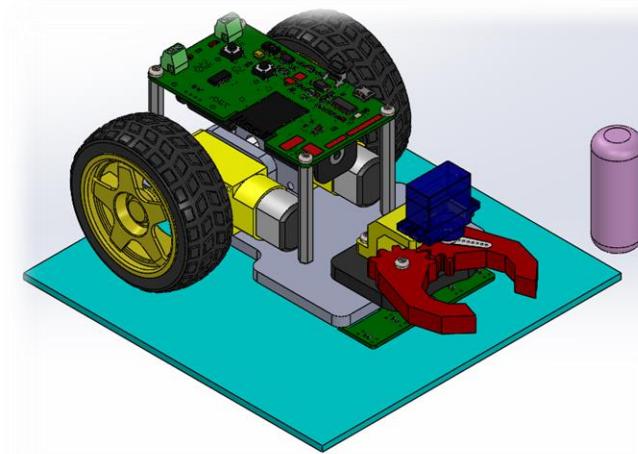
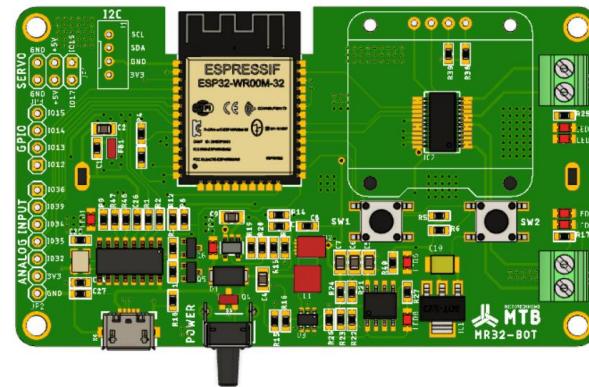


การใช้ KB-IDE ในการเขียนโปรแกรมควบคุมหุ่นยนต์ ด้วยบอร์ด MR32-BOT



KB-IDE
by makerasia



BY MR. META BOONMA

POWERED by

MICROTECHBEING
MTB

ประวัติผู้บรรยาย

ชื่อ นายเมรา บุญมา (Mr. Meta Boonma) การศึกษา

- 2018 : มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ
- 2016 : วิทยาลัยเทคนิคสมุทรสงคราม
- 2013 : วิทยาลัยเทคนิคสมุทรสงคราม

- | ป.ตรี เทคโนโลยีวิศวกรรมอิเล็กทรอนิกส์คอมพิวเตอร์
- | ปวส. เทคโนโลยีระบบโทรคมนาคม
- | ปวช. ช่างอิเล็กทรอนิกส์

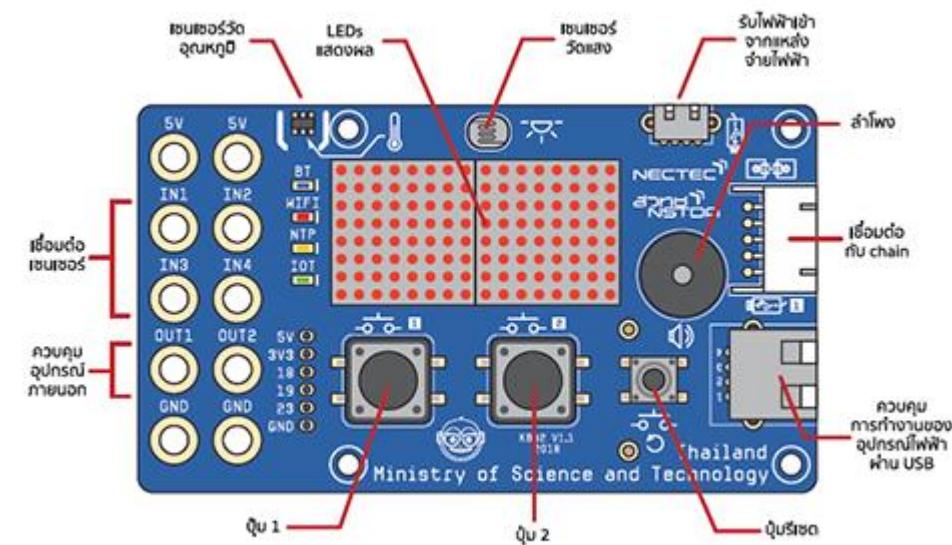
ประสบการณ์การทำงาน

- R&D Engineer @ FORTH TRACKING SYSTEM Co., Ltd.
- IoT Software Engineer @ FreewillFX Co.,Ltd. (ปัจจุบัน)
- Project manager @ **Medical Cannabis Monitoring Platform** powered by MTB (ปัจจุบัน)

ผลงาน

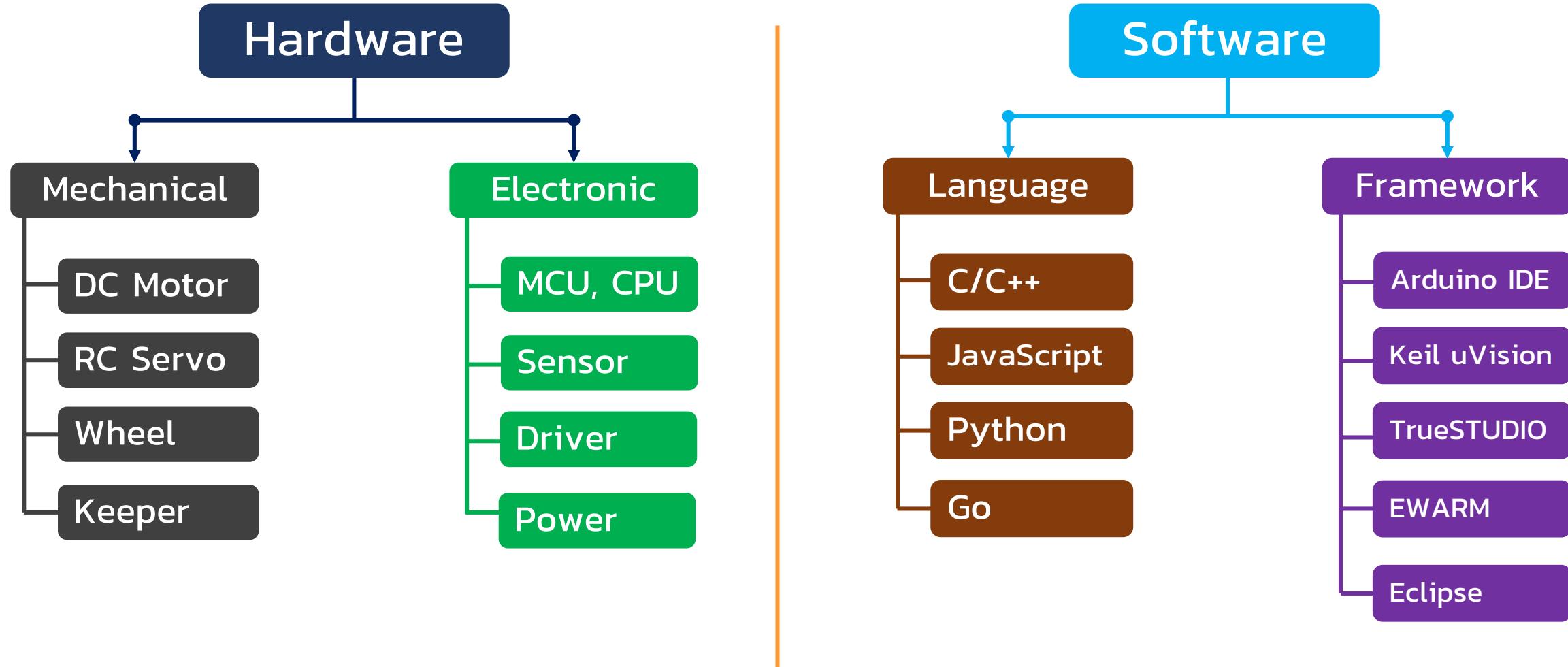
- รางวัลรองชนะเลิศการแข่งขันหุ่นยนต์ ABU Robocon Contest Thailand ในปี 2014 (Hoilod Robot team)
- ออกแบบและพัฒนา Embedded system, Robotic board ด้วย Microcontroller, SoC, SiP
- การใช้ซีปรมะวลดพล DSP เพื่อควบคุมระบบเสียง(Sound System) ด้วย Android application
- ออกแบบ Power Management และ LOW Power สำหรับ Smart devices
- ออกแบบผลิตภัณฑ์ Smart Electronic ด้วยระบบปฏิบัติการ Linux, RTOS (GPS, Fingerprint scan, Face recognition)
- ออกแบบและพัฒนาอุปกรณ์ IoT บนเทคโนโลยีการสื่อสารข้อมูล LORA, NB-IoT, 3G/4G, WLAN
- พัฒนา IoT Platform ในชื่อ **Medical Cannabis Monitoring Platform** powered by MTB

ทำความรู้จักกับ KidBright



KidBright เป็นบอร์ดส่งเสริมการเรียนรู้พื้นฐานของ **การเขียนโปรแกรม (Programming)** จุดเริ่มต้นจากโครงการสื่อการสอนโปรแกรมมิ่งในโรงเรียน **(Coding at School Project)** ของ **กระทรวงวิทยาศาสตร์และเทคโนโลยี** ซึ่งมีจุดมุ่งหมายในการพัฒนาศักยภาพระหว่าง **ความคิดเชิงตรรกศาสตร์ และความคิดริเริ่มสร้างสรรค์** ในลักษณะการเรียนรู้แบบ learn and play

ส่วนประกอบในการเขียนโปรแกรมหุ่นยนต์



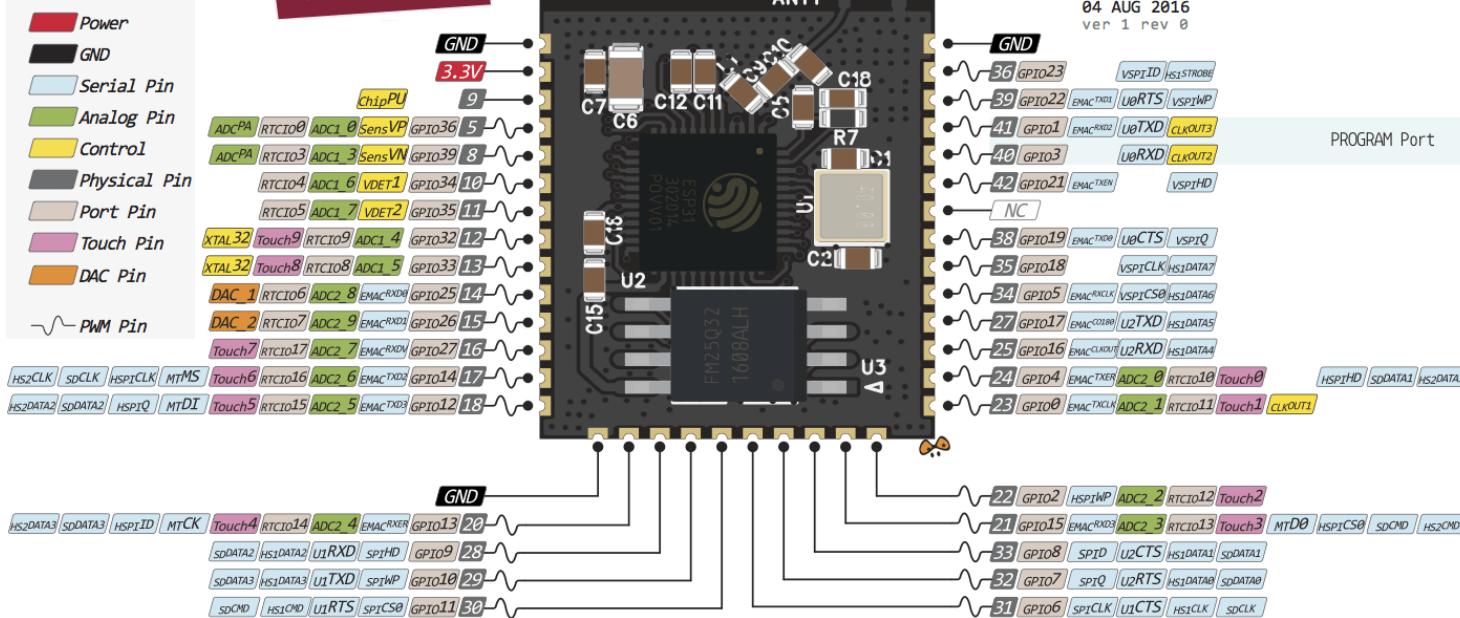
ทำความรู้จักกับ ESP32



ESPRESSIF

WROOM32

PINOUT



04 AUG 2016
ver 1 rev 0

สเปคของ ESP32

- ชิปเซ็ตสถาปัตยกรรม Tensilica LX6 แบบ 2 แกนสมอง สัญญาณนาฬิกา 240MHz
- มีแรมในตัว 512KB
- รองรับการเชื่อมต่อรวมภายในอักษรสูงสุด 16MB
- มาพร้อมกับ WiFi มาตรฐาน 802.11 b/g/n รองรับการใช้งานทั้งในโหมด Station softAP และ Wi-Fi direct
- มีบลูทูธในตัว รองรับการใช้งานในโหมด 2.0 และโหมด 4.0 BLE
- มี GPIO จำนวน 32 ช่อง
- รองรับ UART จำนวน 3 ช่อง
- รองรับ SPI จำนวน 3 ช่อง
- รองรับ I²C จำนวน 2 ช่อง
- รองรับ ADC จำนวน 12 ช่อง
- รองรับ DAC จำนวน 2 ช่อง
- รองรับ I^S จำนวน 2 ช่อง
- รองรับ PWM / Timer ทุกช่อง
- รองรับการเชื่อมต่อกับ SD-Card

