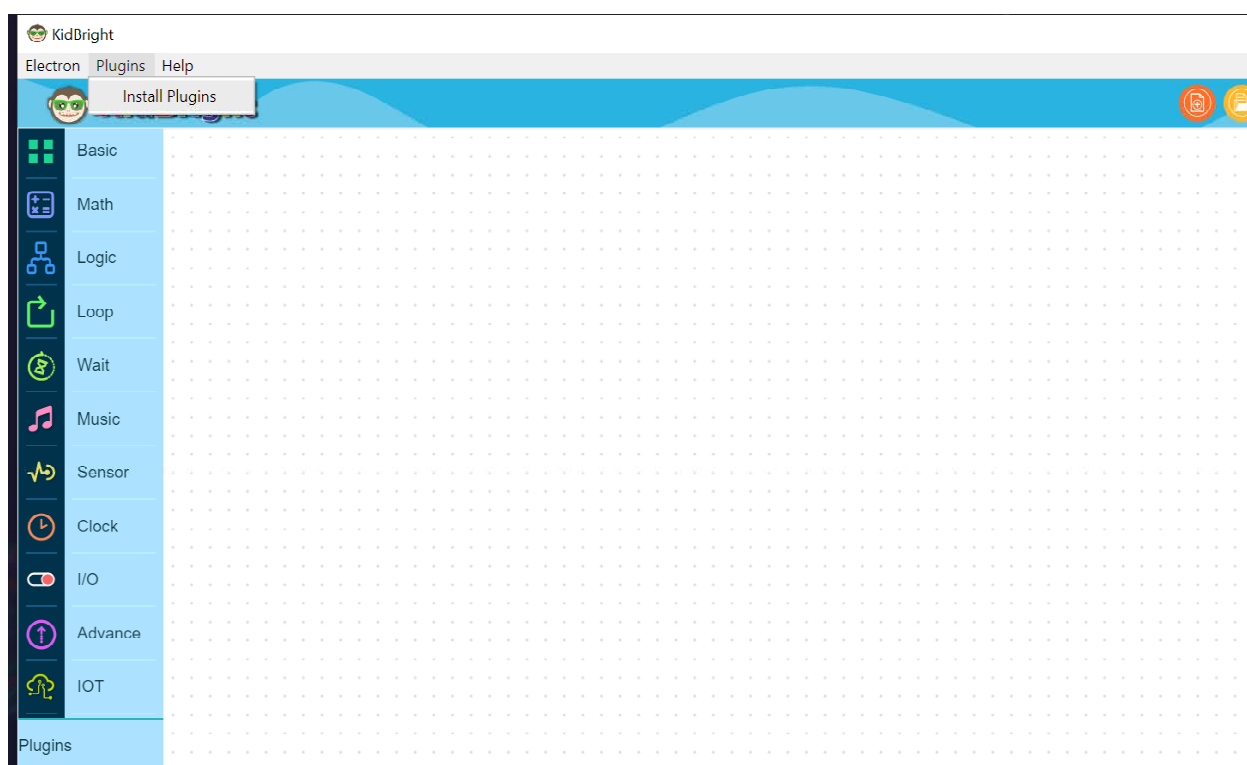
https://store.kidbright.info/download/51/openkb_v1.0.1.zip

(3) จากนั้นเตรียมการติดตั้งปลั๊กอิน ซึ่งทำได้ 2 วิธีคือ

1. ติดตั้งผ่านทางเมนู Plug-in ของ KidBright IDE
2. ติดตั้งด้วย Gitlab



รูปที่ 2-5 เว็บเพจของ KidStore ในหน้าของปลั๊กอิน OpenKB



รูปที่ 2-6 เริ่มต้นการติดตั้งบล็อกส่วนขยาย OpenKB

2.3.1 กรณีติดตั้งผ่านเมนูของโปรแกรม KidBrightIDE

- (1) เปิดโปรแกรม KidBrightIDE ไปที่เมนูคำสั่ง เลือก **Plugins > Install Plugins** ดังรูปที่ 2-6
- (2) เลือกไฟล์ปลั๊กอิน (openkb_vx.x.x.zip) ที่ดาวน์โหลดไว้ ดังรูปที่ 2-7 เมื่อเลือกแล้ว โปรแกรม KidBrightIDE จะปิดแล้วเปิดใหม่ บล็อกคำสั่ง **OpenKB** จะอยู่ในหัวข้อ **Plugins** เมื่อเลือกเปิดรายการ OpenKB จะพบบล็อกคำสั่งหลักตามรูปที่ 2-8

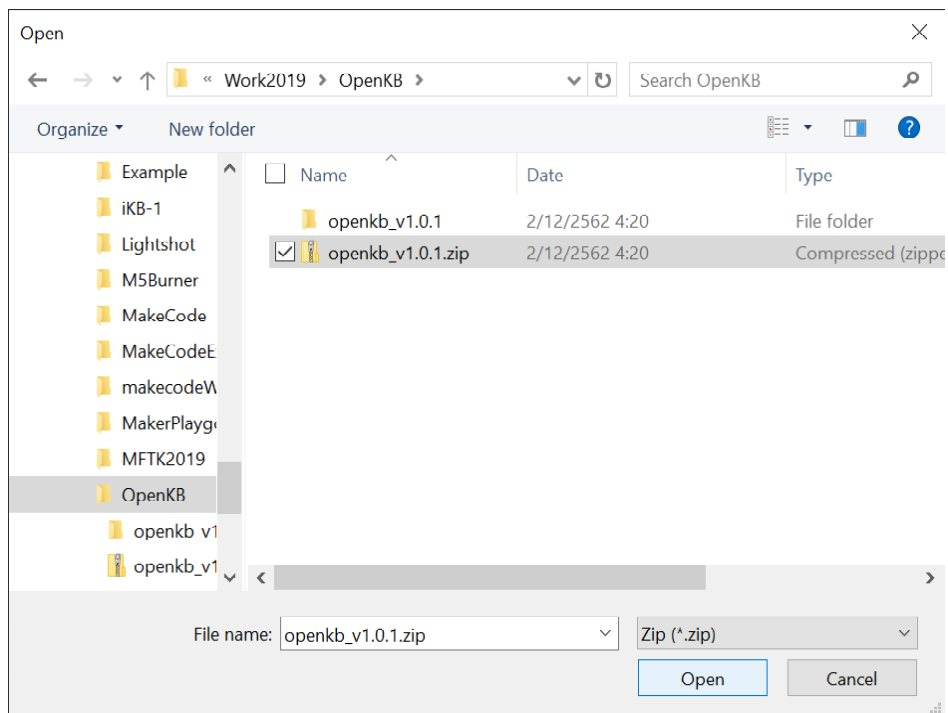
2.3.2 กรณีติดตั้งไปยังโฟลเดอร์ของโปรแกรม KidBrightIDE

- (1) ปิดโปรแกรม KidBright IDE
- (2) คลายไฟล์ **openkb_vx.x.x.zip** ออกมา จะได้โฟลเดอร์ **openkb**
- (3) ย้ายโฟลเดอร์ **openkb** ไปไว้ในโฟลเดอร์ **plugins** ซึ่งจะอยู่ในโฟลเดอร์ **kbide** ตามตำแหน่งดังนี้

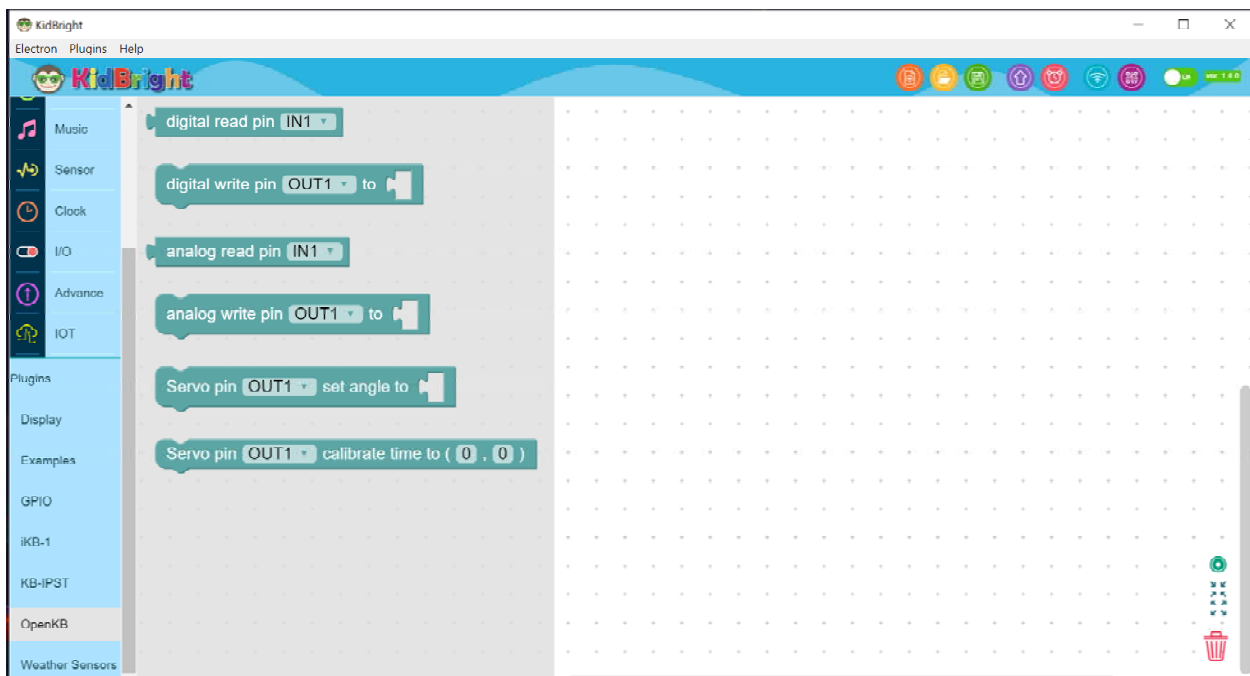
C:\Users\xxx\AppData\Local\KidBright\app-1.3.2\resources\app\kbide

