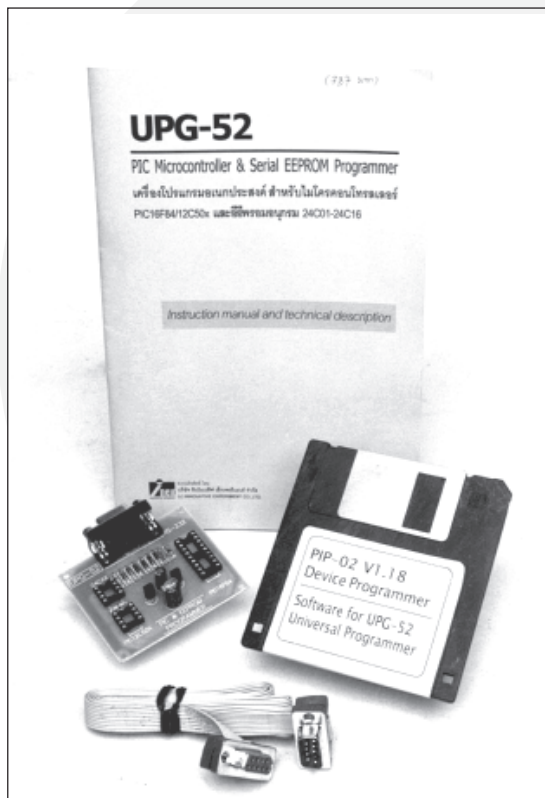


UPG-52

เครื่องโปรแกรมเมอร์
สำหรับไมโครคอนโทรลเลอร์

PIC16F84/12C50x

และอีอีพรอมแบบอนุกรมเบอร์ 24Cxx



งบประมาณ 737 บาท

● กฤษดา ใจเย็น

เครื่องมือพัฒนาระบบไมโครคอนโทรลเลอร์ขนาดเล็ก
ราคาประหยัด ใช้โปรแกรมไมโครคอนโทรลเลอร์ PIC
และหน่วยความจำอีอีพรอมอนุกรม 24 ใช้งานง่าย,
ทำงานโดยไม่ต้องใช้แหล่งจ่ายไฟจากภายนอก

ปัจจุบันไมโครคอนโทรลเลอร์ตระกูล PIC ได้รับความนิยมเพิ่มมากขึ้นมีการนำมาประยุกต์ใช้งานหลากหลายส่งผลให้มีผู้สนใจหันมาเรียนรู้และใช้งานกันอย่างมากมาย หากติดขัดในเรื่องของเครื่องมือที่ใช้ในการเขียนข้อมูลลงบนตัวไมโครคอนโทรลเลอร์ เนื่องจากมีราคาค่อนข้างสูง ข้อจำกัดดังกล่าวกำลังจะหมดไป เนื่องจากคุณๆ สามารถจัดหาเครื่องโปรแกรมไมโครคอนโทรลเลอร์ตระกูล PIC ได้ในราคาไม่ถึงพันบาท

UPG-52 คือเครื่องมือดังกล่าว โดยบนบอร์ด UPG-52 สามารถโปรแกรมไมโครคอนโทรลเลอร์ PIC ได้ 2 อนุกรม คือ PIC18C84/F84 และ PIC12C50x ซึ่งเพียงพอสำหรับนักเรียนที่กำลังศึกษาและทดลองใช้งานไมโครคอนโทรลเลอร์ตระกูลนี้อยู่ เนื่องจาก PIC16x84 เป็นไมโครคอนโทรลเลอร์ที่มีหน่วยความจำแบบสามารถลบและเขียนใหม่ได้ โดย PIC16C84 มีหน่วยความจำโปรแกรมเป็นอีอีพรอม ในขณะที่ PIC16F84 มีหน่วยความจำโปรแกรมเป็นแบบแฟลช แต่ในปัจจุบันมีการผลิตเพียงเบอร์ PIC16F84 ออกสู่ตลาดเท่านั้นและด้วยความสามารถในการลบและเขียนใหม่ได้นับพันครั้ง ทำให้การพัฒนากระบวนการควบคุมขนาดเล็กโดยใช้ PIC16F84 จึงน่าจะเพียงพอ

คุณสมบัติทางเทคนิค

- ใช้โปรแกรมไมโครคอนโทรลเลอร์ PIC16F84 และ PIC12C50x (12C508, 12C508A, 12C509, 12C509A)
- สามารถโปรแกรมหน่วยความจำอีอีพรอมอนุกรมเบอร์ 24C01-24C16 ได้
- ฮาร์ดแวร์มีขนาดเล็กเพียง 2x2.5 นิ้วเท่านั้น
- ไม่ต้องใช้แหล่งจ่ายไฟตรงจากภายนอก
- ซอฟต์แวร์ทำงานบนดอส
- มีฟังก์ชันตรวจสอบข้อมูลและเปรียบเทียบข้อมูล
- สามารถอ่านข้อมูลจากตัวชิป (ในกรณีที่ไม่มีการป้องกัน)
- สามารถบันทึกข้อมูลที่อ่านจากชิปเป็นแฟ้มข้อมูลได้
- เชื่อมต่อกับคอมพิวเตอร์ทางพอร์ตอนุกรม
- ใช้งานกับคอมพิวเตอร์โน้ตบุ๊กได้

ส่วนไมโครคอนโทรลเลอร์ PIC-12C50x ในปัจจุบันมี 2 เบอร์คือ PIC-12C508 และ PIC12C509 มีข้อเด่นในด้านขนาดที่เล็ก มีจำนวนอินพุต/เอาต์พุต 6 บิต มีวงจรกำเนิดสัญญาณนาฬิกาความถี่ 4 เมกะเฮิร์ตซ์อยู่ภายใน เพียงจ่ายไฟให้ก็สามารถทำงานได้แล้ว หน่วยความจำโปรแกรมของ PIC12C50x มีราคาถูกมาก หากมีการนำไปพัฒนาระบบควบคุมที่มีขนาดเล็กมากๆ จึงเหมาะที่จะใช้ PIC-12C50x เป็นตัวควบคุมการทำงาน

นอกจากนั้น UPG-52 ยังสามารถใช้อ่านและเขียนข้อมูล (หรือโปรแกรมข้อมูลนั่นเอง) กับหน่วยความจำอีอีพรอมแบบอนุกรม (Serial EEPROM) เบอร์ 24C01-24C16 ได้อีกด้วย

การโปรแกรมหน่วยความจำโปรแกรมของ PIC16F84

ไมโครคอนโทรลเลอร์ PIC16F84 มีการจัดขาตามรูปที่ 1 ในการโปรแกรม PIC16F84 ต้องจ่ายแรงดันไฟเลี้ยง +4.5

ถึง +5.5 โวลต์เข้าที่ขา VDD และแรงดันโปรแกรม Vpp +12 ถึง +14 โวลต์เข้าที่ขา MCLR สำหรับขาที่ใช้ในการโปรแกรมมีด้วยกัน 5 ขาดังนี้