ESP32 เป็นชิปในโครงสร้างสถาปัตยกรรม WiFi และ Bluetooth 4.2 แบบ SoC (System on Chip) สามารถเขียนโปรแกรมลงไปได้เลยไม่จำเป็นต้องใช้ MCU อื่นมาควบคุม ถูกพัฒนาโดยบริษัท Espressif โดย CPU ใช้สถาปัตยกรรม Tensilica LX6 แบบ 2 แกนสมอง สัญญาณนาฬิกา 240MHz มี GPIO สำหรับต่อใช้งานจำนวน 32 ช่อง

การพัฒนาโปรแกรมบน ESP32



ESP-WROOM-32



Microcontroller



ESP-IDF เป็นคอมโพเนนต์ของ ESP32 ที่ทางบริษัท ESPRESSIF ผู้ที่ออกแบบ SOC ตัวนี้ขึ้นมาจึง
จำเป็นต้องใช้ ESP-IDF ในการเขียนชุดคอมโพเนนต์เพื่อ Upload code ลงใน ESP32



ต่อมามีคนเขียน lib ลงบน Arduino และ PlatformIO ทำให้เขียนโปรแกรมควบคุม ESP32
ง่ายมากยิ่งขึ้น

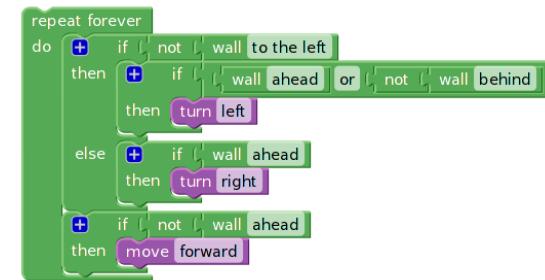


PlatformIO IDE



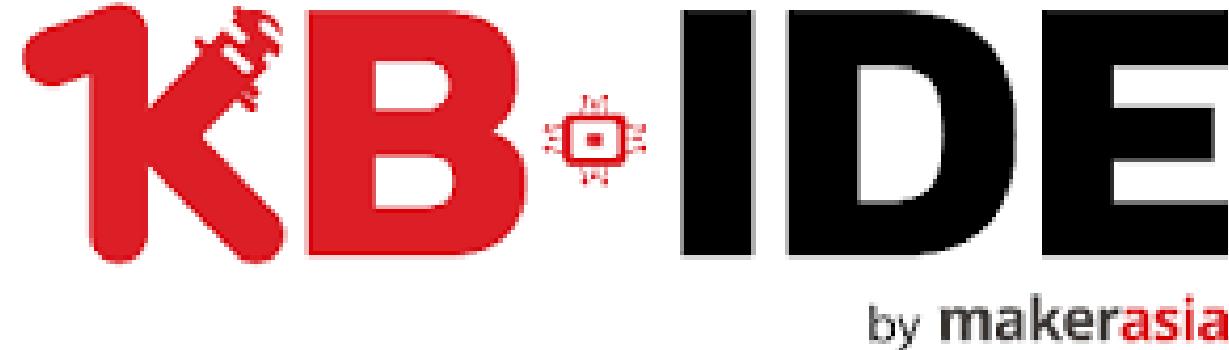
Text coding IDE

Blockly coding IDE



Programing language

ກໍາຄວາມຮູ້ຈັກກັບ KB-IDE



KB-IDE ເປັນໂປຣແກຣມສ້າງຂາດໃຫຍ່ສໍາເຮັບໃຊ້ເຂົ້າເບີນໂປຣແກຣມບັນບອົດ IoT (ແລະ SBC ດ້ວຍໃນອນາຄຕ) ໂດຍເປົາເໝາຍການອອກແບບຄືວ IDE ສໍາເຮັບຖຸກຄນ ຖຸກຮະດັບ ເດືອກໆ ສາມາດໃຊ້ Block Programming ໄດ້ ມີວິທີໂປຣສາມາດໃຊ້ Code Editor ໄດ້ ເປັນ IDE ທີ່ໃຊ້ໄດ້ກັບທຸກບອົດໃນຕາລາດ ທີ່ສໍາຄັນຄືວເປັນ IDE ທີ່ Hackable ນັ້ນໜໍາຍຄວາມວ່ານັກພັດນາສາມາດແປ່ລິ່ຍນແປລັງ ຜັກຕົວຕ່ອຍອັດ IDE ໄດ້ ໄນວ່າຈະເປັນ Board Manager ຢັ້ງPlug-in ຕ່າງໆ ແລະ ເນື່ອງຈາກ KB-IDE ເປັນ Open Source ແລະ Open Architecture.

ความสามารถของ KB-IDE



- เขียนโปรแกรมได้กึ่งในแบบ Blocks Programming และ Code Editor
- เพิ่ม Hardware Board ได้ใน Board Manager
- เพิ่ม Framework ในการเขียนโปรแกรมได้ (ตอนนี้ support ESP-IDF และ Arduino)
- เพิ่ม Plugins และ Library ได้
- คอมโพล์เป็น Native Code ทำให้มีข้อจำกัดในด้านความเร็ว และสามารถคอมโพล์ได้เร็วกว่า IDE ในตลาดปัจจุบันมากรวมกับ Arduino IDE
- เครื่องช่วยในการเขียนโปรแกรมต่าง ๆ เช่น Serial Console และ Serial Graph ที่พร้อมใช้งาน
- สามารถใช้ได้กับ Mac, Windows และ Linux
- ตัวอย่างและ Tutorial พร้อมใช้งาน

Text Programming

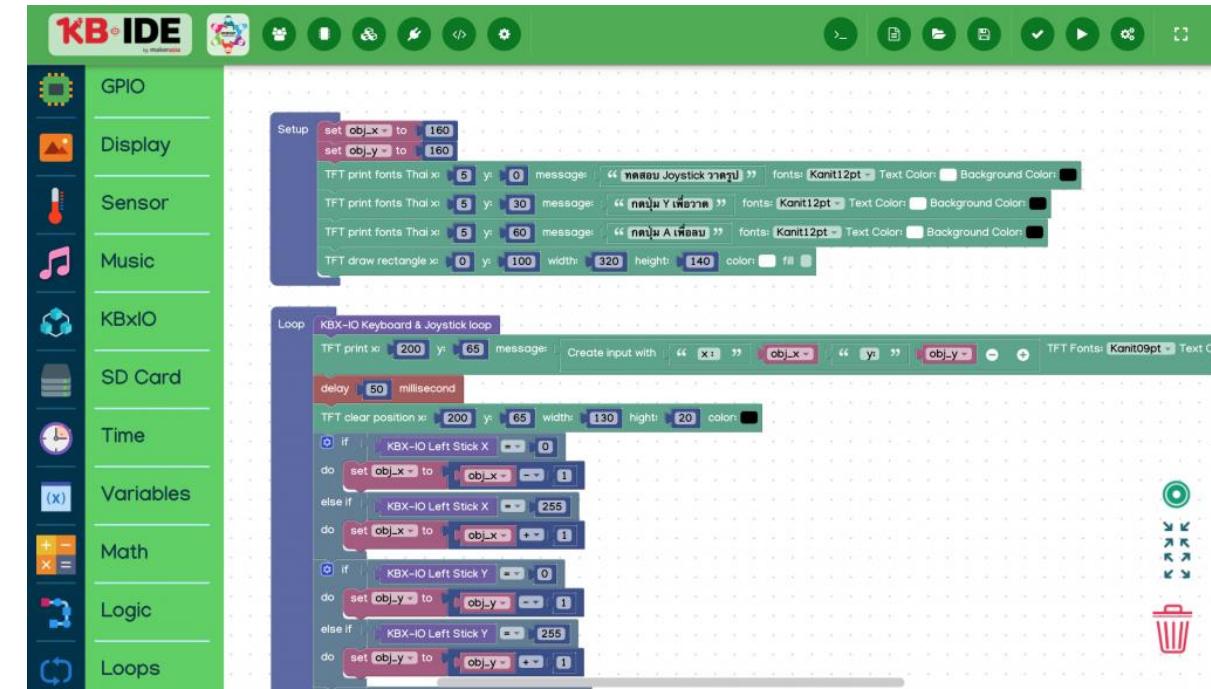
- Speed up coding by converting C/C++ from block.
- Line error hint, indicate and report lines with error.
- 50+ editor themes, dark/light, select your personal preference.
- Fast switch block & code mode.
- External package, add a more powerful tool.

```

1  /*
2   * ESP32 eeprom_class example with EEPROM library
3
4   This simple example demonstrates using EEPROM library to store different data
5   in ESP32 Flash memory in a multiple user-defined EEPROM partition (0x1000 or
6   4KB max size or less).
7
8   Install 'ESP32 Partition Manager' ONCE from
9   https://github.com/francis94c/ESP32Partitions And generate different
10  partitions with 'partition_name' Usage: EEPROMClass
11  ANY_OBJECT_NAME("partition_name", size);
12
13 Generated partition that would work perfectly with this example
14 #Name, Type, Subtype, Offset, Size, Flags
15 nvs, data, nvs, 0x9000, 0x5000,
16 otadata, data, ota, 0xe000, 0x2000,
17 app0, app, ota_0, 0x10000, 0x140000,
18 app1, app, ota_1, 0x150000, 0x140000,
19 eeprom0, data, 0x99, 0x290000, 0x1000,
20 eeprom1, data, 0x9a, 0x291000, 0x500,
21 eeprom2, data, 0x9b, 0x292000, 0x100,
22 spiffs, data, spiffs, 0x293000, 0x16d000,
23
24 Created for arduino-esp32 on 25 Dec, 2017
25 by Elochukwu Ifediora (fedy0)
26 */
27
28 #include "EEPROM.h"
29
30 // Instantiate eeprom objects with parameter/argument names and size same as in
31 // the partition table
32 EEPROMClass NAMES("eeprom0", 0x1000);
33 EEPROMClass HEIGHT("eeprom1", 0x500);
34 EEPROMClass AGE("eeprom2", 0x100);

```

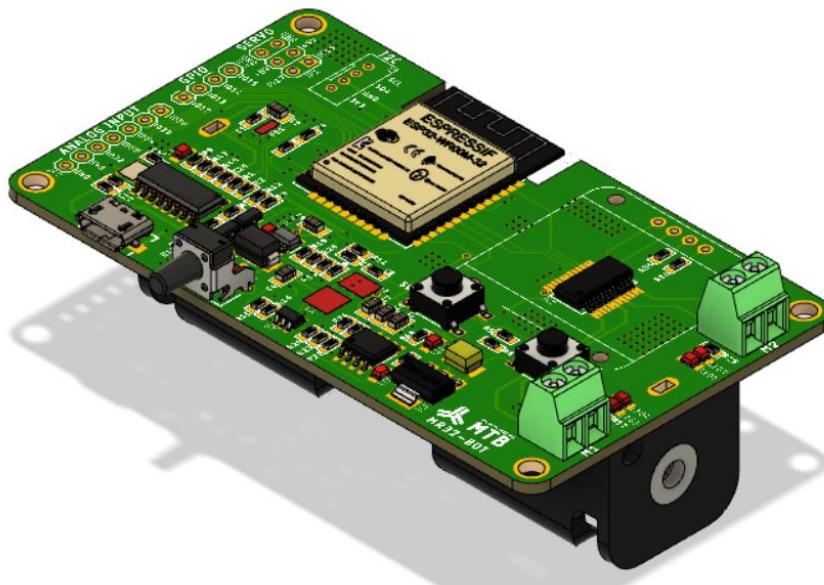
Board : KBX-UI (kbx) Mode : 3



Block Programming

- 100% no hardcoded, just drag & drop with block connection.
- Easy and Intuitive.
- Basic & Advance block coverage 99% of the code that you write.
- Various helping blocks, Images, Music, Text-to-speech.
- Have fun in exploring with your creativity.
- Speed up text-coding, all blocks are 100% convertible ready-compile C/C++