โดยที่ xxx คือชื่อของคอมพิวเตอร์ที่ตั้งไว้

- (4) เปิดโปรแกรม KidBrightIDE ขึ้นมา จะมีรายการ **OpenKB** ในหัวข้อ **Plugins** แสดงขึ้นมา



รูปที่ 2-7 เลือกไฟล์ .zip ของปลั๊กอินที่ทำการดาวน์โหลดมา



รูปที่ 2-8 แสดงการเปิดเมนูเพื่อเลือกใช้งานบล็อกส่วนขยาย OpenKB

2.4 คำอธิบายบล็อกคำสั่งของบล็อกส่วนขยาย OpenKB

2.4.1 บล็อกอ่านค่าจากพอร์ตอินพุตดิจิทัล

บล็อกภาษาไทย



บล็อกภาษาอังกฤษ



ใช้อ่านค่าดิจิทัลจากช่อง IN1, IN2, DAC1, 18, 19, 23 และ 17 โดยให้ค่าเป็นตัวเลข 0 หรือ 1

2.4.2 บล็อกเขียนค่าไปยังพอร์ตเอาต์พุตดิจิทัล

บล็อกภาษาไทย



บล็อกภาษาอังกฤษ



ใช้เขียนค่าดิจิทัลไปที่ช่อง OUT1, OUT2, DAC1, 18, 19, 23 และ 17 โดยรับค่าเป็นตัวเลขจำนวนเต็ม 0 หรือ 1

2.4.3 บล็อกอ่านค่าแอนะล็อก

บล็อกภาษาไทย



บล็อกภาษาอังกฤษ



ใช้อ่านค่าแอนะล็อกจากช่อง IN1, IN2, IN3 และ IN4 หรือขา 32 ถึง 35 โดยให้ค่าเป็นตัวเลข 0 ถึง 4095 (12 บิต)

2.4.4 บล็อกเขียนค่าแอนะล็อก

บล็อกภาษาไทย



บล็อกภาษาอังกฤษ



ใช้สร้างสัญญาณ PWM ที่ช่อง OUT1, OUT2, DAC1, 18, 19, 23 และ 17 โดยบล็อกจะรับค่าเป็นตัวเลขจำนวนเต็ม 0 ถึง 1023

2.4.5 บล็อกควบคุมเซอร์โวมอเตอร์

บล็อกภาษาไทย



บล็อกภาษาอังกฤษ



ใช้ควบคุมองศาการหมุนของเซอร์โวมอเตอร์ที่ช่อง OUT1, OUT2, DAC1, 18, 19, 23 และ 17 บล็อกจะรับค่าเป็นตัวเลขจำนวนเต็ม 0 ถึง 179 (หน่วยองศา) โดยควรต่อใช้งานร่วมกับบอร์ด JCON4 เพื่อปรับชุดต่อให้เป็นขาพอร์ตแบบ IDC 3 ขาสำหรับนำเซอร์โวมอเตอร์มาต่อใช้งาน ดังแสดงข้อมูลในรูปที่ 2-9

JCON-SERVO4

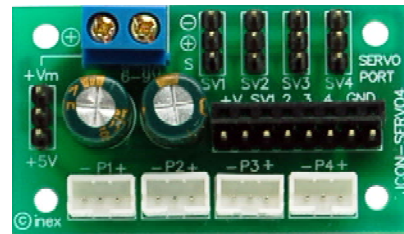
บอร์ดเชื่อมต่อเซอร์โวมอเตอร์

● ใช้เปลี่ยนการต่อสัญญาณจากจุดต่อแบบ JST 2 มิลลิเมตร 3 ขา และ IDC 2.54 มิลลิเมตร เป็นจุดต่อสำหรับเซอร์โวมอเตอร์

● รองรับได้ 4 ช่องต่อบอร์ด หากต่อพ่วงไฟเลี้ยงกัน จะสามารถขยายจำนวนการเชื่อมต่อเซอร์โวมอเตอร์ได้อย่างไม่จำกัด แต่แบตเตอรี่ต้องมีความสามารถในการจ่ายกระแสไฟฟ้ามกเพียงพอ

● มีจุดต่อ +Vm สำหรับต่อไฟเลี้ยง +4.8 ถึง +7.2V สำหรับเซอร์โวมอเตอร์ขนาดเล็ก

● มีจัมเปอร์เลือกระดับแรงดัน +Vm เป็น +5V จากบอร์ดไมโครคอนโทรลเลอร์ที่นำมาต่อใช้งานด้วย เพื่อรองรับการเชื่อมต่อกับอุปกรณ์ของ Parallax (www.parallax.com) ที่ใช้การเชื่อมต่อในแบบเดียวกับสายสัญญาณของเซอร์โวมอเตอร์ แต่ใช้ไฟเลี้ยง +5V

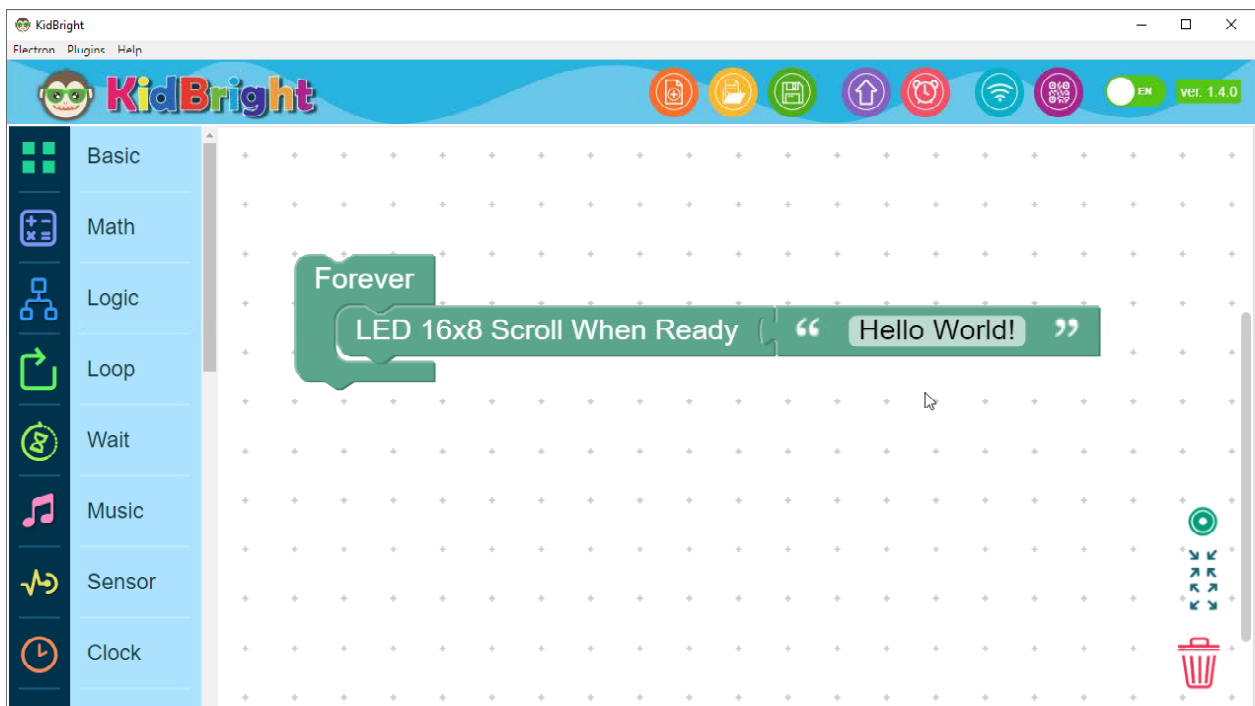


รูปที่ 2-9 ข้อมูลทางเทคนิคเบื้องต้นของ JCON-SERVO4 มินิบอร์ดเชื่อมต่อเซอร์โวมอเตอร์ 4 ช่อง

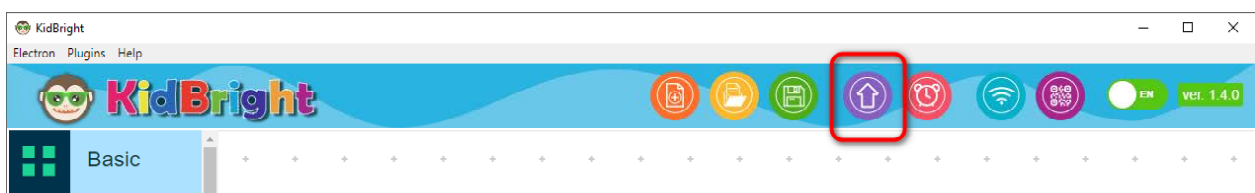
2.5. เริ่มต้นการพัฒนาโค้ดสำหรับบอร์ด OpenKB ด้วย KidBright IDE

เมื่อติดตั้งโปรแกรมและไดรเวอร์ต่างๆ จนเสร็จเรียบร้อยแล้ว ในการเริ่มต้นทดสอบการเขียนโค้ด และดาวน์โหลดเพื่อทดสอบการทำงานกับบอร์ด OpenKB มีขั้นตอนดังนี้