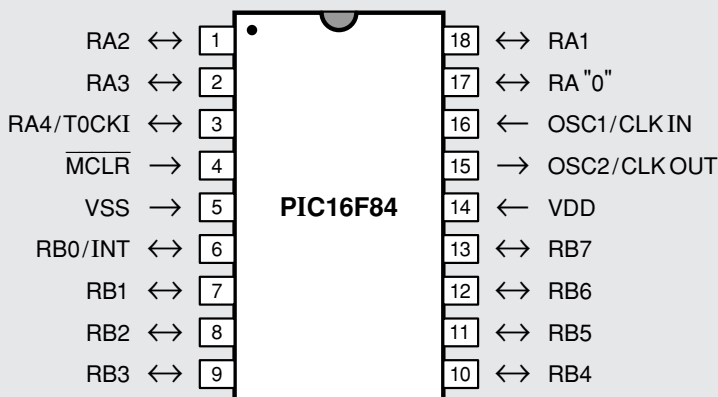
1. ขา RB6 เมื่ออยู่ในโหมดโปรแกรมจะใช้เป็นขาอินพุตสำหรับรับสัญญาณนาฬิกา
2. ขา RB7 เมื่ออยู่ในโหมดโปรแกรมจะใช้เป็นขาอินพุต/เอาต์พุตสำหรับถ่ายทอดข้อมูลในลักษณะอนุกรม
3. ขา $\overline{\text{MCLR}}$ เมื่ออยู่ในโหมดโปรแกรมจะเป็นขา Vpp เพื่อรับแรงดันโปรแกรมและใช้ในการกำหนดให้ PIC-16F84 เข้าสู่โหมดการโปรแกรม
4. ขา VDD เป็นขาต่อไฟเลี้ยงบวก
5. ขา VSS เป็นขากกราวด์

การโปรแกรมหน่วยความจำโปรแกรมของ PIC12C50X

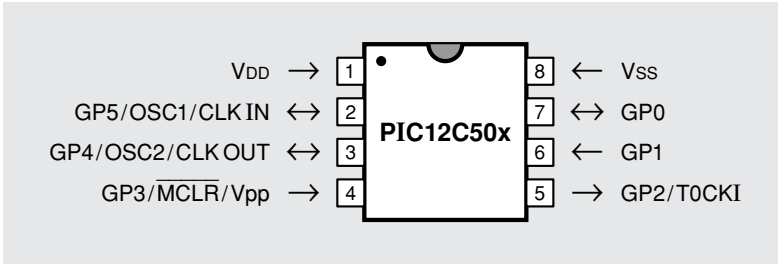
ไมโครคอนโทรลเลอร์ PIC12C50x มีการจัดขาตามรูปที่ 2 ขาที่ใช้ในการโปรแกรมมีทั้งสิ้น 5 ขาดังนี้

1. ขา GP1 เมื่ออยู่ในโหมดโปรแกรมจะใช้เป็นขาอินพุตรับสัญญาณนาฬิกา (clock)
2. ขา GP0 เมื่ออยู่ในโหมดโปรแกรม จะใช้เป็นขาอินพุตสำหรับถ่ายทอดข้อมูลหรือเป็นขาข้อมูล (data)
3. ขา GP3/ $\overline{\text{MCLR}}$ /Vpp เมื่ออยู่ในโหมดโปรแกรม เป็นขารับแรงดันไฟสูงสำหรับการโปรแกรม (+Vpp)
4. ขา VDD ต่อกับไฟเลี้ยง
5. ขา VSS ต่อกับกราวด์

ไมโครคอนโทรลเลอร์ PIC12C50x ต้องการไฟสูงในการโปรแกรม (Vpp) อยู่ในช่วง +12 ถึง +13.25 โวลต์ ค่าแรงดัน Vpp ที่เหมาะสมที่สุดคือ 13-13.25 โวลต์ กระแสไฟที่ต้องการประมาณ 40 มิลลิแอมป์ ส่วนไฟเลี้ยงชิปอยู่ในช่วง 2-6.5 โวลต์



รูปที่ 1 การจัดขาของไมโครคอนโทรลเลอร์ PIC16F84/PIC16C84



รูปที่ 2 การจัดขาของไมโครคอนโทรลเลอร์ PIC12C50x

ใน PIC12C50x มีด้วยกัน 2 รุ่น คือ รุ่นที่เป็นหน่วยความจำแบบอีพรอม สามารถเขียนและลบข้อมูลในหน่วยความจำโปรแกรมได้ โดยการลบต้องใช้แสงอัลตราไวโอเลตเหมือนกับการลบข้อมูลในอีพรอมทั่วไปและรุ่นที่สามารถโปรแกรมได้ครั้งเดียว (OTP : One Time Programmable) ความแตกต่างของ PIC-12C50x ทั้ง 2 รุ่น สังเกตได้จากตัวถังภายนอก หากเป็นรุ่นที่ลบได้จะมีกระจกรูปวงกลมติดตั้งอยู่ด้านบนตัวถัง

PIC12C50x มีด้วยกัน 4 เบอร์ คือ PIC12C508, PIC12C508A, PIC-12C509 และ PIC12C509A แต่ละเบอร์จะมีความแตกต่างกันในเรื่องขนาดของหน่วยความจำโปรแกรมและหน่วยความจำข้อมูลภายในตัวไมโครคอนโทรลเลอร์

รายละเอียดเบื้องต้นของ อีพรอมอนุกรม 24Cxx

อีพรอมอนุกรมเบอร์ 24CXX เป็นหน่วยความจำอีพรอมที่มีการติดต่อเป็นแบบ I²C บัส 2 สาย สำหรับอีพรอมอนุกรมที่เครื่อง UPG-52 สามารถโปรแกรมได้มี 5 เบอร์คือ 24C01, 24C02, 24C04, 24C08 และ 24C16 ซึ่งแต่ละเบอร์จะมีความจุแตกต่างกันตั้งแต่ 1 กิโลบิต จนถึง 16 กิโลบิต มีการจัดขาเหมือนกันหมดตามรูปที่ 3 และมีรายละเอียดของอีพรอมแต่ละเบอร์ดังนี้

- 24C01 มีความจุ 1 กิโลบิต หรือ 128 ไบต์มีการจัดสรรหน่วยความ

จำเป็น 128 เพจ แต่ละเพจมีขนาด 1 ไบต์ (8 บิต) ต้องการข้อมูลเพื่ออ้างอิงแอดเดรสจำนวน 7 บิต

- 24C02 มีความจุ 2 กิโลบิต หรือ 256 ไบต์ มีการจัดสรรหน่วยความจำเป็น 250 เพจ เพจละ 1 ไบต์ ต้องการข้อมูลเพื่ออ้างอิงแอดเดรสจำนวน 8 บิต

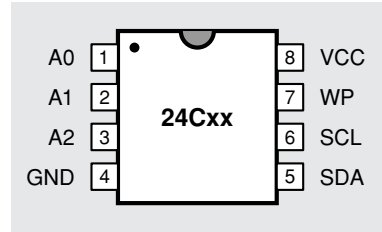
- 24C04 มีความจุ 4 กิโลบิต หรือ 512 ไบต์ มีการจัดสรรหน่วยความจำเป็น 256 เพจ เพจละ 2 ไบต์ ต้องการข้อมูลเพื่ออ้างอิงแอดเดรสจำนวน 9 บิต

- 24C08 มีความจุ 8 กิโลบิต หรือ 1,024 ไบต์ มีการจัดสรรหน่วยความจำเป็น 4 บล็อก บล็อกละ 256 เพจ แต่ละเพจบรรจุ 4 ไบต์ ต้องการข้อมูลเพื่ออ้างอิงแอดเดรสจำนวน 10 บิต

