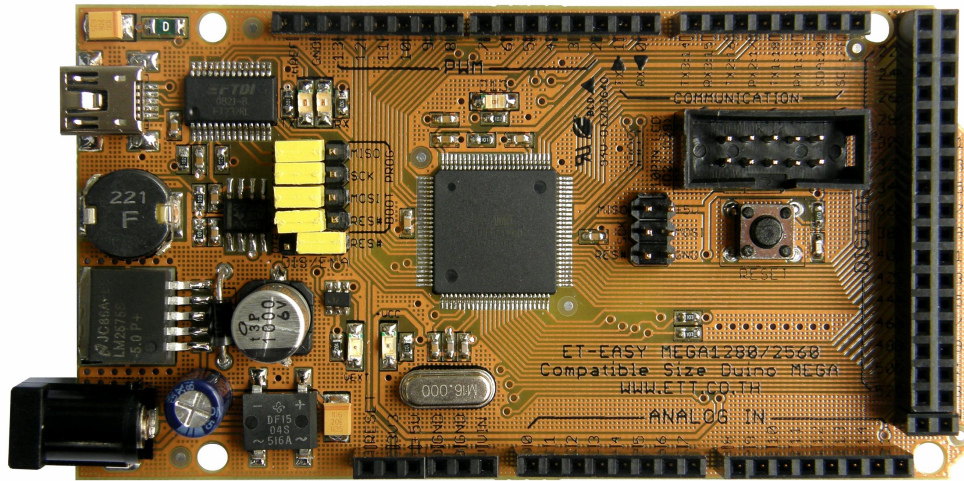


ET-EASY MEGA1280 (Duino Mega)

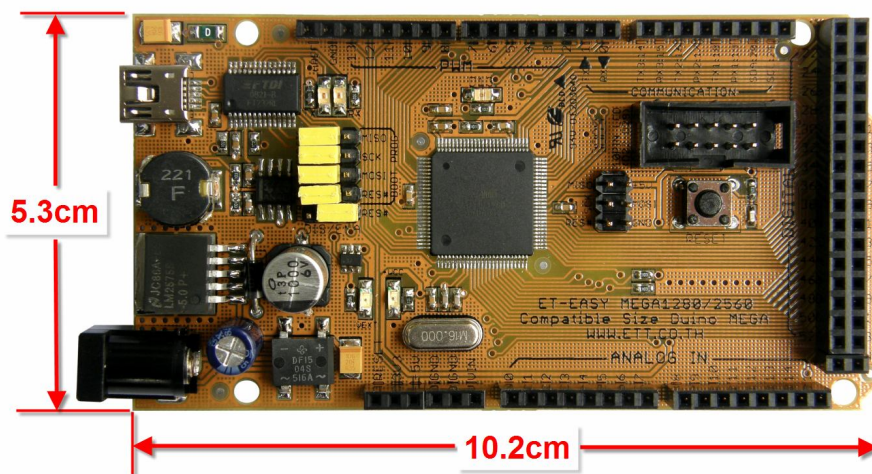


จากการที่ Arduino ที่เป็นการพัฒนาระบบ MCU ของ AVR แบบ Open Source ได้รับการแนะนำเผยแพร่ออกมาสู่สาธารณะ ซึ่งได้รับความนิยมกันอย่างแพร่หลายจากผู้คนทั่วโลก ภายในระยะเวลาอันรวดเร็ว ทางด้านของ Software ก็มีการพัฒนากันอย่างต่อเนื่อง ซึ่งในขณะนี้ (สิงหาคม 2552) โปรแกรมของ Arduino ได้รับการปรับปรุงเป็น Version "arduino-0016" แล้ว โดยทางด้าน Hardware เองก็ได้มีการพัฒนาปรับปรุงอย่างต่อเนื่องควบคู่กันไปด้วยเช่นเดียวกัน ซึ่งจากเดิมที่มีการพัฒนาโปรแกรมให้รองรับกับการใช้งานได้กับชิพ MCU รุ่นเล็ก 28 ขา อย่าง ATMEGA8, ATMEGA88/ATMEGA168/ATMEGA328 สำหรับเป็นจุดเริ่มต้นให้ผู้สนใจได้ใช้เป็นเครื่องมือและอุปกรณ์ในการศึกษาทดลองเรียนรู้ ตลอดจนถึงนำไปประยุกต์ใช้งานกันนั้น มาถึงวันนี้ ขนาดของทรัพยากรต่างๆ รวมทั้งขนาดของหน่วยความจำ สำหรับเขียนโปรแกรม ที่มีอยู่ในชิพ AVR รุ่นเล็กที่มีอยู่เริ่มไม่เพียงพอต่อการประยุกต์ใช้งานในงานบางประเภทแล้ว ทาง Arduino เอง จึงได้ทำการพัฒนาให้ Arduino สามารถรองรับการใช้งานขนาดใหญ่ขึ้นอีก โดยปรับปรุงโปรแกรมให้ใช้กับชิพ AVR รุ่นใหญ่ขึ้น เพื่อให้มีจำนวน I/O ทั้ง Digital, Analog, PWM, UART และ ขนาดหน่วยความจำเพิ่มมากขึ้นกว่าเดิม แต่ก็ยังคงใช้แนวทางในการพัฒนาโปรแกรม แบบเดียวกันกับรุ่นเล็กทุกประการ โดยได้เลือกใช้ชิพ AVR เบอร์ ATMEGA1280 และออกแบบพัฒนา Hardware บอร์ดขึ้นมารองรับ โดยใช้รหัสชื่อรุ่นว่า "Arduino Mega" ออกวางจำหน่าย และได้มีการเผยแพร่รายละเอียดทาง Hardware ต่างๆให้ผู้สนใจนำไปพัฒนาต่อยอดกันตัวเอง แต่เนื่องจากชิพดังกล่าวมีโครงสร้างตัวถังเป็นแบบ SMD จึงทำให้เป็นอุปสรรคสำหรับผู้ใช้งานจำนวนไม่น้อยในการที่จะผลิตหรือสร้างบอร์ดขึ้นใช้งานเอง