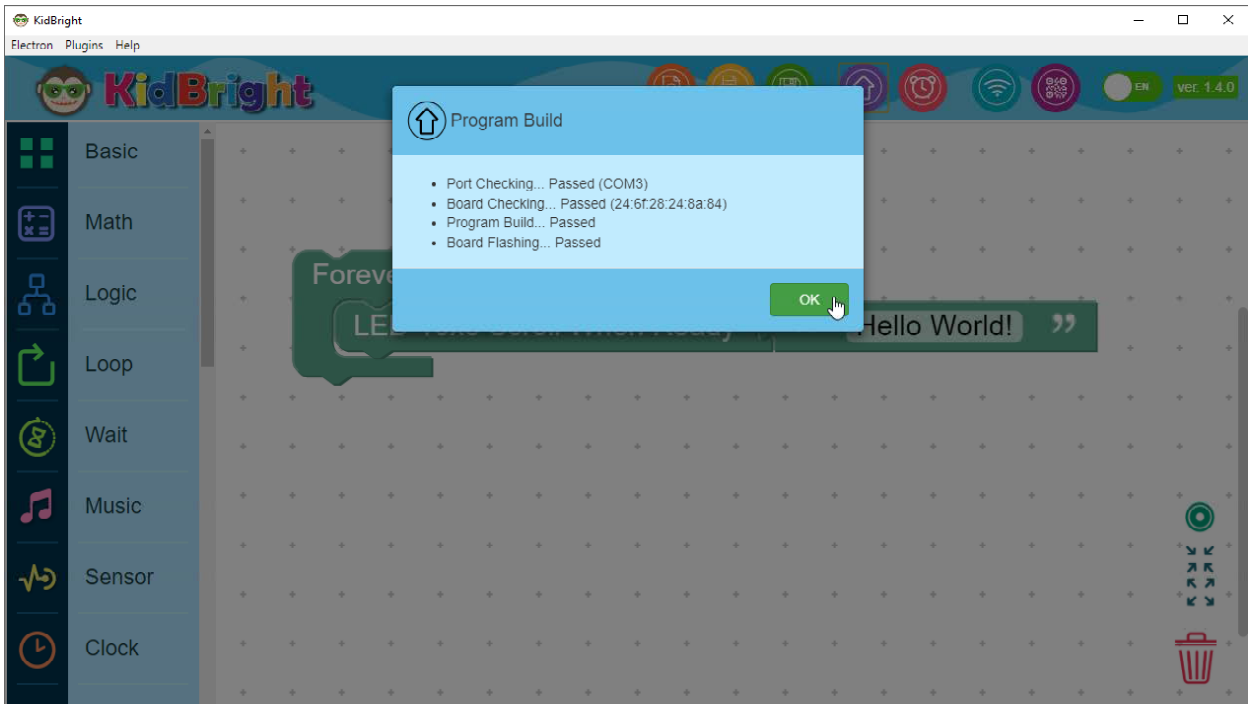
- (1) เชื่อมต่อบอร์ด OpenKB กับคอมพิวเตอร์ผ่านทางพอร์ต USB
- (2) รอสักครู่หนึ่งเพื่อให้คอมพิวเตอร์ทำการเชื่อมต่อกับอุปกรณ์ผ่านพอร์ต USB สำเร็จหรือเรียกว่า การอินั้มเมอเรชั่น (enumeration) **ซึ่งสังเกตได้จาก LED สีฟ้าที่อยู่ทางด้านขวาของบอร์ด OpenKB ติดสว่าง**
- (3) เปิดโปรแกรม KidBright IDE แล้วสร้างโค้ดทดสอบ ดังนี้



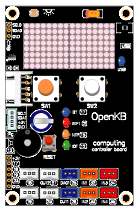
- (4) ทำการอัปโหลดไปยังบอร์ด OpenKB โดยคลิกที่ปุ่ม **Build & Run** ตามรูป



(5) จากนั้นหน้าต่างแสดงสถานะการคอมไพล์และอัปโหลดปรากฏขึ้นมาตามรูป คลิกปุ่ม OK เพื่อตอบรับ



เมื่ออัปโหลดโค้ดเสร็จ บอร์ด OpenKB จะทำงานทันที ที่จอแสดงผล LED ของบอร์ด OpenKB แสดงข้อความ Hello World! ในลักษณะข้อความเลื่อนจากขวาซ้าย เมื่อได้ดังนี้แสดงว่า บอร์ด OpenKB พร้อมทำงาน



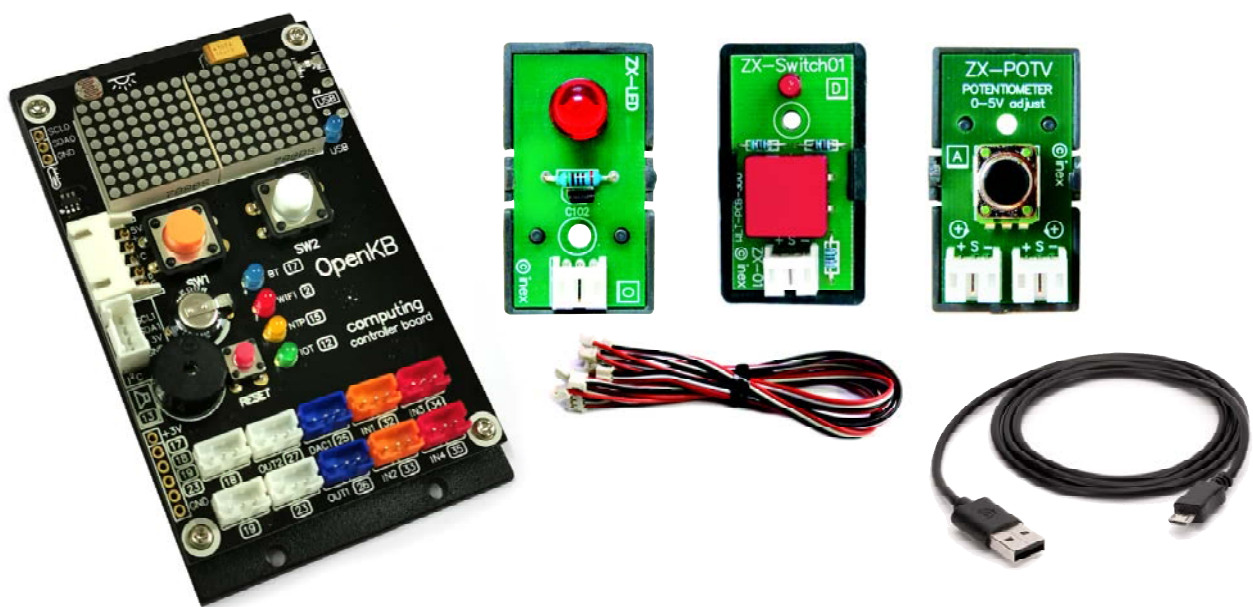
บทที่ 3

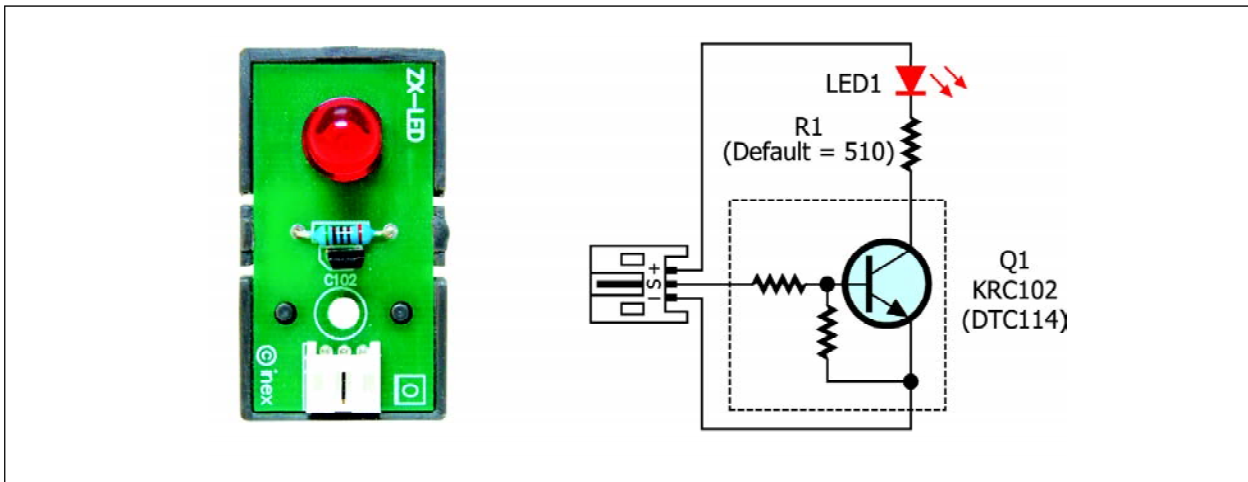
ตัวอย่างการพัฒนาโค้ดเพื่อใช้งาน OpenKB ด้วย KidBright IDE

3.1 แนะนำชุดอุปกรณ์ OpenKB Starter Kit

เพื่อให้การเริ่มต้นใช้งานบอร์ด OpenKB เป็นไปอย่างมีประสิทธิภาพ จึงมีการจัดชุดของอุปกรณ์ต่อพ่วงเพื่อใช้ในการทดสอบและทดลองใช้งาน OpenKB ทั้งในแบบทำงานด้วยตัวเองและเชื่อมต่อกับอุปกรณ์ภายนอก ชุดอุปกรณ์นี้ชื่อว่า OpenKB Starter Kit ประกอบด้วย