- 24C016 มีความจุ 16 กิโลบิต หรือ 2,048 ไบต์ มีการจัดสรรหน่วยความจำเป็น 8 บล็อก บล็อกละ 256 เพจ แต่ละเพจบรรจุ 8 ไบต์ ต้องการข้อมูลเพื่ออ้างอิงแอดเดรสจำนวน 11 บิต

ขาสัญญาณอีพรอมอนุกรม 24Cxx ที่ใช้ในการโปรแกรม

อีพรอมอนุกรม 24Cxx มีขาต่อใช้งาน 8 ขาคือ ขาค่อไฟเลี้ยง 2 ขา (+VCC และ GND), ขากำหนดแอดเดรส 3 ขา (A0-A2) และขาที่ใช้ในการติดต่อแบบ I²C บัส 2 ขา (SCL และ SDA) สำหรับขาที่ใช้ในการโปรแกรม มี 2 ขาคือ SCL และ SDA



รูปที่ 3 การจัดขาของหน่วยความจำอีพรอมอนุกรมเบอร์ 24Cxx (24C01-24C16)

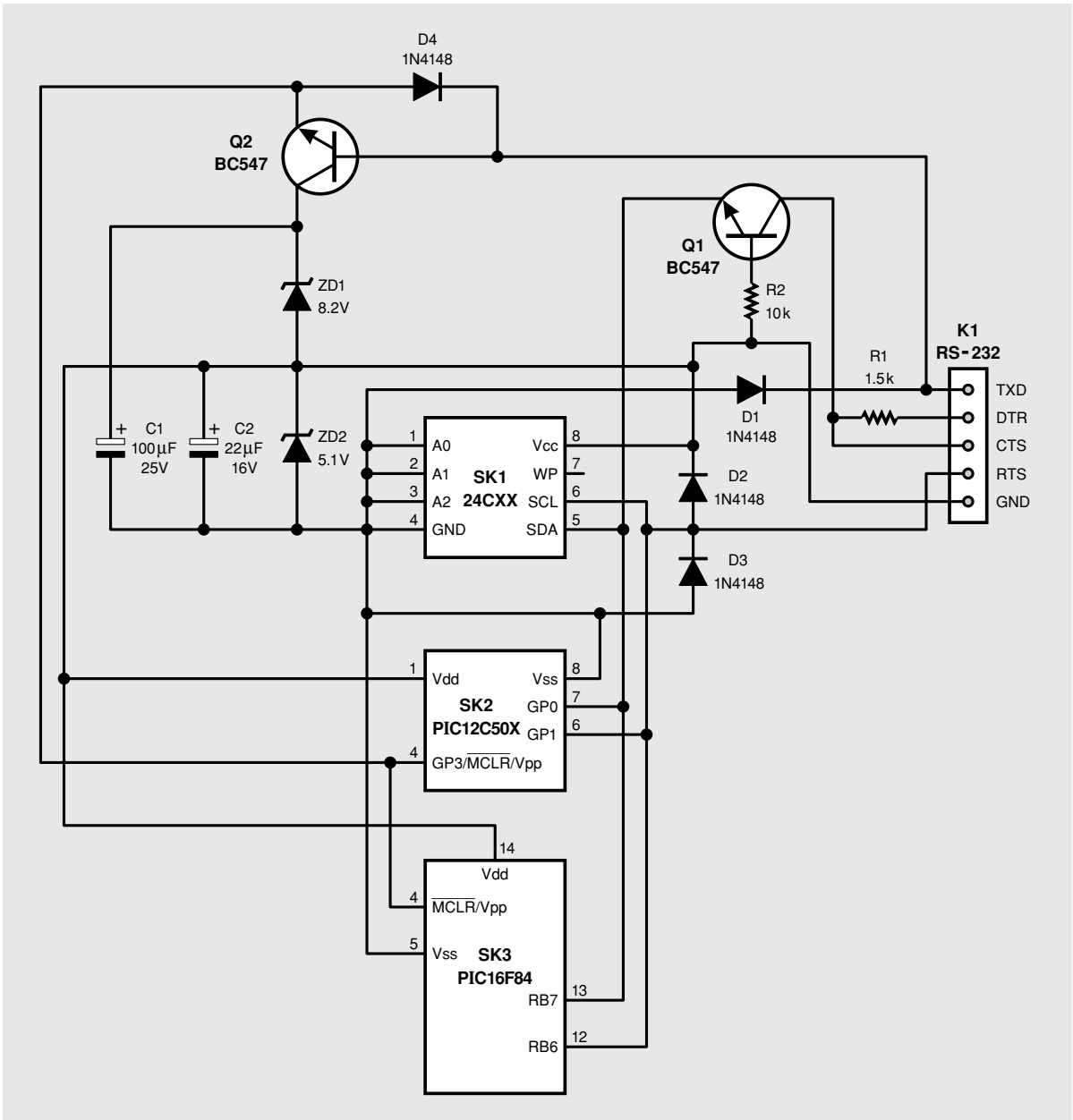
ขา SCL (Serial Clock input) เป็นขาสำหรับป้อนสัญญาณนาฬิกา หากต้องการป้อนข้อมูลเข้าอีพรอมอนุกรมจะต้องป้อนสัญญาณนาฬิกาขอบขาขึ้นเข้าที่ขา SCL นี้ ในทางตรงกันข้ามเมื่อต้องการอ่านข้อมูลออกจากหน่วยความจำต้องกำหนดจังหวะการทำงานที่ขอบขาลงของสัญญาณนาฬิกาให้แก่ขา SCL นี้

ขา SDA (Serial Data) ขานี้จะเป็นขาค่ายทอดข้อมูลในลักษณะอนุกรมข้อมูลสามารถผ่านเข้าและออกได้

อันที่จริงสาระประกอบในการเรียนรู้การโปรแกรมตัวไมโครคอนโทรลเลอร์ PIC16F84/12C50x และอีพรอมอนุกรม 24C01-24C16 ยังมีอีกมากไม่ว่าจะเป็นเรื่องการจัดสรรหน่วยความจำโปรแกรม, กระบวนการโปรแกรมไปจนถึงคำสั่งการโปรแกรม แต่ทว่าเราไม่สามารถที่จะนำเสนอได้หมดในบทความชุดนี้เพียงชุดเดียว เราจึงได้นำเสนอเพียงฮาร์ดแวร์ที่จำเป็นในการโปรแกรมของอุปกรณ์แต่ละตัวเท่านั้น ท่านผู้อ่านสามารถที่จะหาข้อมูลเพิ่มเติมได้ที่ www.microchip.com หรือเอกสารข้อมูลที่เป็นภาษาไทยก็มีแถมให้กับชุดสำเร็จที่สามารถดูรายละเอียดได้ที่ท้ายบทความ

รายละเอียดการทำงานของวงจร

หลังจากอธิบายหลักการโปรแกรม PIC16F84, PIC12C50x และ 24Cxx กันอย่างคร่าว ๆ แล้ว ต่อไปจะเป็นรายละเอียด