ทำความรู้จักกับบอร์ด MR32-BOT

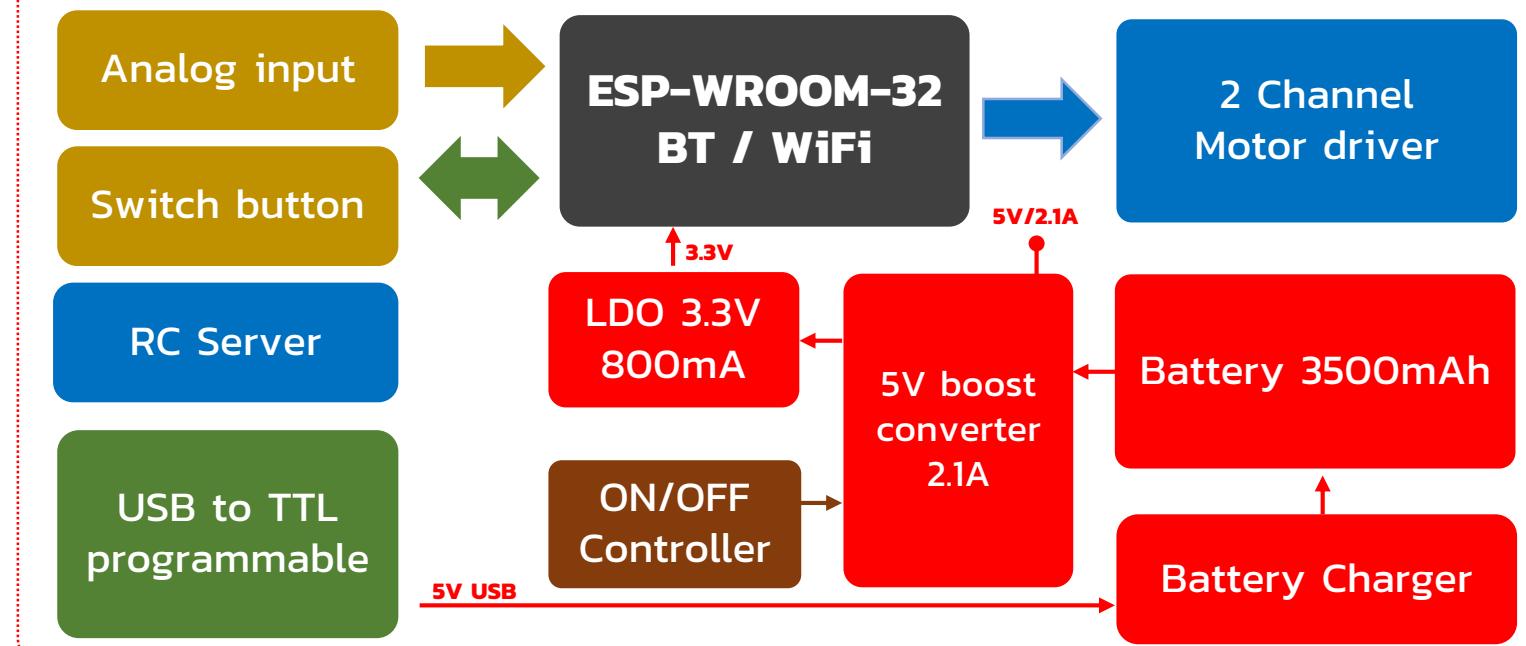


MR32-BOT Rev. A

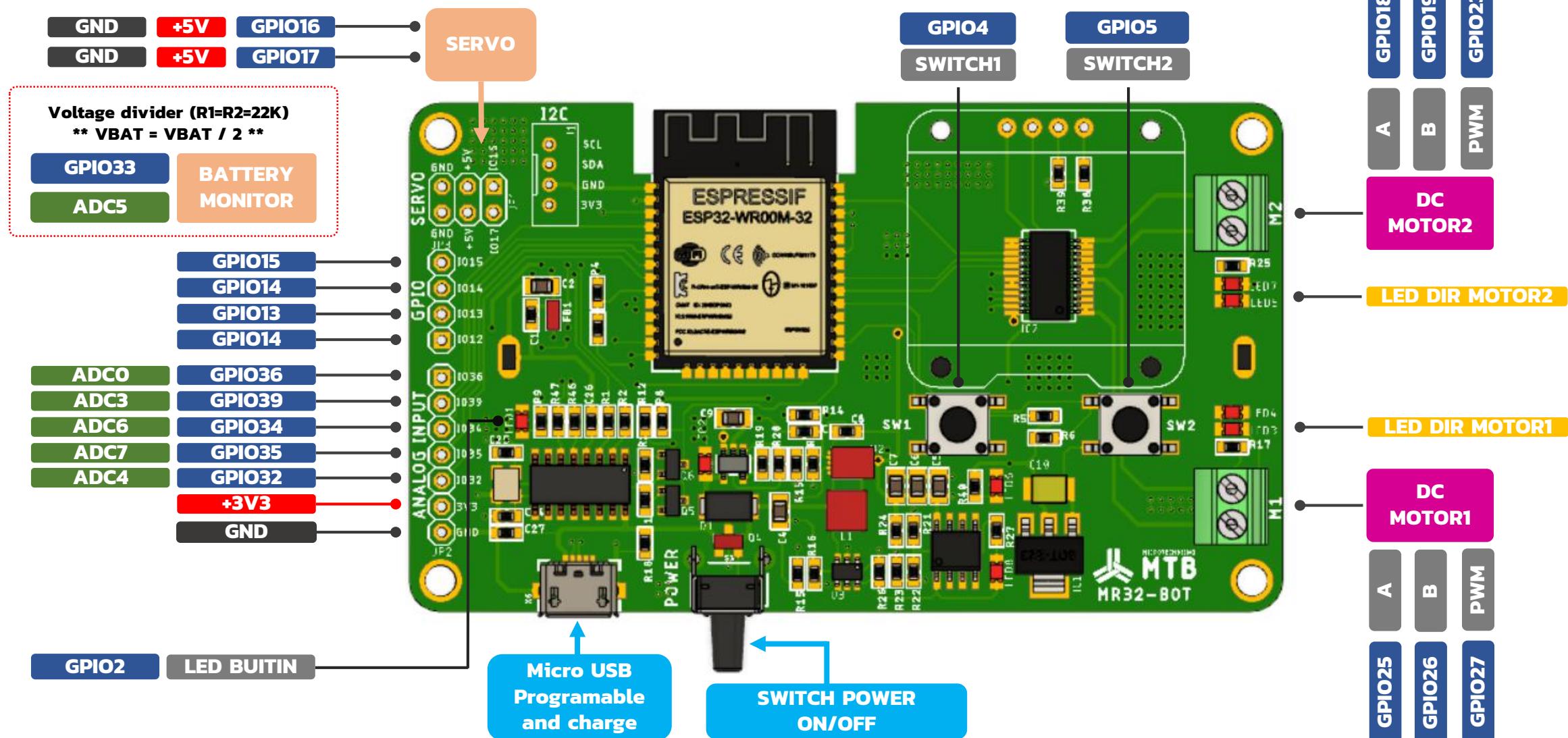
บอร์ดไมโครคอนโทรลเลอร์ **MR32-BOT** เป็นบอร์ดที่พัฒนาขึ้นมาด้วยวัตถุประสงค์ที่
อยากให้การสร้างหุ่นยนต์ขนาดเล็กง่ายมากยิ่งขึ้นโดยไม่จำเป็นต้องหาบอร์ดอื่นๆ มาต่อพ่วงกัน
เพื่อที่จะลดปัญหาของ การซื้อมต่อสายไฟผิดพลาด, ต่อแบตเตอรี่ผิดขั้ว, ขนาดของบอร์ด
สำเร็จรูปในก้อนตลาดมีขนาดใหญ่และราคาสูง

บอร์ดไมโครคอนโทรลเลอร์ **MR32-BOT** จึงได้ถูกออกแบบมาให้ช่วยแก้ปัญหาที่กล่าวมาใน
ตอนต้นและยังช่วยลดเวลาในการพัฒนาโปรแกรมลงค่อนข้างมาก

MR32-BOT Architecture



បាត់ងណុខោ MR32-BOT



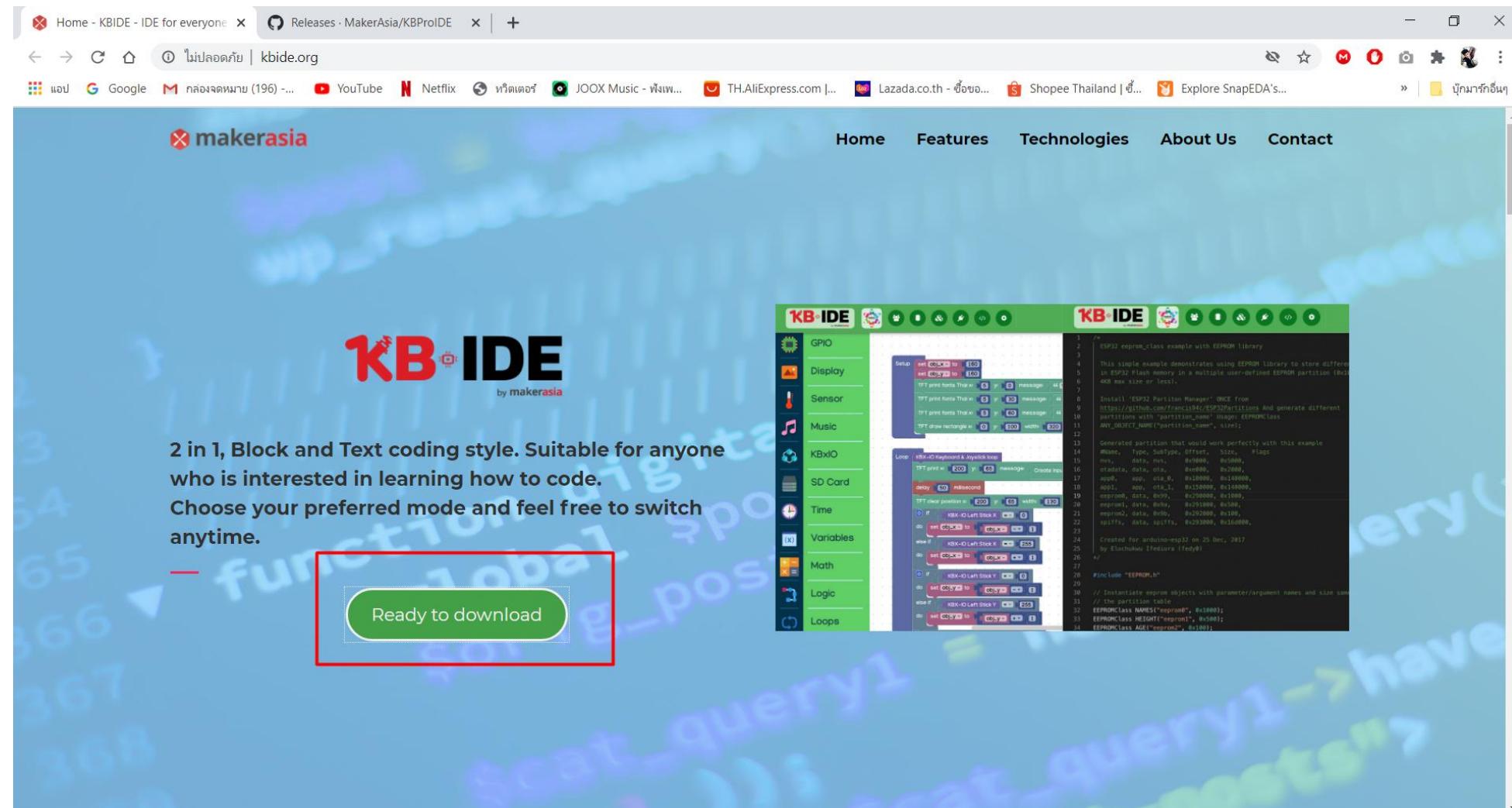
Introduction

POWERED by

MICROTECHBEING
MTB

การติดตั้งโปรแกรม KB-IDE

<https://kbide.org/>



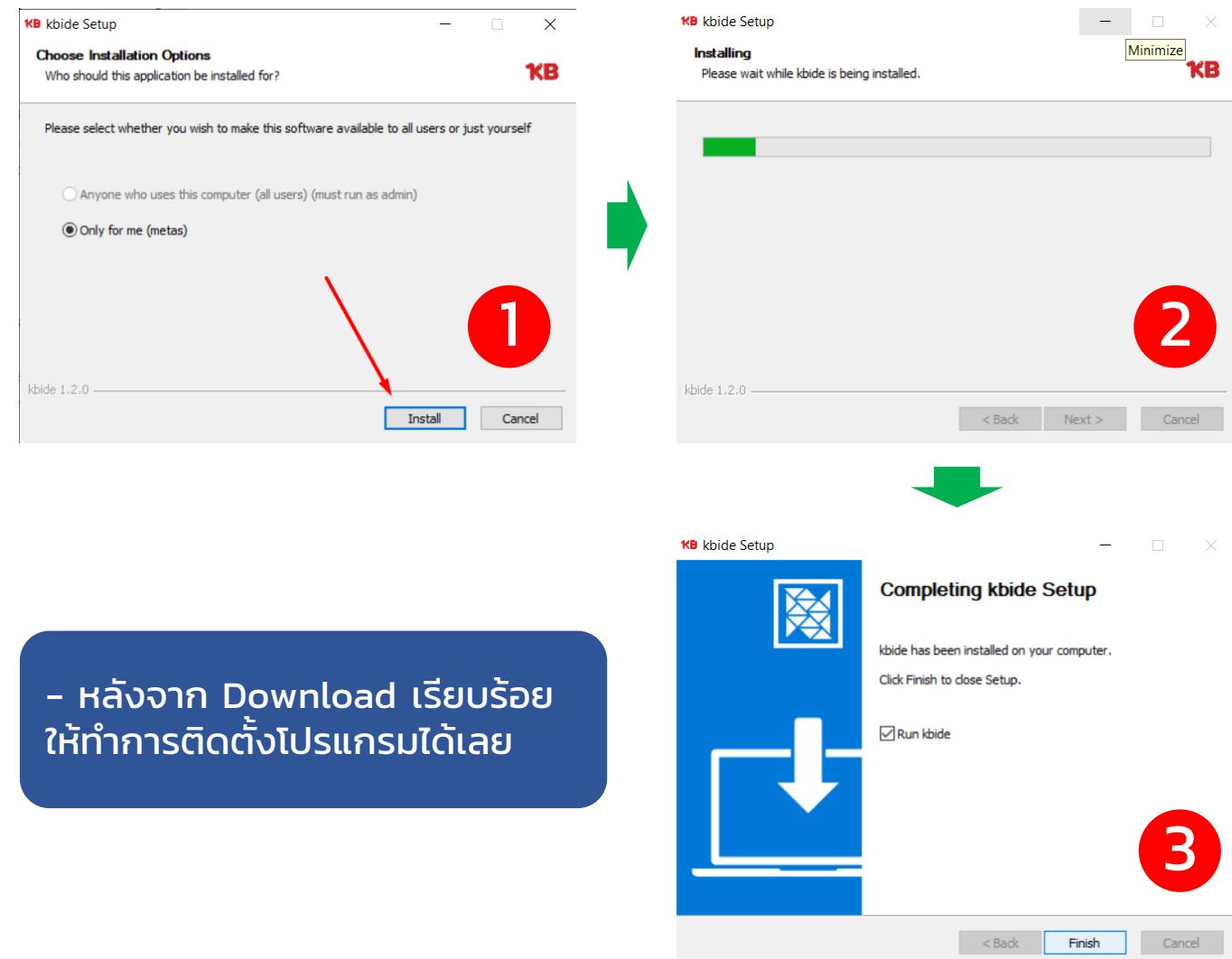
การติดตั้งโปรแกรม KB-IDE

- Fix example cannot open C/C++ sourcecode.
- Add submodule boards/openkb , kbm5stack

