ET-PGMPIC PK3

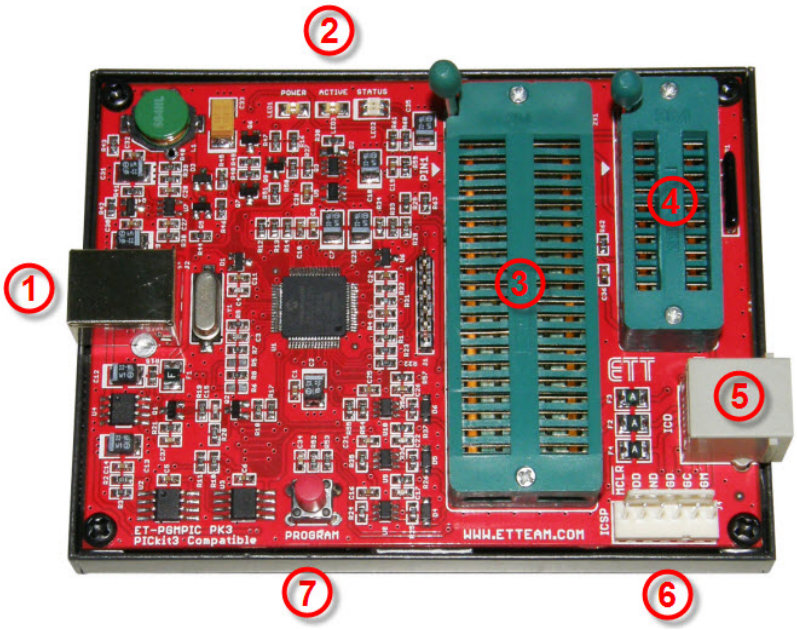
ET-PGMPIC PK3 เป็นเครื่องโปรแกรมและดีบักไมโครคอนโทรลเลอร์ PIC ซึ่งเป็นไมโครคอนโทรลเลอร์ของบริษัท Microchip โดย ET-PGMPIC PK3 นี้จะมีคุณสมบัติเทียบเท่า PICkit 3 ของบริษัท Microchip โดยสามารถโปรแกรมและดีบักไมโครคอนโทรลเลอร์ที่มีหน่วยความจำแบบ Flash Memory ได้หลากหลายเบอร์ไม่ว่าจะเป็น ไมโครคอนโทรลเลอร์ PIC10 , PIC12 , PIC16 , PIC18 , dsPIC30 , dsPIC33 , PIC24 และ PIC32

คุณสมบัติของ ET-PGMPIC PK3

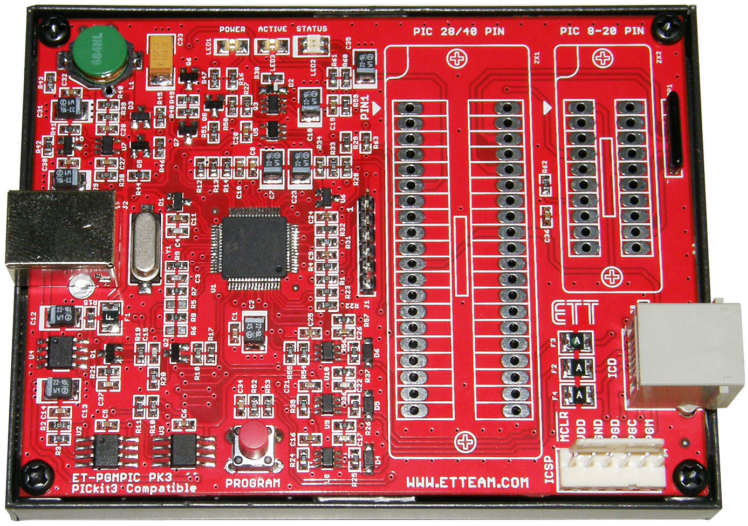
- การเชื่อมต่อเป็น USB (Full Speed 12 Mbits/s)
- มีวงจรป้องกันไฟเกินและป้องกันการลัดวงจร
- สามารถโปรแกรมและดีบักไมโครคอนโทรลเลอร์ PIC และ dsPIC ได้
- ใช้งานร่วมกับโปรแกรม MPLAB IDE และโปรแกรม PICkit 3 Programmer (ดาวน์โหลดฟรี)
- สามารถอัพเกรด Firmware ผ่านทางคอมพิวเตอร์
- สามารถใช้งานร่วมกับบอร์ดเป้าหมายที่มีไฟเลี้ยงตั้งแต่ 2.0 - 6.0 V ได้
- มี LED แสดงผลการทำงาน POWER, ACTIVE, STATUS
- สามารถอ่านและเขียนพื้นที่ในหน่วยความจำและส่วนของพื้นที่อีอีพรอมได้
- คุณสมบัติเทียบเท่า PICkit 3 ของบริษัท Microchip
- มีหน่วยความจำแบบ Flash 512 Kbyte สำหรับการใช้งานโหมด Programmer-To-Go
- สามารถโปรแกรมผ่านทาง TEXTOOL 40 PIN และ 20 PIN ได้ (เฉพาะรุ่น PLUS)
- ใช้แจ๊กแบบโมดูล่าร์ 6 ขา และจัดเรียงขาตามมาตรฐานแจ๊ก ICD ของ Microchip ทำให้สามารถใช้งานกับบอร์ดของ Microchip หรือบอร์ดที่แจ็ก ICD ได้ทันที

1. โครงสร้างของบอร์ด ET-PGMPIC PK3

ET-PGMPIC PK3 จะมีอยู่ 2 รุ่นคือ ET-PGMPIC PK3 และ ET-PGMPIC PK3 PLUS โดยคุณสมบัติจะเหมือนกันทุกอย่าง จะแตกต่างกันเพียง ET-PGMPIC PK3 PLUS จะติดตั้ง TEXTOOL 40 PIN และ 20 PIN มาด้วย ดังรูป



รูปที่ 1-1 แสดงโครงสร้างของบอร์ด ET-PGMPIC PK3 PLUS



รูปที่ 1-2 แสดงโครงสร้างของบอร์ด ET-PGMPIC PK3

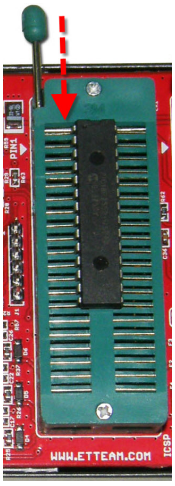
2. โครงสร้างของบอร์ดสามารถอธิบายตามหมายเลขต่าง ๆ ได้ดังนี้

หมายเลข 1 ช่องต่อ USB สำหรับเชื่อมต่อ ET-PGMPIC PK3 เข้ากับคอมพิวเตอร์

หมายเลข 2 ไฟแสดงสถานะต่างๆ คือ POWER , ACTIVE , STATUS

1. POWER (สีเขียว) แสดงสถานะว่ามีไฟเลี้ยงบอร์ด ET-PGMPIC PK3
2. ACTIVE (สีเขียว) แสดงสถานะว่าการเชื่อมต่อ ET-PGMPIC PK3 กับคอมพิวเตอร์เสร็จสมบูรณ์
3. STATUS เป็น LED สองสี จะมีสถานะคือ
   - BUSY (สีเหลือง) แสดงสถานะว่า ET-PGMPIC PK3 กำลังทำงาน เช่นกำลังโปรแกรมอยู่
   - ERROR (สีแดง) จะแสดงเมื่อ ET-PGMPIC PK3 เกิดข้อผิดพลาด

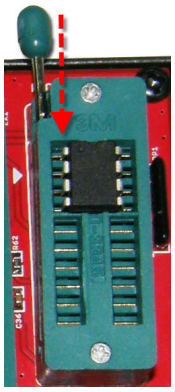
หมายเลข 3 TEXTOOL ขนาด 40 PIN รองรับไมโครคอนโทรลเลอร์ PIC ขนาด 28 ขา และ 40 ขา ซึ่งการใส่ไอซี ให้ดูตำแหน่งขา 1 ของไอซี ตรงกับขา 1 ของ TEXTOOL คือ ชิดขอบด้านบนดังรูป และจะต้องกดคันโยกเพื่อล๊อคไอซีให้แน่นก่อนใช้งาน



รูปที่ 2-1 แสดงตัวอย่างการติดตั้งไอซี 28 ขา เพื่อโปรแกรม

หมายเหตุ การโปรแกรมผ่านทาง TEXTOOL 40 PIN นี้ไม่สามารถใช้กับไมโครคอนโทรลเลอร์ แบบ DIP ตระกูล dsPIC30 , dsPIC33 , PIC24 และ PIC32 ถ้าต้องการโปรแกรมตระกูลเหล่านี้ ให้ใช้การโปรแกรมทางพอร์ต ICD หรือ ICSP แทน

หมายเลข 4 TEXTOOL ขนาด 20 PIN รองรับไมโครคอนโทรลเลอร์ PIC ตั้งแต่ 8 ขา ถึง 20 ขา ซึ่งการใส่ไอซี ให้ดูตำแหน่งขา 1 ของไอซี ตรงกับขา 1 ของ TEXTOOL คือ ชิดด้านบนดังรูป และจะต้องกดคันโยกเพื่อล๊อคไอซีให้แน่นก่อนใช้งาน



รูปที่ 2-2 แสดงตัวอย่างการติดตั้งไอซี 8 ขา เพื่อโปรแกรม

หมายเหตุ การโปรแกรมผ่านทาง TEXTOOL 40 PIN นี้ไม่สามารถใช้กับไมโครคอนโทรลเลอร์ แบบ DIP ตระกูล dsPIC30 , dsPIC33 , PIC24 และ PIC32 ถ้าต้องการโปรแกรมตระกูลเหล่านี้ ให้ใช้การโปรแกรมทางพอร์ต ICD หรือ ICSP แทน

หมายเลข 5 ช่องต่อสัญญาณ ICD เป็นพอร์ตของสัญญาณโปรแกรมที่จัดเรียงตามมาตรฐานของบริษัท Microchip ดังนั้นจึงสามารถนำไปใช้กับบอร์ดต่างๆ ที่ออกแบบโดยใช้มาตรฐานนี้ ได้ทันที ซึ่งตำแหน่งและชื่อขาสัญญาณแสดงดังรูป

ตำแหน่งขา	ชื่อสัญญาณ
1	MCLR
2	VDD
3	GND
4	PGD
5	PGC
6	PGM

รูปที่ 2-3 แสดงตำแหน่งขาสัญญาณโปรแกรม

โดยที่หน้าที่ของขาต่างๆ มีดังนี้

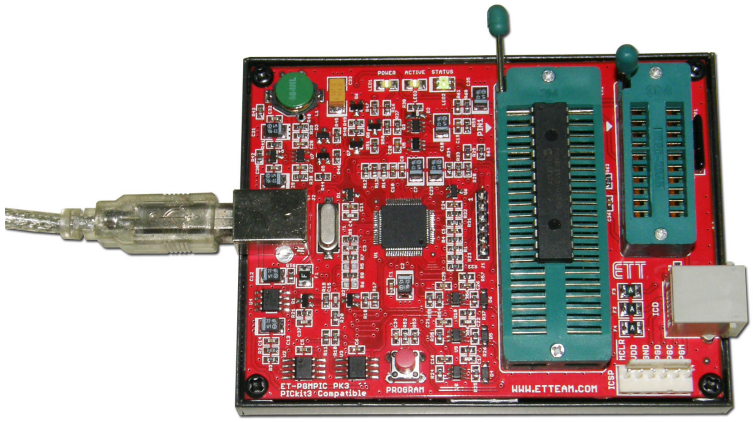
1. MCLR เป็นขาโปรแกรมแรงดันไฟสูงจะต่อกับ RESET หรือขา MCLR ของ MCU
2. VDD เป็นขาไฟเลี้ยงของ MCU ซึ่งจะต้องต่อขานี้กับไฟเลี้ยงของบอร์ดเป้าหมาย
3. GND เป็นขากราวด์ ซึ่งจะต้องต่อกับกราวด์ของบอร์ดเป้าหมาย
4. PGD เป็นขา PROGRAM DATA ซึ่งจะต้องต่อกับขา PGD ของ MCU
5. PGC เป็นขา PROGRAM CLOCK ซึ่งจะต้องต่อกับขา PGC ของ MCU
6. PGM เป็นขาสำหรับการโปรแกรมแบบแรงดันไฟต่ำ (LVP) ซึ่งปกติไม่ได้ใช้งาน

หมายเลข 6 ช่องต่อสัญญาณ ICSP เป็นพอร์ตของสัญญาณโปรแกรมเช่นเดียวกับ ICD แต่เปลี่ยนรูปแบบโดยใช้ คอนเน็คเตอร์แบบ 6 PIN แทนเพื่อให้ผู้ใช้สามารถเลือกต่อขาสัญญาณโปรแกรมได้ตามความต้องการ

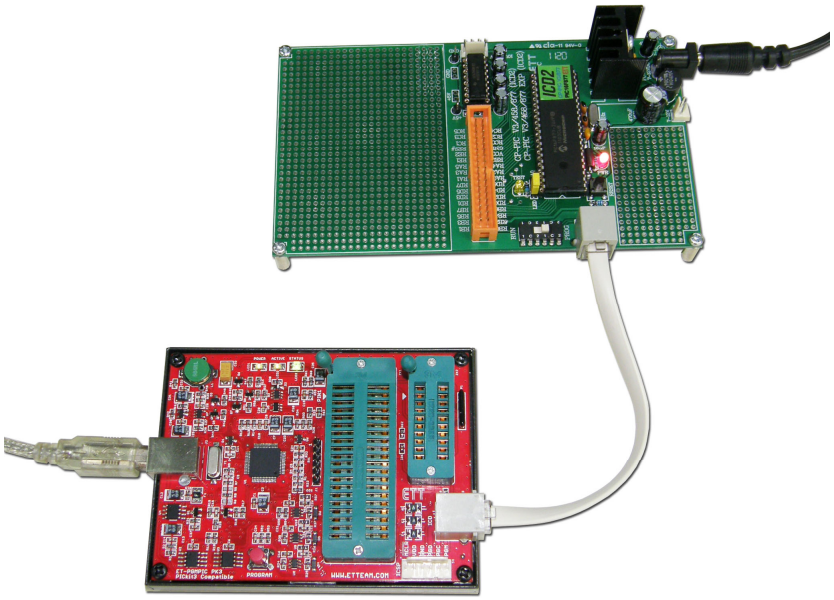
หมายเลข 7 สวิตช์สั่งโปรแกรม สวิตช์นี้จะใช้งานเมื่อมีการใช้การ ET-PGMPIC PK3 ในโหมด **Programmer-To-Go** ซึ่งการใช้ในโหมดนี้คือ การเอา Hex File ที่จะโปรแกรมไปเก็บไว้ในหน่วยความจำ Flash ของ ET-PGMPIC PK3 หลังจากนั้นก็สามารถโปรแกรมข้อมูลเข้าไมโครคอนโทรลเลอร์ PIC โดยการกดสวิตช์ PROGRAM ได้ทันทีโดยไม่ต้องเปิดโปรแกรม MPLAB IDE หรือ PICkit 3 Programmer