รูปที่ 4 วงจรสมบูรณของ UPG-52 เครื่องโปรแกรมอเนกประสงค์

เียะการทํางานของ UPG-52 เครื่องโปรแกรมอเนกประสงค์ ในรูปที่ 4 เป็นวงจรสุมบูรณ์ของ UPG-52 จะเห็นได้ว่ามีการแบ่งแยกช็อกเก็ตสำหรับติดตั้งอุปกรณ์ที่ต้องการโปรแกรม โดย SK1 เป็นช็อกเก็ต 8 ขา สำหรับติดตั้ง 24Cxx ส่วน SK2 เป็นช็อกเก็ต 8 ขา เช่นกัน สำหรับติดตั้ง PIC12C50x และ SK3 เป็นช็อกเก็ต 18 ขา สำหรับติดตั้ง PIC16F84

UPG-52 เชื่อมต่อกับคอมพิวเตอร์พีซีผ่านทางพอร์ตอนุกรม RS-232 ใช้ขาสัญญาณ TXD, DTR, CTS, RTS และ GND รวม 5 ขา ในการโปรแกรมและอ่านค่าของข้อมูลจากตัวอุปกรณ์ และจะเห็นได้ว่าใน UPG-52 มีอุปกรณ์เพียงไม่กี่ตัว แต่สามารถสร้างเป็นเครื่องโปรแกรมไมโครคอนโทรลเลอร์และอีมูเลชันออนบอร์ดแสดงให้เห็นถึง ความสะดวกใน

การพัฒนาไมโครคอนโทรลเลอร์ PIC ที่ง่ายและประหยัด

ใน UPG-52 นี้ จะกำหนดให้ CTS เป็นขาสำหรับรับข้อมูลจากชิปที่ทำการโปรแกรม ขา DTR เป็นขาส่งข้อมูลที่ต้องการโปรแกรม ส่วนขา RTS เป็นขาป้อนสัญญาณนาฬิกาให้แก่ชิปที่ทำการโปรแกรมขา TXD ใช้ในการอินพุตแรงดันไฟสูงสำหรับโปรแกรม PIC16F84

และ PIC12C50x สุดท้ายขา GND ใช้ในการกำหนดไฟเลี้ยงให้แก่ชิปที่ทำการโปรแกรมทุกตัว การกำหนดหน้าที่ต่างๆ ขาสัญญาณพอร์ตอนุกรมมาจากซอฟต์แวร์ที่ใช้ในการโปรแกรม นั่นคือ PIP-02 Device programmer ซึ่งจะได้อ่านถึงการใช้งานต่อไป

ไดโอด D2 และ D3 ทำหน้าที่จำกัดระดับแรงดันที่ขาอินพุตสัญญาณนาฬิกาของอุปกรณ์ทั้งหมด ไม่ว่าจะเป็นขา SCL สำหรับ 24Cxx ขา GP1 สำหรับ PIC12C50x และขา RB6 สำหรับ PIC16F84 ทั้งนี้เนื่องจากพอร์ตอนุกรมของคอมพิวเตอร์อาจมีระดับแรงดันสูงถึง +12 โวลต์ที่ลอจิกสูงสำหรับคอมพิวเตอร์บางเครื่อง

ทรานซิสเตอร์ Q1 ใช้ในการขยายระดับสัญญาณของ RS-232 เพื่อให้ขาข้อมูลของอุปกรณ์ทั้งหมด ไม่ว่าจะเป็นขา SDA ของ 24Cxx, ขา GP0 ของ PIC-12C50x และขา RB7 ของ PIC16F84 มีระดับลอจิกที่แน่นอน ส่งผลให้ข้อมูลที่ทำการอ่านและเขียนมีความถูกต้องแม่นยำ โดย Q2 ต้องวงจรเป็นลักษณะเบสรวมทำหน้าที่เป็นวงจรขยาย 2 ทิศทางไปในตัว เนื่องจากข้อมูลต้องมีการถ่ายทอดสัญญาณเข้าและออกตลอดเวลาที่ทำการโปรแกรม เมื่อต้องการเขียนข้อมูลมายังตัวชิป ขา TXD จะมีระดับลอจิกสูง ทำให้ Q2 ทำงาน จนถึงจุดอิ่มตัวขาอิมิตเตอร์และคอลเล็กเตอร์ของ Q2 เสมือนลัดวงจรเกิดแรงดันไฟสูง +13.3 โวลต์ป้อนมาเข้าที่ขา \overline{MCLR} ของ PIC12C50x และ PIC16F84 นอกจากนั้นยังทำให้เกิดแรงดันไฟเลี้ยงตัวชิปทั้งหมดด้วย โดยมีซีเนอร์ไดโอด ZD2 (5.1V) เป็นตัวควบคุมให้คงที่ที่ประมาณ 5 โวลต์

ในขณะเดียวกัน ทรานซิสเตอร์ Q1 จะถูกกระตุ้นให้ทำงาน ข้อมูลจะถูกส่งมาจากขา DTR ผ่าน R1 ผ่าน Q1

เข้าไปยังขาข้อมูล ในขณะที่สัญญาณนาฬิกาจะถูกส่งมาจากขา RTS ผ่านการควบคุมระดับแรงดันด้วย D2 จริงอยู่ที่สัญญาณที่ RTS เป็นพัลส์ แต่เมื่อผ่าน D2 ก็เหมือนกับผ่านการเร็กติไฟร์มี C2 ช่วยกรองแรงดันให้เรียบขึ้น และมีซีเนอร์ไดโอด ZD2 ควบคุมให้แรงดันคงที่ ที่ประมาณ +5 โวลต์

ทรานซิสเตอร์ Q1 ยังคงได้รับการไบแอสอยู่ ข้อมูลที่ออกจากชิปจึงเป็น "1" จะผ่าน Q1 ออกทางขา CTS ถ้าเป็น "0" ข้อมูลจะไม่ผ่าน Q1 แต่จะใช้สถานะที่ขา DTR ซึ่งเป็นลอจิกต่ำส่งข้อมูล "0" ออกไปทางขา CTS แทน ดังนั้น UPG-52 จึงสามารถโปรแกรมหรืออ่านข้อมูลจากตัวชิปได้ครั้งละ 1 เบอร์เท่านั้น และต้องคิดตั้งชิปเบอร์ที่ต้องการเขียนหรืออ่านเพียงตัวเดียว ส่วนข้อผิดพลาดที่เหลือให้เว้นว่างไว้ โดยที่

- ไมโครคอนโทรลเลอร์ PIC-16F84 และ PIC16C84 ใช้ข้อผิดพลาด SK3 บนบอร์ด UPG-52
- ไมโครคอนโทรลเลอร์ PIC-12C508 และ PIC12C509 ใช้ข้อผิดพลาด SK2
- หน่วยความจำอีพ롬แบบ I2C บัสเบอร์ 24C01 ถึง 24C16 ใช้ข้อผิดพลาด SK3

การใช้งานเครื่องโปรแกรม UPG-52 และซอฟต์แวร์ PIP-02 Device Programmer

ซอฟต์แวร์ที่ใช้ในการโปรแกรมข้อมูลสำหรับบอร์ด UPG-52 จะใช้โปรแกรม PIP-02 Device Programmer ร่วมกับ โปรแกรมไดรเวอร์ JDM84 โดยเริ่มต้นด้วยการโหลดโปรแกรมไดรเวอร์ก่อน แล้วจึงค่อยเรียกโปรแกรม PIP-02

โปรแกรมไดรเวอร์ JDM84

โปรแกรมไดรเวอร์ JDM84 นั้นเป็นโปรแกรมประเภทฝั่งตัวในหน่วยความจำออกแบบมาเพื่อใช้กับบอร์ด UPG-52 โดยเฉพาะมีรูปแบบการใช้งานดังนี้