▼ Assets 10

	KB-IDE_Portable_v1.2.0_linux64.7z	171 MB
	KB-IDE_Portable_v1.2.0_win32.7z	177 MB
	KB-IDE_Portable_v1.2.0_win64.7z	181 MB
	KB-IDE_Setup_v1.2.0_mac.dmg	369 MB
	KB-IDE_Setup_v1.2.0_win32.exe	244 MB
	KB-IDE_Setup_v1.2.0_win64.exe	248 MB
	KB-IDE_v1.2.0_linux-arm.tar.xz	241 MB
	KB-IDE_v1.2.0_linux-arm64.tar.xz	299 MB
	Source code (zip)	
	Source code (tar.gz)	

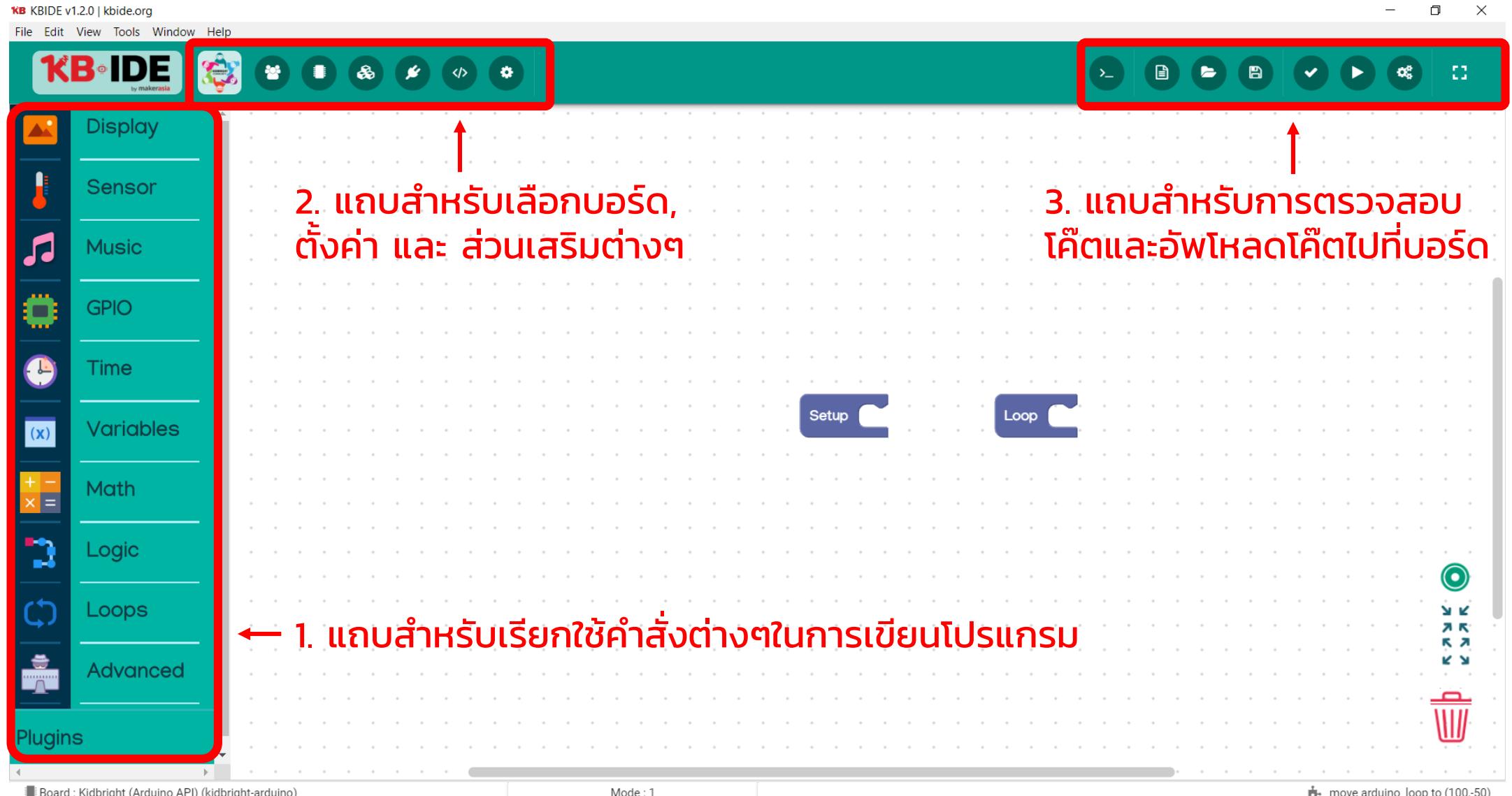
เลือก Download ให้ตรงกับระบบปฏิบัติการของเรา
เช่น Windows 10 64bit ให้เลือก KB-IDE_win64.exe



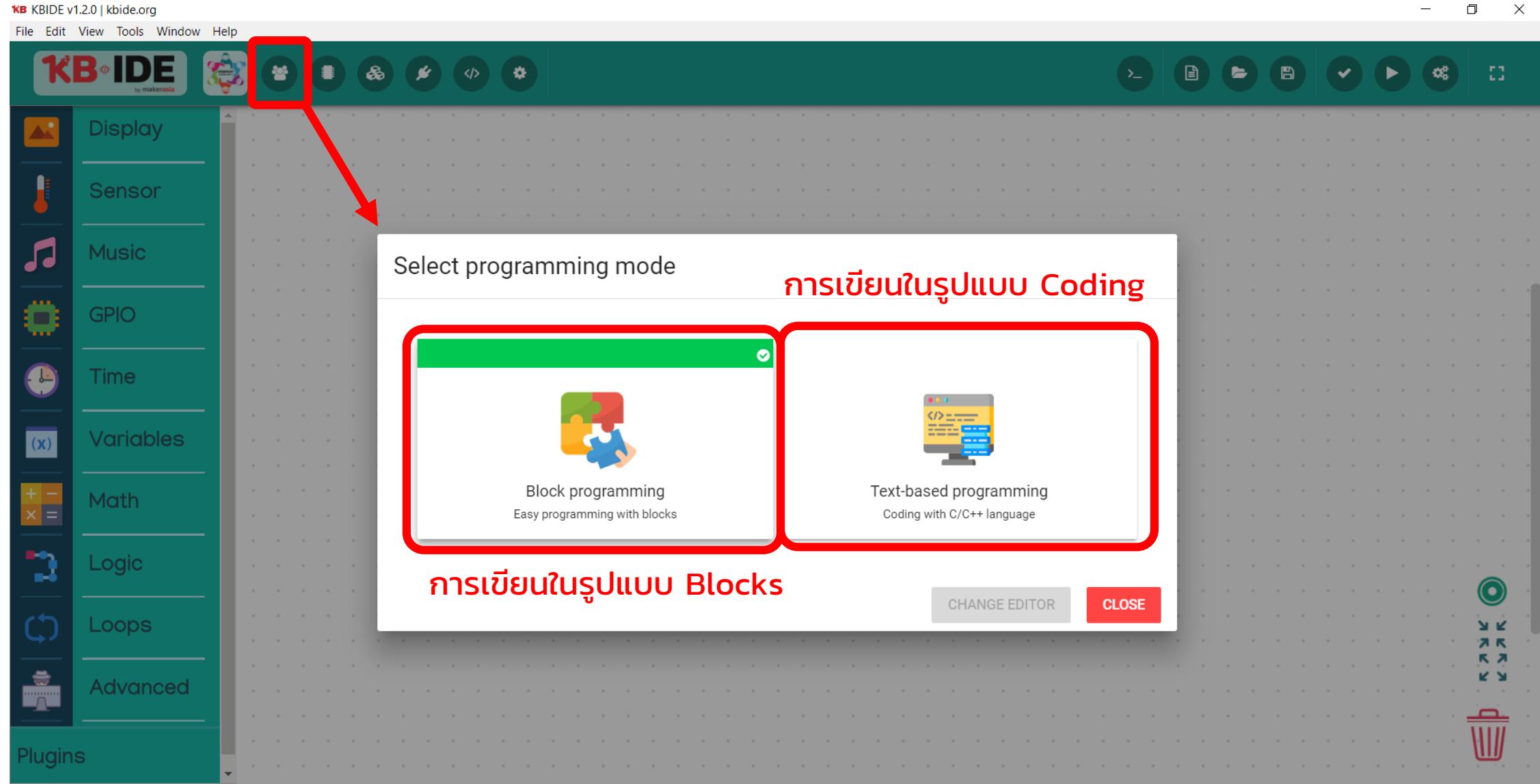
- หลังจาก Download เรียบร้อย
ให้ทำการติดตั้งโปรแกรมได้เลย

โปรแกรม KB-IDE

KB IDE
by makerasia



การเปลี่ยนรูปแบบการเขียนโปรแกรม



ติดตั้งบอร์ด Generic ESP32 Wroom

1



Board Manager

คลิกที่เมนู Board Manager

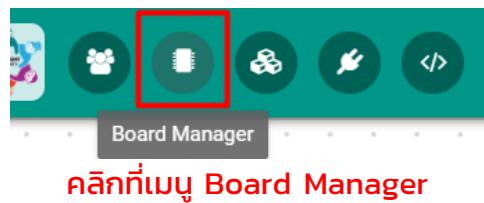
KB-IDE v1.2.0 | kbide.org
File Edit View Tools Window Help

The KB-IDE interface is shown with the 'Board Manager' window open. The 'Current board : Kidbright (Arduino API)' is displayed. The 'Generic ESP32 Wroom' board is selected. A large red arrow points from a red box containing the text 'คลิกเพื่อ Download บอร์ด' (Click to Download board) to the download icon (a green circle with a white downward arrow) for the Generic ESP32 Wroom board. The board's version is listed as 1.0.1. Other boards visible include IOXGD4, NKP_ONE, ESP32 DOIT DEVKIT V2, Generic ESP32 Wrover, Node32-Lite, and TTGO-T1 V1.3. The bottom status bar shows 'Board : Kidbright (Arduino API) (kbright-arduino)' and 'Mode : 1'.

คลิกเพื่อ Download บอร์ด

การเลือกบอร์ดที่จะใช้เขียนโปรแกรม

1



KB KBIDE v1.2.0 | kbide.org
File Edit View Tools Window Help

KB IDE by makes

GPIO Time Variables Math Logic Loops Advanced Plugins

Current board : Generic ESP32 Wroom

Board name

Arduino MEGA2560 Arduino NANO Arduino Uno R3 ESP32 DOIT DEVKIT V2

ESP32-my Doit.

Generic ESP32 Wroom IPST-WiFi IOXESP32 NX-M5Stack

ESP32-Wroom development board (Compatible with all ESP32)

Mode : 1

Board : Generic ESP32 Wroom (esp32-wroom)

POWERED by

move arduino_loop to (100,50)

A screenshot of the KBIDE software interface showing the 'Board Manager'. The 'Generic ESP32 Wroom' board is selected and highlighted with a green checkmark. A large blue arrow points from the 'Generic ESP32 Wroom' board towards the 'ESP32 DOIT DEVKIT V2' board, which is also highlighted with a red border. Both boards are version 1.1.0.

เลือกบอร์ดเป็น

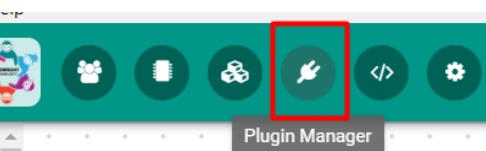
- Generic ESP32 Wroom
- ESP32 DOIT DEVKIT V2

อย่างใดอย่างหนึ่ง

****แนะนำ**
Generic ESP32 Wroom

การติดตั้ง Plugin

1



คลิกที่เมนู Plugin Manager

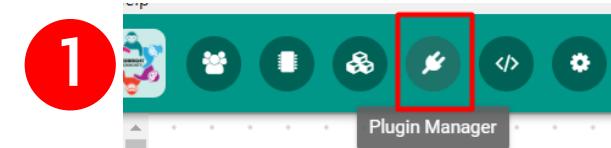
ในช่องค้นหาให้พิมพ่า "Servo" แล้วกด Enter จะแสดง Plugin ขึ้นมา

คลิกเพื่อ Download Servo Plugin

Plugin ที่จำเป็น

- Blynk
- Servo
- L298

การติดตั้ง Plugin (ต่อ)



คลิกที่เมนู Plugin Manager

Plugin Manager

plugin name

Installed

- Blynk** [v1.0.0 by Volodymyr Shymansky] [block by Comdet] [git]
Build a smartphone app for your project in minutes. Blynk allows creating IoT solutions easily.
- C/C++ Code** [v1.0.0 by Comdet] [git]
KBIDE Plugin for add C/C++ code to block
- Servo motor for ESP32** [v1.0.7 by original by Jaroslav Paral | migrate by Comdet] [git]
Generate RC servo signal on a selected pin.
- I298** [v1.0.0 by cmmakerclub] [git]
I298 plugin for KB-IDE

Online available

- IFTTT Webhook** [v1.0.0 by John Romkey | blocks by Comdet] [git]
Library for triggering IFTTT web hooks, sending up to three values. Does not use Arduino String library.