ทาง ETT จึงได้นำ ATMEGA1280 มาพัฒนาเป็นบอร์ด โดยให้มีโครงสร้างการทำงานเช่นเดียวกันกับ Arduino Mega ขึ้นมา โดยใช้ชื่อว่า "ET-EASY MEGA1280" โดยได้ออกแบบให้มีการจัดสรร Pin I/O ต่างๆ รวมทั้งขนาดให้ตรงตามมาตรฐานของบอร์ด "Arduino Mega" เพียงแต่ได้มีการปรับปรุงข้อจำกัดบางอย่างให้ดีขึ้นกว่า Arduino Mega ฐานมาตรฐาน เพื่อเพิ่มความสะดวกกับผู้ใช้จำนวนมากยิ่งขึ้น

คุณสมบัติของบอร์ด

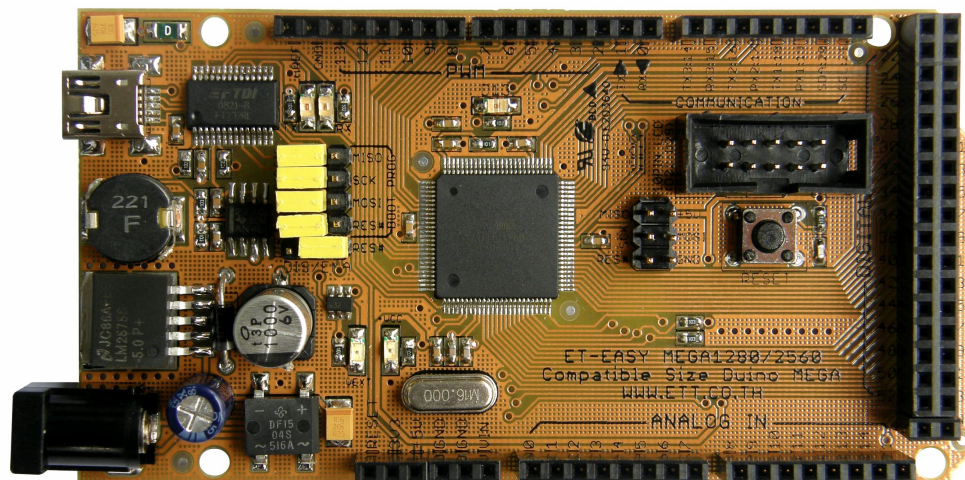
- ใช้ ATMEGA1280 เป็น MCU ประจําบอร์ด Run ความถี่ 16MHz จาก Crystal Oscillator
- 128KByte Flash(สงวนไว้ 4KByte สำหรับ Bootloader) / 8KByte SRAM / 4KByte EEPROM
- รองรับการพัฒนาโปรแกรมด้วยภาษา C++ ของ Arduino ตามแบบ Arduino Mega ได้ 100%
- ใช้ USB Bridge ของ FTDI เบอร์ FT232RL พร้อม Over Current Protection สำหรับติดต่อสื่อสาร และ Download Code จากคอมพิวเตอร์ให้บอร์ด พร้อม Jumper สำหรับปรับใช้งานบอร์ดเป็นการ Program Bootloader ให้กับ MCU จากพอร์ต USB ในบอร์ดได้เอง โดยไม่ต้องใช้เครื่องโปรแกรม AVRISP จากภายนอก
- 54 Pin Digital I/O โดยมี 14Pin สามารถโปรแกรมหน้าที่เป็น PWM ได้
- 16 Pin Analog Input (ADC ขนาด 10 บิต 16 ช่อง)
- 4 UART(Hardware Serial Port) แบบ TTL Logic
- ขนาดของ PCB บอร์ด และ ตำแหน่ง Pin Connector ต่างๆ ตรงกันกับ Arduino Mega ทั้งหมด ทำให้สามารถนำไปติดตั้งใช้งานร่วมกับบอร์ด Shield แบบต่างๆที่มีการผลิตขึ้นมาใช้งานร่วมกันกับ บอร์ด Arduino Mega ได้ทั้งหมด โดยบอร์ดมีขนาด PCB Size 5.3cm x 10.2cm
- มีขั้ว Header 10Pin IDE ของ 8 บิต Digital I/O(D22...D29) สำหรับเชื่อมต่อกับ LCD หรือ บอร์ด I/O แบบต่างๆ ของ อีทีที ที่เพิ่มความสะดวกในการใช้งาน
- รองรับการใช้งานกับ External Supply ทั้งแบบ AC และ DC ขนาด 7-20V โดยเลือกใช้ Regulate แบบ Switching ขนาด 1A (LM2575-5V) ลดปัญหาเรื่องความร้อนเมื่อมีการใช้กระแสสูงๆ สามารถใช้แหล่งจ่ายจากพอร์ต USB ได้ในกรณีใช้กระแสไม่เกิน 500mA โดยมีวงจรเลือกแหล่งจ่ายอัตโนมัติ โดยจะตัดการใช้ไฟเลี้ยงจาก USB โดยอัตโนมัติ เมื่อมีการต่อแหล่งจ่ายจากภายนอกให้บอร์ด