1. บอร์ด OpenKB
2. สาย microUSB
3. ZX-LED มินิบอร์ดขับ LED
4. ZX-SWITCH01 มินิบอร์ดสวิตช์กดติดปลายนิ้ว
5. ZX-POT มินิบอร์ดวงจรตัวต้านทานปรับค่าได้
6. สายเชื่อมต่อกับมินิบอร์ดกับ OpenKB 3 เส้น





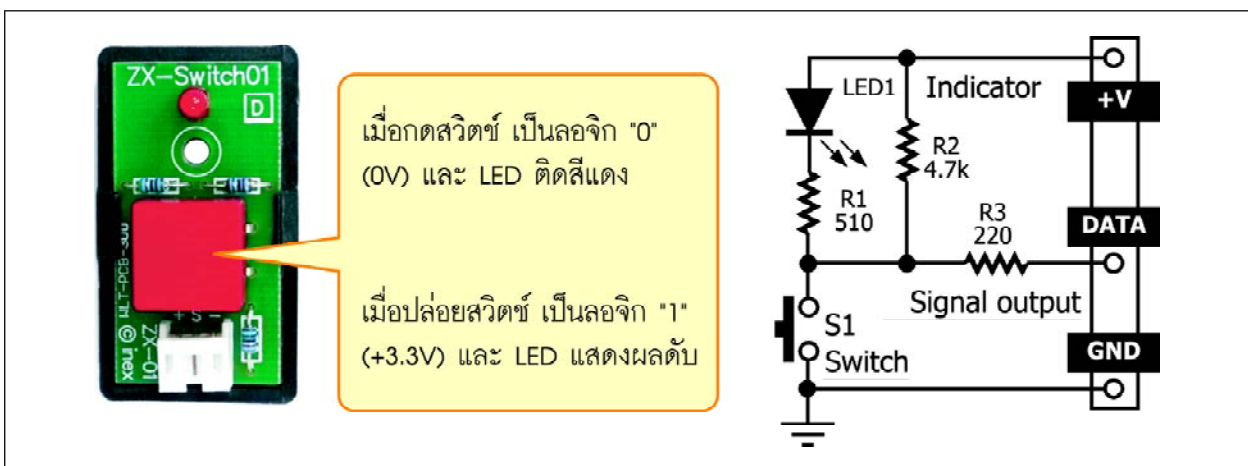
รูปที่ 3-1 รูปร่างและวงจรของมินิบอร์ด ZX-LED

3.1.1 ZX-LED มินิบอร์ดวงจรขับ LED 8 มม.

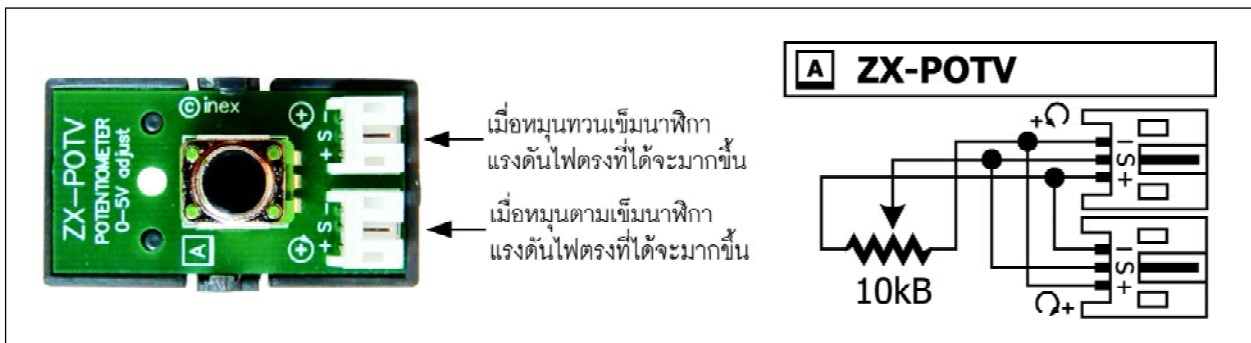
มีหน้าตาและวงจรแสดงในรูปที่ 3-1 เป็นบอร์ดวงจรขับ LED ด้วยทรานซิสเตอร์ ใช้ LED ขนาด 8 มิลลิเมตร ต้องการลอจิก “1” ในการขับให้สว่าง รองรับการใช้งานกับสายสัญญาณแบบ JST3AA

3.1.2 ZX-SWITCH01 มินิบอร์ดวงจรสวิตช์กดติดปล่อยดับ

มีวงจรและรูปร่างแสดงในรูปที่ 3-2 ประกอบด้วยสวิตช์พร้อม LED แสดงผล ถ้ากดสวิตช์จะส่งลอจิก “0” ออกไปยังเอาต์พุต และ LED1 สีแดงติดสว่าง จนกว่าจะปล่อยสวิตช์



รูปที่ 3-2 รูปร่างและวงจรของมินิบอร์ดวงจรสวิตช์ ZX-SWITCH01



รูปที่ 3-3 แสดงรูปร่างและวงจรของมินิบอร์ดวงจรตัวต้านทานปรับค่าได้แบบแกนหมุน ZX-POT

3.1.3 ZX-POT มินิบอร์ดวงจรตัวต้านทานปรับค่าได้

ใช้กำหนดแรงดัน 0 ถึง ไฟเลี้ยง (ในที่นี้คือ 3.3V) ตามการหมุนแกนของตัวต้านทานปรับค่าได้ นำไปใช้วัดค่ามุมและระยะทางได้ มีเอาต์พุต 2 แบบคือ ให้แรงดันไฟฟ้ามากขึ้นเมื่อหมุนทวนเข็มนาฬิกาหรือเมื่อหมุนตามเข็มนาฬิกา

มีวงจรและหน้าตาของมินิบอร์ด ZX-POT แสดงในรูปที่ 3-3

3.2 ตัวอย่างการทดลอง

3.2.1 ทดลองการแสดงผล

มีด้วยกัน 2 ตัวอย่าง

- (1) เชื่อมต่อบอร์ด OpenKB กับคอมพิวเตอร์ รออนกระแสทั้ง LED สีฟ้าที่ตำแหน่ง USB ของบอร์ด OpenKB ติดสว่าง เพื่อแจ้งว่า คอมพิวเตอร์ติดต่อกับ OpenKB ได้แล้ว
- (2) เปิดโปรแกรม KidBright IDE สร้างโค้ดตามรูปที่ A3-1 บันทึกไฟล์ชื่อ **OpenKB-Hello.txt**
- (3) อัปโหลดโค้ดไปยังบอร์ด OpenKB ด้วยการคลิกที่ปุ่ม **Build & Run**
- (4) เมื่ออัปโหลดโค้ดเรียบร้อยแล้ว บอร์ด OpenKB จะทำงานทันที ดูผลการทำงานที่จอแสดงผล LED ของ OpenKB

บอร์ด OpenKB แสดงข้อความ Hello World โดยเลื่อนจากขวาซ้ายเพียง 1 ครั้ง



รูปที่ A3-1 โค้ดของไฟล์ OpenKB-Hello.txt สำหรับทดสอบการแสดงผลข้อความอย่างง่าย



รูปที่ A3-2 โค้ดของไฟล์ **OpenKB-HelloForever.txt** สำหรับทดสอบการแสดงผลข้อความต่อเนื่องอย่างง่าย

(5) สร้างโค้ดตามรูปที่ A3-2 บันทึกไฟล์ชื่อ **OpenKB-HelloForever.txt**

(6) อัปโหลดโค้ดไปยังบอร์ด OpenKB ด้วยการคลิกที่ปุ่ม **Build & Run** รอจนกระทั่งการอัปโหลดเสร็จสิ้น

(7) เมื่ออัปโหลดโค้ดเรียบร้อยแล้ว บอร์ด OpenKB จะทำงานทันที ผลการทำงานที่จอแสดงผล LED ของ OpenKB

บอร์ด OpenKB แสดงข้อความ Hello World โดยเลื่อนจากขวามาซ้าย ซ้าย แล้วหยุดรอ 0.5 วินาที จากนั้นวนแสดงข้อความอย่างต่อเนื่อง