CLOSE

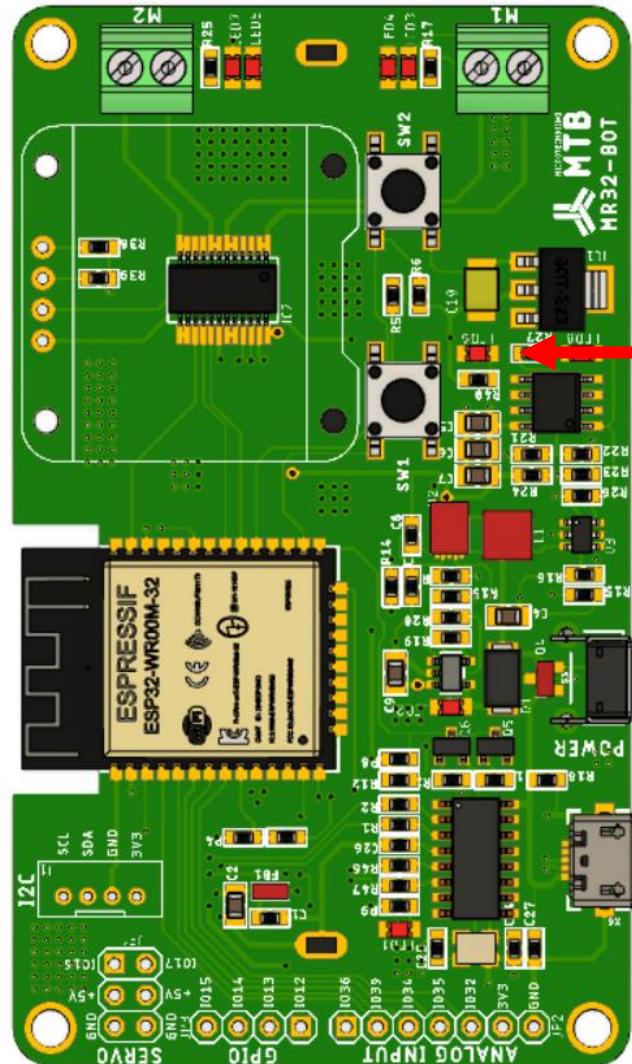
เมื่อติดตั้ง Plugin สำเร็จ
ปรากฏ Plugin ที่ติดตั้งแล้ว
ในหัวข้อ Installed

สรุปขั้นตอนการติดตั้งและตั้งค่าต่างๆ

1. ติดตั้งโปรแกรม **KB-IDE** สามารถดาวน์โหลดได้จาก <https://kbide.org/>
 2. ติดตั้งบอร์ด **Generic ESP32 Wroom**
 3. เลือกบอร์ดที่เราจะใช้ในการเขียนโปรแกรมในที่นี่คือ **Generic ESP32 Wroom**
 4. ติดตั้ง **Plugin** กี่จำเป็นต้องใช้
 - **Blynk**
 - **Servo motor for ESP32**
 - **L298**
-

**** หากทำการติดตั้งครบแล้วมาเริ่มเขียนโปรแกรมกัน ****

การเชื่อมต่อ MR32-BOT เพื่ออัพโหลดโปรแกรม

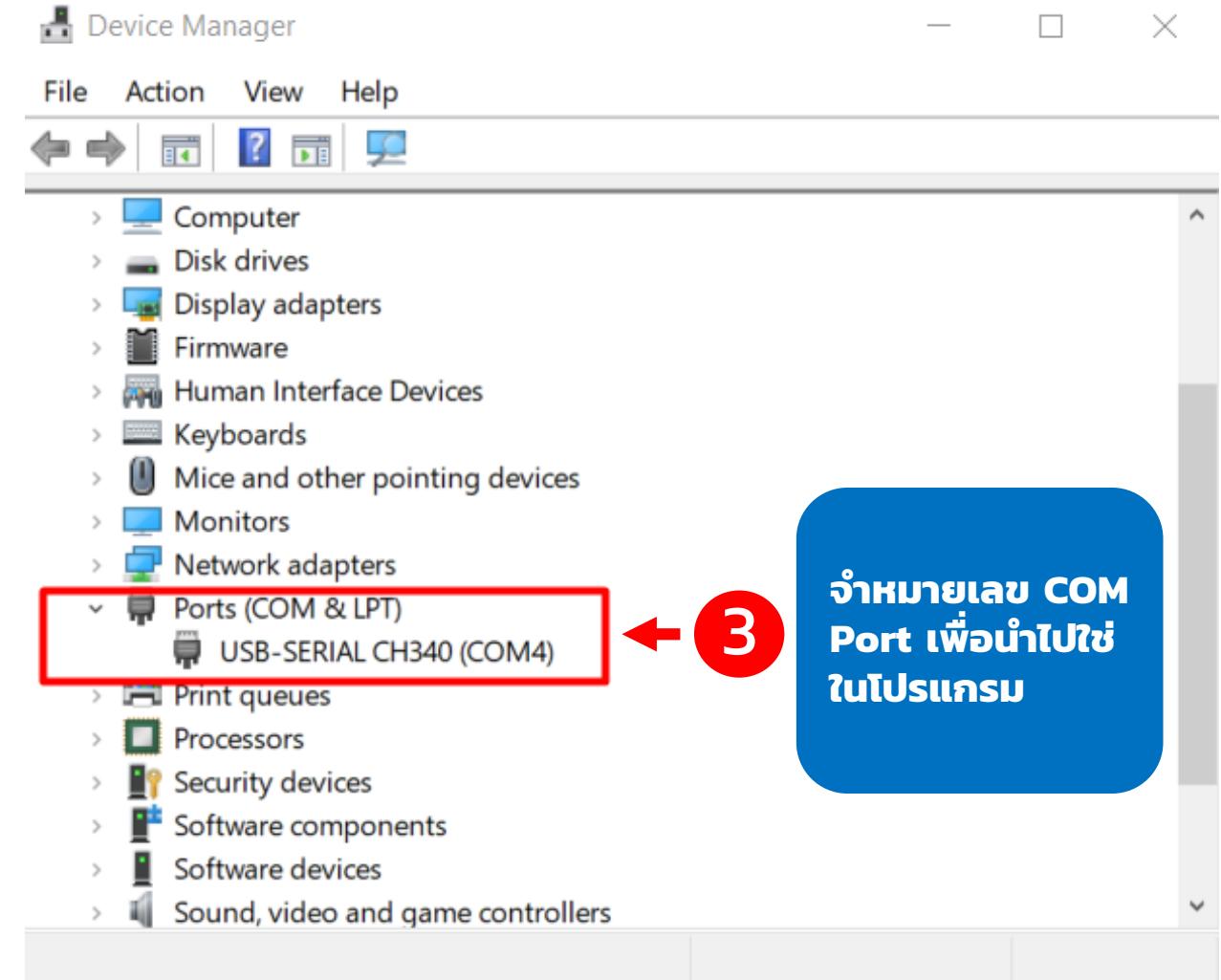
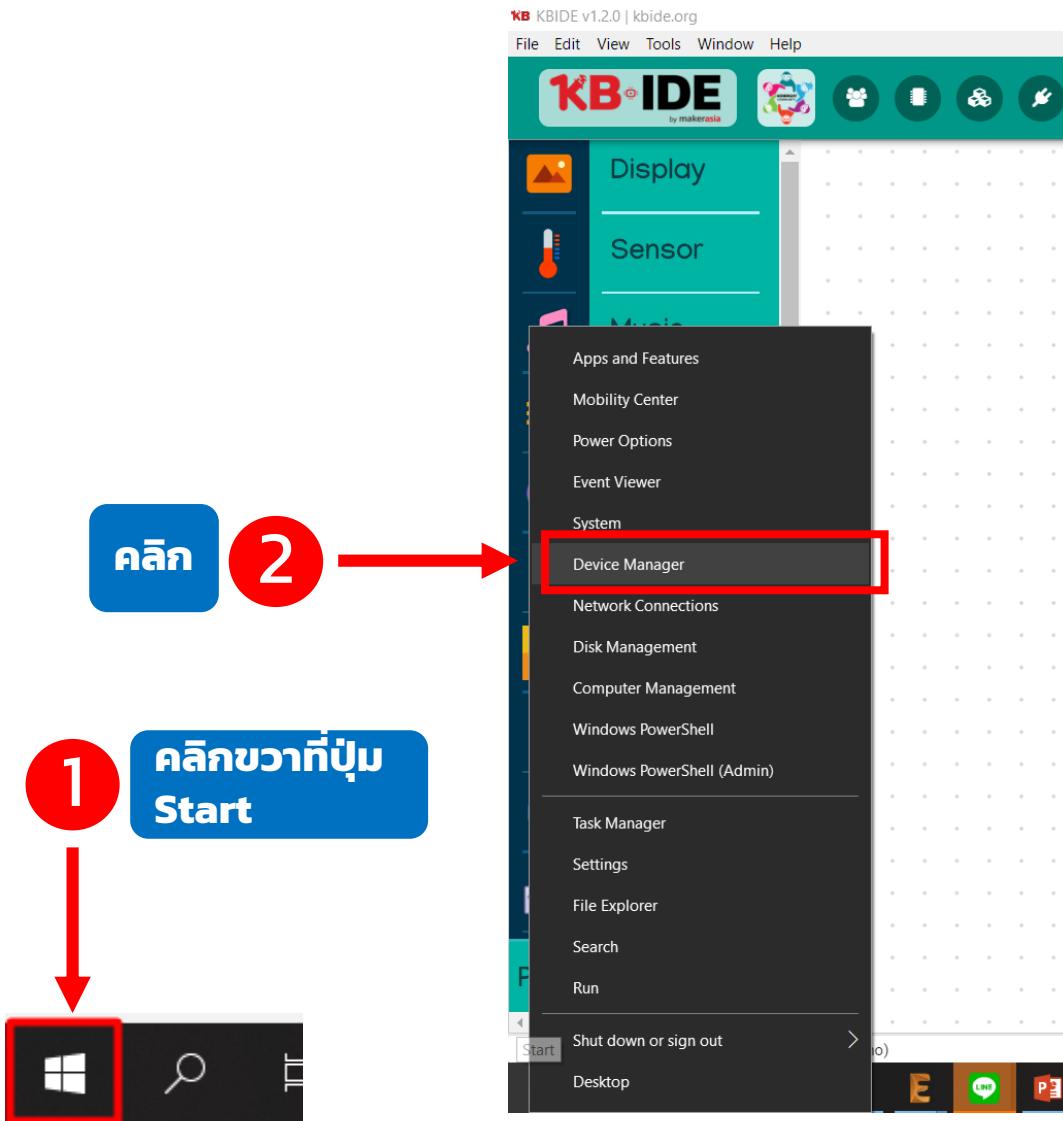


3 LED สีฟ้าแสดงสถานะการทำงานของบอร์ด

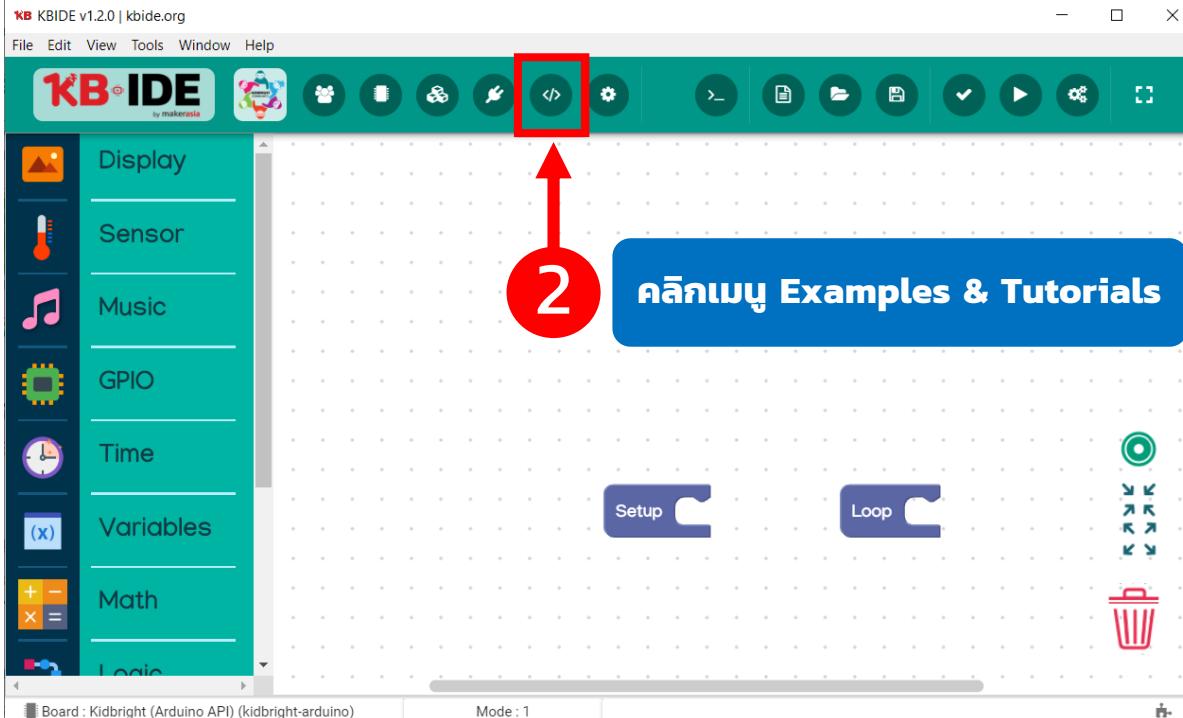
2 กดสวิตซ์ 1 ครั้งให้ LED สีฟ้าสว่างเพื่อเปิด
การทำงานบอร์ด (กดอีกครั้งเพื่อปิดบอร์ด)