JDM84 [COMn], [REMOVE]

โดย COMn หมายถึง พอร์ตอนุกรมที่ใช้งานโดยกำหนดได้ตั้งแต่ COM1 ถึง COM4

REMOVE หมายถึงการลบโปรแกรมที่ฝังตัวอยู่ในหน่วยความจำออกโปรแกรมที่ฝังตัวอยู่ในหน่วยความจำออกโปรแกรม JDM84 เพียงอย่างเดียวที่คอสพร้อมท์ โปรแกรม JDM84 จะยังไม่โหลดข้อมูลลงหน่วยความจำ แต่จะแสดงข้อความเพื่อให้ใส่คำสั่งต่อท้ายลงไป ดังนี้

C:\jdm84>jdm84

Syntax:

JDM84 COMn - Install on COMn

JDM84 REMOVE - Remove JDM84

ในกรณีที่ป้อนคำสั่ง JDM84 COM2 แต่ที่จอภาพแสดงข้อความแจ้งความผิดพลาด ดังตัวอย่างต่อไปนี้

C:\jdm84>jdm84 COM2

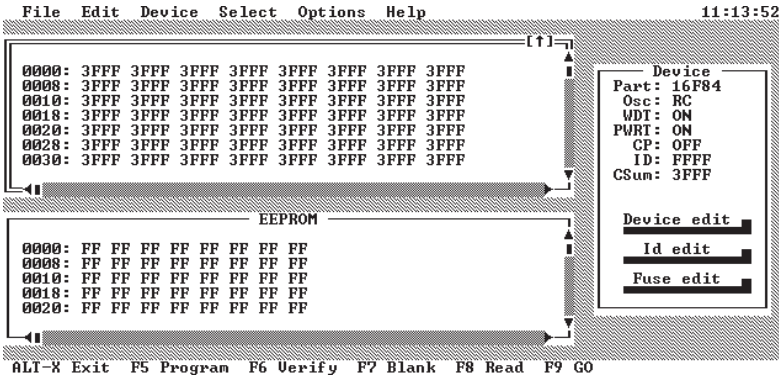
Programmer not found at COM2

หมายความว่า ขณะนี้ยังไม่ได้ต่อสายจากพอร์ตอนุกรม COM2 ไปยังบอร์ด UPG-52 ให้ทำการต่อสายให้เรียบร้อยก่อน จากนั้นพิมพ์คำสั่ง JDM84 COM2 ใหม่อีกครั้ง ที่จอภาพจะปรากฏข้อความ

C:\jdm84>jdm84 COM2

PINAPI Driver Installed on COM2

แสดงว่าโปรแกรมไดรเวอร์ได้ถูกโหลดและฝังตัวอยู่ในหน่วยความจำเรียบร้อยแล้ว หลังจากนั้นสามารถเรียกโปรแกรม



รูปที่ 5 หน้าตาของโปรแกรม PIP-02 เมื่อเรียกเข้าครั้งแรก

PIP02.EXE ขึ้นมาใช้งานได้ทันทีโดยพิมพ์ PIP02 ที่คอสพร้อมท์ ซึ่งจะปรากฏโปรแกรม PIP-02 ขึ้นมาที่หน้าจอแสดงในรูปที่ 5

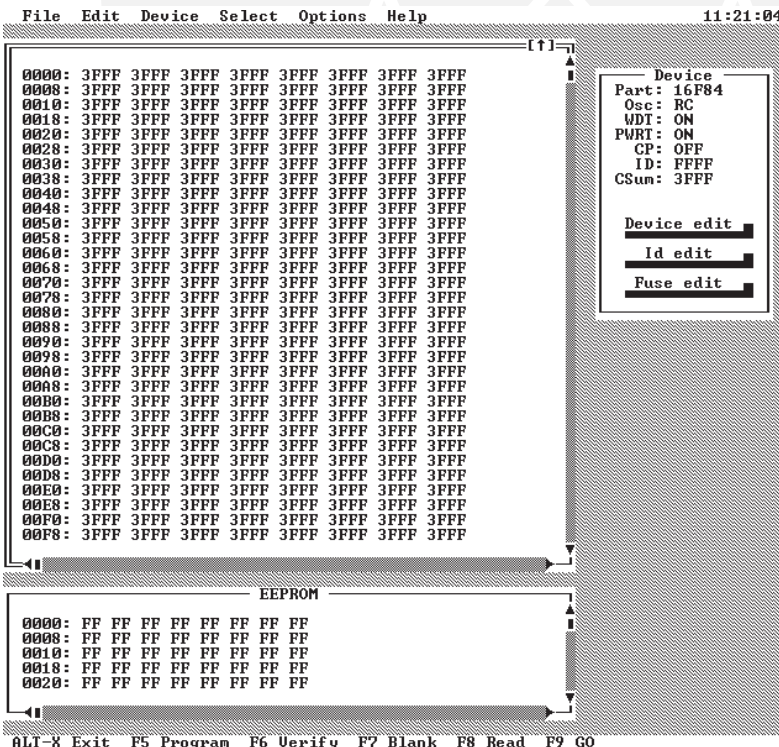
เมื่อเลิกใช้งานโปรแกรม PIP-02 แล้วจะต้องลบโปรแกรม JDM84 ออกจากหน่วยความจำด้วย โดยพิมพ์

```
C:\jdm84>jdm84 REMOVE
ที่จอภาพจะแสดงข้อความ
Driver uninstalled
```

ขั้นตอนต่าง ๆ ดังที่กล่าวมานี้ จะเป็นเรื่องยุ่งยาก ดังนั้นคำสั่งต่าง ๆ ทั้งหมดจึงถูกบรรจุอยู่ในแบตเตอรี่ฟลैชชื่อ UPG-52.BAT โดยการเรียกใช้งานในกรณีที่เครื่องโปรแกรม UPG-52 ต่ออยู่กับพอร์ต COM2 ทำได้ดังนี้

```
C:\jdm84>UPG-52 COM2
```

เมื่อออกจากโปรแกรม PIP-02 แล้วโปรแกรมจะทำการลบโปรแกรม JDM84 ออกจากหน่วยความจำโดยอัตโนมัติ



รูปที่ 6 หน้าตาของโปรแกรม PIP-02 ในขณะที่เลือกเป็น PIC16F84

การใช้งาน

PIP-02 Device Programmer

PIP-02 Device Programmer เป็นโปรแกรมที่ทำงานบนระบบปฏิบัติการดอส แต่ก็ให้คุณสมบัติของเมนูต่างๆ ใกล้เคียงกับการใช้งานโปรแกรมบนวินโดวส์ ผู้ใช้งานไม่จำเป็นต้องมีความเชี่ยวชาญในการใช้งานโปรแกรมก็สามารถใช้งานได้ โดยหน้าตาของ PIP-02 Device Programmer กรณีใช้งานกับ PIC16F84 แสดงในรูปที่ 6

เมื่อนำมาใช้งานกับไมโครคอนโทรลเลอร์ PIC16C84 และ PIC16F84 โปรแกรม PIP-02 Device Programmer จะปรากฏหน้าต่าง 3 หน้าต่างด้วยกันคือ