3. การใช้งาน ET-PGMPIC PK3

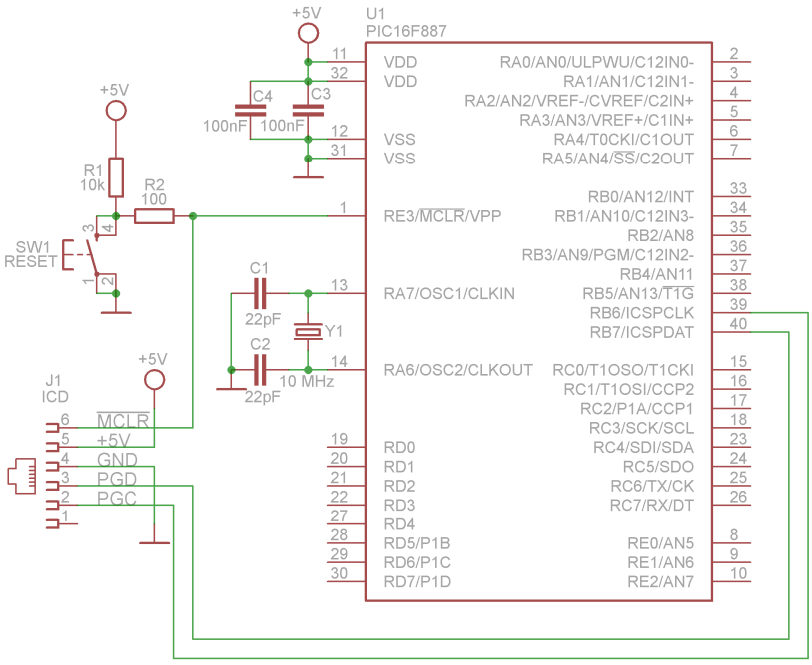
การใช้ ET-PGMPIC PK3 จะสามารถใช้ได้กับโปรแกรม PICkit 3 Programmer และโปรแกรม MPLAB IDE โดยสามารถโปรแกรมผ่านทาง TEXTOOL หรือ ผ่านทางช่องต่อสัญญาณแบบ ICD ดังรูปที่ 3-1 และรูปที่ 3-2 ส่วนตัวอย่าง การจัดวงจรโปรแกรมของไมโครคอนโทรลเลอร์ PIC แสดงดังรูปที่ 3-3



รูปที่ 3-1 แสดงการใช้งาน ET-PGMPIC PK3 ผ่านทาง TEXTOOL



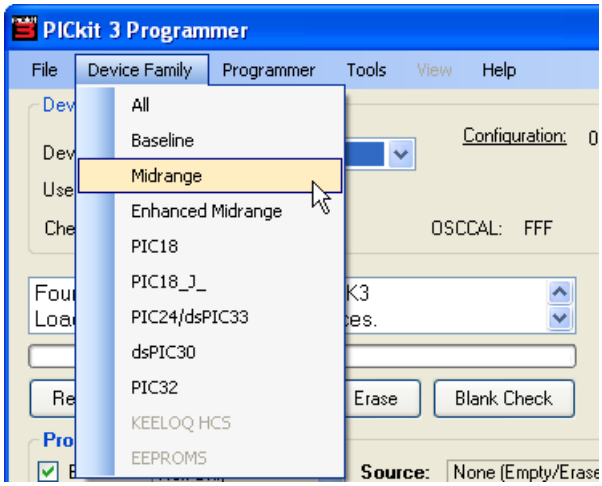
รูปที่ 3-2 แสดงการใช้งาน ET-PGMPIC PK3 ร่วมกับบอร์ดที่ใช้ ช่องต่อสัญญาณแบบ ICD



รูปที่ 3-3 แสดงตัวอย่างการจัดวงจรการโปรแกรมของไมโครคอนโทรลเลอร์ PIC

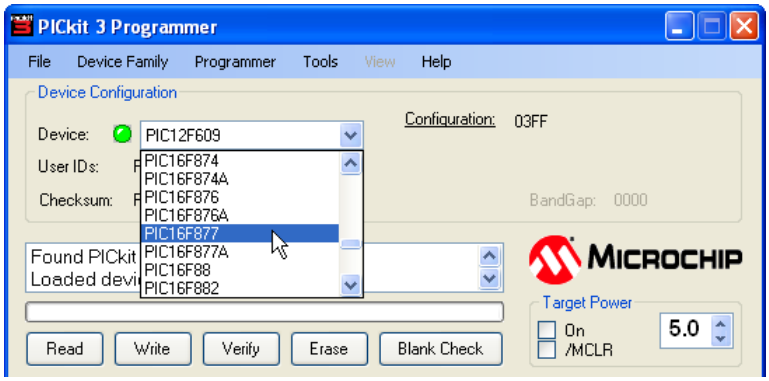
4. การใช้งาน ET-PGMPIC PK3 ร่วมกับโปรแกรม PICkit 3 Programmer

4.1 ให้ทำการติดตั้งโปรแกรม PICkit3 Programmer จากนั้นให้เปิดโปรแกรมขึ้นมา ตัวอย่างนี้จะแสดงการโปรแกรม PIC16F877 ให้ทำการเลือก **Device Family** เป็น **Midrange** ดังรูปที่ 4-1



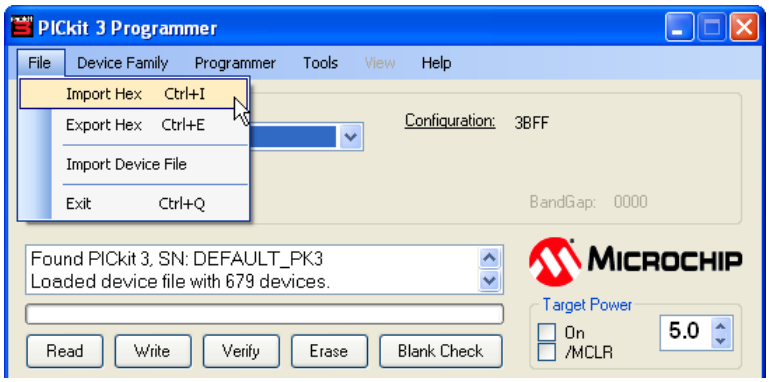
รูปที่ 4-1 แสดงการเลือก Device Family ของไมโครคอนโทรลเลอร์

4.2 เลือก Device เป็น PIC16F877 ดังรูปที่ 4-2



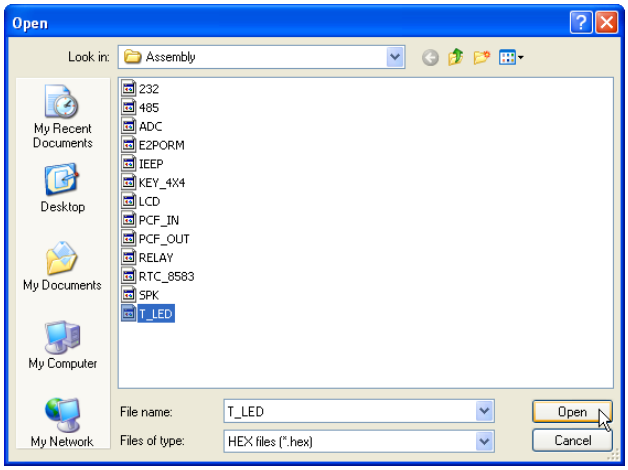
รูปที่ 4-2 แสดงการเลือกเบอร์ของไมโครคอนโทรลเลอร์

4.3 เลือกที่เมนู **File>Import Hex** เพื่อเลือก Hex File ที่จะทำการโปรแกรม ดังรูปที่ 4-3



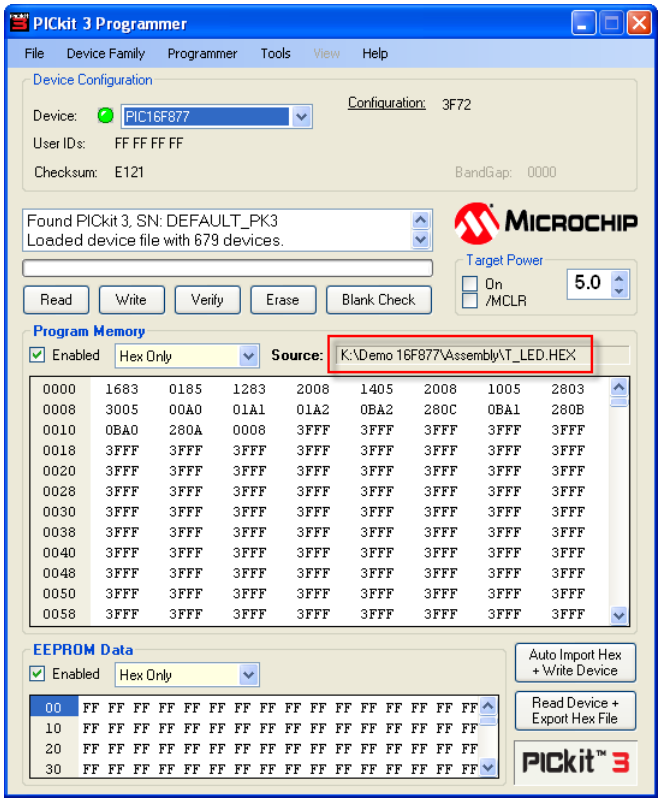
รูปที่ 4-3

4.4 เลือก Hex File ที่ต้องการจะโปรแกรมจากนั้นคลิก Open ดังรูปที่ 4-4



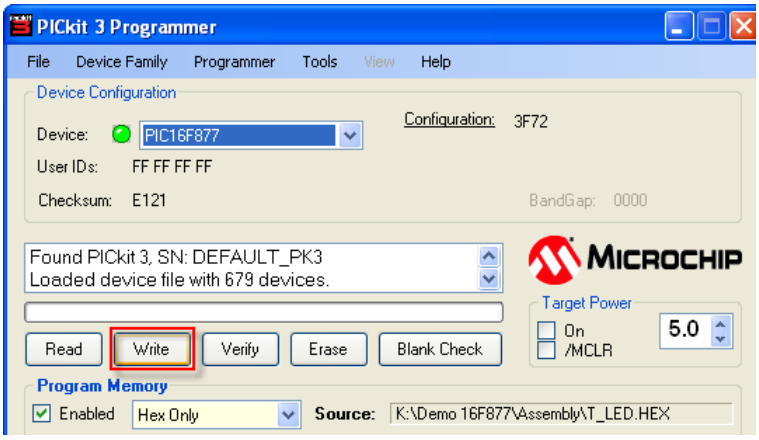
รูปที่ 4-4 แสดงการเลือก Hex File ที่ต้องการโปรแกรม

4.5 จากนั้นจะเห็นรายละเอียดต่างของ Hex File ที่จะโปรแกรมดังรูปที่ 4-5



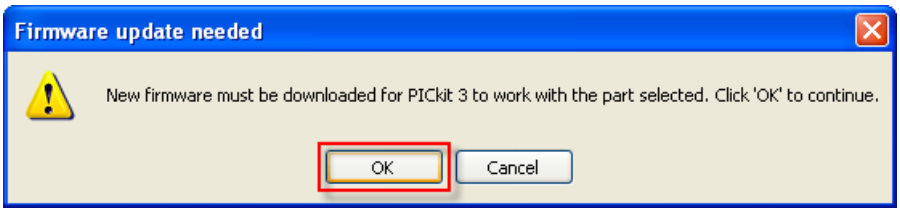
รูปที่ 4-5 แสดงรายละเอียดต่างของ Hex File

4.6 คลิกที่ปุ่ม **Write** เพื่อเริ่มการโปรแกรม ดังรูปที่ 4-6



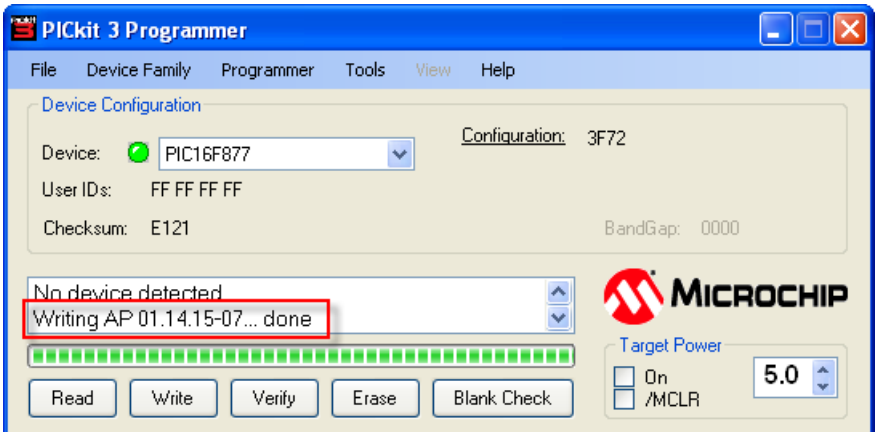
รูปที่ 4-6

4.7 ในการกรณีที่ก่อนหน้า ET-PGMPIC PK3 ผ่านการใช้งานกับไมโครคอนโทรลเลอร์ตระกูลอื่นมาที่ไม่ใช่ Midrange จะปรากฏข้อความว่าต้องการอัพเดท Firmware ดังรูปที่ 4-7 ให้เลือก **OK** เพื่อทำการอัพเดท Firmware



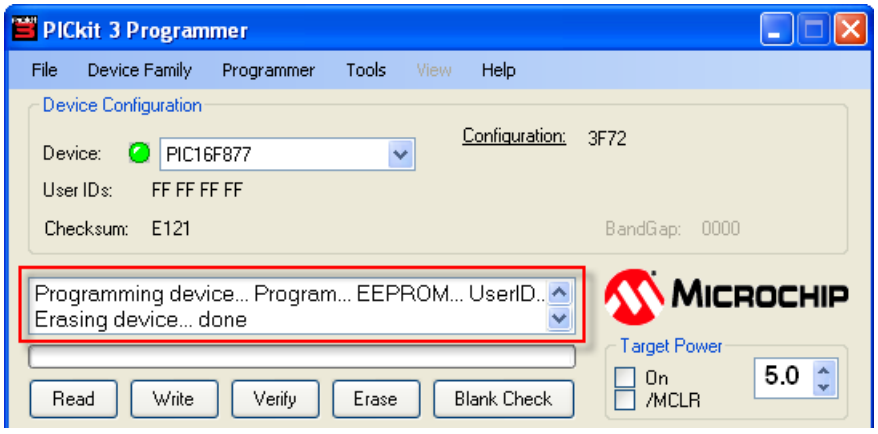
รูปที่ 4-7 **แสดงการอัพเดท** Firmware

4.8 ระหว่างการอัพเดท Firmware ห้ามถอดสาย USB หรือ ปิดโปรแกรมเป็นอันขาด เพราะจะทำให้ ET-PGMPIC PK3 เสียหายได้ ให้รอจนการอัพเดท Firmware เสร็จเรียบร้อย ดังรูปที่ 4-8



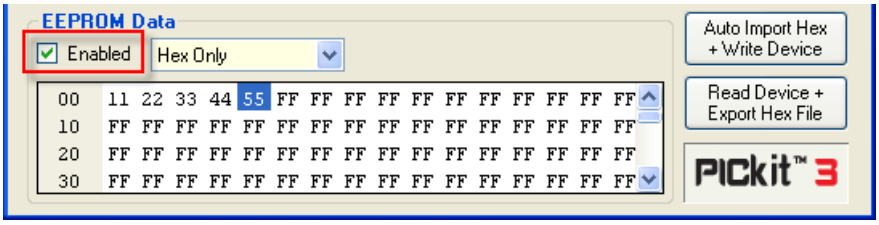
รูปที่ 4-8 แสดงเมื่อการอัพเดท Firmware เสร็จเรียบร้อย