1 เสียบสาย Micro USB กับคอมพิวเตอร์จะเห็น
LED สีแดงที่บอร์ดสว่างขึ้นมา

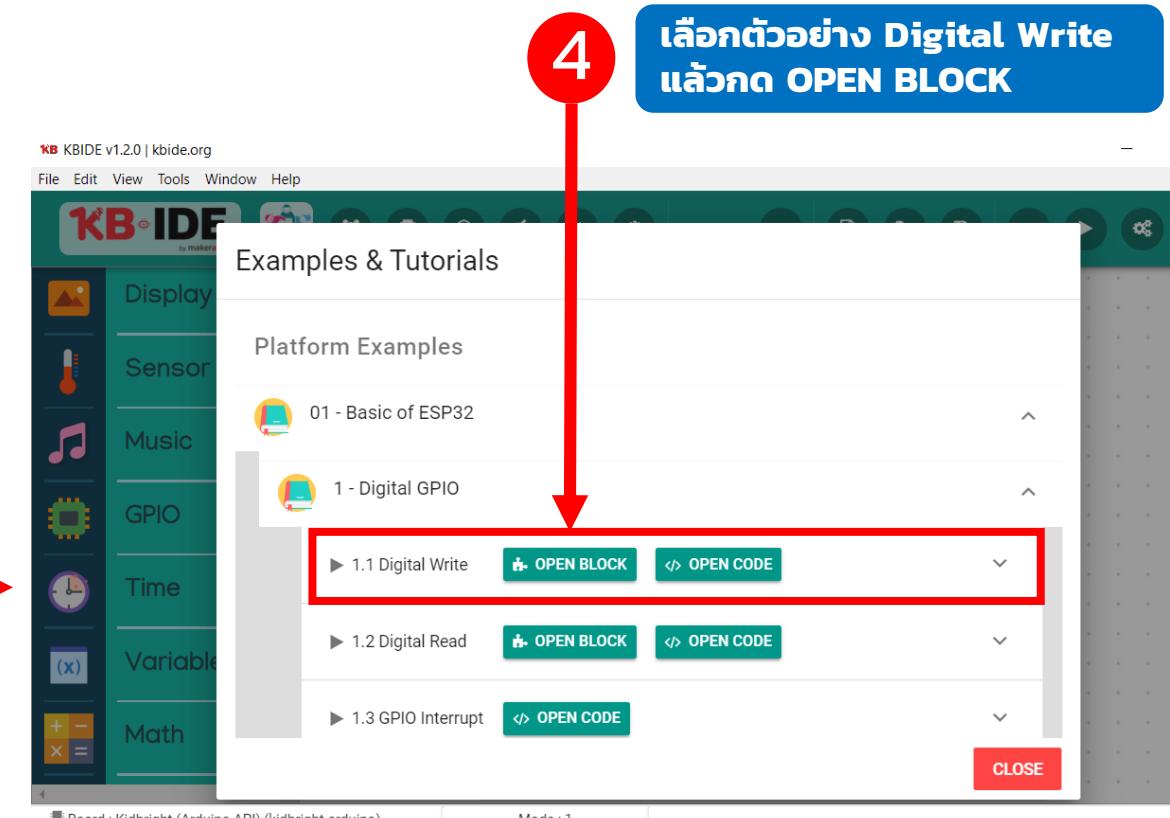
การเชื่อมต่อ MR32-BOT เพื่ออัพโหลดโปรแกรม(ต่อ)



ทดลองอัพโหลดโปรแกรมลง MR32-BOT

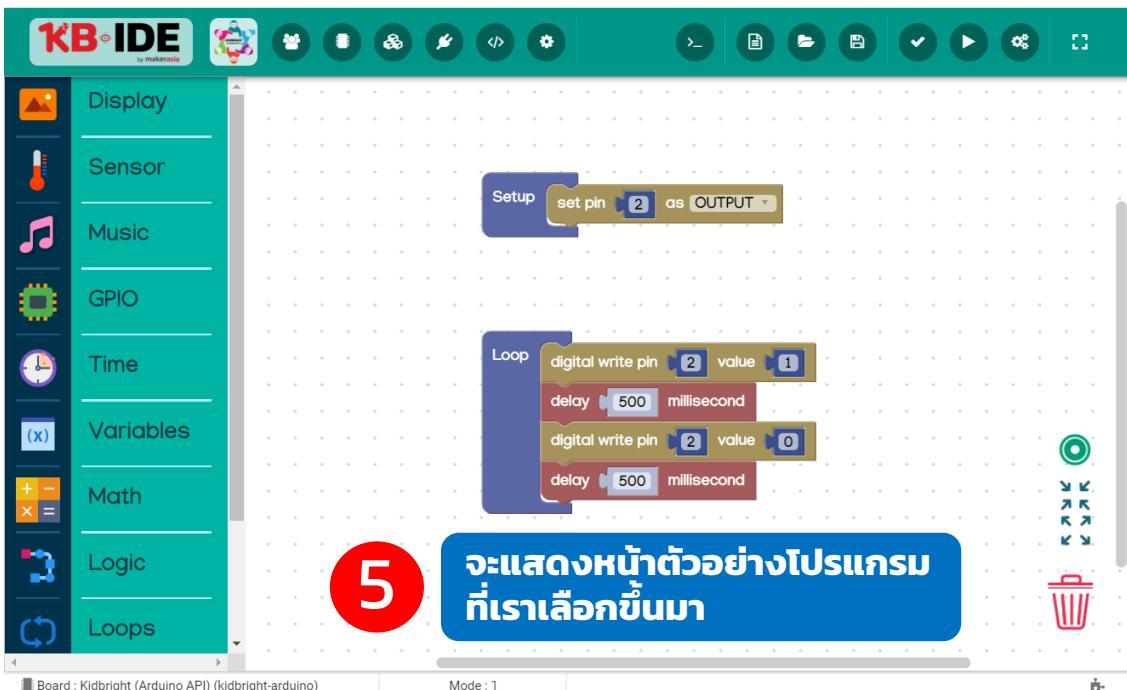


1
เปิดโปรแกรม KB-IDE ขึ้นมา



3
จะแสดงหน้าตัวอย่างโปรแกรม

ทดลองอัพโหลดโปรแกรมลง MR32-BOT



5

เมื่ออัพโหลดโปรแกรมสำเร็จให้สังเกตุที่บอร์ด MR32-BOT จะเกิดการกระพริบของ LED สีขาวบนบอร์ด

KB-IDE

Compile & Run

8

Finding board
Finding board using COM4 MAC fc:f5:c4:06:14:10

2 Compile the code
compiling... KB_music.cpp ok.

3 Upload program and Run

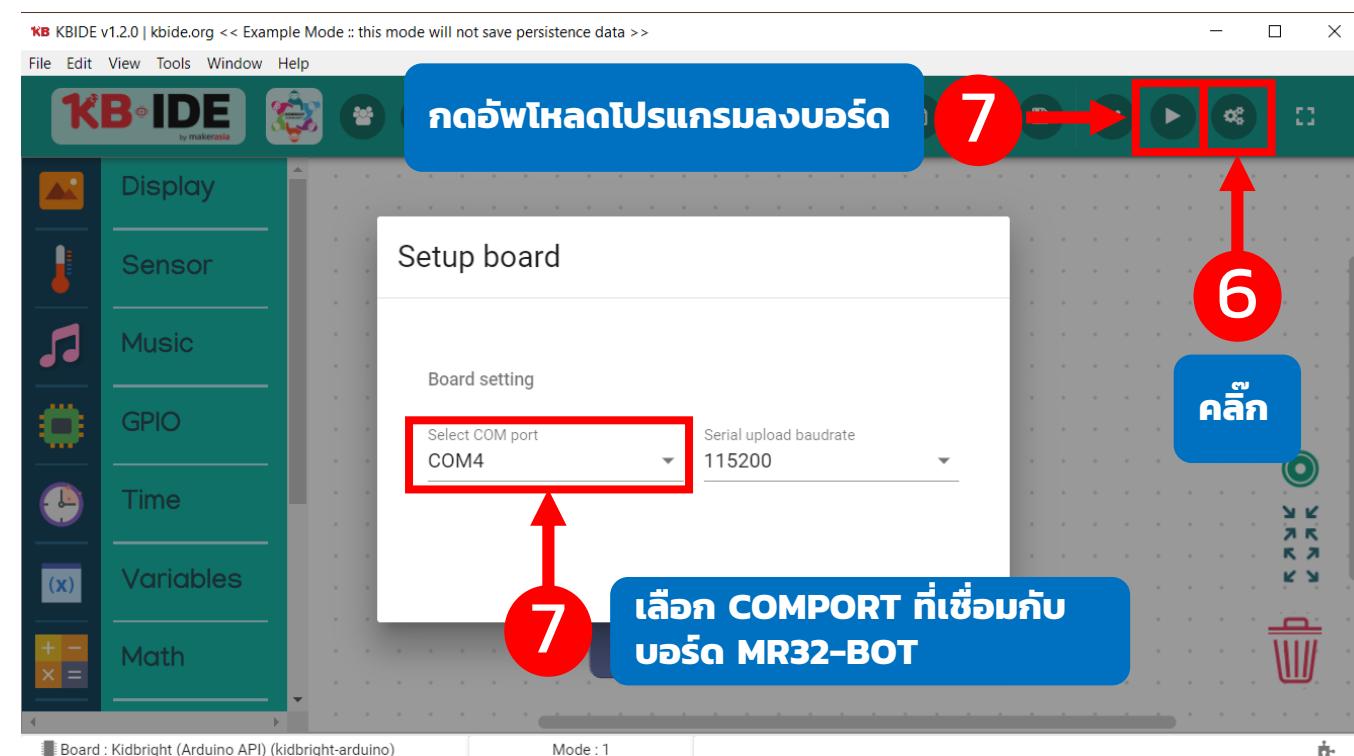
SD

Compile & Run

9



เสร็จ



POWERED by

MICROTECHBEING
MTB

ลองเขียนโปรแกรมบอร์ด MR32-BOT

LED Blink

KB-IDE by makerasia

Display
Sensor
Music
GPIO
Time
Variables
Math
Logic
Loops
Advanced
Plugins

Board : Kidbright (Arduino API) (kidbright-arduino)

Mode : 1

deselected

Setup set pin 2 as OUTPUT

Loop

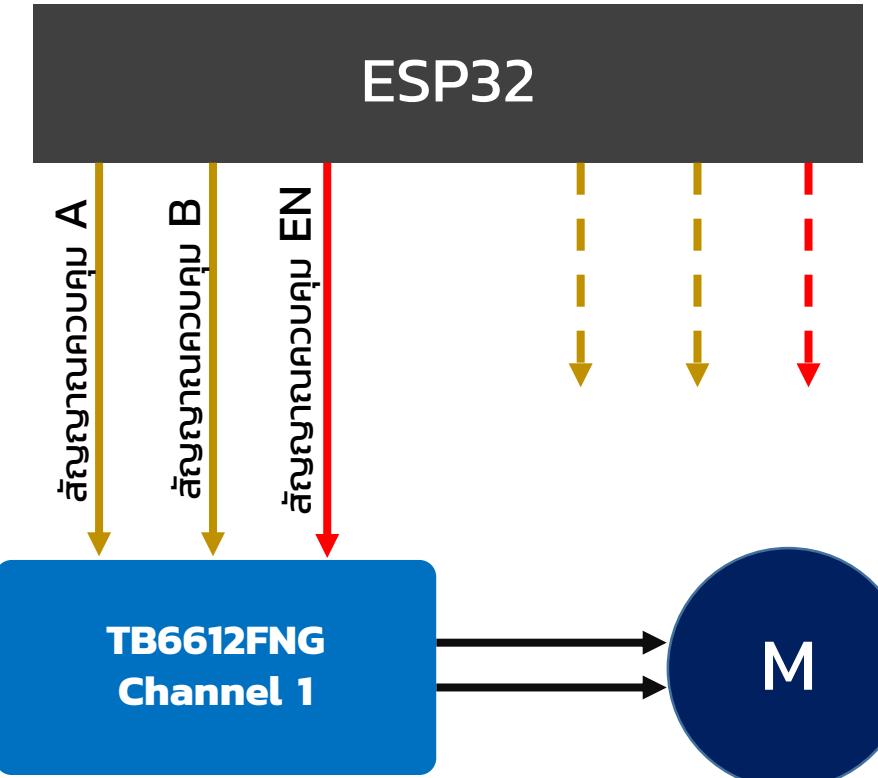
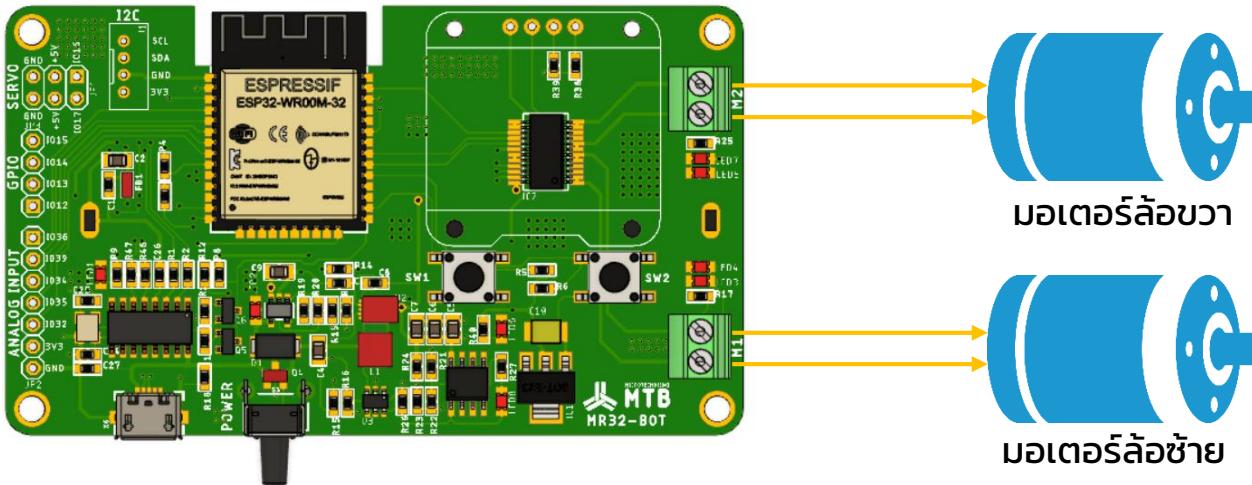
- digital write pin 2 value 1
- delay 100 millisecond
- digital write pin 2 value 0
- delay 100 millisecond