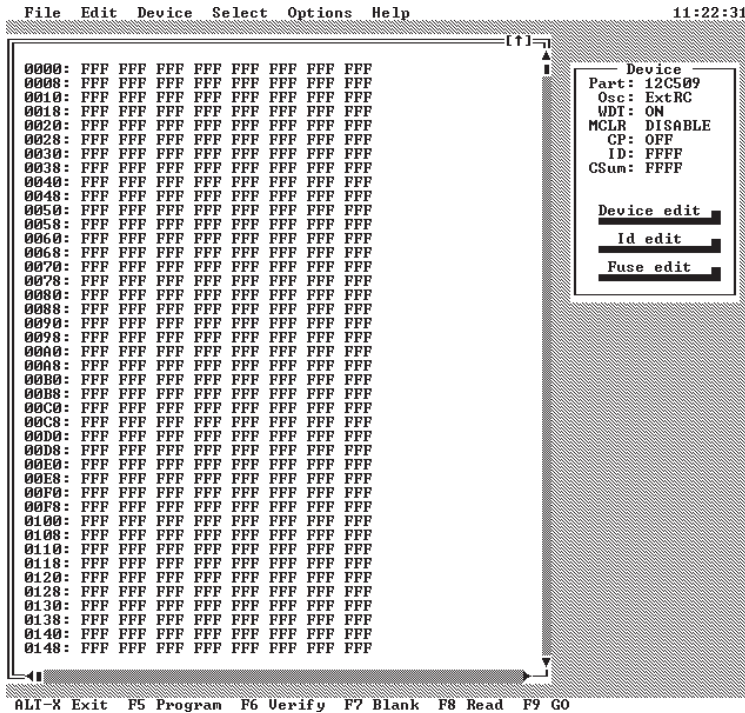
1. หน้าต่างของหน่วยความจำโปรแกรม หรือไฟล์นามสกุล .HEX ที่จะใช้โปรแกรมลงบน PIC16F84 หรือ PIC16C84 นั่นเอง โดยหน้าต่างนี้เมื่อโหลดข้อมูลเข้ามาแล้วยังสามารถแก้ไขข้อมูลก่อนที่จะทำการโปรแกรมได้อีก

2. หน้าต่างของหน่วยความจำข้อมูลอีพ롬 ใช้ในการแสดงค่าของข้อมูลในหน่วยความจำข้อมูลอีพ롬ภายใน PIC16F84 หรือ PIC16C84 ข้อมูลที่แสดงในหน้าต่างนี้ผู้ใช้งานสามารถทำการแก้ไขก่อนหน้าที่จะทำการโปรแกรมได้อย่างอิสระอีกด้วย

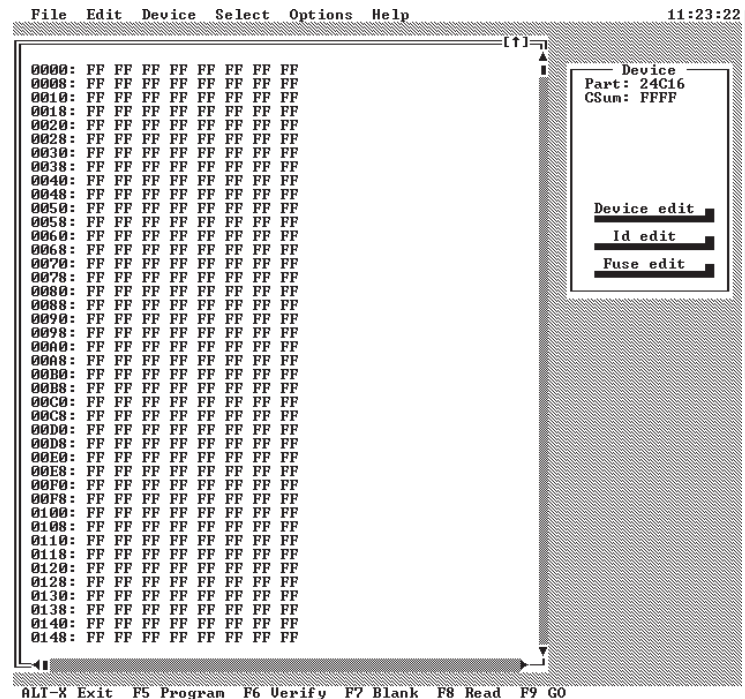
3. หน้าต่าง Device ที่หน้าต่างนี้ผู้ใช้งานสามารถเลือกเบอร์ของไมโครคอนโทรลเลอร์ที่ต้องการโปรแกรมพร้อมกับกำหนดค่าพารามิเตอร์ที่เกี่ยวข้อง อาทิ เลือกชนิดของวงจรรักษาเน็ดสัญญาณนาฬิกา หรือสซิทเลเตอร์ ต้องการเลือกใช้วอร์คซ์-ด็อกไทเมอร์และเพาเวอร์อัปไทเมอร์หรือไม่ รวมทั้งเลือกการป้องกันการอ่านหน่วยความจำโปรแกรม

สำหรับการใช้งานไมโครคอนโทรลเลอร์ PIC12C508 และ PIC12C509

ซึ่งไม่มีหน่วยความจำอีพროมภายใน ต่างเท่านั้นคือ หน้าต่างของหน่วยความจำโปรแกรมและหน้าต่างของ DEVICE ดังรูปที่ 7



รูปที่ 7 หน้าตาของโปรแกรม PIP-02 ในขณะที่เลือกเป็น PIC12C509



รูปที่ 8 หน้าตาของโปรแกรม PIP-02 เมื่อใช้งานกับหน่วยความจำอีพโรม 24Cxx

สำหรับการใช้งานกับอีพโรม 24C01-24C16 ก็เช่นเดียวกัน จะเหลือเพียง 2 หน้าต่างคือหน้าต่างของหน่วยความจำอีพโรมและหน้าต่างของ DEVICE ดังรูปที่ 8

เมนูใช้งานของ PIP-02

Device Programmer

เมนู FILE

เช่นเดียวกับการใช้โปรแกรมบนวินโดวส์ เมนู File ก็จะประกอบไปด้วยคำสั่งต่างๆ ดังนี้

LOAD เป็นคำสั่งเพื่อเรียกไฟล์นามสกุล .HEX ที่ต้องการโปรแกรมลงไปในตัวไมโครคอนโทรลเลอร์

SAVE และ **SAVE AS** เป็นคำสั่งสำหรับบันทึกไฟล์นามสกุล .HEX ที่อ่านจากไมโครคอนโทรลเลอร์หรือบันทึกไฟล์ที่ทำการแก้ไขข้อมูลโดยตรงด้วย PIP-02 Device Programmer

Change Dir เป็นคำสั่งเพื่อเปลี่ยนไดเรกทอรีไปยังตำแหน่งที่ต้องการ

EXIT (Alt+X) ใช้เพื่อออกจากโปรแกรม PIP-02

เมนู EDIT

เมนูนี้ประกอบด้วยคำสั่ง 3 คำสั่งเพื่อแก้ไขข้อมูลของหน่วยความจำโปรแกรมได้แก่

Clear buffer เป็นคำสั่งเพื่อใช้เคลียร์ค่าในหน่วยความจำโปรแกรมที่โหลดมาเก็บไว้ก่อนการโปรแกรมทั้งหมด

Clear fuses เป็นคำสั่งสำหรับการเคลียร์ค่าของพารามิเตอร์ต่างๆ ให้เป็นค่าเริ่มต้นทั้งหมด (default)