ความแตกต่างระหว่าง Arduino Mega กับ ET-EASY MEGA1280(Duino Mega)



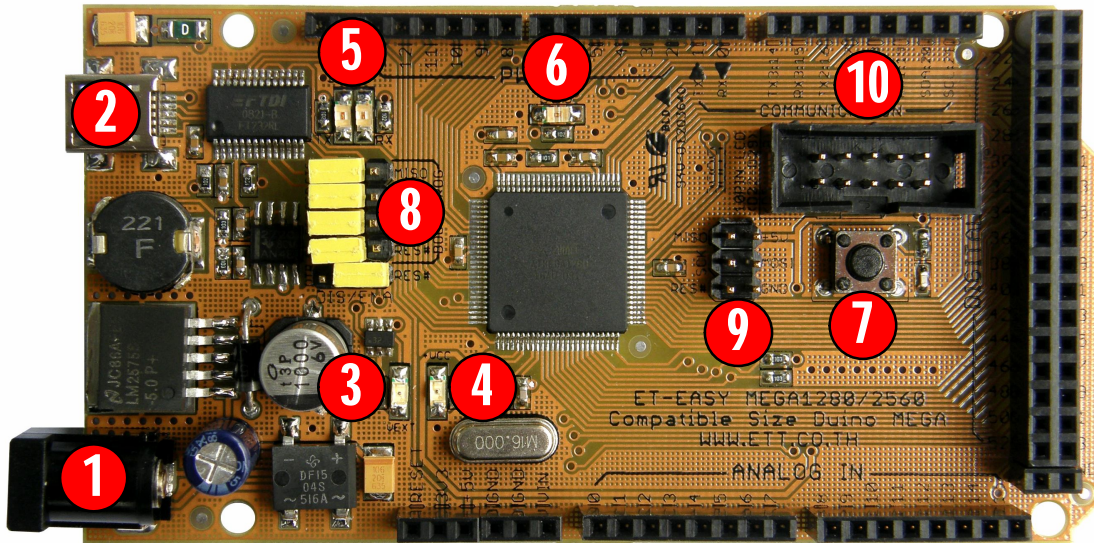
รูปแสดง ลักษณะของบอร์ด Arduino Mega



รูปแสดง ลักษณะของบอร์ด ET-EASY MEGA1280

1. วงจรภาค Supply ของบอร์ด ET-EASY MEGA1280 ปรับปรุงให้ใช้วงจร Bridge Rectifier และ ใช้ Switching Regulate ทำให้สามารถใช้ได้กับแหล่งจ่ายทั้งแบบ AC และ DC โดยไม่จำกัดขั้วไฟฟ้า และ ช่วยลดความร้อนที่เกิดจากไอซี Regulate ได้เป็นอย่างมาก ในขณะที่วงจร Arduino Mega ใช้ Half wave Rectifier ซึ่งต้องใช้กับ DC Adapter แบบ แกนใน เป็น ขั้วบวกเท่านั้น และใช้ไอซี Regulate ธรรมดา ซึ่งจะเกิดความร้อนสูงกว่าเมื่อต้องจ่ายกระแสมากๆ
2. บอร์ด ET-EASY MEGA1280 ออกแบบให้มีระบบ Jumper สำหรับใช้ทำหน้าที่เป็นตัวเลือกการทำ Auto Reset และการ Program Bootloader ด้วย FTDI BitBang ไว้ให้ด้วยทำให้มีความสะดวกมากขึ้นในการใช้งาน ในขณะที่ Arduino Mega ต้องใช้การต่อสาย Jump เอง