3.2.2 ทดสอบอ่านค่าจากตัวตรวจจับแสง

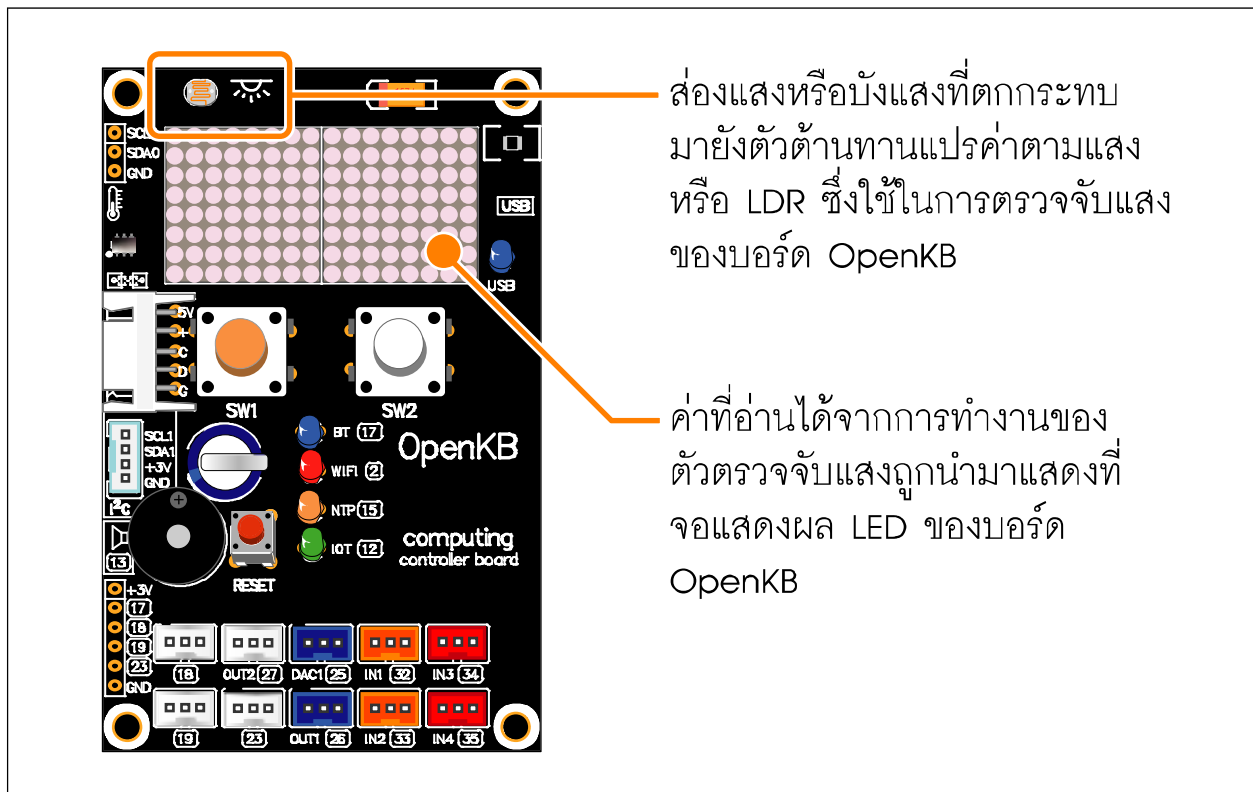
ในตัวอย่างนี้นำเสนอการสร้างโค้ดเพื่ออ่านค่าจากตัวตรวจจับแสงบนบอร์ด OpenKB

(1) เชื่อมต่อบอร์ด OpenKB กับคอมพิวเตอร์ รอจนกระทั่ง LED สีฟ้าที่ตำแหน่ง USB ของบอร์ด OpenKB ติดสว่าง เพื่อแจ้งว่า คอมพิวเตอร์ติดต่อกับ OpenKB ได้แล้ว

(2) เปิดโปรแกรม KidBright IDE สร้างโค้ดตามรูปที่ A3-3 บันทึกไฟล์ชื่อ **OpenKB-LightRead.txt**



รูปที่ A3-3 โค้ดของไฟล์ **OpenKB-LightRead.txt** สำหรับทดสอบการอ่านค่าจากตัวตรวจจับแสงบนบอร์ด OpenKB อย่างง่าย



รูปที่ A3-4 แสดงตำแหน่งของตัวตรวจจับแสงที่ใช้ในการตรวจจับแสงของบอร์ด OpenKB

(3) อัปโหลดโค้ดไปยังบอร์ด OpenKB ด้วยการคลิกที่ปุ่ม **Build & Run** รอจนกระทั่งการอัปโหลดเสร็จสิ้น

(4) เมื่ออัปโหลดโค้ดเรียบร้อยแล้ว บอร์ด OpenKB จะทำงานทันที ทดลองใช้มือบังแสงที่ตกกระทบตัวตรวจจับแสงบนบอร์ด OpenKB ตามรูปที่ A3-4 หรือใช้ไฟฉายส่องแสงไปยังตัวตรวจจับแสง ดูผลการทำงานที่จอแสดงผล LED ของ OpenKB

บอร์ด OpenKB แสดงระดับของความเข้มแสงที่ตรวจจับได้ โดยมีค่า 00 ถึง 100 โดยค่ายิ่งมากหากตรวจจับแสงได้มาก



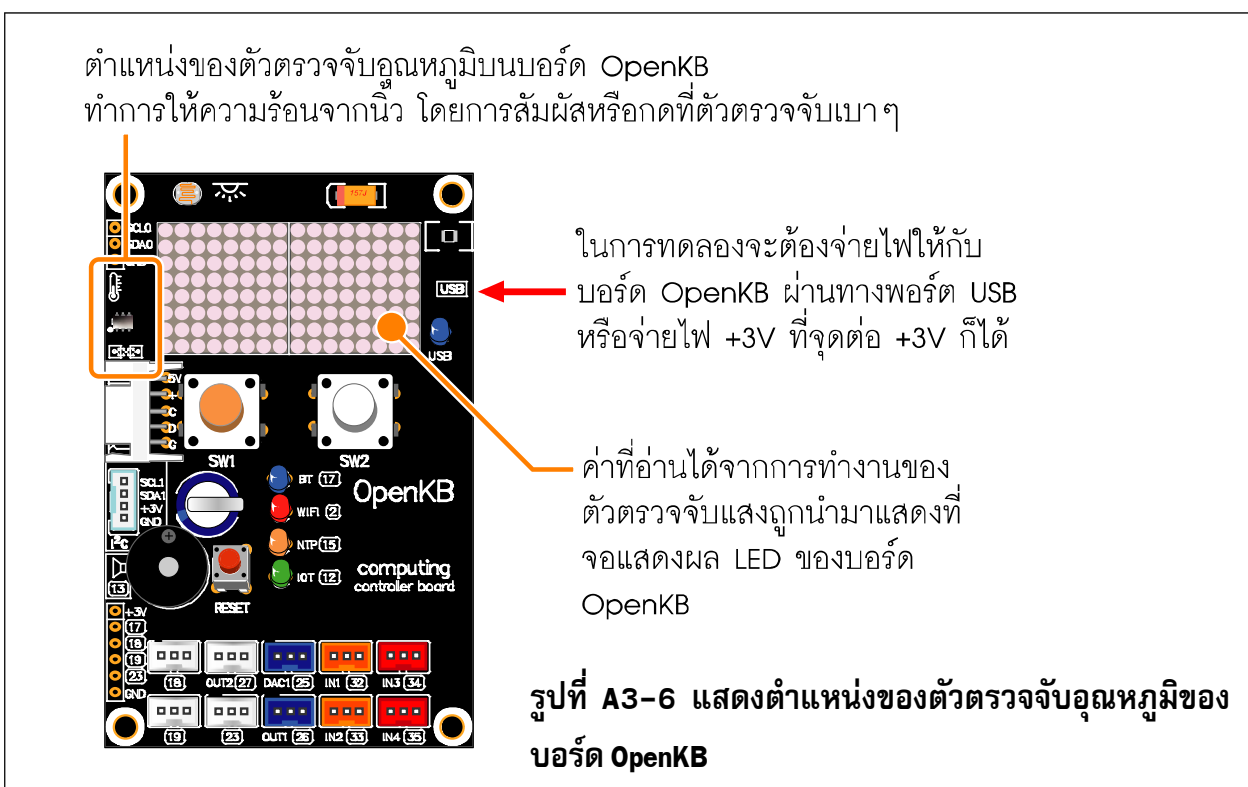
รูปที่ A3-5 โค้ดของไฟล์ OpenKB-TempRead.txt สำหรับทดสอบการอ่านค่าจากตัวตรวจจับอุณหภูมิบนบอร์ด OpenKB อย่างง่าย

3.2.3 ทดสอบอ่านค่าจากตัวตรวจจับอุณหภูมิ

ในตัวอย่างนี้นำเสนอการสร้างโค้ด สำหรับอ่านค่าอุณหภูมิในอากาศจากตัวตรวจจับอุณหภูมิที่ติดตั้งบนบอร์ด OpenKB เพื่อนำมาแสดงผลที่จอแสดงผล LED ของบอร์ด OpenKB โดยแสดงเป็นค่าอุณหภูมิในหน่วยองศาเซลเซียส ความละเอียดทศนิยม 1 ตำแหน่ง และแสดงผลเป็นแบบตัวอักษรเลื่อน

(1) เชื่อมต่อบอร์ด OpenKB กับคอมพิวเตอร์ รอจนกระทั่ง LED สีฟ้าที่ตำแหน่ง USB ของบอร์ด OpenKB ติดสว่าง เพื่อแจ้งว่า คอมพิวเตอร์ติดต่อกับ OpenKB ได้แล้ว

(2) เปิดโปรแกรม KidBright IDE สร้างโค้ดตามรูปที่ A3-5 บันทึกไฟล์ชื่อ **OpenKB-TempRead.txt**



(3) อัปโหลดโค้ดไปยังบอร์ด OpenKB ด้วยการคลิกที่ปุ่ม **Build & Run** รอจนกระทั่งการอัปโหลดเสร็จสิ้น