4.9 จากนั้นให้กดปุ่ม **Write** เพื่อทำการโปรแกรม Hex File อีกครั้ง เมื่อการโปรแกรมเสร็จเรียบร้อย จะแสดงข้อความ ดังรูปที่ 4-9



รูปที่ 4-9 แสดงเมื่อการโปรแกรมเสร็จเรียบร้อย

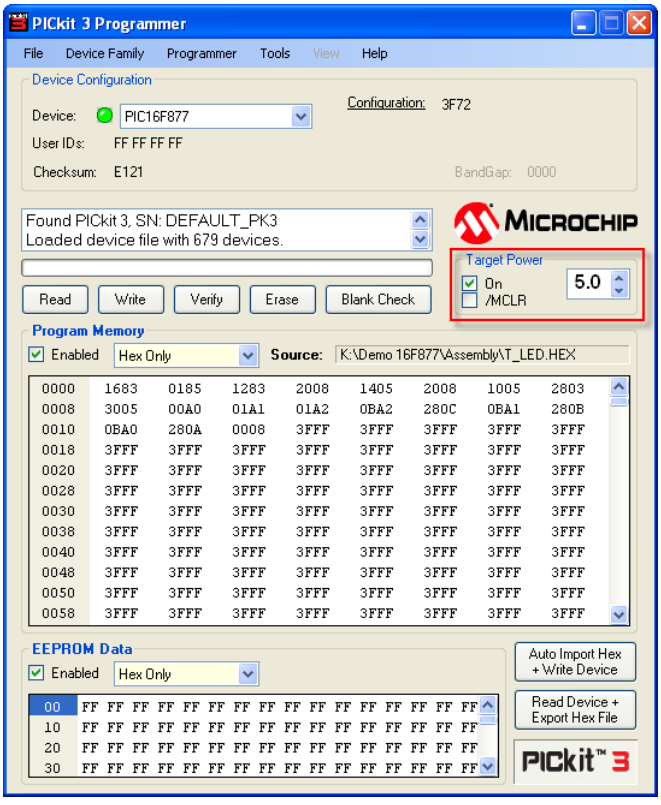
4.10 ในการกรณีที่ผู้ใช้ต้องการโปรแกรมค่า EEPROM ภายในของไมโครคอนโทรลเลอร์ PIC ก็สามารถเลือก **Enable** และใส่ค่าข้อมูลที่ต้องการได้เลย พร้อมกับการโปรแกรม Hex File ในตอนแรก ก่อนที่จะกดปุ่ม **Write** ดังรูปที่ 4-10



รูปที่ 4-10 แสดงต้องการโปรแกรมค่า EEPROM ภายใน

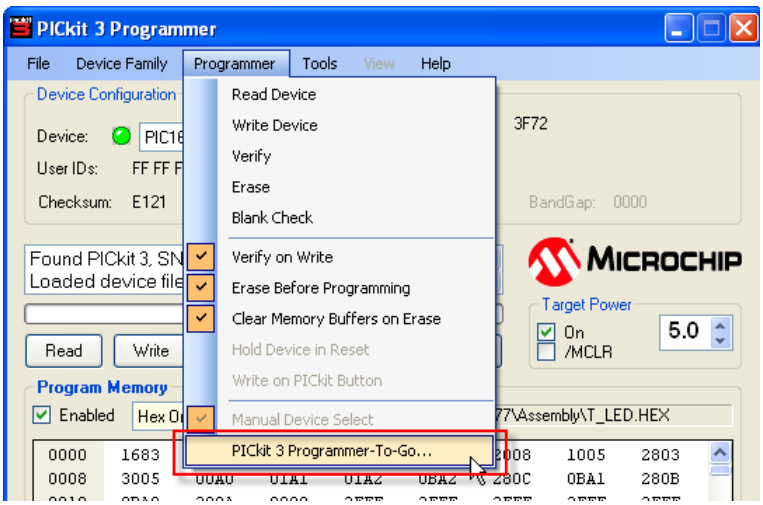
5. การใช้งานโหมด Programmer-To-Go ด้วยโปรแกรม PICkit 3 Programmer

5.1 เปิดโปรแกรม PICkit 3 Programmer เลือกเบอร์ที่ต้องการจะโปรแกรม และ เลือก Hex File ที่ต้องการจะโปรแกรม เหมือนขั้นตอนการโปรแกรมปกติ ในกรณีที่โปรแกรมผ่านทาง TEXTOOL หรือใช้ไฟเลี้ยงจาก ET-PGMPIC PK3 เลี้ยงบอร์ดปลายทาง ในช่อง **Target Power** ให้เลือก **On** ด้วย ดังรูปที่ 5-1



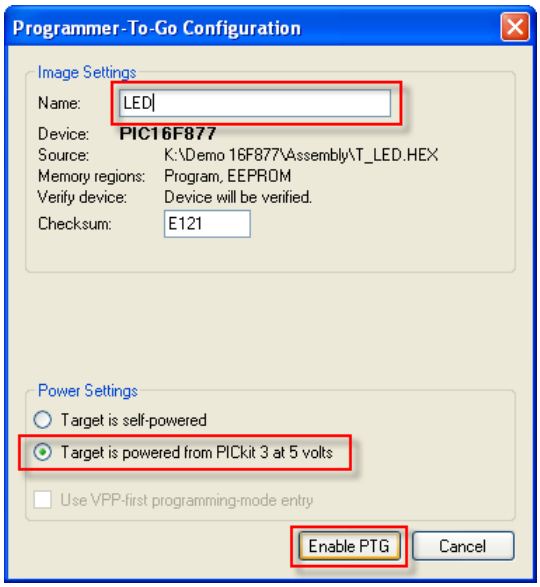
รูปที่ 5-1 แสดงเลือกเบอร์ที่ต้องการจะโปรแกรม และ เลือก Hex File

5.2 เลือกที่เมนู **Programmer>PICkit 3 Programmer-To-Go..** ดังรูปที่ 5-2



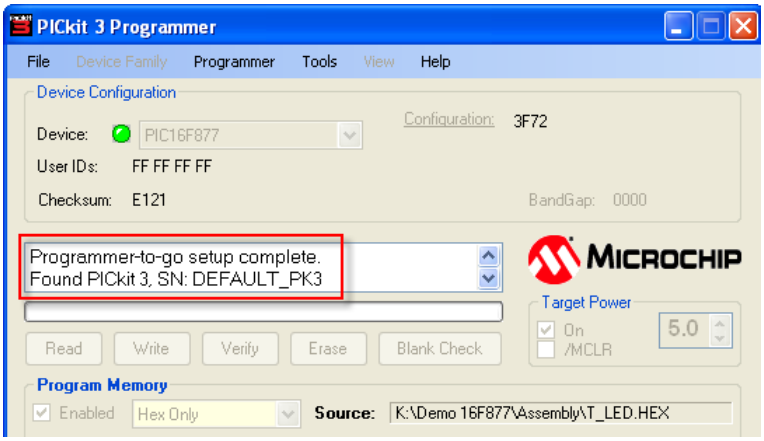
รูปที่ 5-2 แสดงการเลือกใช้งานโหมด Programmer-to-go

5.3 จากนั้นจะปรากฏหน้าต่าง Programmer-To-Go Configuration ให้ตั้งชื่อในช่อง Image Settings ตามความต้องการ ส่วนในช่อง Power Settings จะมีตัวเลือกอยู่ 2 ตัวเลือกคือ **Target is self-powered** หมายถึง บอร์ดปลายทางมีไฟเลี้ยงเป็นของตัวเอง และ **Target is powered from PICkit 3 at 5 volts** หมายถึง บอร์ดปลายทางใช้ไฟเลี้ยงจาก PICkit 3 ให้ผู้ใช้เลือกให้ตรงกับการใช้งาน ในกรณีที่เป็นการโปรแกรมผ่าน TEXTOOL ให้เลือก **Target is powered from PICkit 3 at 5 volts** จากนั้นให้คลิกที่ปุ่ม **Enable PTG** เพื่อเริ่มการทำงาน ดังรูปที่ 5-3



รูปที่ 5-3 แสดงการตั้งค่าการใช้งาน Programmer-To-Go

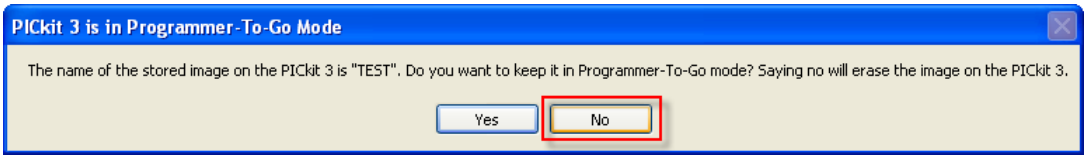
5.4 เมื่อการโหลดค่าต่างเสร็จเรียบร้อย จะแสดงข้อความ ดังรูปที่ 5-4 พร้อมกันนั้น LED ACTIVE จะกระพริบ แสดงว่าการใช้งาน **Programmer-To-Go** พร้อมใช้งาน



รูปที่ 5-4 แสดงเมื่อการโหลดค่าต่างเสร็จเรียบร้อย

5.5 เมื่อทุกอย่างพร้อม การใช้งานผู้ใช้ก็เพียงแค่กดสวิตช์ **PROGRAM** บน ET-PGMPIC PK3 จะสังเกตเห็น LED ACTIVE หยุดกระพริบ และ LED STATUS จะติดเป็นสีส้ม เมื่อการโปรแกรมเสร็จเรียบร้อย LED ACTIVE จะกลับมากระพริบ และ LED STATUS จะกลายเป็นสีเขียว หลังจากนั้นผู้ใช้ก็สามารถปิดโปรแกรม PICkit 3 Programmer และนำไปโปรแกรมบอร์ดอื่นๆ ได้เลยโดยไม่ต้องเปิดโปรแกรม

5.6 ในกรณีที่ต้องการโปรแกรม Hex File ตัวใหม่ หรือ กลับไปใช้งานในโหมดปกติ ให้ทำการเชื่อมต่อ ET-PGMPIC PK3 เข้ากับคอมพิวเตอร์ และเปิดโปรแกรม PICkit 3 Programmer จากนั้นจะปรากฏข้อความบอกว่า PICkit 3 อยู่ในโหมด Programmer-To-Go ดังรูปที่ 5-5 ให้เลือก No เพื่อเข้าสู่โหมดปกติ ซึ่งข้อมูลต่างๆ จะถูกลบ

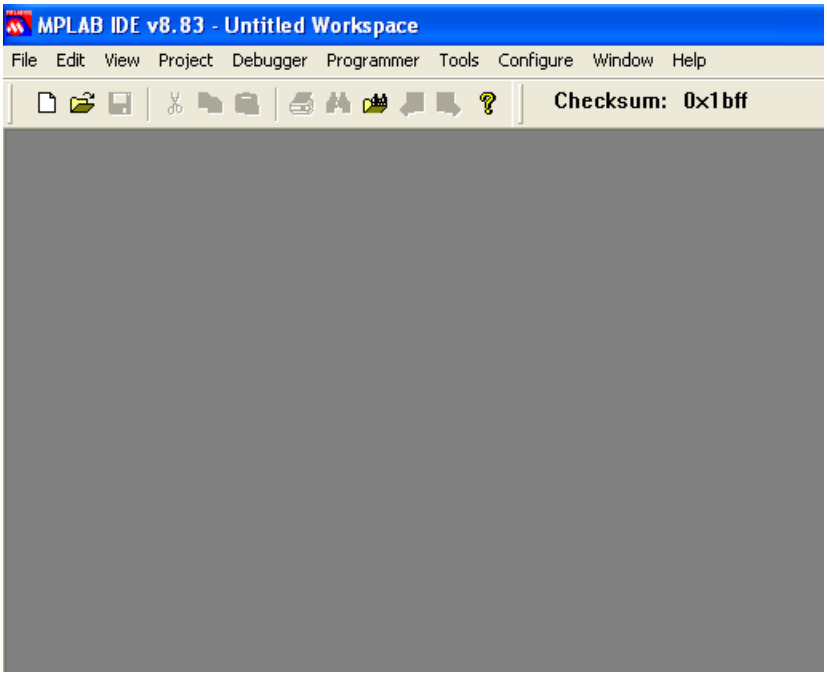


รูปที่ 5-5 แสดงการกลับไปใช้งานในโหมดปกติ

6. การโปรแกรมและดีบักด้วยภาษาแอสแซมบลี

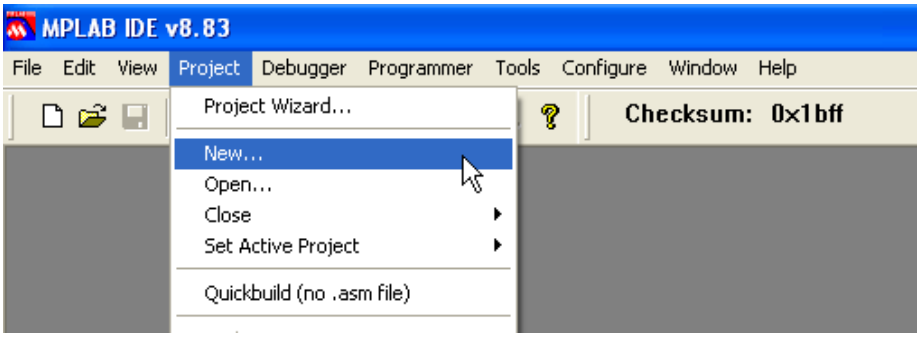
ในที่จะยกตัวอย่างโดยใช้บอร์ด CP-PIC V4(ICD2) ซึ่งใช้ MCU PIC16F877 ก่อนการใช้งานนั้นต้องต่อบอร์ดต่างๆ เรียบร้อยก่อนและ เลื่อน SW MODE ของบอร์ด CP-PIC V3(ICD2) ไปที่ตำแหน่ง PROG ซึ่งขั้นตอนการใช้งานต่าง ๆ มีดังนี้

6.1 หลังจากต่อ ET-PGMPIC PK3 เข้ากับคอมพิวเตอร์และต่อสายต่างเข้ากับบอร์ดเป้าหมายแล้วก็ให้เปิดโปรแกรม MPLAB IDE จากนั้นจะปรากฏหน้าต่างของโปรแกรม MPLAB IDE ดังรูปที่ 6-1



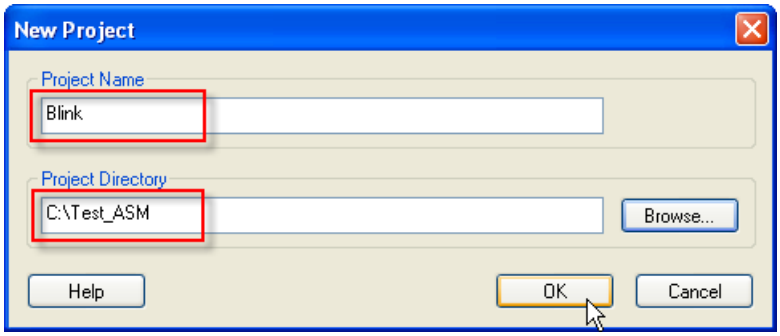
รูปที่ 6-1 แสดงหน้าต่างของโปรแกรม MPLAB IDE