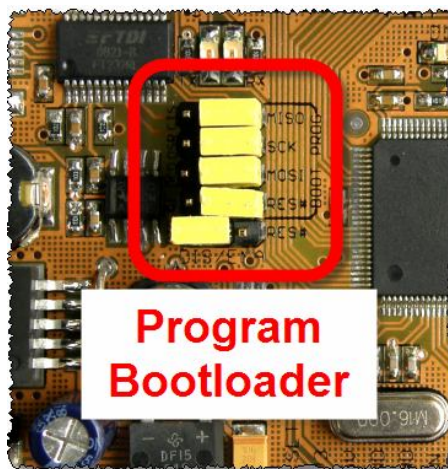
โครงสร้างบอร์ด ET-EASY MEGA1280 (Duino Mega)



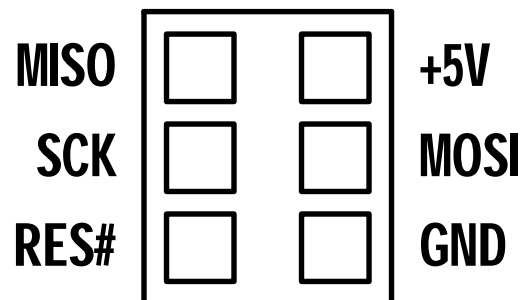
รูปแสดง โครงสร้างของบอร์ด ET-EASY MEGA1280 (Duino MeGA)

- หมายเลข 1 คือ ขั้วต่อแหล่งจ่ายไฟเลี้ยงจากภายนอก สามารถใช้ได้กับแหล่งจ่ายทั้งแบบ AC และ DC พร้อมวงจร Bridge Rectifier และ Regulate แบบ Switching ช่วยลดความร้อนของ IC Regulate เมื่อมีการดึงกระแสมากๆได้เป็นอย่างดี สามารถใช้กับแรงดัน Input 7-20V
- หมายเลข 2 เป็นขั้วต่อ USB สำหรับติดต่อสื่อสารกับคอมพิวเตอร์ PC โดยใช้ FT232RL เป็น USB Bridge ในการเชื่อมต่อระหว่างคอมพิวเตอร์ PC และ MCU ในบอร์ด และยังสามารถใช้ไฟจาก พอร์ต USB เป็นแหล่งจ่ายให้กับบอร์ดได้ด้วย โดยจะมี Poly Fuse ขนาด 500mA สำหรับป้องกันการดึงกระแสเกินจากพอร์ต USB ด้วย และที่พิเศษคือมีวงจรสำหรับตรวจสอบแหล่งจ่ายเพื่อสลับการใช้งานแหล่งจ่ายจาก USB ไปเป็น External Supply ได้เอง โดยอัตโนมัติ โดยเมื่อไม่ได้ต่อ External Supply บอร์ดจะใช้ไฟจากพอร์ต USB เป็นแหล่งจ่ายในการทำงาน แต่เมื่อมีการต่อ External Supply วงจรจะสลับไปใช้แหล่งจ่ายจาก External Supply เองโดยอัตโนมัติ
 - LED +VCC ใช้แสดงสถานะเมื่อมีการจ่ายไฟให้กับบอร์ด
 - LED VEXT ใช้แสดงสถานะเมื่อมีการจ่ายไฟจาก External Supply
- หมายเลข 3 เป็น LED VEXT ใช้แสดงสถานะเมื่อมีการจ่ายไฟเลี้ยงจาก External Supply
- หมายเลข 4 เป็น LED +VCC ใช้แสดงสถานะของแหล่งจ่ายไฟเลี้ยง (+VCC) ของบอร์ด โดยเมื่อบอร์ดใช้แหล่งจ่ายจาก External Supply จะแสดงสถานะโดยการให้ LED VEXT และ LED +VCC ติดสว่างพร้อมกันทั้งคู่ แต่ถ้าบอร์ดใช้แหล่งจ่ายจากพอร์ต USB จะแสดงสถานะโดยการให้ LED +VCC ติดสว่างเพียงดวงเดียว