กดลงเปลี่ยนระยะเวลา
เพื่อกำหนดความเร็วการ
กระพริบของ LED
เช่น 100 จะทำให้ LED
กระพริบเร็วขึ้น

ເນື້ອຫາສໍາຮັບການສັ່ງຈານຫຸ່ນຍົນຕີ

1. ເຮັດວຽກສັ່ງຈານມອເຕວຣ໌ 1 ລ້ອ
2. ສັ່ງຈານມອເຕວຣ໌ 2 ລ້ອເພື່ອກຳໃຫ້ຫຸ່ນຍົນຕີເຄລື່ອນທີ່
3. ການປັບແຕ່ງທົກການການເຄລື່ອນທີ່
4. ການປັບຄວາມເຮົວຂອງລ້ອ (ການໃໝ່ PWM)
5. ການສັ່ງຈານມື້ອຈັບ (ຈັບ - ປລ່ອຍ)
6. ການອ່ານຄ່າຈາກ Switch
7. ສັ່ງຈານມື້ອຈັບດ້ວຍ Switch
8. ສັ່ງຈານໃຫ້ຫຸ່ນຍົນຕີເຄລື່ອນທີ່ດ້ວຍກາດ Switch
9. **** Demo ການສັ່ງຈານຫຸ່ນຍົນຕີດ້ວຍ Smart phone**

การสั่งงานมอเตอร์



A	B	EN	OUTPUT
0	1	1	CW
1	0	1	CCW
0	0	1	Stop

เรียนเขียนโปรแกรมบอร์ด MR32-BOT

DC motor (Single motor)

KB KBIDE v1.2.0 | kbide.org

File Edit View Tools Window Help

KB IDE by makerasia

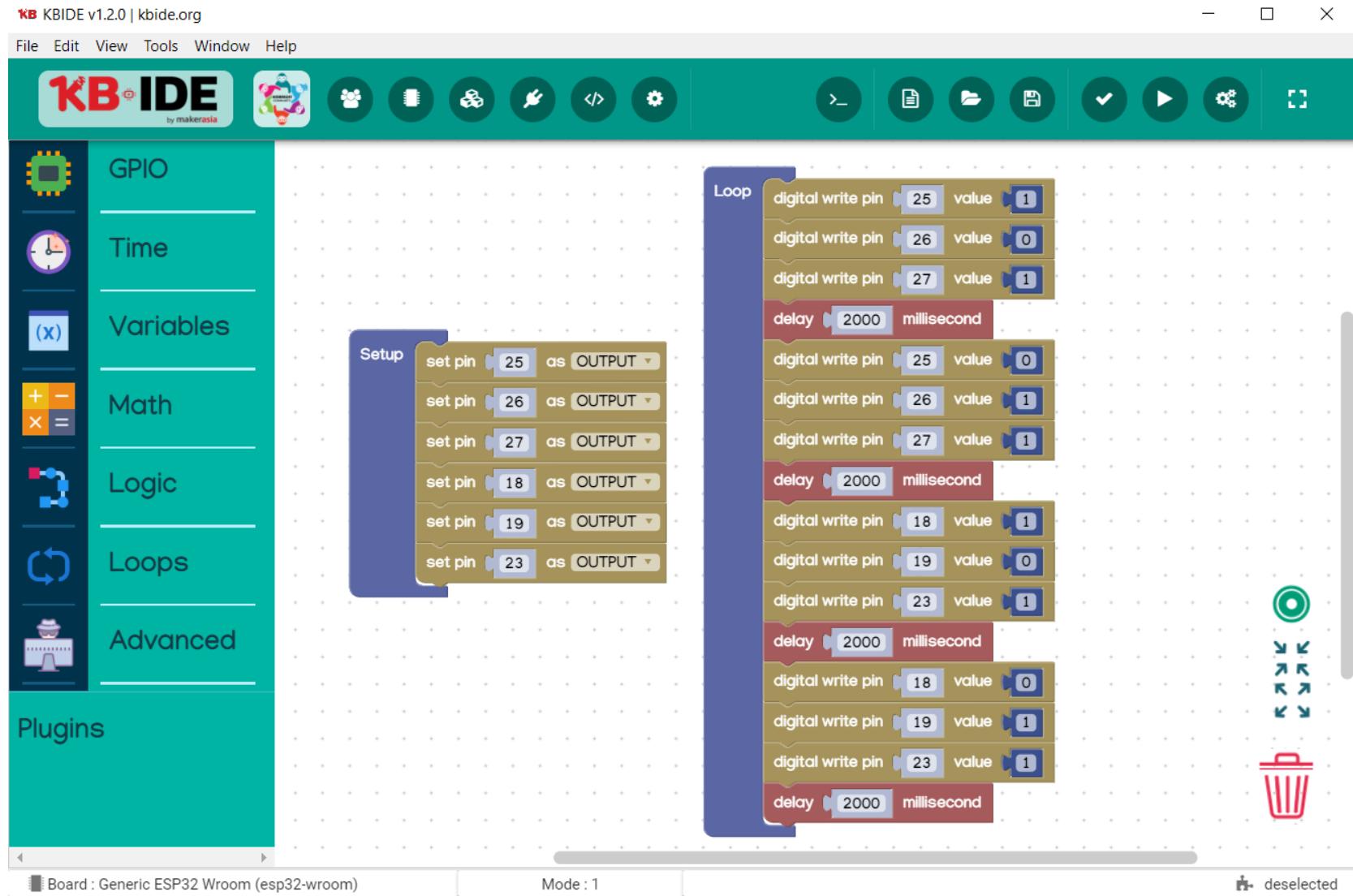
GPIO Time Variables Math Logic Loops Advanced Plugins

Board : Generic ESP32 Wroom (esp32-wroom) Mode : 1 deselected

```
Setup: set pin 25 as OUTPUT  
        set pin 26 as OUTPUT  
        set pin 27 as OUTPUT  
  
Loop: digital write pin 25 value 1  
      digital write pin 26 value 0  
      digital write pin 27 value 1  
      delay 2000 millisecond  
      digital write pin 25 value 0  
      digital write pin 26 value 1  
      digital write pin 27 value 1  
      delay 2000 millisecond
```

เรียนเขียนโปรแกรมบอร์ด MR32-BOT

DC motor (Dual motor)



NOTE !

DC
MOTOR1

A
B
PWM

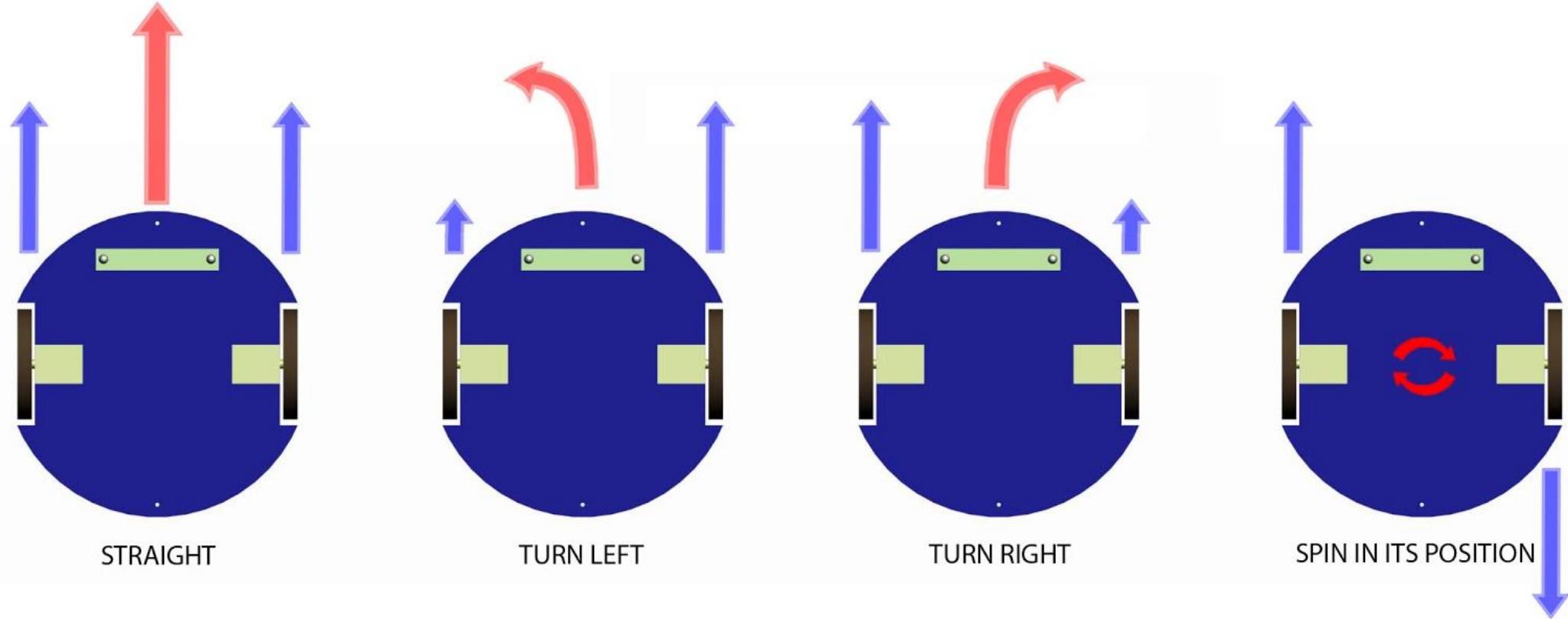
GPIO25
GPIO26
GPIO27

DC
MOTOR1

A
B
PWM

GPIO18
GPIO19
GPIO23

การเคลื่อนที่ของหุ่นยนต์



เรียนเขียนโปรแกรมบอร์ด MR32-BOT

DC motor (Dual motor)

The screenshot shows the KB-IDE v1.2.0 interface with the following components:

- Top Bar:** File, Edit, View, Tools, Window, Help.
- Toolbar:** Includes icons for file operations, project management, and execution.
- Left Sidebar (Categories):** GPIO, Time, Variables, Math, Logic, Loops, Advanced, Functions, Text, WiFi.
- Setup Block:** A stack of six "set pin [pin] as [OUTPUT]" blocks for pins 25, 26, 27, 18, 19, and 23.
- Loop Block:** A stack of four pairs of movement blocks: L_Forward, R_Forward, delay 2000 ms, L_Backward, R_Backward, and another delay 2000 ms.
- Scratch-like Script Area:** Contains two main sections for left and right motors, each with three digital write pin blocks (e.g., digital write pin 25 value 1).
- Right Panel (Component Labels):**
 - DC MOTOR1:** A green circle icon with arrows pointing up, down, left, and right.
 - DC MOTOR2:** A red trash can icon.
 - GPIO25:** A grey rectangle labeled "A".
 - GPIO26:** A grey rectangle labeled "B".
 - GPIO27:** A grey rectangle labeled "PWM".
 - GPIO18:** A blue rectangle labeled "A".
 - GPIO19:** A blue rectangle labeled "B".
 - GPIO23:** A blue rectangle labeled "PWM".
- Bottom Status Bar:** Board: Generic ESP32 Wroom (esp32-wroom), Mode: 1, deselected.

NOTE !