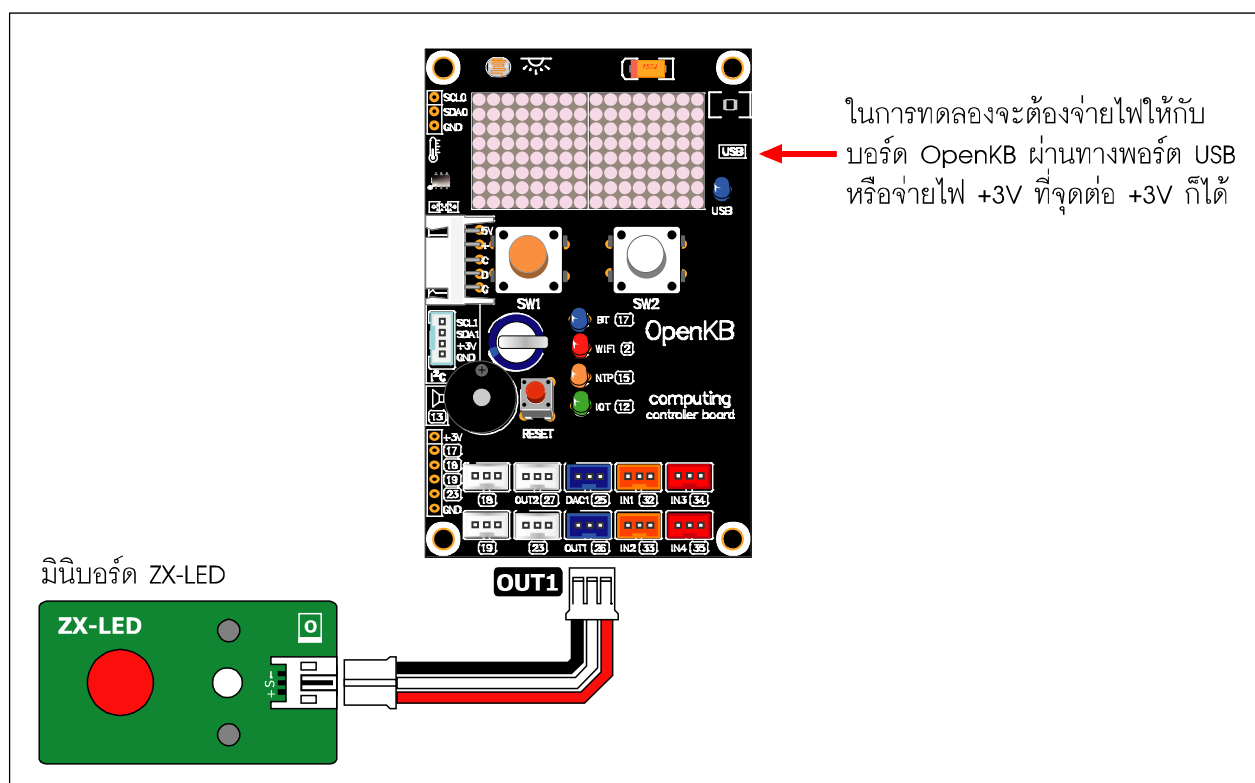
(4) เมื่ออัปโหลดโค้ดเรียบร้อยแล้ว บอร์ด OpenKB จะทำงานทันที ทดลองทดลองให้ความร้อนแก่ตัวตรวจจับอุณหภูมิที่อยู่ทางซ้ายของบอร์ด OpenKB ตามรูปที่ A3-6 ด้วยการใช้นิ้วสัมผัส ผลการทำงานที่จอแสดงผล LED ของ OpenKB

บอร์ด OpenKB แสดงค่าอุณหภูมิเพิ่มสูงขึ้นเมื่อมีการให้ความร้อนแก่ตัวตรวจจับอุณหภูมิด้วยนิ้ว และลดลงเมื่อนำนิ้วออก หรือจะใช้ลมเป่าไปยังตัวตรวจจับอุณหภูมิเมื่อได้รับลม ทำให้อากาศเย็นลง ค่าของอุณหภูมิก็จะลดลง

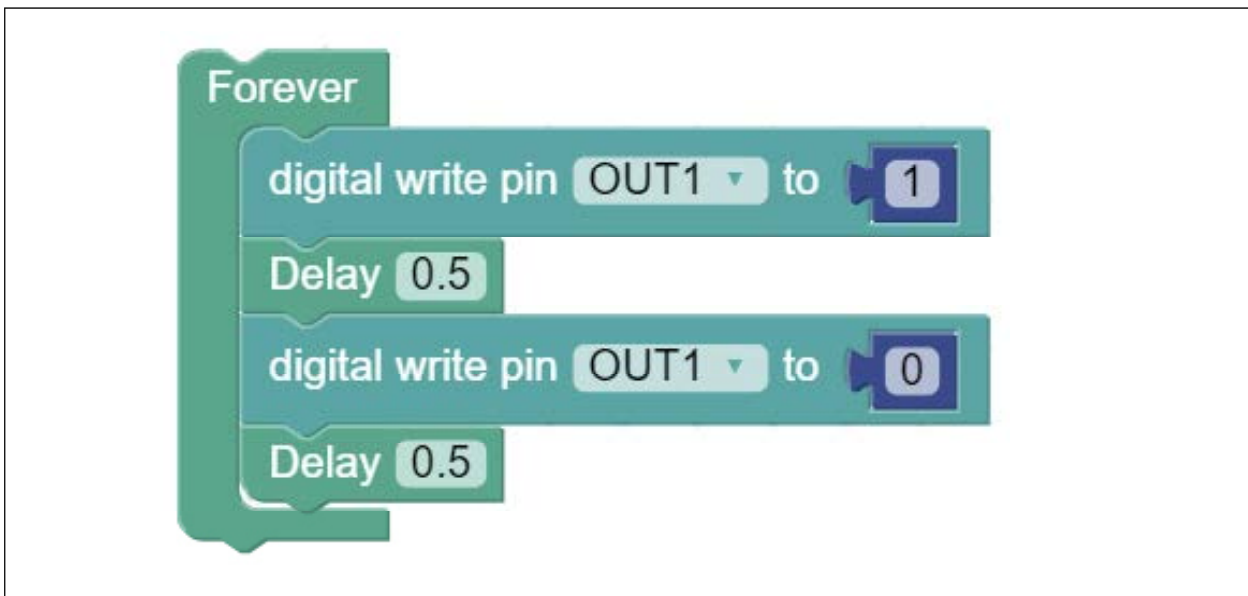
3.2.4 ติดต่ออุปกรณ์เอาต์พุตภายนอกพื้นฐาน

ในหัวข้อนี้เป็นตัวอย่างเริ่มต้นสำหรับการติดต่อกับอุปกรณ์ภายนอกของ OpenKB ผ่านพอร์ตเอาต์พุตดิจิทัล นั่นคือ พอร์ต OUT1 หรือ GPIO26 โดยทำการต่อมินิบอร์ด ZX-LED เข้าที่พอร์ต OUT1 ตามรูปที่ A3-7

(1) เชื่อมต่อบอร์ด OpenKB กับคอมพิวเตอร์ รอจนกระทั่ง LED สีฟ้าที่ตำแหน่ง USB ของบอร์ด OpenKB ติดสว่าง เพื่อแจ้งว่า คอมพิวเตอร์ติดต่อกับ OpenKB ได้แล้ว



รูปที่ A3-7 แสดงการเชื่อมต่ออุปกรณ์เพื่อทดสอบการติดต่อกับอุปกรณ์เอาต์พุตภายนอกของบอร์ด OpenKB



รูปที่ A3-8 โค้ดของไฟล์ **OpenKB-BlinkLED.txt** สำหรับทดสอบการติดต่อกับอุปกรณ์เอาต์พุตดิจิทัลของบอร์ด OpenKB อย่างง่าย

(2) เปิดโปรแกรม KidBright IDE สร้างโค้ดตามรูปที่ A3-8 บันทึกไฟล์ชื่อ **OpenKB-BlinkLED.txt**

(3) อัปโหลดโค้ดไปยังบอร์ด OpenKB ด้วยการคลิกที่ปุ่ม **Build & Run** รอจนกระทั่งการอัปโหลดเสร็จสิ้น

(4) เมื่ออัปโหลดโค้ดเรียบร้อยแล้ว บอร์ด OpenKB จะทำงานทันที ดูผลการทำงานที่มินิบอร์ด ZX-LED

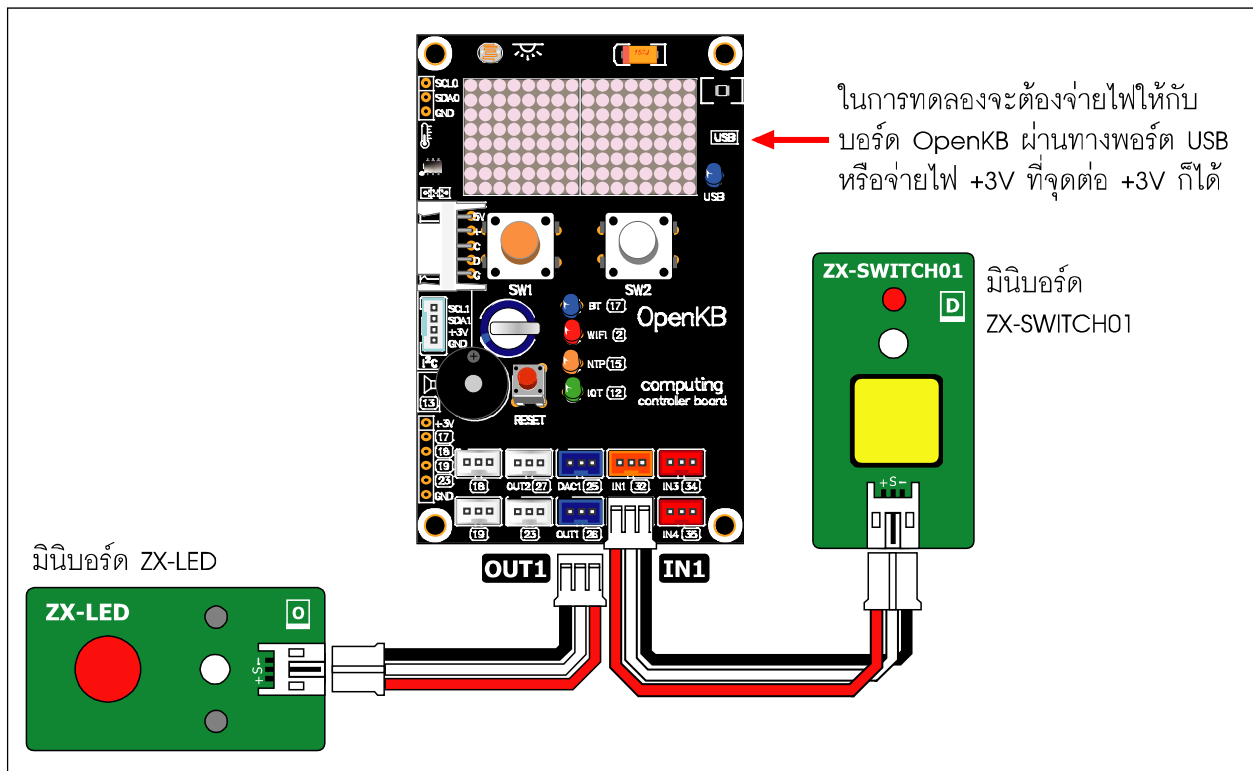
LED บนมินิบอร์ด ZX-LED ที่ต่อกับพอร์ต OUT1 หรือ GPIO26 ติดกะพริบทุกๆ 0.5 วินาที