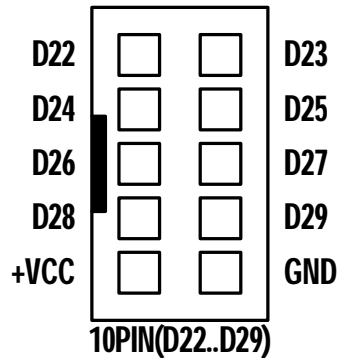
- หมายเลข 5 เป็น LED แสดงสถานะของ RX และ TX ใช้สำหรับแสดงการรับส่งข้อมูลระหว่างบอร์ด ET-EASY MEGA1280 กับคอมพิวเตอร์ PC ผ่านทางพอร์ต USB
- หมายเลข 6 เป็น LED D13 ใช้สำหรับทดสอบการทำงานของ Bootloader และ ใช้ทดสอบการทำงานของบอร์ดจากการควบคุมของ Pin Digital-13 ทำงานด้วย Logic "1" และ หยุดทำงานด้วย Logic "0"
- หมายเลข 7 เป็นสวิตช์ Reset ใช้สำหรับสั่ง Reset การทำงานของบอร์ด
- หมายเลข 8 เป็นชุด Jumper สำหรับเลือก การ Program Bootloader ผ่าน USB Port และ การใช้งานตามปกติ



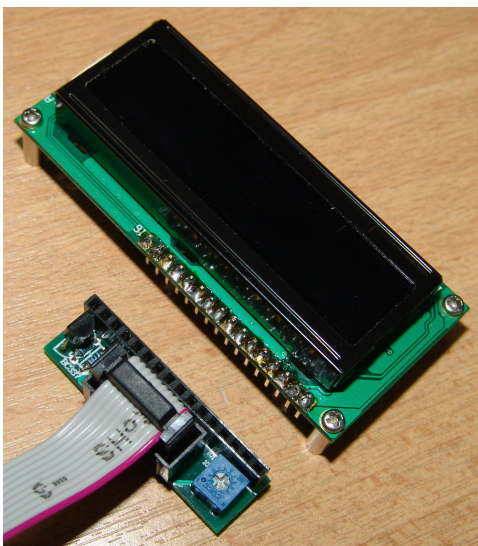
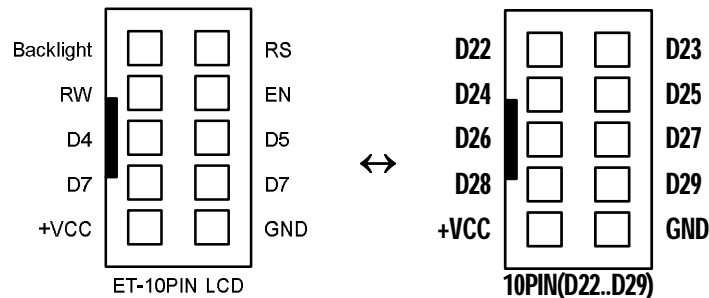
- หมายเลข 9 เป็นขั้วต่อ AVRISP ใช้สำหรับ Download Code ให้กับ MCU โดยขั้วต่อ AVRISP นี้จะสามารถใช้งานได้กับเครื่องโปรแกรมทุกรุ่นที่รองรับการใช้งานกับ ATMEGA1280 และใช้ขั้วต่อตรงตามมาตรฐาน AVRISP ดังรูป