Fill buffer เป็นคำสั่งสำหรับการแก้ไขค่าหน่วยความจำโปรแกรมครั้งละหลายๆ ตำแหน่งในคราวเดียว โดยจะต้องกำหนดแอดเดรสเริ่มต้นและแอดเดรส

สิ้นสุดของหน่วยความจำก่อน จากนั้นจึงป้อนค่าที่ต้องการเปลี่ยนแปลงลงในช่อง DATA สำหรับอ็ีพროมค่าข้อมูลที่ป้อนให้กับช่อง DATA จะต้องเป็นข้อมูลเลขฐาน 16 ขนาด 8 บิต

เมนู Device

เมนูนี้ถ้ายังไม่โหลดใครเวอร์ JDM84 คำสั่งต่างๆ ภายในเมนูจะมีสีจางลง ซึ่งหมายความว่าไม่สามารถเรียกใช้คำสั่งใดๆ ได้เมื่อโหลดใครเวอร์แล้วเมนูนี้มีคำสั่งต่างๆ ให้ใช้งานดังนี้

Program (F5) เป็นคำสั่งเพื่อเขียนข้อมูลที่เก็บอยู่ในหน่วยความจำโปรแกรม, หน่วยความจำข้อมูลอ็ีพโรมและค่าพารามิเตอร์ต่างๆ ลงบนไมโครคอนโทรลเลอร์หรืออ็ีพโรมซึ่งใช้งานเฉพาะกับไมโครคอนโทรลเลอร์ เมื่อใช้งานกับอ็ีพโรมเมนูนี้จะมีสีจางลง เมื่อสั่งโปรแกรมแล้ว หากมีข้อผิดพลาดจะปรากฏหน้าต่างแสดงความผิดพลาดขึ้นมา แต่วิธีสังเกตอาจจะใช้วิธีการฟังจากเสียงหลังจากการโปรแกรมเรียบร้อยแล้วซึ่งจะมีเสียงที่แตกต่างกัน

Program fuses คำสั่งนี้จะทำการโปรแกรมเฉพาะค่าพารามิเตอร์สำหรับไมโครคอนโทรลเลอร์เท่านั้น

Erase คำสั่งนี้จะทำการลบข้อมูลในไมโครคอนโทรลเลอร์ PIC16F84 และ PIC16C84 เท่านั้น ส่วนไมโครคอนโทรลเลอร์เบอร์อื่นๆ และอ็ีพโรมเมนูนี้จะมีสีจางลง

Verify (F6) เป็นคำสั่งสำหรับการเปรียบเทียบข้อมูลที่เก็บอยู่ในไมโครคอนโทรลเลอร์หรือ อ็ีพโรมกับข้อมูลที่เก็บอยู่ใน PIP-02 Device Programmer ว่ามีค่าตรงกันหรือไม่เพื่อเป็นการตรวจสอบความถูกต้องหลังการโปรแกรมข้อมูลลงบนไมโครคอนโทรลเลอร์และอ็ีพโรม

Blank Check (F7) เป็นคำสั่งเพื่อตรวจสอบหน่วยความจำโปรแกรมภายในไมโครคอนโทรลเลอร์หรือหน่วยความจำอ็ีพโรม ว่ามีข้อมูลอยู่หรือไม่ ถ้ามีข้อมูลอยู่ PIP-02 Device Programmer จะแสดงข้อความ “Device NOT Blank”

Read (F8) เป็นคำสั่งเพื่ออ่านค่าข้อมูลหน่วยความจำจากไมโครคอนโทรลเลอร์หรืออ็ีพโรมขึ้นมาเก็บไว้ที่ หน่วยความจำของโปรแกรม PIP-02 Device Programmer

Reset/go (F9) คำสั่งนี้ใช้สำหรับการโปรแกรมไมโครคอนโทรลเลอร์ในวงจร (In-circuit programming) เมื่อใช้คำสั่งนี้จะเป็นการออกจากโหมด Program ไปทำงานที่โหมด Run แทน มีใช้สำหรับเครื่องโปรแกรมบางรุ่นเท่านั้น

เมนู Select

ในเมนูนี้คำสั่งให้ใช้งานทั้งสิ้น 3 คำสั่ง

Device เป็นคำสั่งสำหรับเลือกเบอร์ของไมโครคอนโทรลเลอร์หรืออ็ีพโรม

Fuse Word เป็นคำสั่งสำหรับกำหนดค่าพารามิเตอร์ต่างๆ ของไมโครคอนโทรลเลอร์ อาทิ วอตซ์ต็อกไทเมอร์-เพาเวอร์อัปไทเมอร์ และวงจรกำเนิดสัญญาณนาฬิกา แต่เมื่อเลือกเมนู Device เป็นอ็ีพโรมเมนูนี้จะจางลงและเรียกใช้งานไม่ได้

ID Loc เป็นคำสั่งสำหรับการกำหนดค่า ID ภายในตัวไมโครคอนโทรลเลอร์ เมื่อเลือกเมนู Device เป็นอ็ีพโรมเมนูนี้จะจางลงและเรียกใช้งานไม่ได้

คำสั่งทั้งสามนี้จะเป็นคำสั่งเดียวกับที่ปรากฏอยู่ในหน้าต่าง Device ผู้ใช้งานสามารถเลือกใช้ได้จากหน้าต่าง Device และจากเมนู Select นี้

เมนู Option

Output file type ใช้เพื่อกำหนดชนิดของไฟล์นามสกุล .HEX ที่ต้องการจะบันทึก ได้แก่ INHX8M อันเป็นมาตรฐานที่ไมโครชิปใช้ สามารถใช้งานกับเครื่องโปรแกรมที่พัฒนาโดยไมโครชิปในเครื่องโปรแกรม UPG-52 ก็ใช้มาตรฐานนั้นเช่นกัน INHX16 เป็นไฟล์มาตรฐานของอินเทลขนาด 16 บิต, INHX32 เป็นไฟล์มาตรฐานของอินเทลขนาด 32 บิต และ RAW BINARY เป็นไฟล์ในรูปแบบไบนารีซึ่งเหมาะกับการเก็บข้อมูลของอ็ีพโรม

Toggle video เป็นคำสั่งสำหรับการเลือกโหมดของหน้าจอแสดงผล เพื่อให้มีพื้นที่ในการแสดงข้อมูลของหน่วยความจำได้มากขึ้น หรือใช้งานในโหมดการแสดงผลปกติ ซึ่งมีขนาด 80 คอลัมน์ 25 บรรทัด

การใช้งาน PIP-02 Device Programmer กับบอร์ด UPG-52

1. เชื่อมต่อสายระหว่างพอร์ตอนุกรมของคอมพิวเตอร์ เข้ากับบอร์ด UPG-52 จากนั้นเรียกโปรแกรม UPG-52.BAT ตามด้วยชื่อพอร์ตอนุกรมที่ต้องการติดต่อ
2. เลือกเบอร์ของไมโครคอนโทรลเลอร์หรืออ็ีพโรมที่ต้องการโปรแกรมโดยไปที่เมนู Select เลือกคำสั่ง Device
3. ในกรณีที่ต้องการอ่านข้อมูลจากไมโครคอนโทรลเลอร์หรืออ็ีพโรมสามารถใช้คีย์ F8 หรือเมนู DEVICE เรียก READ เพื่อดึงโค้ดหรือข้อมูลที่เก็บอยู่ภายในไมโครคอนโทรลเลอร์หรืออ็ีพโรมออกมาเก็บไว้ในหน่วยความจำของโปรแกรม PIP-02 เพื่อพร้อมที่จะทำการแก้ไขหรือโปรแกรมลงบนไมโคร-