6.2 ทำการสร้างโปรเจคใหม่โดยเลือกที่เมนู Project>New… ดังรูปที่ 6-2



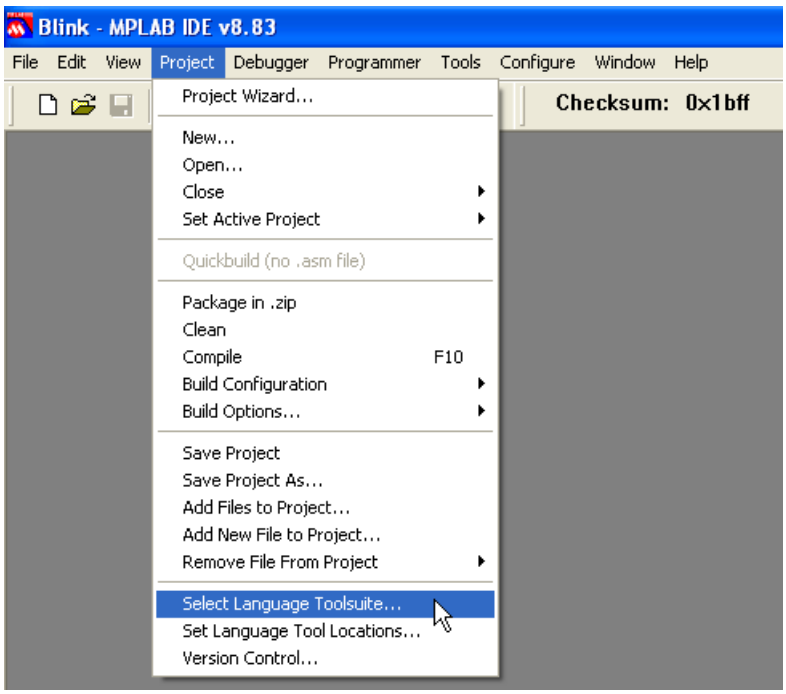
รูปที่ 6-2 แสดงหน้าต่างการสร้างโปรเจคใหม่

6.3 จากนั้นจะปรากฏหน้าต่าง New Project ให้ทำการตั้งชื่อ Project Name และระบุตำแหน่งที่จะเก็บโปรเจคโดยการคลิกปุ่ม Browse จากนั้นคลิก OK ดังรูปที่ 6-3



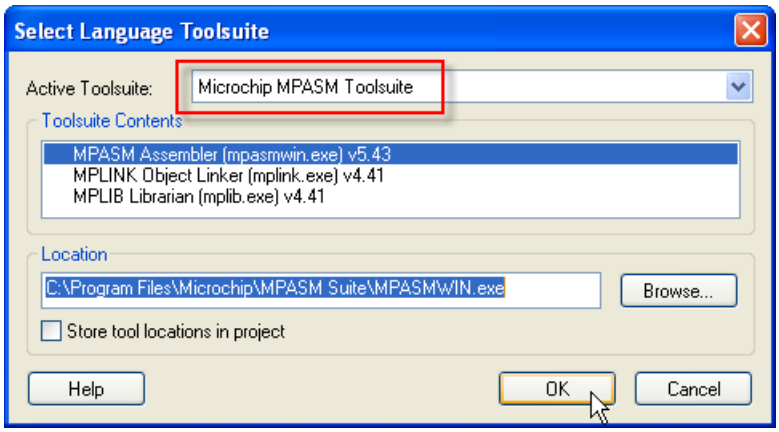
รูปที่ 6-3 แสดงหน้าต่าง New Project

6.4 ทำการเลือกเครื่องมือที่จะใช้ในการเขียนโปรแกรม โดยเลือกที่ **Project>Select Language Toolsuite…** ดังรูปที่ 6-4



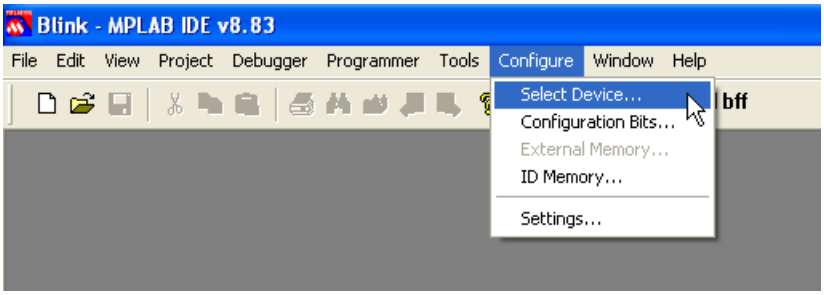
รูปที่ 6-4 แสดงการเลือกเครื่องมือที่จะใช้ในการเขียนโปรแกรม

6.5 จากนั้นจะปรากฏหน้าต่าง Select Language Toolsuite ให้เลือก Active Toolsuite เป็น Microchip MPASM Toolsuite เพราะว่าตัวอย่างนี้จะเขียนด้วยภาษาแอสแซมบลี จากคลิกปุ่ม OK ดังรูปที่ 6-5



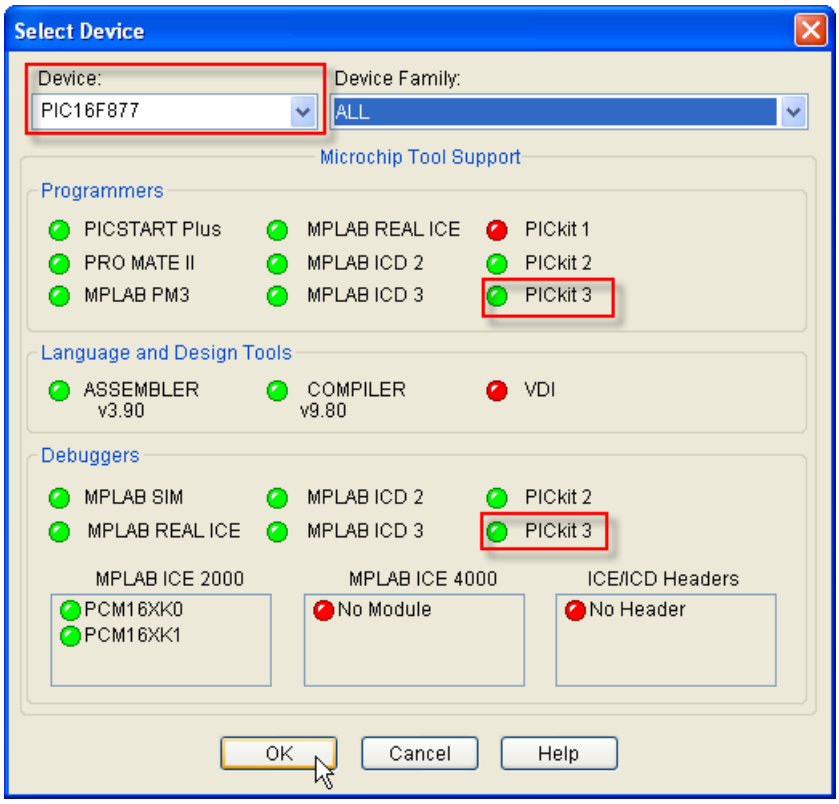
รูปที่ 6-5 แสดงหน้าต่าง Select Language Toolsuite

6.6 ทำการเลือกเบอร์ของ MCU ที่ใช้ในโดยการเลือกที่เมนู **Configure>Select Device…** ดังรูปที่ 6-6



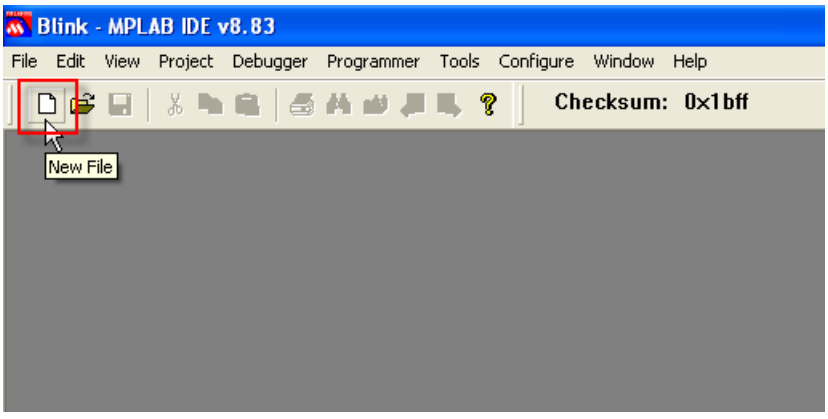
รูปที่ 6-6 แสดงการเลือกเบอร์ของ MCU

6.7 จากนั้นจะปรากฏหน้าต่าง Select Device ให้ทำการเลือก Device เป็น 16F877 จากนั้นคลิกปุ่ม OK ดังรูปที่ 6-7 ข้อสังเกตจากหน้าต่าง Select Device ถ้า MCU เบอร์ไหนสามารถโปรแกรมและดีบักได้ด้วย PICkit 3 จะมีปุ่มสีเขียวหน้าเครื่องมือ จากรูปจะเห็นเป็นสีเขียวทั้ง Programmers และ Debuggers แสดงว่า เบอร์ 16F877 สามารถทั้งโปรแกรมและดีบักได้นั่นเอง แต่ถ้าเป็นสีเหลืองแสดงว่าสามารถใช้ได้แต่อาจจะยังไม่สมบูรณ์ ส่วนถ้าเป็นสีแดงก็ไม่สามารถใช้ได้นั่นเอง



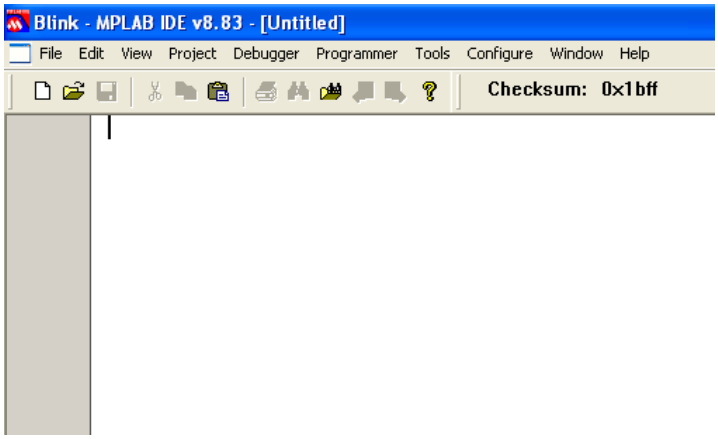
รูปที่ 6-7 แสดงหน้าต่าง Select Device

6.8 ทำการสร้างไฟล์ที่จะเขียนโค๊ดภาษาแอสแซมบลีโดยไปที่เมนู **File>New** หรือคลิกที่ไอคอน New File ดังรูปที่ 6-8



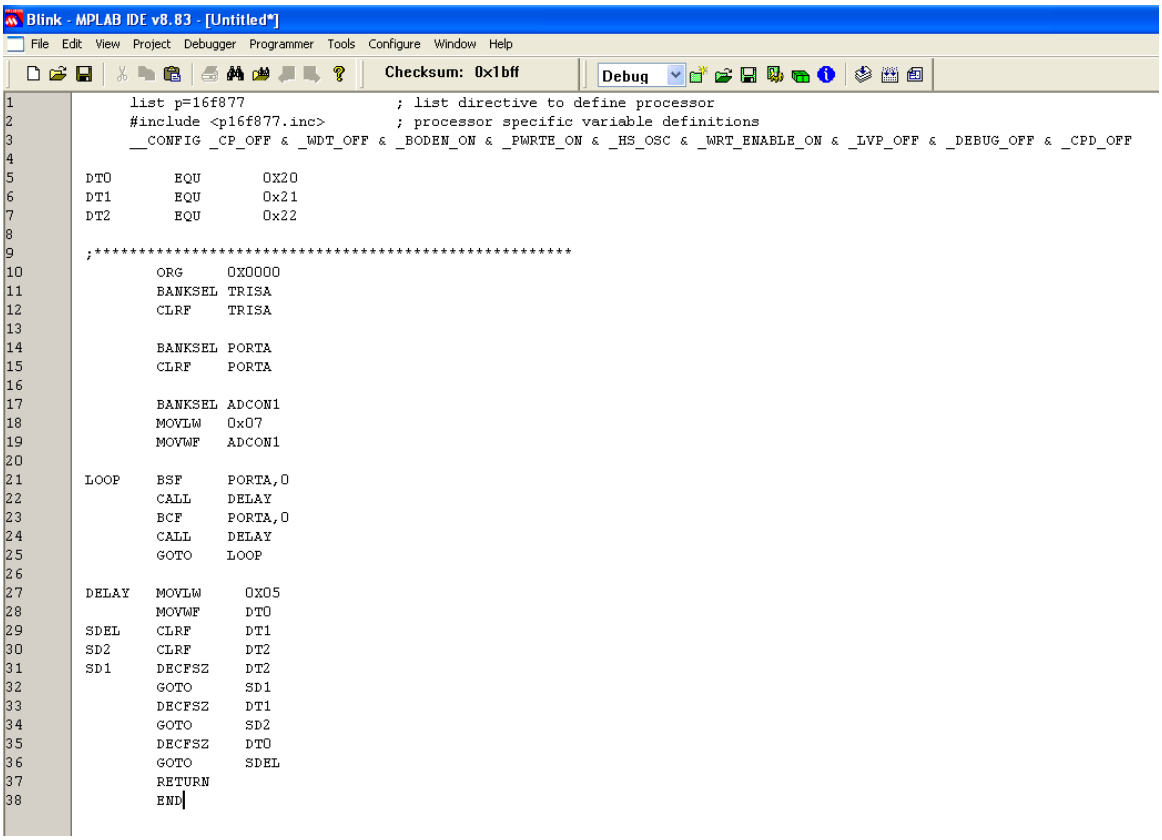
รูปที่ 6-8 แสดงการสร้างไฟล์ใหม่

6.9 จากนั้นหน้าต่างว่างๆ เพื่อใช้สำหรับเขียนโค๊ดดังรูปที่ 6-9



รูปที่ 6-9 แสดงหน้าต่างที่ใช้สำหรับเขียนโค๊ด

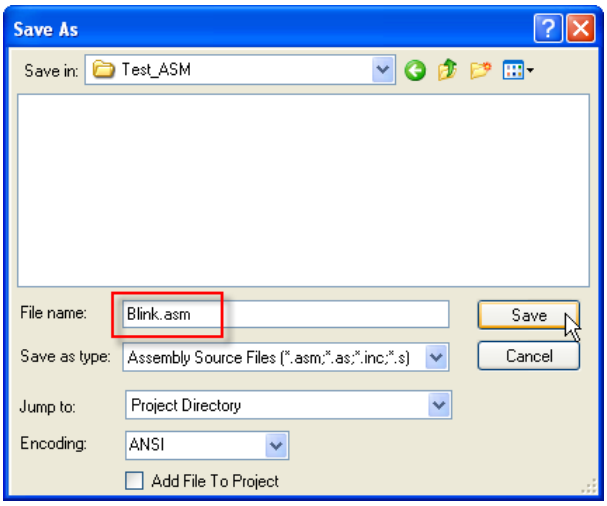
6.10 ทำการเขียนโค๊ดภาษาแอสแซมบลีดังรูปที่ 6-10 ซึ่งจากตัวอย่างจะเป็นโปรแกรมไฟกระพริบออกที่ PORT RA0




```
1         list p=16f877                ; list directive to define processor
2         #include <p16f877.inc>       ; processor specific variable definitions
3         __CONFIG _CP_OFF & _WDT_OFF & _BODEN_ON & _PWRTE_ON & _HS_OSC & _WRT_ENABLE_ON & _LVP_OFF & _DEBUG_OFF & _CPD_OFF
4
5       DT0       EQU       0X20
6       DT1       EQU       0x21
7       DT2       EQU       0x22
8
9       ;*****************************************************
10              ORG     0X0000
11              BANKSEL TRISA
12              CLRF    TRISA
13
14              BANKSEL PORTA
15              CLRF    PORTA
16
17              BANKSEL ADCON1
18              MOVLW   0x07
19              MOVWF   ADCON1
20
21      LOOP    BSF     PORTA,0
22              CALL    DELAY
23              BCF     PORTA,0
24              CALL    DELAY
25              GOTO    LOOP
26
27      DELAY   MOVLW     0X05
28              MOVWF     DT0
29      SDEL    CLRF      DT1
30      SD2     CLRF      DT2
31      SD1     DECFSZ    DT2
32              GOTO      SD1
33              DECFSZ    DT1
34              GOTO      SD2
35              DECFSZ    DT0
36              GOTO      SDEL
37              RETURN
38              END
```