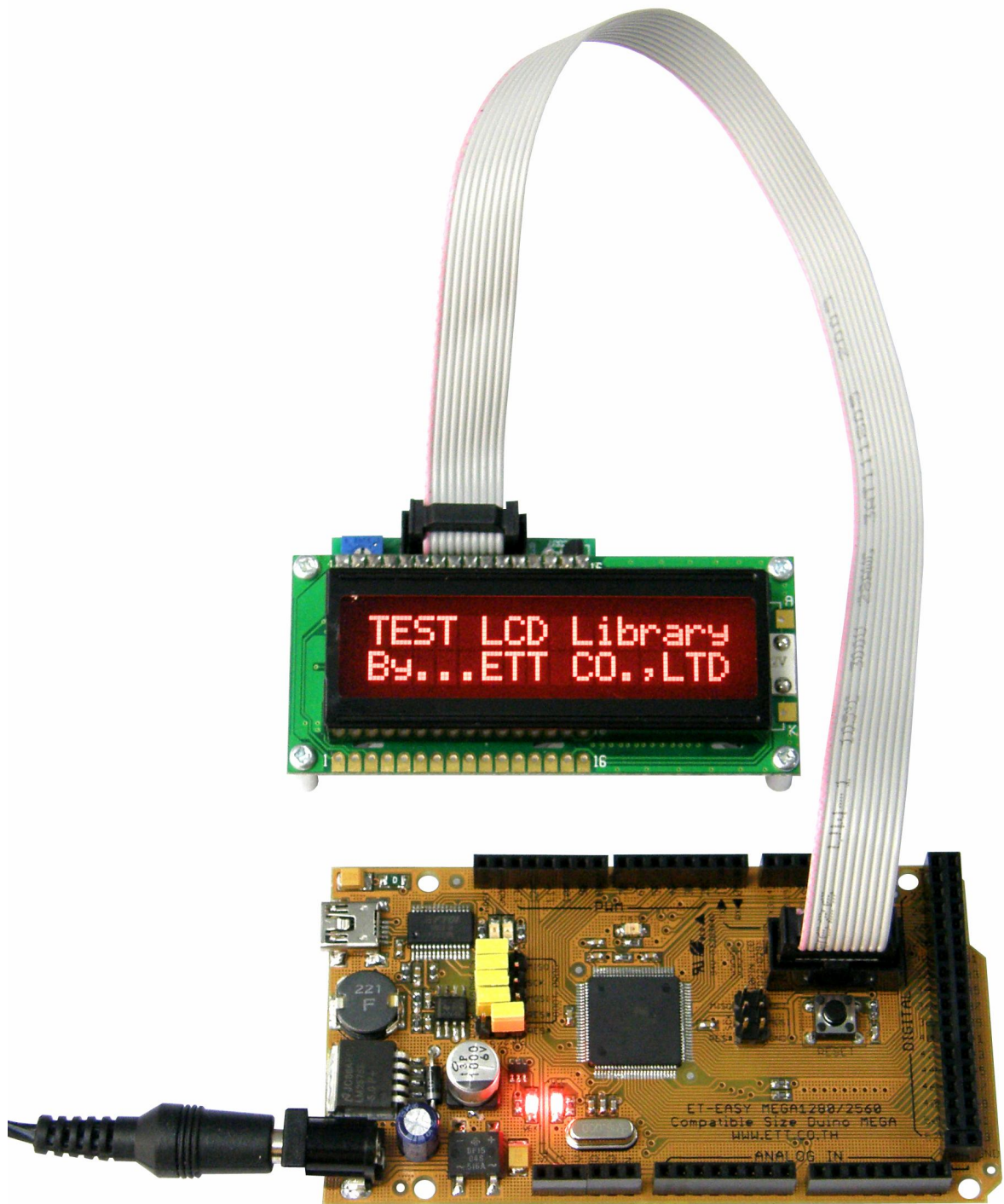


- หมายเลข **10** เป็นหัวต่อสัญญาณจาก D[22..29] สำหรับเชื่อมต่อกับบอร์ด I/O ของ อีทีที รวมทั้งจอแสดงผล LCD โดยใช้ร่วมกับ 10PIN LCD หรือ ET-CONV SPI TO LCD