3.2.5 ติดต่อกับอุปกรณ์อินพุตเอาต์พุตภายนอกพื้นฐาน

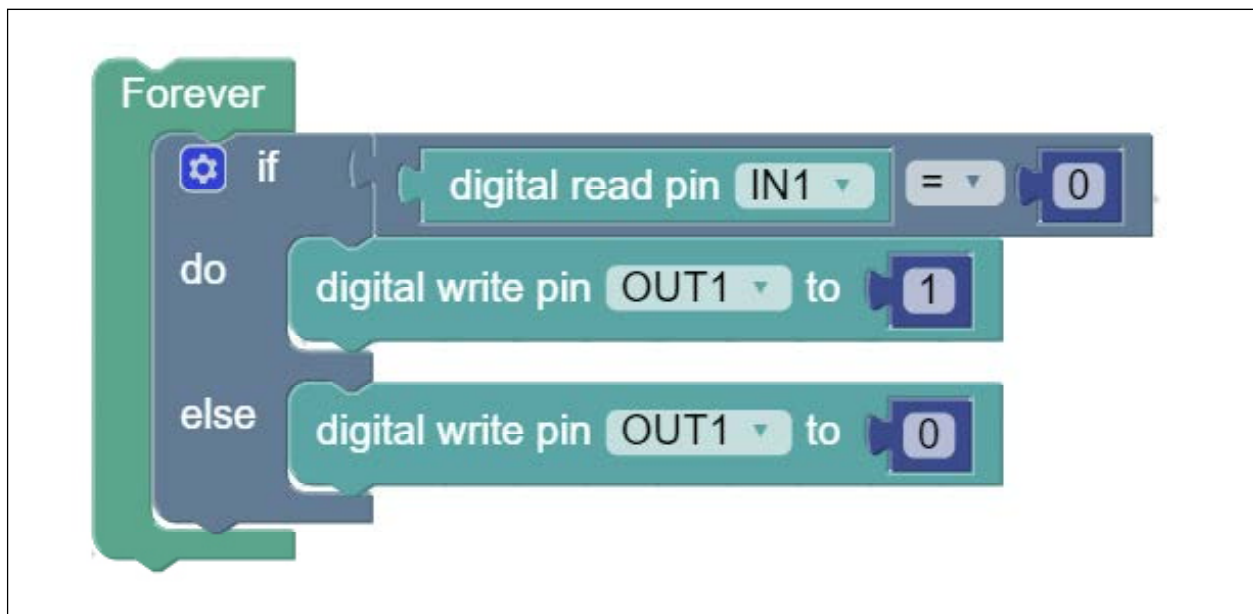
ในหัวข้อนี้เป็นตัวอย่างเริ่มต้นสำหรับการติดต่อกับอุปกรณ์ภายนอกของ OpenKB ผ่านพอร์ตอินพุตเอาต์พุตดิจิทัล นั่นคือ พอร์ต OUT1 หรือ GPIO26 กับ IN1 หรือ GPI32 โดยทำการต่อมินิบอร์ด ZX-LED เข้าที่พอร์ต OUT1 และมินิบอร์ดสวิตช์ ZX-SWITCH01 เข้าที่พอร์ต IN1 ตามรูปที่ A3-9

(1) เชื่อมต่อบอร์ด OpenKB กับคอมพิวเตอร์ รอจนกระทั่ง LED สีฟ้าที่ตำแหน่ง USB ของบอร์ด OpenKB ติดสว่าง เพื่อแจ้งว่า คอมพิวเตอร์ติดต่อกับ OpenKB ได้แล้ว

(2) เปิดโปรแกรม KidBright IDE สร้างโค้ดตามรูปที่ A3-10 บันทึกไฟล์ชื่อ **OpenKB-InOutTest.txt**



รูปที่ A3-9 แสดงการเชื่อมต่ออุปกรณ์เพื่อทดสอบการติดต่ออุปกรณ์อินพุตเอาต์พุตภายนอกพื้นฐานของบอร์ด OpenKB



รูปที่ A3-10 โค้ดของไฟล์ OpenKB-InOutTest.txt สำหรับทดสอบการติดต่อกับอุปกรณ์อินพุตเอาต์พุตดิจิทัลของบอร์ด OpenKB อย่างง่าย

(3) อัปโหลดโค้ดไปยังบอร์ด OpenKB ด้วยการคลิกที่ปุ่ม **Build & Run**

(4) เมื่ออัปโหลดโค้ดเรียบร้อยแล้ว บอร์ด OpenKB จะทำงานทันที ทดลองกดสวิตช์แล้วปล่อย
ดูผลการทำงานที่มินิบอร์ด ZX-LED

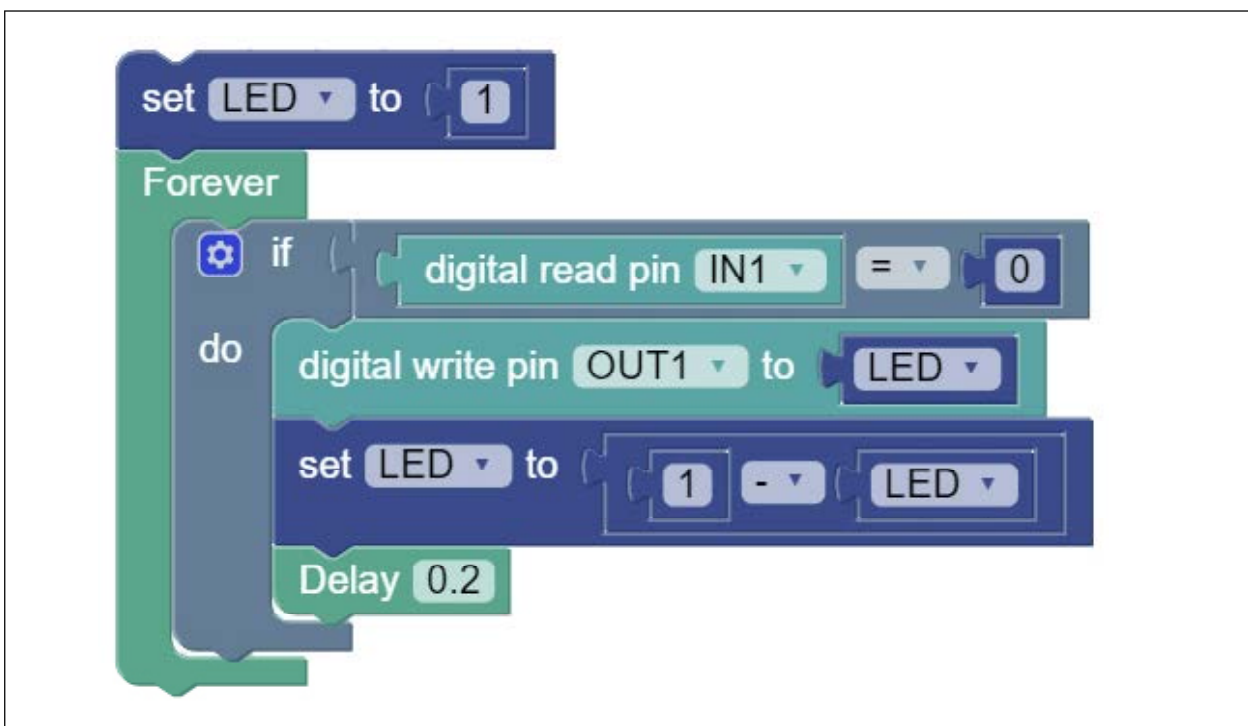
เมื่อโปรแกรมทำงาน OpenKB จะรอการกดสวิตช์ที่พอร์ต IN1

เมื่อเกิดการกดสวิตช์ทำให้สถานะลอจิกเป็น “0” เงื่อนไขการตรวจสอบจึงเป็นจริง OpenKB จะส่งสัญญาณลอจิก “1” ไปยังพอร์ต OUT1 ทำให้ LED ที่ต่ออยู่ติดสว่าง