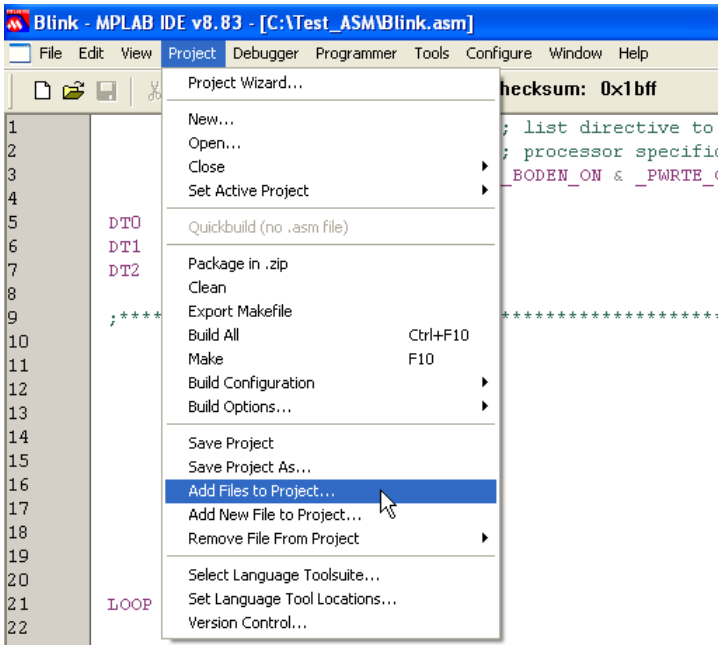
รูปที่ 6-10 แสดงการเขียนโค๊ดภาษาแอสแซมบลี

6.11 ทำการบันทึกไฟล์ที่เขียนขึ้นมาโดยไปที่เมนู **File>Save** หรือคลิกที่ไอคอน Save จากนั้นจะปรากฏหน้าต่าง Save As ให้ตั้งชื่อไฟล์เป็น **Blink.asm** และเก็บไว้ที่เดียวกับที่เก็บโปรเจคตอนแรก จากนั้นคลิกปุ่ม Save ดังรูปที่ 6-11



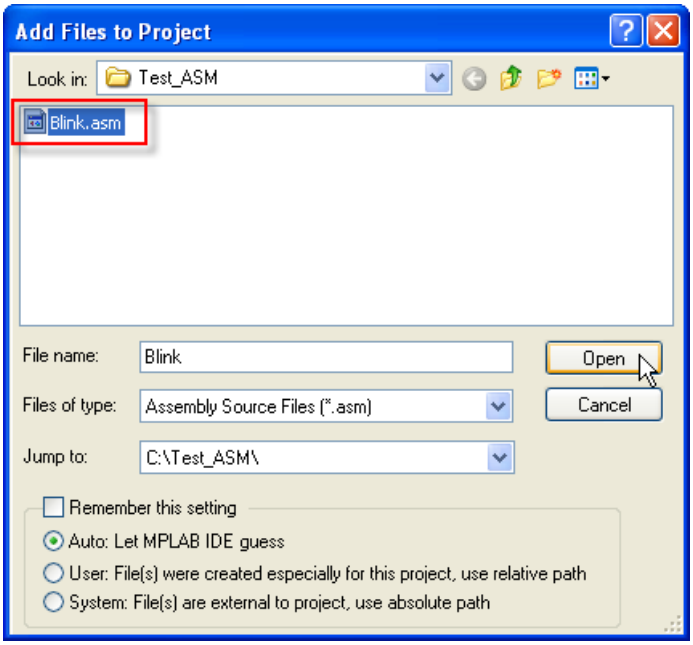
รูปที่ 6-11 แสดงหน้าต่าง Save As

6.12 ทำการเพิ่มไฟล์ภาษาแอสแซมบลีที่เขียนเข้าไปในโปรเจคโดยการเลือกที่เมนู **Project>Add Files to Project** ดังรูปที่ 6-12



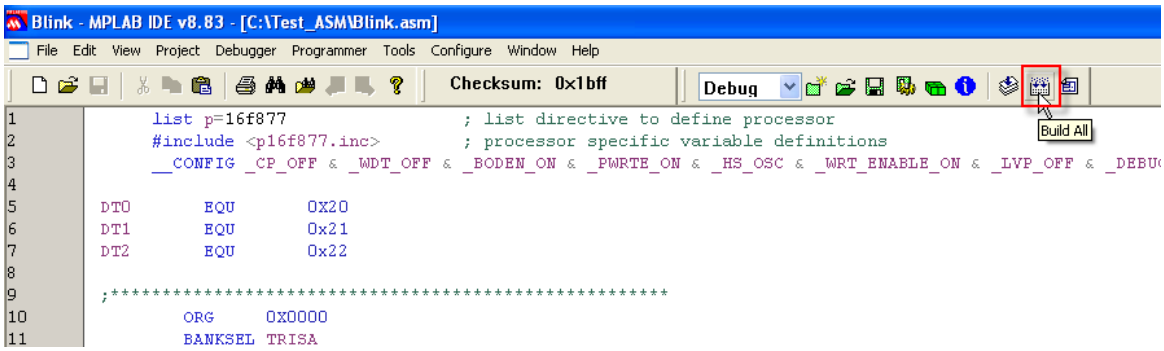
รูปที่ 6-12 แสดงวิธีการเพิ่มไฟล์เข้าไปในโปรเจค

6.13 จากนั้นจะปรากฏหน้าต่าง Add Files to Project ให้ทำการเลือกที่ได้ทำการบันทึกไว้จากขั้นตอนที่ผ่านมาจากนั้นคลิก Open ดังรูปที่ 6-13



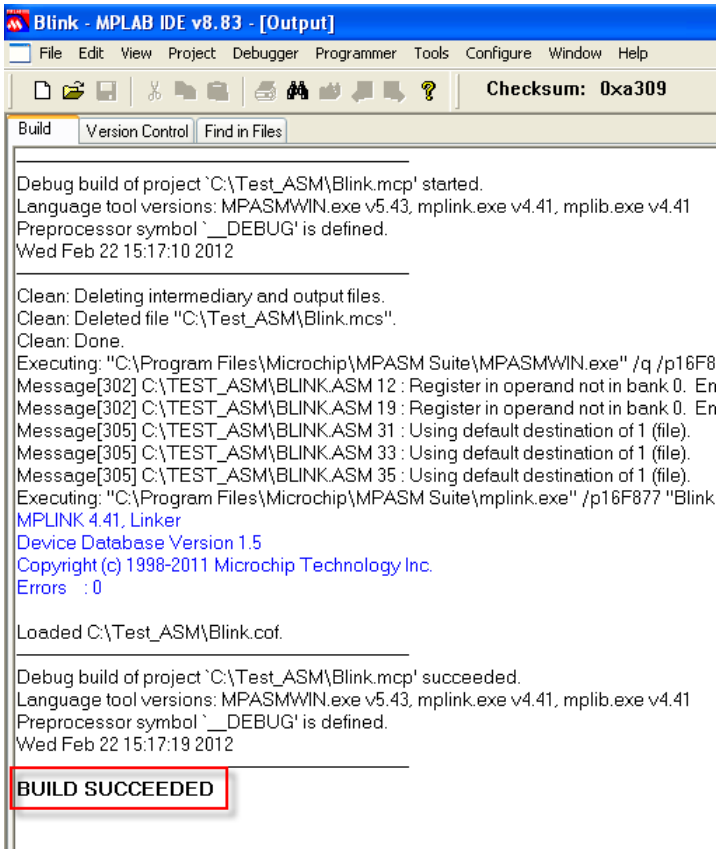
รูปที่ 6-13 แสดงหน้าต่าง Add Files to Project

6.14 ทำการคอมไพล์โค๊ดที่เขียนขึ้นโดยเลือกที่เมนู **Project>Build All** หรือคลิกที่ไอคอน Build All ดังรูปที่ 6-14



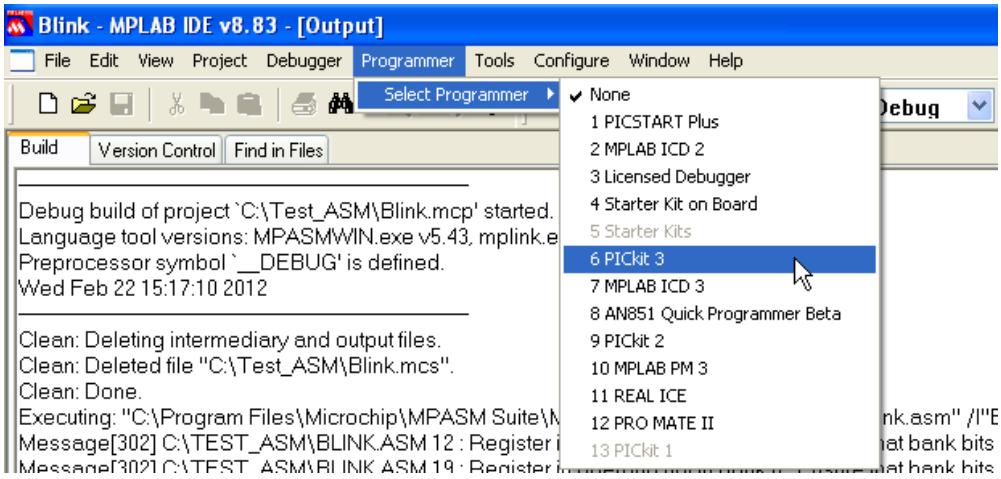
รูปที่ 6-14 แสดงวิธีการคอมไพล์โค๊ดที่เขียนขึ้น

6.15 ถ้าโค๊ดที่เขียนขึ้นถูกต้องตามหลักไวยกรณ์ (Syntex) จะปรากฏข้อความดังรูปที่ 6-15 ถ้าไม่ถูกต้อง ก็จะมีข้อความแจ้งเตือนข้อผิดพลาดขึ้นมาให้ทำการแก้ไขและคอมไพล์ใหม่



รูปที่ 6-15 แสดงเมื่อโค๊ดที่เขียนขึ้นถูกต้องตามหลักไวยกรณ์

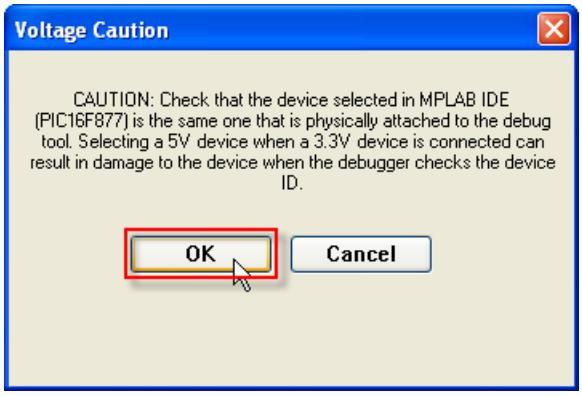
6.16 เมื่อทุกอย่างเรียบร้อยให้ทดลองโปรแกรมไฟล์ “Blink.hex” ที่ได้จากการคอมไพล์เข้าสู่ MCU โดยผู้ใช้จะต้องเลือกเครื่องที่จะใช้ในการโปรแกรม โดยเลือกไปที่เมนู Programmer>Select Programmer>Select Programmer>PICkit 3 ดังรูปที่ 6-16



รูปที่ 6-16 แสดงการเลือกเครื่องที่จะใช้ในการโปรแกรม

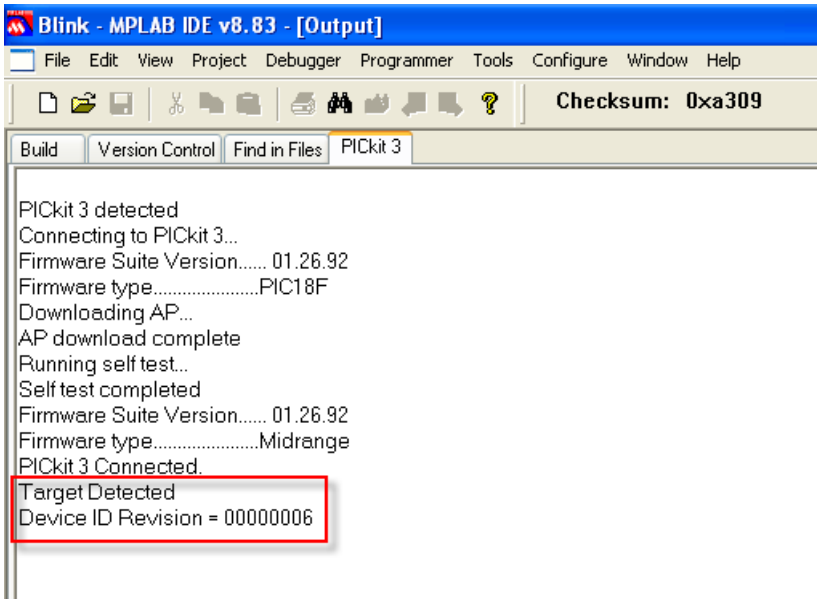
6.17 หลังจากนั้นหาก PGMPIC PK3 ผ่านการใช้งานกับไมโครคอนโทรลเลอร์ตระกูลอื่นมาที่ไม่ใช่ Midrange จะมีการอัพเดท Firmware ระหว่างการอัพเดท Firmware ห้ามถอดสาย USB หรือปิดโปรแกรมเป็นอันขาด เพราะจะทำให้ ET-PGMPIC PK3 เสียหายได้ ให้รอจนการอัพเดท Firmware เสร็จเรียบร้อย

6.18 จากนั้นจะปรากฏหน้าต่างคำเตือนให้ตรวจสอบการต่อไฟเลี้ยง ให้เลือก OK ดังรูปที่ 6-17



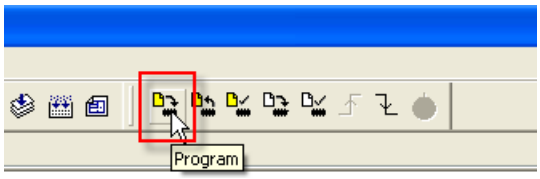
รูปที่ 6-17 แสดงคำเตือนให้ตรวจสอบการต่อไฟเลี้ยง