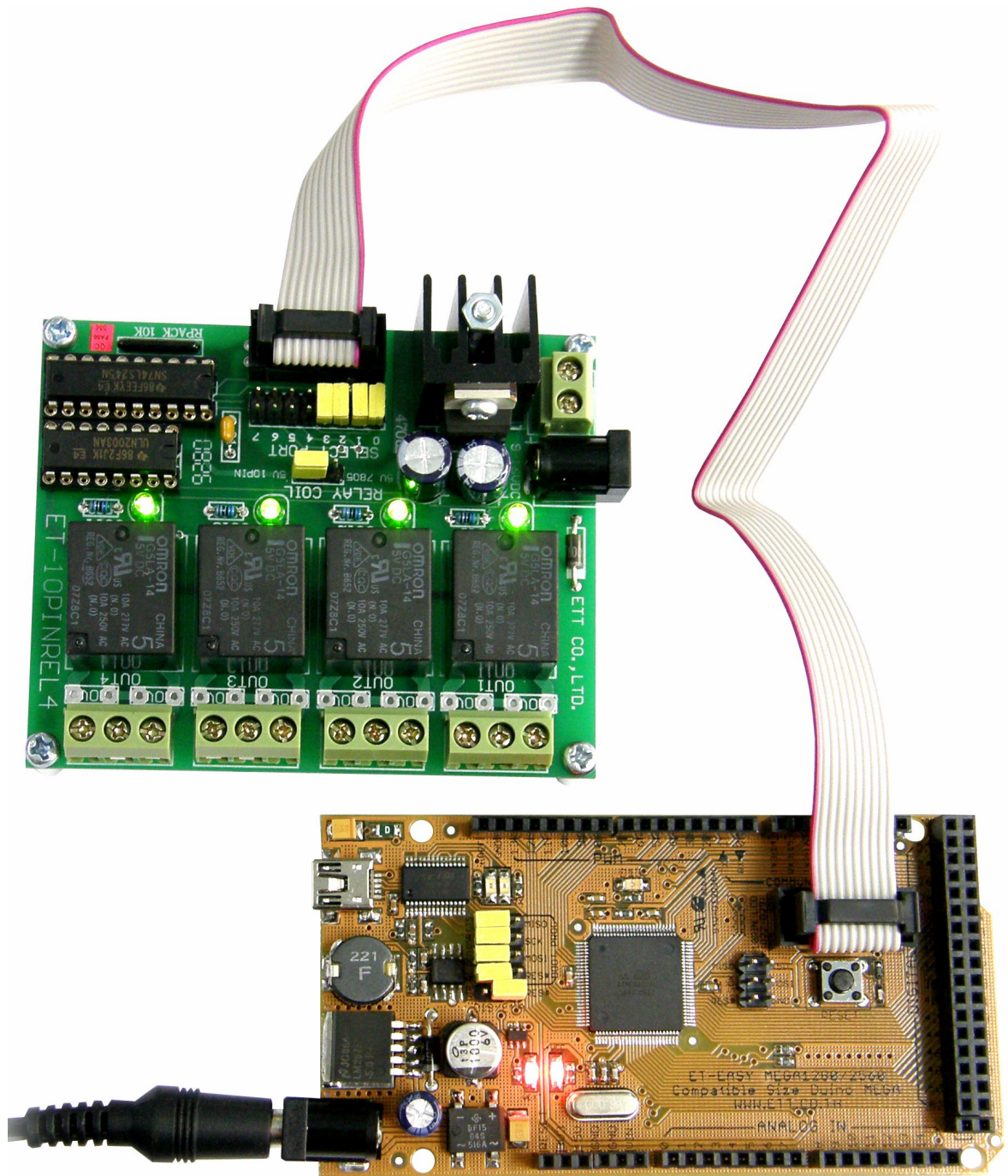


โดยในกรณีที่ต้องการนำหัวต่อ 10PIN ไปเชื่อมต่อเพื่อใช้ควบคุมการแสดงผลของ LCD นั้น เพื่อความสะดวกในการใช้งาน ขอแนะนำให้จัดหาชุด ET-10PIN CLCD (ET-CONV 10 TO LCD) มาเป็นตัวกลางในการเชื่อมต่อระหว่าง หัว IDE 10PIN ของบอร์ด ET-EASY MEGA1280 ดังตัวอย่างโดยเมื่อนำ LCD มาติดตั้งเข้ากับชุด ET-10PIN CLCD เรียบร้อยแล้วจะทำให้สามารถเชื่อมต่อสายสัญญาณระหว่างหัว ET-10PIN CLCD กับ 10PIN ได้โดยง่าย โดยใช้การเชื่อมต่อผ่านทางสายแพร 10PIN ได้ทันทีดังตัวอย่าง





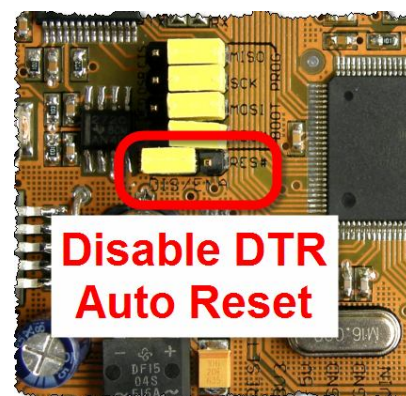
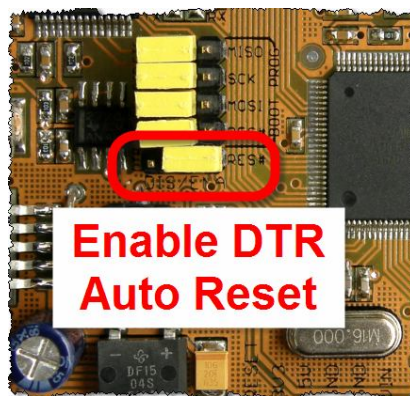
รูปแสดง ตัวอย่างการต่อ LCD ด้วยขั้ว 10Pin กับ ET-CONV 10 TO LCD



รูปแสดง ตัวอย่างการต่อ บอร์ด I/O ของ อีทีที ด้วยหัว 10Pin

คุณสมบัติของสัญญาณต่าง ๆ ของบอร์ด ET-EASY MEGA1280

- **RESET#** เป็นสัญญาณ Input Logic Reset ของ MCU เมื่อเป็น Logic Low จะทำให้ MCU อยู่ในสภาวะรีเซ็ต เมื่อเป็น Logic High จะทำให้ MCU อยู่ในสภาวะทำงานตามปกติ โดยสัญญาณ RESET# นี้จะถูกควบคุมจาก 2 แหล่ง คือ จาก สวิตช์ RESET ภายในบอร์ด และ จากสัญญาณ DTR ของ FT232RL ถ้ามีการเลือก Enable Jumper ของ Auto Reset จาก DTR ไว้