เมื่อปล่อยสวิตช์ สถานะลอจิกที่ขา IN1 เป็น “1” ทำให้เงื่อนไขเป็นเท็จ OpenKB จะส่งสัญญาณลอจิก “0” ไปยังพอร์ต OUT1 ทำให้ LED ที่ต่ออยู่ดับ

(5) สร้างโค้ดสร้างโค้ดตามรูปที่ A3-11 บันทึกไฟล์ชื่อ OpenKB-Out1ToggleSwitch.txt

เริ่มต้นด้วยการสร้างตัวแปรชื่อ LED และกำหนดสถานะเริ่มต้นเป็น “1” จากนั้นจะวนตรวจสอบการกดสวิตช์ที่ขาพอร์ต IN1 หรือ GPI32 ด้วยบล็อกคำสั่ง digital read pin หากเป็น “0” แสดงว่า มีการกดสวิตช์เกิดขึ้น ก็จะนำค่าของ ตัวแปร LED ที่เป็น “1” เขียนไปยังขาพอร์ต OUT1 ด้วยคำสั่ง digital write pin ทำให้ LED บนมินิบอร์ด ZX-LED ติดสว่าง จากนั้น จะทำการกำหนดค่าของตัวแปร LED ใหม่ ด้วยการนำไปลบกับเลข 1 จะได้ $1-1 = 0$ นำไปเก็บไว้ในตัวแปร LED



รูปที่ A3-11 โค้ดของไฟล์ OpenKB-Out1ToggleSwitch.txt สำหรับทดสอบการติดต่อกับอุปกรณ์อินพุต
เอาต์พุตดิจิทัลของบอร์ด OpenKB เพื่อทำงานเป็นสวิตช์ควบคุม LED แบบกดติดกดดับ

เมื่อวนทำงานในรอบใหม่ หากยังไม่มีเกิดการกดสวิตช์ จะยังไม่มีเขียนค่าใหม่ไปยังพอร์ต OUT1 ทำให้ LED ยังคงติดสว่าง จนกระทั่งมีการกดสวิตช์ครั้งใหม่ ค่าของตัวแปร LED ที่ขณะนี้ เป็น “0” แล้ว จะถูกเขียนไปยังพอร์ต OUT1 ทำให้ LED บนมินิบอร์ด ZX-LED ดับลง และมีการปรับปรุงค่าของตัวแปร LED ใหม่อีกครั้งเป็น $1 - 0 = 1$ ทำให้ค่าของตัวแปร LED เป็น “1” เพื่อนำไปใช้เขียนค่าไปยังพอร์ต OUT1 หากเกิดการกดสวิตช์ในครั้งต่อไป

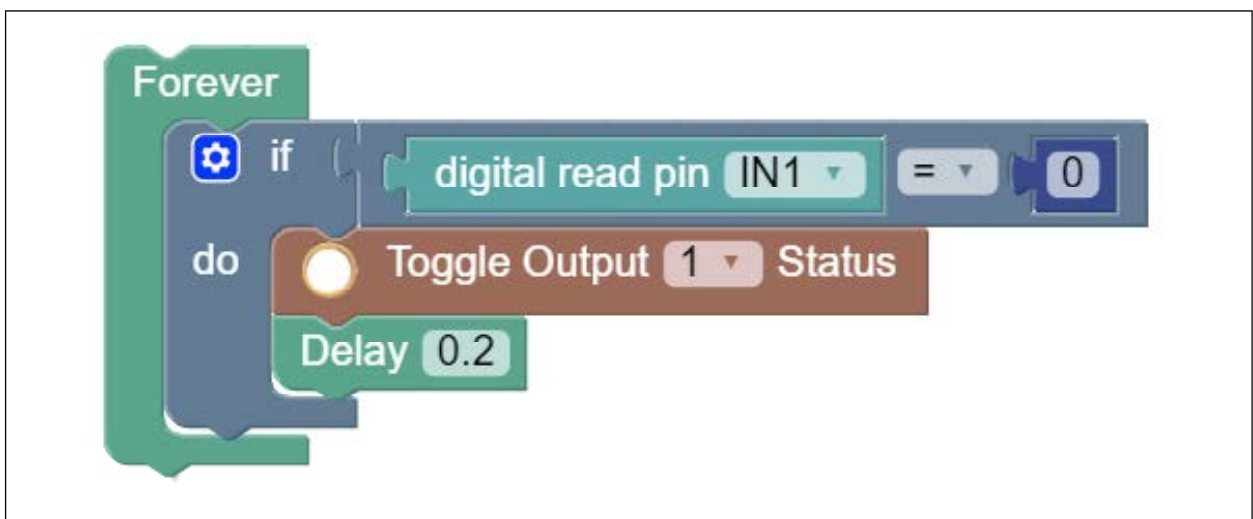
(6) อัปโหลดโค้ดไปยังบอร์ด OpenKB ด้วยการคลิกที่ปุ่ม **Build & Run** รอจนกระทั่งการอัปโหลดเสร็จสิ้น

(7) เมื่ออัปโหลดโค้ดเรียบร้อยแล้ว บอร์ด OpenKB จะทำงานทันที ทดลองกดสวิตช์แล้วปล่อย ผลการทำงานที่มินิบอร์ด ZX-LED จากนั้นทำการกดสวิตช์แล้วปล่อย ทำหลายๆ ครั้ง เพื่อตรวจสอบการทำงาน

ผลการทำงานของโค้ดในรูปที่ A3-11 เป็นการควบคุมอุปกรณ์เอาต์พุตอย่าง LED ด้วยอุปกรณ์อินพุตอย่างสวิตช์ โดยทำงานในแบบเมื่อกดสวิตช์แล้วปล่อย LED ติดค้าง หากกดสวิตช์แล้วปล่อยอีกครั้ง LED จะดับ สลับไปเช่นนี้ในทุกครั้งที่มีการกดสวิตช์

(9) สร้างโค้ดสร้างโค้ดตามรูปที่ A3-12 บันทึกไฟล์ชื่อ **OpenKB-Out1ToggleSwitch2.txt**

ตัวอย่างต่อไปนี้เป็น การควบคุมอุปกรณ์เอาต์พุตอย่าง LED ด้วยอุปกรณ์อินพุตอย่างสวิตช์ โดยทำงานในแบบเมื่อกดสวิตช์แล้วปล่อย LED ติดค้าง หากกดสวิตช์แล้วปล่อยอีกครั้ง LED จะดับ สลับไปเช่นนี้ในทุกครั้งที่มีการกดสวิตช์



รูปที่ A3-12 โค้ดของไฟล์ **OpenKB-Out1ToggleSwitch2.txt** สำหรับทดสอบการติดต่อกับอุปกรณ์อินพุตเอาต์พุตดิจิทัลของบอร์ด OpenKB เพื่อทำงานเป็นสวิตช์ควบคุม LED แบบกดติดกดดับ

จากโค้ดตัวอย่างก่อนหน้านี้การเปลี่ยนแปลงค่าของตัวแปร *LED* เพื่อเขียนไปยังขาพอร์ต *OUT1* ใช้การคำนวณทางคณิตศาสตร์ แต่สำหรับตัวอย่างนี้ใช้บล็อกคำสั่ง *Toggle Output Status* มาทำงานแทน ทำให้โค้ดสั้นลง การทำงานของโค้ดจะวนตรวจสอบการกดสวิตช์ที่ขาพอร์ต *IN1* หรือ *GPI32* ด้วยบล็อกคำสั่ง *digital read pin* หากเป็น “0” แสดงว่า มีการกดสวิตช์เกิดขึ้น ก็จะทำการสลับสถานะลอจิกที่ขาพอร์ต *OUT1* ทันที หากสถานะเดิมเป็น “0” ก็จะเปลี่ยนเป็น “1” หรือหากเดิมเป็น “1” ก็จะเปลี่ยนเป็น “0” ทำให้ *LED* ของมินิบอร์ด *ZX-LED* ติดดับตามสถานะลอจิกที่เกิดจากการทำงานของบล็อกคำสั่ง *Toggle Output Status*

โค้ดนี้ง่ายขึ้นมาก เพราะไม่ต้องสร้างตัวแปรเพื่อเก็บค่าสถานะและเขียนไปยังพอร์ตเอาต์พุต ไม่ต้องมีการปรับปรุ่ค่าของตัวแปร เพียงตรวจสอบสถานะการกดสวิตช์ แล้วกลับสถานะลอจิกที่ขาพอร์ตเอาต์พุต ก็จะได้ผลการทำงานตามที่ต้องการ

(10) อัปโหลดโค้ดไปยังบอร์ด OpenKB ด้วยการคลิกที่ปุ่ม *Build & Run* รอจนกระทั่งการอัปโหลดเสร็จสิ้น

(11) เมื่ออัปโหลดโค้ดเรียบร้อยแล้ว บอร์ด OpenKB จะทำงานทันที ทดลองกดสวิตช์แล้วปล่อย ดูผลการทำงานที่มินิบอร์ด *ZX-LED* จากนั้นทำการกดสวิตช์แล้วปล่อย ทำหลายๆ ครั้ง เพื่อตรวจสอบการทำงาน

ผลการทำงานของโค้ดในรูปที่ A3-12 เป็นการควบคุมอุปกรณ์เอาต์พุตอย่าง *LED* ด้วยอุปกรณ์อินพุตอย่างสวิตช์เหมือนกับโค้ดในรูปที่ A3-11 ความแตกต่างมีเพียงการสร้างโค้ดที่ง่ายและกระชับขึ้น