6.19 ถ้าไม่มีข้อผิดพลาดโปรแกรมจะตรวจสอบเจอไมโครคอนโทรลเลอร์ PIC ที่มาเชื่อมต่อด้วยดังรูปที่ 6-18



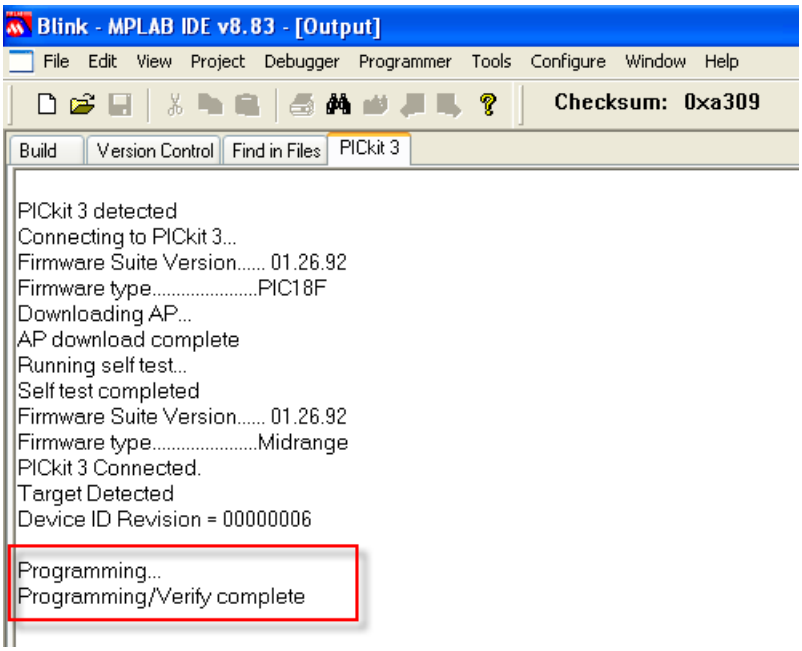
รูปที่ 6-18 แสดงเมื่อโปรแกรมตรวจสอบเจอไมโครคอนโทรลเลอร์ PIC ที่มาเชื่อมต่อด้วย

6.20 ในกรณีที่ทำการติดต่อแล้วไม่เจอ อาจจะมีสาเหตุได้หลายอย่างเช่นไม่มีไฟเลี้ยงบอร์ดเป้าหมาย ,ต่อสายผิด หรือไม่ก็ MCU เสีย ให้ทำการตรวจสอบและแก้ไข จากนั้นก็ทำการเชื่อมต่อใหม่อีกครั้ง เมื่อทุกอย่างเรียบร้อยก็ให้ทำการโปรแกรม hex ไฟล์ที่ได้จากการคอมไพล์เข้าสู่ MCU โดยคลิกที่ไอคอน Program target device ดังรูปที่ 6-19



รูปที่ 6-19 แสดงการโปรแกรม Hex File

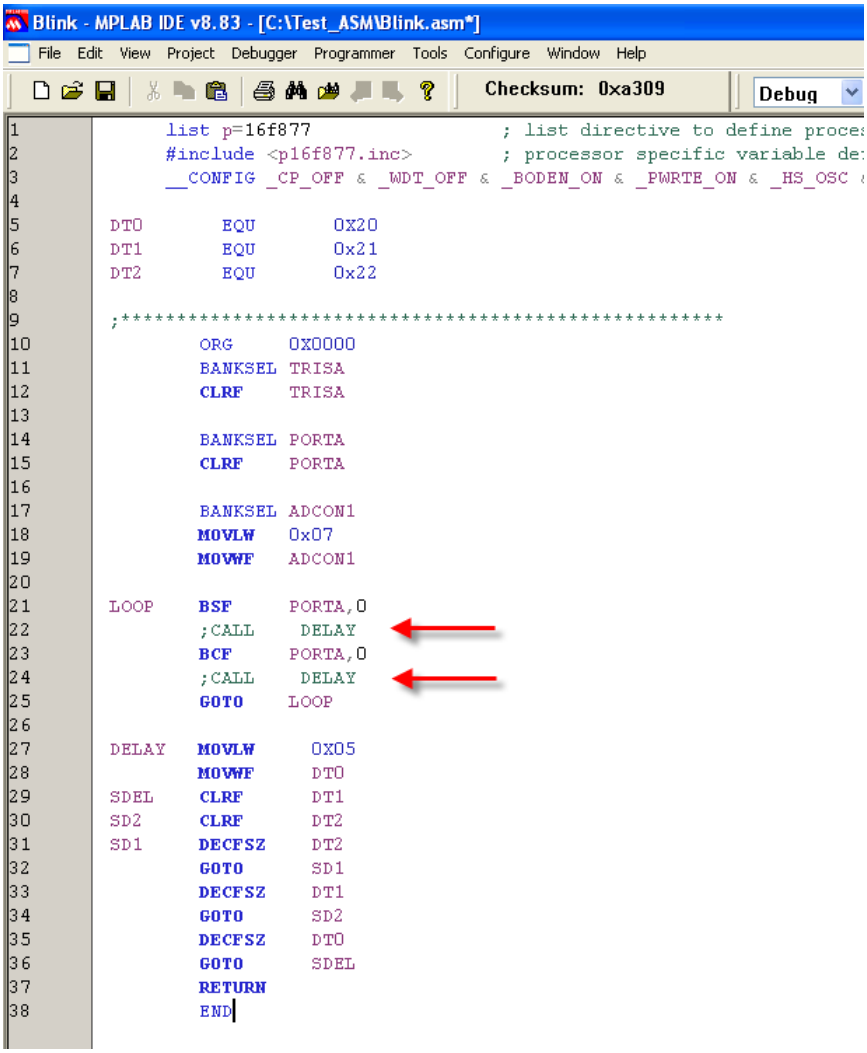
6.21 ถ้าการโปรแกรมไม่พบข้อผิดพลาดก็จะปรากฏข้อความ Programming/Verify complete ดังรูปที่ 6-20



รูปที่ 6-20 แสดงเมื่อการโปรแกรมเสร็จสมบูรณ์

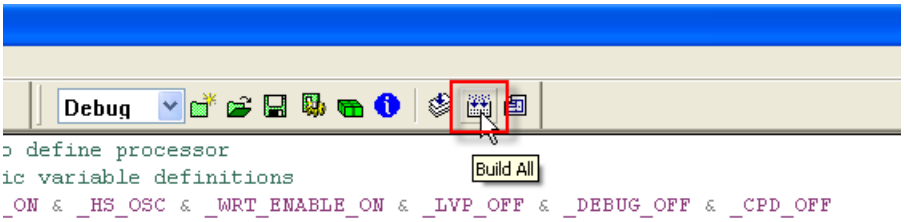
6.22 จากนั้นเลื่อน SW MODE ของบอร์ด CP-PIC V3(ICD2) ไปที่ตำแหน่ง RUN จะเห็นว่า LED ที่พอร์ต RA0 จะกระพริบตามโปรแกรมที่เขียนไว้

6.23 หลังจากได้ทดลองโปรแกรมกันแล้วขั้นต่อไป จะทำการทดลองดีบักดูค่าต่างๆ กัน โดยจะทำการปิดลูบโปรแกรมหน่วงเวลาเพื่อให้ง่ายต่อการดีบัก ดังรูปที่ 6-21



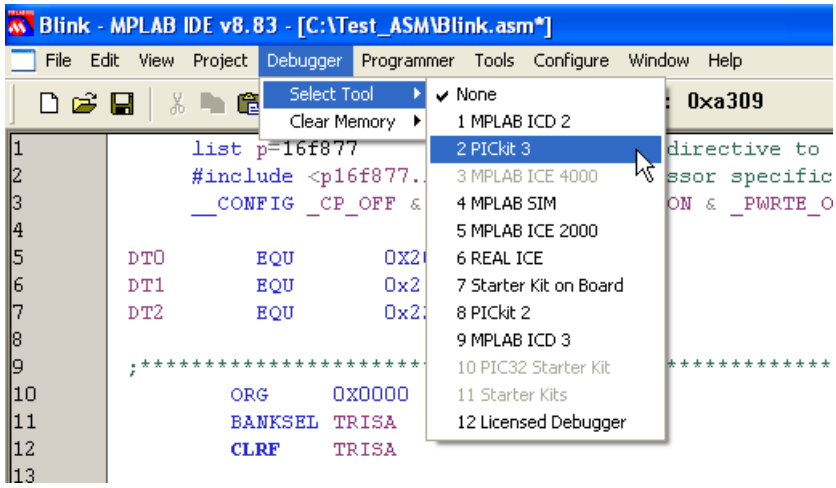
รูปที่ 6-21 แสดงการปิดลูปหน่วงเวลา

6.24 จากนั้นให้คอมไพล์โปรแกรมที่แก้ไขใหม่โดยคลิกที่ปุ่ม Build All ดังรูปที่ 6-22



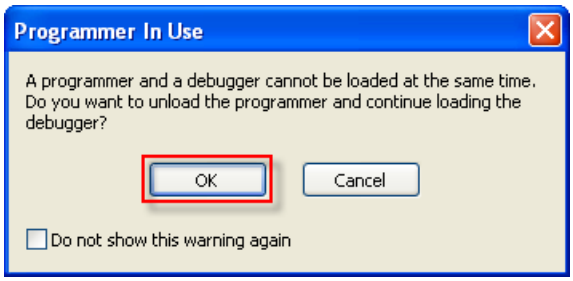
รูปที่ 6-22 แสดงการคอมไพล์โปรแกรมที่แก้ไข

6.25 ทำการเลือก Debugger เป็น PICkit 3 ดังรูปที่ 6-23



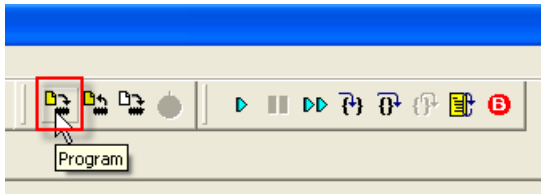
รูปที่ 6-23 แสดงการเลือก Debugger

6.26 จากนั้นจะปรากฏหน้าต่าง Programmer In Use ขึ้นมาถามว่าต้องการจะยกเลิกการใช้งานในโหมด Programmer หรือไม่ ให้ตอบตกลงโดยคลิกที่ปุ่ม OK ดังรูปที่ 6-24



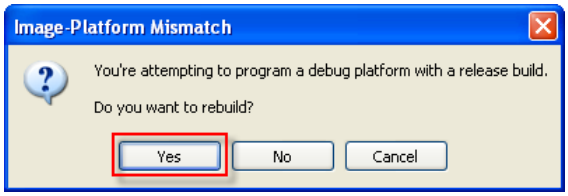
รูปที่ 6-24 แสดงหน้าต่าง Programmer In Use

6.27 จากนั้นโปรแกรมจะทำการติดต่อกับบอร์ดปลายทางอีกครั้ง ให้ทำการโปรแกรม Hex File ที่จะใช้ในการดีบัก โดยคลิกที่ปุ่ม Program ดังรูปที่ 6-25



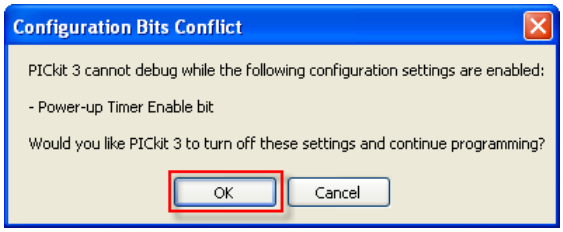
รูปที่ 6-25 แสดงการโปรแกรม hex ไฟล์

6.28 จากนั้นจะปรากฏหน้าต่าง Image-Platform Mismatch ให้เลือก Yes ดังรูปที่ 6-26



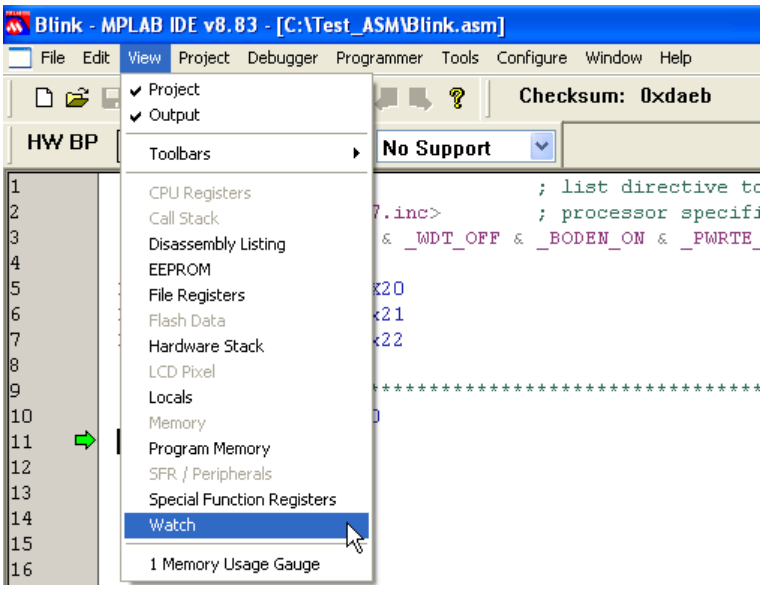
รูปที่ 6-26 แสดงหน้าต่าง Image-Platform Mismatch

6.29 จากนั้นจะปรากฏหน้าต่าง Configuration Bits Conflict ฟ้องว่าไม่สามารถ debug ได้ เนื่องจากได้เปิดใช้งาน Power-up Timer ได้เปิดใช้งานอยู่ ให้เลือก OK เพื่อปิดการใช้งานดังรูปที่ 6-27



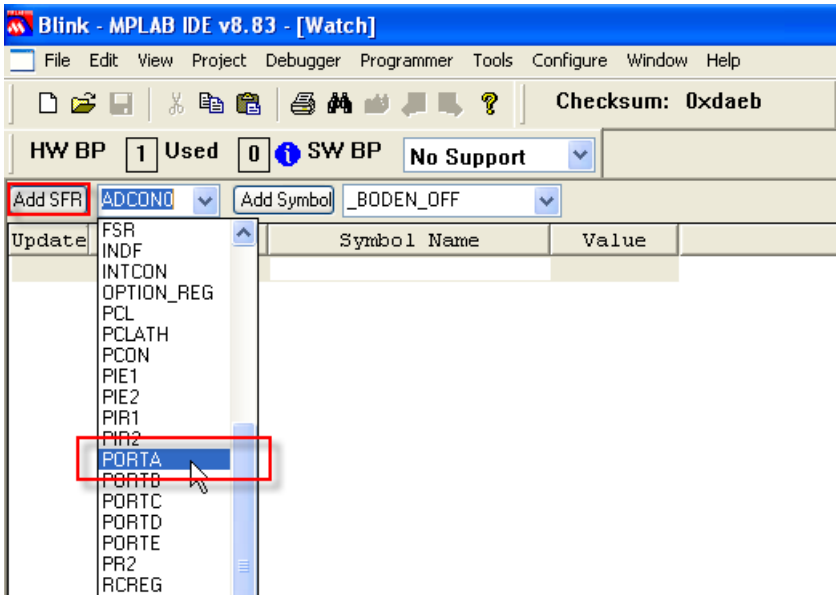
รูปที่ 6-27 แสดงหน้าต่าง Configuration Bits Conflict