คอนโทรลเลอร์หรืออ็ีพროมตัวอื่น ๆ ได้นอกจากนี้ข้อมูลที่อ่านขึ้นมาสามารถบันทึกลงในไฟล์เพื่อเก็บไว้โปรแกรมในคราวต่อไปได้

4. ในกรณีที่ มีไฟล์นามสกุล .HEX สำหรับไมโครคอนโทรลเลอร์หรือนามสกุลอื่น ๆ สำหรับอ็ีพროมอยู่แล้ว ให้ไปที่เมนู File เลือกคำสั่ง LOAD เลือกไฟล์ที่ต้องการจากนั้นกดปุ่ม OPEN

5. กรณีไมโครคอนโทรลเลอร์ PIC16F84 หรือ PIC16C84 ให้ทำการแก้ไขค่าพารามิเตอร์ โดยไปที่เมนู Select เลือก Fuse word เพื่อเลือกชนิดของวงจรกำเนิดสัญญาณนาฬิกาหรือออสซิลเลเตอร์ เลือกว่าจะใช้วอตซ์ต็อกไทเมอร์ และเพาเวอร์อัปไทเมอร์หรือไม่ และเลือกว่าจะป้องกันการอ่านหน่วยความจำ (code protect) หรือไม่

6. ทำการเสียบไมโครคอนโทรลเลอร์หรืออ็ีพโรมที่ต้องการโปรแกรมลงในซ็อกเก็ต

7. ทำการตรวจสอบว่าไมโครคอนโทรลเลอร์หรืออ็ีพโรมที่ต้องการโปรแกรมมีข้อมูลเดิมอยู่หรือไม่โดยใช้คำสั่ง BLANK CHECK หรือกดปุ่ม F7

8. ถ้าต้องการโปรแกรมข้อมูลลงบนไมโครคอนโทรลเลอร์หรืออ็ีพโรม โดยใช้ไฟล์ซึ่งเก็บอยู่ในดิสก์หรือฮาร์ดดิสก์ ให้เลือกเมนู FILE เลือก LOAD จากนั้นเลือกไฟล์ที่ต้องการเมื่อเลือกไฟล์เรียบร้อยแล้ว ข้อมูลต่างๆ ที่อยู่ในไฟล์จะปรากฏที่หน้าต่างของหน่วยความจำ โปรแกรม และหน่วยความจำข้อมูล และถ้ามีการกำหนดค่าพารามิเตอร์ไว้แล้ว ค่าข้อมูลภายในเมนู Device จะมาปรากฏด้วย

9. ตรวจสอบค่าพารามิเตอร์ในหน้าต่าง DEVICE ว่าเป็นค่าที่ตรงตามความต้องการหรือไม่ ถ้าไม่จะต้องทำการแก้ไข

10. ทำการโปรแกรมข้อมูลลงบนไมโครคอนโทรลเลอร์หรืออ็ีพโรมโดยกดปุ่ม F5 หรือเลือกเมนู DEVICE เลือก PROGRAM หน้าจอจะแสดงเปอร์เซ็นต์ของการโปรแกรมและเมื่อครบ 100 เปอร์เซ็นต์ ถ้ามีข้อผิดพลาดจะแสดงข้อความ “Error Programming Device?” ขึ้นที่หน้าจอ ให้ทำการตรวจสอบสายต่อและชนิดของไมโครคอนโทรลเลอร์หรืออ็ีพโรมใหม่อีกครั้ง จากนั้นจึงทำการโปรแกรมใหม่

11. การใช้งานบอร์ด UPG-52 กับไมโครคอนโทรลเลอร์ PIC12C508 และ PIC12C509 นั้นจะมีข้อแตกต่างจาก PIC16F84 คือ PIC12C50x มีออสซิลเลเตอร์แบบ Internal RC เพิ่มเติมขึ้นมา ดังนั้นตัวเลือกของชนิดออสซิลเลเตอร์จะมีเพิ่มขึ้นและ PIC12C50x ได้มีการเตรียมค่าสำหรับปรับแต่งค่าความถี่ ซึ่งเรียกว่า Calibration Data โดยจะเก็บค่านี้ไว้ที่ตำแหน่งหน่วยความจำโปรแกรมไบต์สุดท้าย ในรูปของคำสั่ง `movlw 0xXX` โดย XX คือค่าข้อมูลสำหรับที่ใช้ในการปรับแต่งเมื่อมีการเขียนโปรแกรมและใช้วงจรกำเนิดสัญญาณนาฬิกาแบบ Internal RC และต้องการให้ค่าของความถี่มีความเที่ยงตรงในไบต์แรกของหน่วยความจำโปรแกรม จะต้องป้อนคำสั่ง `movwf OSCCAL` เพื่อโหลดค่าในไบต์สุดท้ายของหน่วยความจำโปรแกรมไว้ที่รีจิสเตอร์ที่ใช้ปรับแต่ง ซึ่งจากคุณสมบัติดังกล่าวนี้ทำให้ไมโครคอนโทรลเลอร์ PIC12C50x ทุกตัวเมื่อออกมาจากโรงงานในไบต์สุดท้ายของหน่วยความจำโปรแกรมมีค่า Calibration Data เก็บอยู่ไม่ได้มีค่าเป็น FFF เหมือนไมโครคอนโทรลเลอร์เบอร์อื่นๆ ทำให้ในขณะทำการโปรแกรม ค่า Check Sum ของโปรแกรมที่ PIP-02 กับค่า Check Sum

ที่อ่านได้จากหน่วยความจำโปรแกรมของ PIC12C50x มีค่าไม่ตรงกัน ทำให้โปรแกรม PIP-02 แสดงข้อผิดพลาดทั้งที่ไม่มีข้อผิดพลาดในการโปรแกรม

สามารถแก้ไขข้อผิดพลาดเหล่านี้ได้โดยการอ่านค่าจากหน่วยความจำโปรแกรมของ PIC12C50x ออกมาก่อนที่จะโปรแกรม (กดปุ่ม F8) จากนั้นจำค่าในไบต์สุดท้ายของหน่วยความจำโปรแกรมไว้ ทำการโหลดโปรแกรมที่ต้องการมาไว้ที่โปรแกรม PIP-02 เปลี่ยนค่าในไบต์สุดท้ายของหน่วยความจำ PIP-02 เป็นค่าที่อ่านได้จากหน่วยความจำโปรแกรมทำการโปรแกรม (กดปุ่ม F5) โปรแกรม PIP-02 จะไม่แสดงข้อผิดพลาดขึ้น

ชุดคิดหรือชุดสำเร็จของเครื่องโปรแกรมมอเนกประสงค์ชุดนี้ พร้อมด้วยซอฟต์แวร์ควบคุมการทำงานและเอกสารประกอบที่สำคัญในการเรียนรู้จัดทำหน่วยที่บริษัท อินโนเวทีฟ แอ็กเพอริเมนต์ จำกัด โทรศัพท์ 374-8621 (สายตรง) 732-0014-7 (ต่อไอเน็ก)