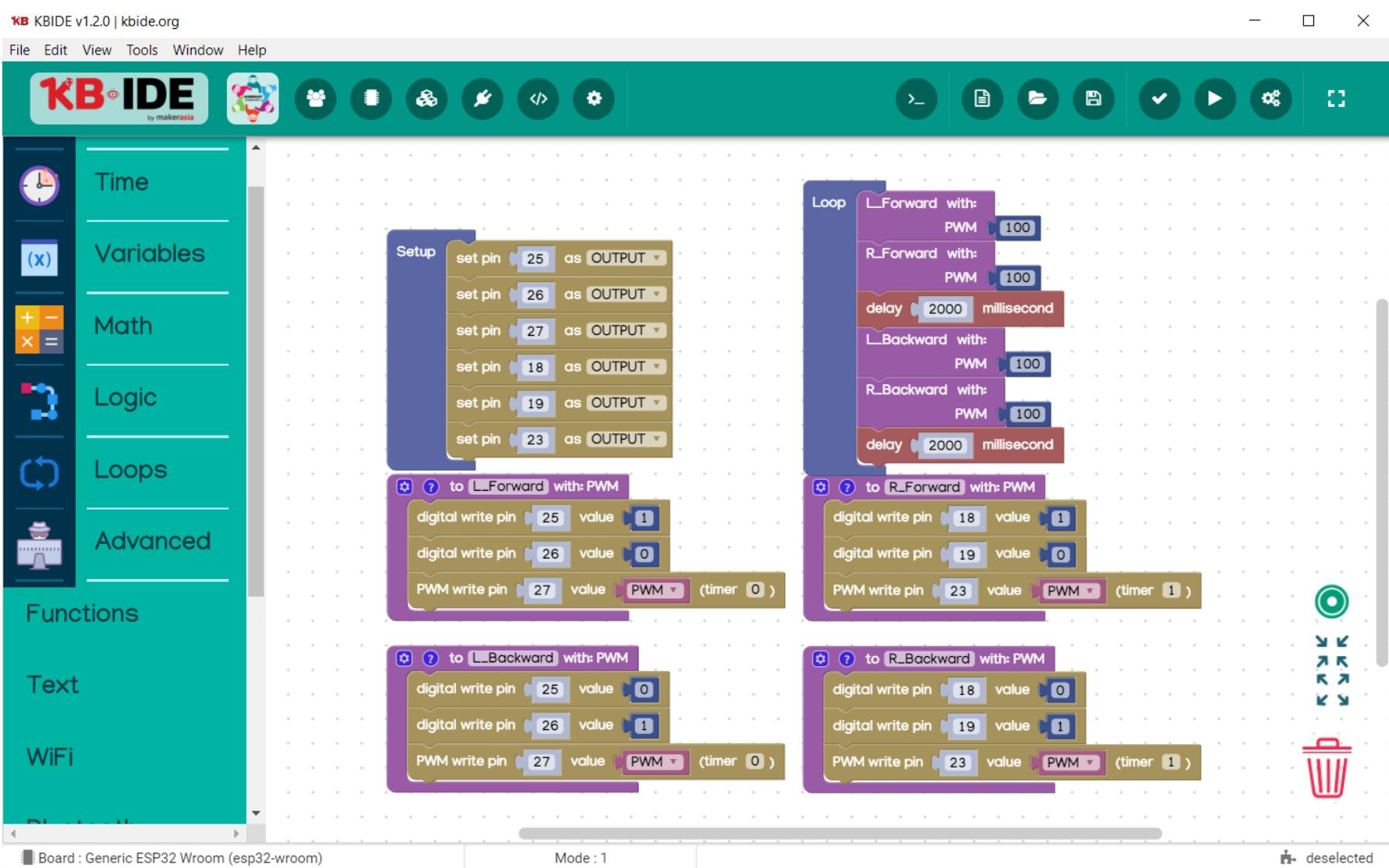
ทดลองเขียนโปรแกรมแบบ Blocky บน KB-IDE ดังนี้

1. ปรับแก้โปรแกรมจากตัวอย่างก่อนหน้านี้เรื่อง **DC Motor** ให้ทดลองเขียนโปรแกรมให้หุ่นยนต์สามารถเคลื่อนที่ได้ตามเงื่อนไขต่อไปนี้

- หุ่นยนต์ เคลื่อนที่ไปข้างหน้าในระยะเวลา 3000ms
- หุ่นยนต์ เคลื่อนที่ถอยหลังในระยะเวลา 3000ms
- หุ่นยนต์ เลี้ยวซ้ายในระยะเวลา 500ms
- หุ่นยนต์ เลี้ยวขวาในระยะเวลา 500ms
- หุ่นยนต์ หยุดเคลื่อนที่

เรียนเขียนโปรแกรมบอร์ด MR32-BOT

DC motor (PWM)



NOTE !

DC
MOTOR1

A B PWM
GPIO25 GPIO26 GPIO27

DC
MOTOR1

A B PWM
GPIO18 GPIO19 GPIO23

เรียนเขียนโปรแกรมบอร์ด MR32-BOT

RC Servo motor

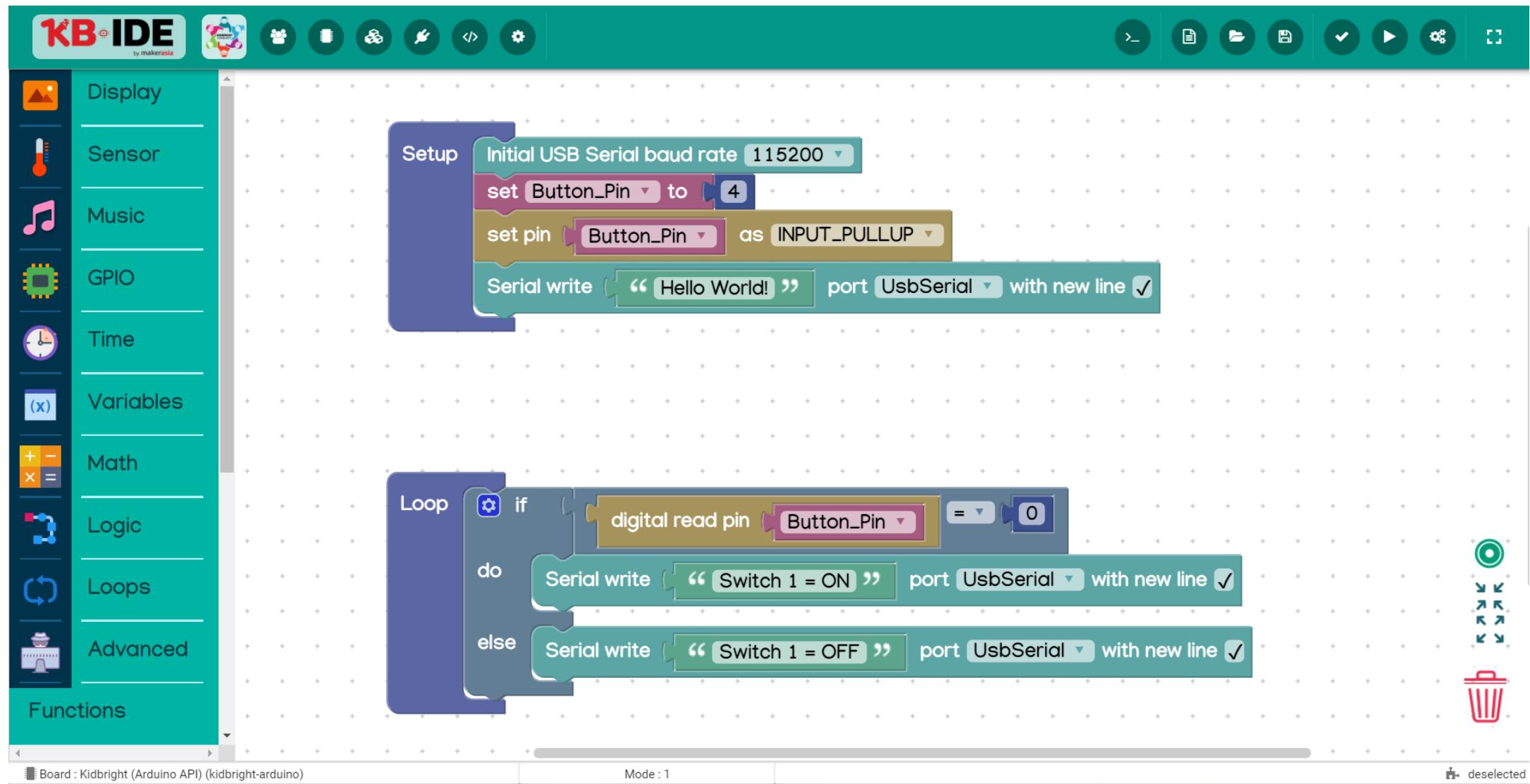
The screenshot shows the KB-IDE interface for programming the MR32-BOT. The left sidebar contains categories: Display, Sensor, Music, GPIO, Time, Variables, Math, Logic, Loops, Advanced, and Functions. The main workspace displays a Scratch-like script:

- Setup:** Servo1 attach pin 16
- Loop:** count with *i* from 0 to 90 by 1
 - do:
 - Servo1 write angle *i* degree
 - delay 50 millisecond
- Loop:** count with *i* from 90 to 0 by -1
 - do:
 - Servo1 write angle *i* degree
 - delay 50 millisecond

The bottom status bar indicates: Board : Kidbright (Arduino API) (kidbright-arduino), Mode : 1, and move arduino_loop to (-63,-38).

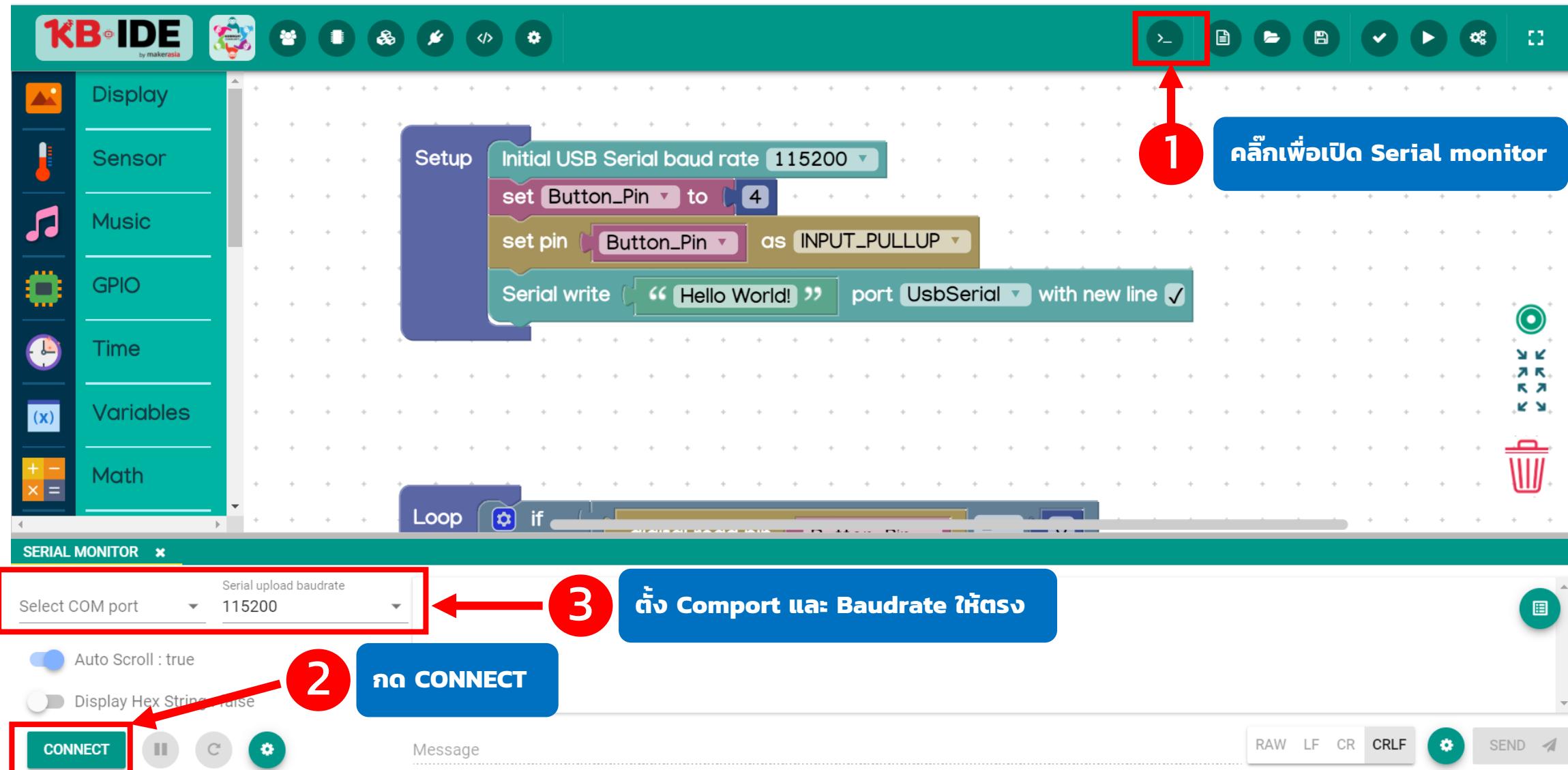
ເຮັດເຂົ້າໂປຣແກຣມບວດ MR32-BOT

Button read serial



เรียนเขียนโปรแกรมบอร์ด MR32-BOT

Button read serial (ต่อ)



ทดลองเขียนโปรแกรมแบบ Blocks บน KB-IDE ดังนี้

ปรับแก้โปรแกรมจากตัวอย่างก่อนหน้านี้เรื่อง RC Servo และการอ่านค่า Switch ให้สามารถทำงานได้ตามเงื่อนไขต่อไปนี้

- กด Switch 1 แล้วมือจับ
- กด Switch 2 แล้วมือปล่อย

ทดลองเขียนโปรแกรมแบบ Blocks บน KB-IDE ดังนี้

1. ปรับแก้โปรแกรมจากตัวอย่างก่อนหน้านี้เรื่อง **DC motor** และ **Switch** ให้สามารถทำงานได้ดังต่อไปนี้

- เมื่อกด **Switch1** = ล้อของหุ่นยนต์หมุนไปด้านหน้า
- เมื่อกด **Switch2** = ล้อของหุ่นยนต์หมุนไปด้านหลัง