6.30 จากนั้นโปรแกรมจะโปรแกรม Hex File ให้ไมโครคอนโทรลเลอร์ รอจนเร็จเรียบร้อย จากนั้นให้ทำการเพิ่มรีจิสเตอร์หรือตัวแปรที่ต้องการดูค่าโดยไปที่เมนู View>Watch ดังรูปที่ 6-28



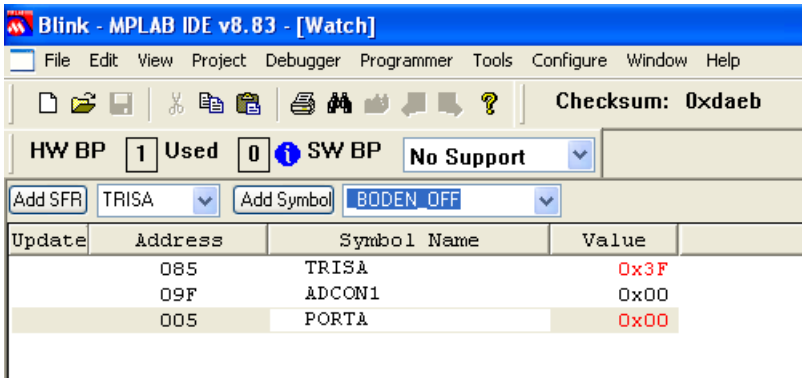
รูปที่ 6-28 แสดงการเพิ่มรีจิสเตอร์หรือตัวแปร

6.31 จากนั้นจะปรากฏหน้าต่าง Watch ดังรูปที่ 6-29 ให้ทำการเลือกรีจิสเตอร์ที่ต้องการดูค่าจากนั้นคลิกที่ปุ่ม Add SFR ในที่นี้จะเลือก TRISA,ADCON1 และ PORTA



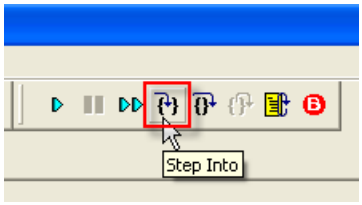
รูปที่ 6-29 แสดงหน้าต่าง Watch

6.32 เมื่อได้ทำการเพิ่มรีจิสเตอร์ต่างๆ เสร็จเรียบร้อยจะได้ดังรูปที่ 6-30

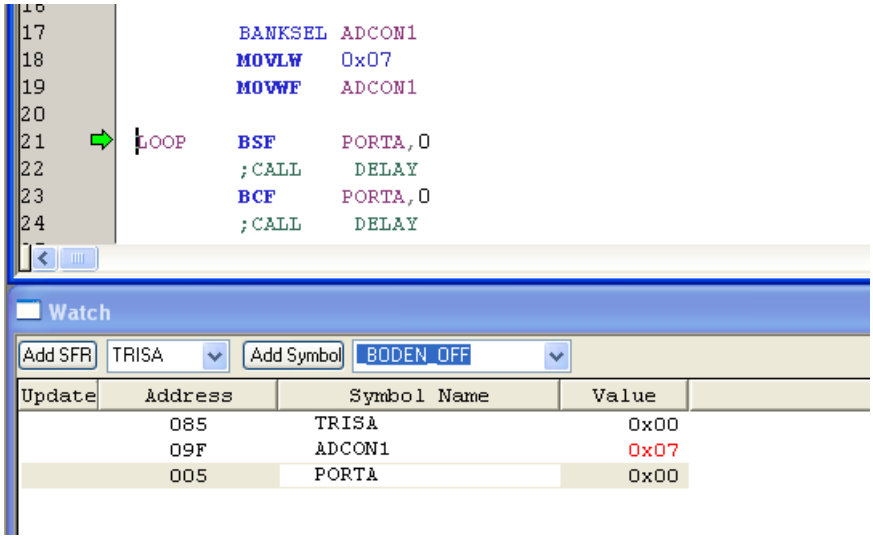


รูปที่ 6-30 แสดงเมื่อได้ทำการเพิ่มรีจิสเตอร์ต่างๆ เสร็จเรียบร้อย

6.33 เริ่มต้นการดีบักโดยการกดปุ่ม Step Into ดังรูปที่ 6-31 ไปเรื่อยๆ จะสังเกตเห็นลูกศรสีเขียวชี้ไปบรรทัดที่ทำงานอยู่และค่ารีจิสเตอร์ ก็จะเปลี่ยนแปลงตามที่เขียนโปรแกรมไว้ด้วย ดังรูปที่ 6-32

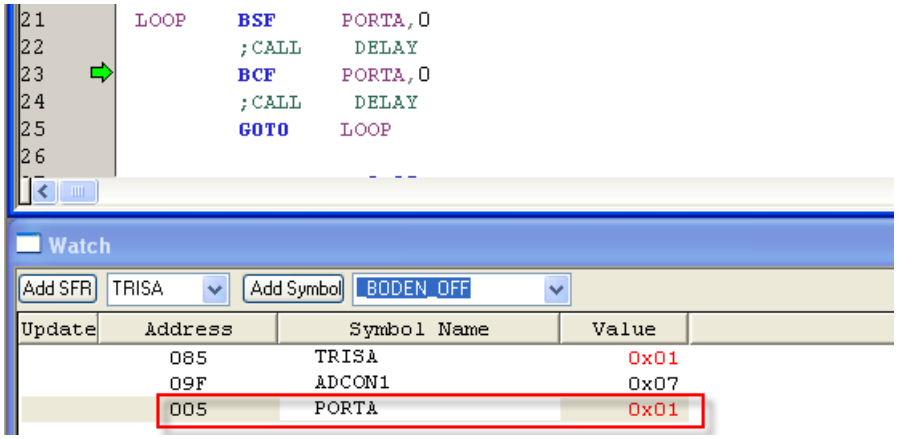


รูปที่ 6-31 แสดงปุ่ม Step Into



รูปที่ 6-32 แสดงค่าต่างๆ ระหว่างการดีบัก

6.34 เมื่อได้ดีบักผ่านโปรแกรมบรรทัด BSF PORTA,0 และ BCF PORTA,0 จะเห็นว่าค่าของ PORTA ก็จะเปลี่ยนแปลงดังรูปที่ 6-33 และพร้อมกันนั้น LED ที่ต่ออยู่กับ PORT RA0 ก็จะเปลี่ยนสภาวะติดดับตามโปรแกรมพร้อมกันไปด้วย



รูปที่ 6-33 แสดงค่าต่างๆ ระหว่างการดีบัก

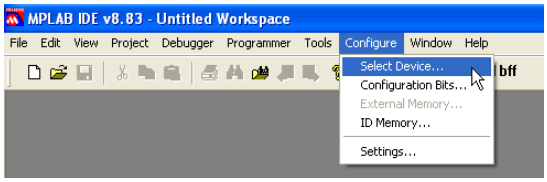
หมายเหตุ ในกรณีที่ผู้ใช้เขียนโปรแกรมด้วยภาษา C ไม่ว่าจะเป็น CCS,HI-TECH,C18,C30,C32 หรืออื่นๆ การโปรแกรมและดีบักก็มีขั้นตอนเหมือนกัน จะต่างกันเพียงแค่การเลือก Select Language Toolsuite เท่านั้น

7. การนำเข้า Hex File จากภายนอกมาโปรแกรม

ในกรณีผู้ใช้ต้องการใช้ ET-PGMPIC PK3 เป็นเครื่องโปรแกรมโดยนำ Hex File จากโปรแกรมอื่นๆ ที่ไม่รองรับการเชื่อมต่อ PICkit 3 เช่น picbasic pro ก็สามารถทำได้โดยการใช้วิธี Import เข้ามาซึ่งมีวิธีการดังนี้

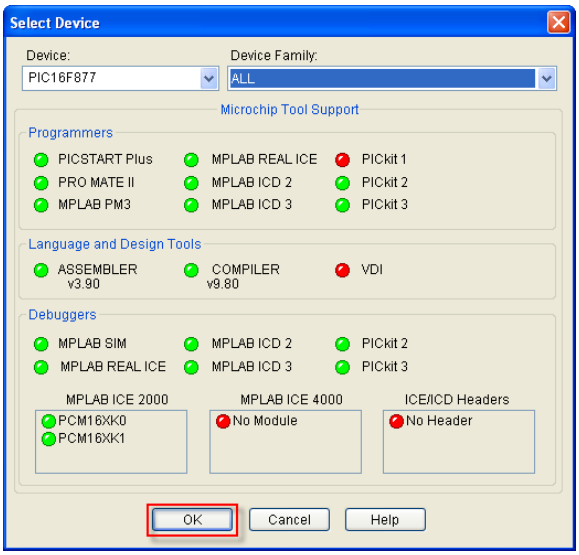
7.1 ทำการเชื่อมต่อ ET-PGMPIC PK3 เข้ากับคอมพิวเตอร์และเชื่อมต่อบอร์ดเป้าหมายให้เรียบร้อย

7.2 จากนั้นทำการเปิดโปรแกรม MPLAB IDE และไปที่เมนู **Configure>Select Device…** เพื่อเลือกเบอร์ของไมโครคอนโทรลเลอร์ ที่จะใช้โปรแกรม ดังรูปที่ 7-1



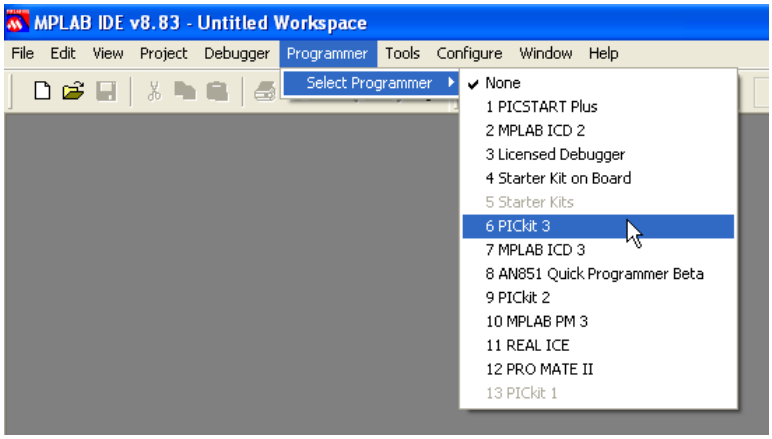
รูปที่ 7-1 แสดงวิธีการเลือกเบอร์ของไมโครคอนโทรลเลอร์

7.3 จากนั้นจะปรากฏหน้าต่าง Select Device ให้ทำการเลือกเบอร์ของ MCU ที่ต้องการโปรแกรมในที่นี้จะเลือกเป็นเบอร์ 16F877 จากนั้นคลิกปุ่ม OK ดังรูปที่ 7-2



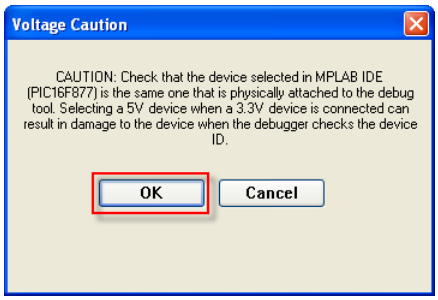
รูปที่ 7-2 แสดงหน้าต่าง Select Device

7.4 ทำการเลือกเครื่องโปรแกรมเป็น PICkit 3 โดยเลือกที่เมนู **Programmer>Select Programmer>PICkit 3** ดังรูปที่ 7-3



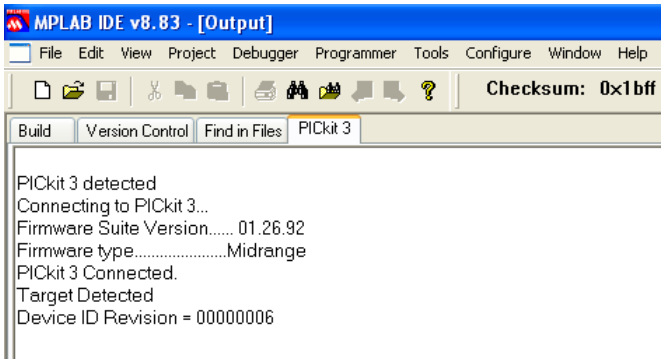
รูปที่ 7-3 แสดงการเลือกเครื่องมือในการโปรแกรม

7.5 จากนั้นจะปรากฏหน้าต่างคำเตือนให้ตรวจสอบการต่อไฟเลี้ยง ให้เลือก OK ดังรูปที่ 7-4



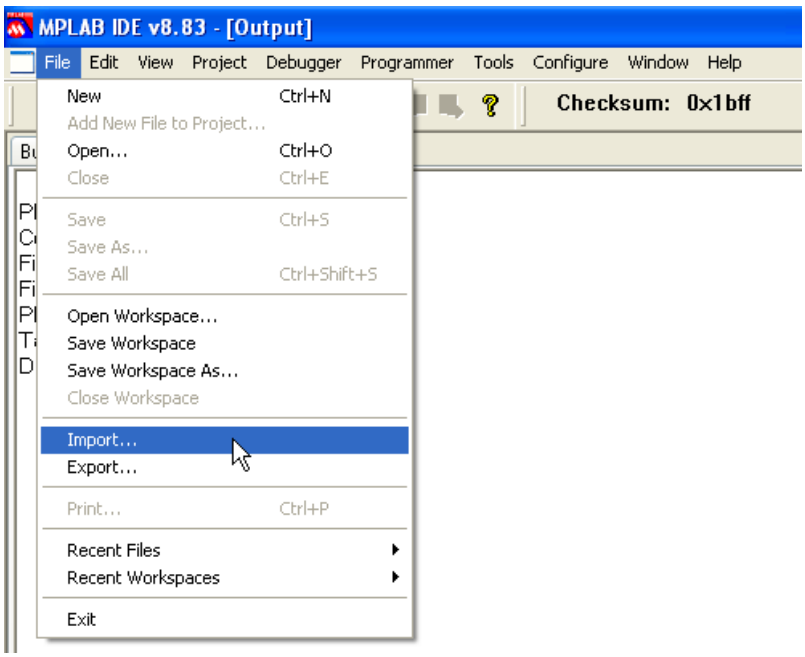
รูปที่ 7-4 แสดงคำเตือนให้ตรวจสอบการต่อไฟเลี้ยง

7.6 ถ้าบอร์ดเป้าหมายไม่มีปัญหาจะเจอเบอร์ของ MCU ที่เลือกใช้ดังรูปที่ 7-5



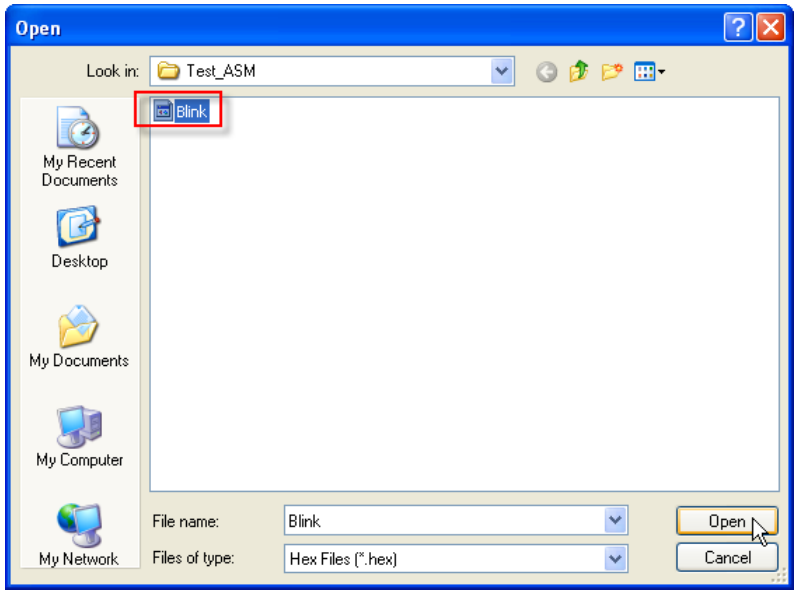
รูปที่ 7-5 แสดงเมื่อ ET-PGMPIC PK3 ติดต่อกับบอร์ดเป้าหมายสำเร็จ

7.7 จากนั้นเลือกที่เมนู File>Import ดังรูปที่ 7-6



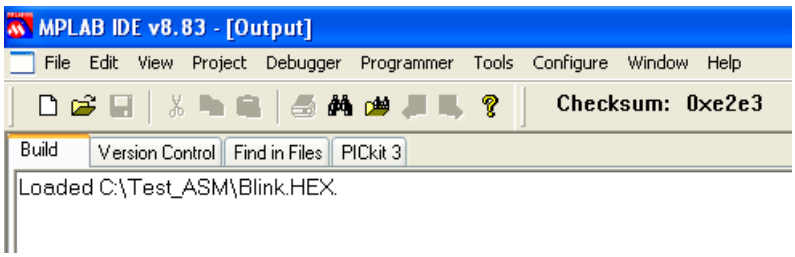
รูปที่ 7-6 แสดงวิธีการ Import ไฟล์

7.8 จากนั้นจะปรากฏหน้าต่าง Open ให้ทำการเลือก Hex File ที่ต้องการโปรแกรมและคลิกที่ปุ่ม Open ดังรูปที่ 7-7



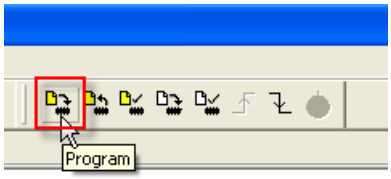
รูปที่ 7-7 แสดงการเลือก Hex File ที่ต้องการโปรแกรม

7.9 เมื่อการ Import Hex File เสร็จสมบูรณ์ก็จะปรากฏข้อความดังรูปที่ 7-8



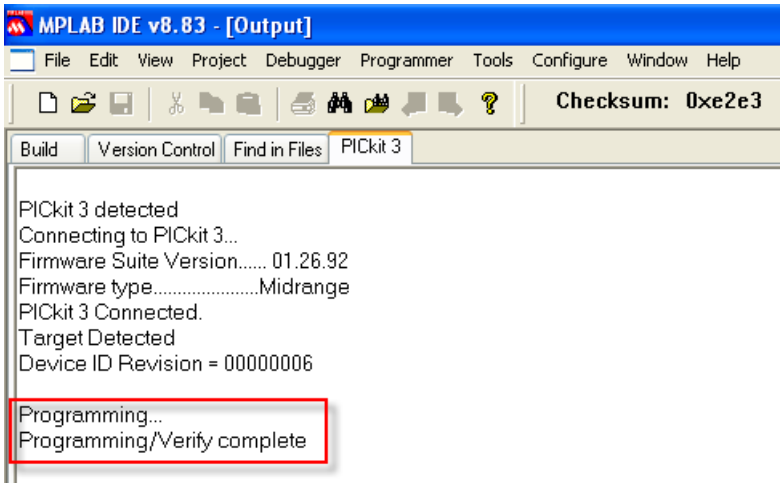
รูปที่ 7-8 แสดงเมื่อการ Import Hex File เสร็จสมบูรณ์

7.10 คลิกที่ไอคอน Program เพื่อโปรแกรม Hex File ดังรูปที่ 7-9



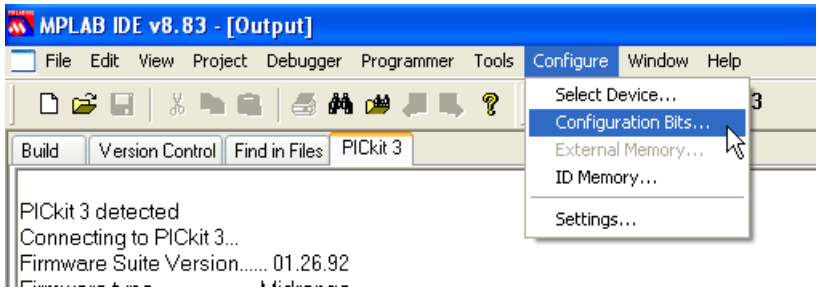
รูปที่ 7-9 แสดงวิธีการโปรแกรม Hex File

7.11 ถ้าการโปรแกรม Hex File ไม่พบข้อผิดพลาดก็ปรากฏข้อความดังรูปที่ 7-10



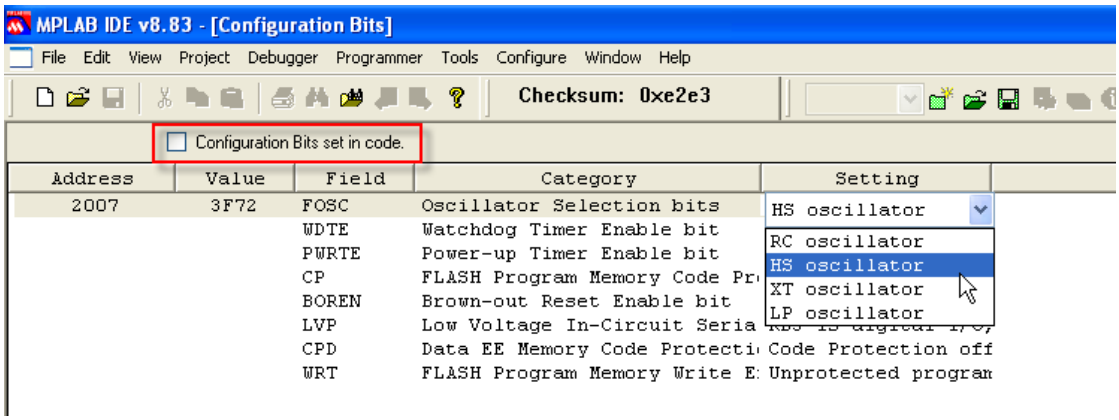
รูปที่ 7-10 แสดงเมื่อการโปรแกรม Hex File ไม่พบข้อผิดพลาด

7.12 โดยปกติแล้วค่า Configuration Bits ของ MCU จะเอามาจาก Hex File แต่ในกรณีที่ผู้ใช้ต้องการแก้ไขค่า Configuration Bits เองก็สามารถทำได้โดย เลือกที่เมนู **Configure>Configuration Bits…** ดังรูปที่ 7-11



รูปที่ 7-11 แสดงวิธีการแก้ไขค่า Configuration Bits

7.13 จากนั้นจะปรากฏหน้าต่าง Configuration Bits ให้เอาเครื่องหมายถูกหน้าช่อง Configuration Bits set in code. ออก เพื่อให้ผู้สามารถกำหนดค่า Configuration Bits เองได้ดังรูปที่ 7-12 เพื่อทำการแก้ไขค่า Configuration Bits เรียบร้อย ก็ให้คลิกที่ไอคอน Program target device เพื่อโปรแกรม Hex File ตามปกติ

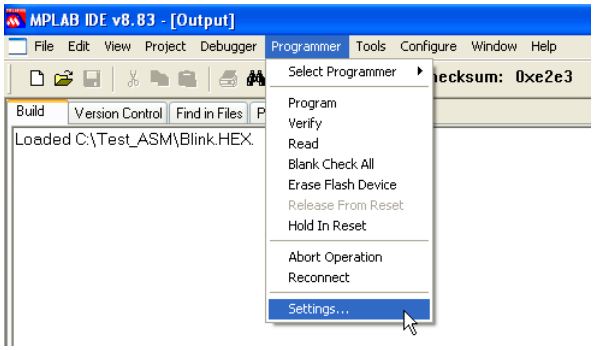


รูปที่ 7-12 แสดงวิธีการแก้ไขค่า Configuration Bits

8. การใช้งานโหมด Programmer-To-Go ด้วยโปรแกรม MPLAB IDE

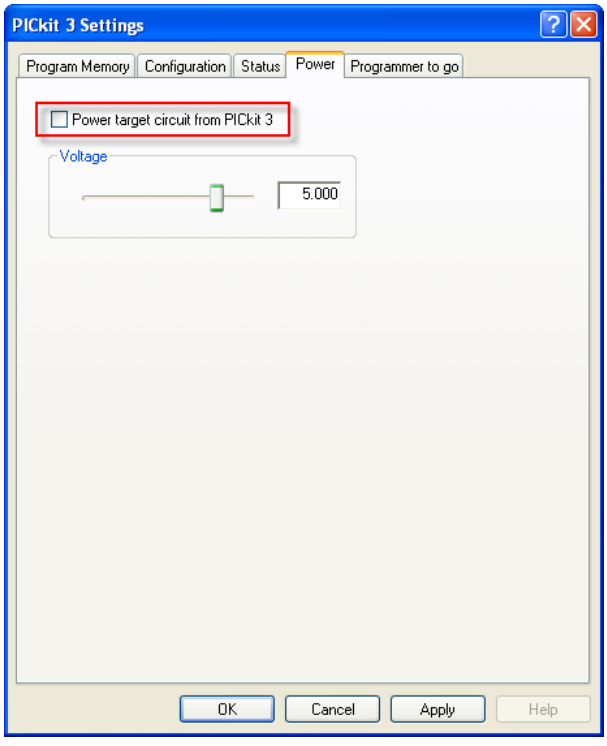
8.1 เปิดโปรแกรม MPLAB IDE เลือกเบอร์ที่ต้องการจะโปรแกรม และ Import Hex File ที่ต้องการจะโปรแกรม เหมือนขั้นตอนการโปรแกรมปกติ

8.2 จากนั้นให้เลือกเมนู **Programmer>Settings..** ดังรูปที่ 8-1



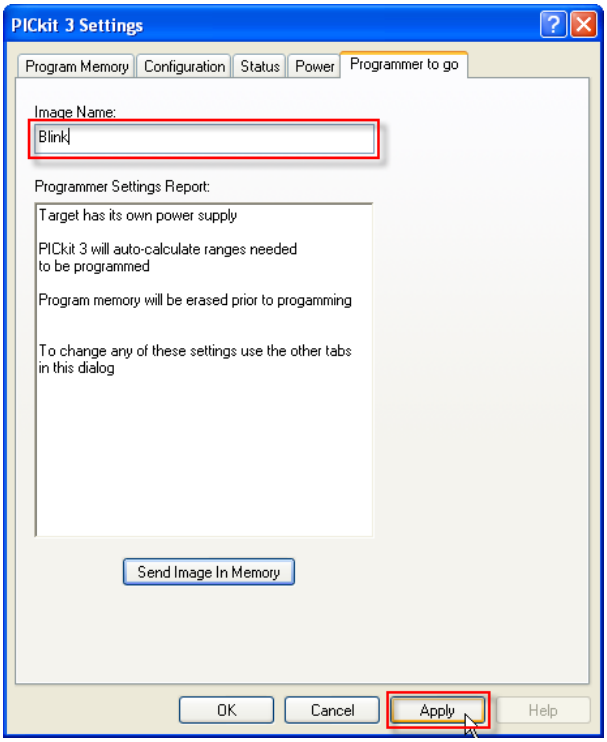
รูปที่ 8-1

8.3 จากนั้นจะปรากฏหน้าต่าง PICkit 3 Settings ให้เลือกไปที่แท็บ Power ดังรูปที่ 8-2 ถ้ากรณีที่โปรแกรมผ่านทาง TEXTOOL หรือใช้ไฟเลี้ยงจาก ET-PGMPIC PK3 เลี้ยงบอร์ดปลายทาง ในช่อง Power target circuit from PICkit 3 ให้เลือกถูกไว้ด้วย



รูปที่ 8-2 แสดงหน้าต่าง PICkit 3 Settings

8.4 จากนั้นไปที่แท็บ Programer to go ทำการตั้งชื่อ Image Name ตามต้องการ จากนั้นคลิกที่ปุ่ม Apply ดังรูปที่ 8-3



รูปที่ 8-3 แสดงหน้าต่าง PICkit 3 Settings

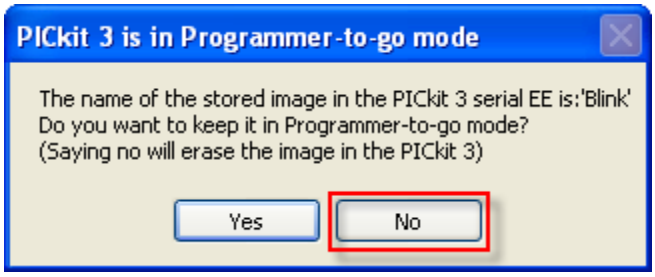
8.5 จากนั้นคลิกที่ปุ่ม **Send Image In Memory** โปรแกรมจะเริ่มโหลด image เมื่อโหลดเสร็จเรียบร้อยจะปรากฏข้อความ PICkit 3 removed ดังรูปที่ 8-4 พร้อมกันนั้น LED ACTIVE จะกระพริบ แสดงว่าการใช้งาน **Programmer-To-Go** พร้อมใช้งาน

```
Target Detected
Device ID Revision = 00000006

Downloading image...
Download of Programmer-to-go image complete
PICkit 3 is now in Programmer to go mode.

PICkit 3 will be disconnected from this session.
The next time you connect (or re-connect) to this unit, you will have the choice to take it off Programmer to go mode.

PICkit 3 removed
```

รูปที่ 8-4 แสดงเมื่อการโหลด image เสร็จเรียบร้อย

8.6 เมื่อทุกอย่างพร้อม การใช้งานผู้ใช้ก็เพียงแค่กดสวิตช์ **PROGRAM** บน ET-PGMPIC PK3 จะสังเกตเห็น LED ACTIVE หยุดกระพริบ และ LED STATUS จะติดเป็นสีส้ม เมื่อการโปรแกรมเสร็จเรียบร้อย LED ACTIVE จะกลับมากระพริบ และ LED STATUS จะกลายเป็นสีเขียว หลังจากนั้นผู้ใช้ก็สามารถปิดโปรแกรม MPLAB IDE และนำไปโปรแกรมบอร์ดอื่นๆ ได้เลยโดยไม่ต้องเปิดโปรแกรม

8.7 กรณีที่ต้องการโปรแกรม Hex File ตัวใหม่ หรือ กลับไปใช้งานในโหมดปกติ ให้เปิดโปรแกรม MPLAB IDE ทำการเชื่อมต่อ ET-PGMPIC PK3 เข้ากับคอมพิวเตอร์ จากนั้นจะปรากฏข้อความบอกว่า PICkit 3 อยู่ในโหมด Programmer-To-Go ดังรูปที่ 8-5 ให้เลือก No เพื่อเข้าสู่โหมดปกติ และข้อมูลต่างๆ จะถูกลบ



รูปที่ 8-5 แสดงการกลับไปใช้งานในโหมดปกติ