- **+3V3** เป็นแหล่งจ่ายไฟขนาด +3.3V ที่ได้จากวงจร Regulate ภายในของ FT232RL สามารถจ่ายกระแสได้สูงสุด 50mA ซึ่งเมื่อต้องการนำแหล่งจ่าย +3.3V นี้ไปใช้งานเป็นแหล่งจ่ายให้กับอุปกรณ์ต่างๆ ต้องระมัดระวังเรื่องการดึงกระแสของโหลดด้วย ถ้าโหลดมีการดึงกระแสมากกว่า 50mA อาจทำให้ FT232RL เกิดความเสียหายได้
- **+VIN** เป็นไฟ DC ที่รับมาจาก Jack VIN(External Supply) แต่ผ่านการ Rectifier และ Filter เป็น DC แล้ว มีขนาดแรงดันเฉลี่ยตามขนาดแรงดันที่ป้อนให้กับบอร์ดทาง Jack VIN

- **+5V** เป็นจุดต่อแหล่งจ่ายไฟของบอร์ดออกไปใช้งาน ซึ่งมาจากแหล่งกำเนิด 2 แหล่ง คือ จากพอร์ต USB และจาก External Supply ซึ่งถ้าต่อแหล่งจ่ายให้บอร์ดจาก External Supply ผ่านทาง Jack VIN แหล่งจ่าย +5V นี้จะมาจาก Switching Regulate (LM2575-5V) สามารถจ่ายกระแสได้สูงสุดถึง 1A แต่ถ้าใช้แหล่งจ่ายจากพอร์ต USB แหล่งจ่าย +5V นี้จะมาจากพอร์ต USB โดยตรงโดยจะมีฟิวส์ แบบ Poly ขนาด 500mA ต่อบริเวณการดึงกระแสเกินเพื่อป้องกันความเสียหายของพอร์ต USB โดยจะจ่ายกระแสได้สูงสุดไม่เกิน 500mA ขึ้นอยู่กับคุณสมบัติในการจ่ายกระแสของพอร์ต USB และการ Configure ค่าให้กับ FT232RL ด้วย
- **A0-A15** เป็นขาสัญญาณ Analog Input แบบ ADC มีขนาดความละเอียด 10บิต มี 16 Pin สามารถรับแรงดัน Analog Input ได้ 0-5VDC
- **D0-D53** เป็นขาสัญญาณ Digital Input/Output แบบ TTL มีทั้งหมด 54 Pin สามารถใช้ทำหน้าที่เป็น Input หรือ Output ตามการกำหนดจากโปรแกรม โดยมีบาง Pin สามารถกำหนดหน้าที่ใช้งานเป็นฟังก์ชันพิเศษต่างๆเพิ่มเติมได้อีก
 - **D0-D1** ถูกสงวนไว้ใช้ทำหน้าที่เป็นพอร์ตสื่อสารอนุกรม RS232 (UART0) โดยได้ทำการเชื่อมต่อกับ USB Bridge ของ FT232RL เพื่อใช้ Upload Code ให้กับบอร์ด และยังสามารถใช้ทดลองติดต่อสื่อสารรับส่งข้อมูลระหว่างบอร์ดกับคอมพิวเตอร์ PC ได้ด้วย
 - **D2-D13** สามารถ โปรแกรมหน้าที่เป็น PWM ขนาด 8 บิต มี 14 Pin ได้
 - **D14** สามารถ โปรแกรมหน้าที่เป็น TX3 สำหรับ ส่งข้อมูลของ UART3 ได้ด้วย
 - **D15** สามารถ โปรแกรมหน้าที่เป็น RX3 สำหรับ รับข้อมูลให้กับ UART3 ได้ด้วย
 - **D16** สามารถ โปรแกรมหน้าที่เป็น TX2 สำหรับ ส่งข้อมูลของ UART2 ได้ด้วย
 - **D17** สามารถ โปรแกรมหน้าที่เป็น RX2 สำหรับ รับข้อมูลให้กับ UART2 ได้ด้วย
 - **D18** สามารถ โปรแกรมหน้าที่เป็น TX1 สำหรับ ส่งข้อมูลของ UART1 ได้ด้วย
 - **D19** สามารถ โปรแกรมหน้าที่เป็น RX1 สำหรับ รับข้อมูลให้กับ UART1 ได้ด้วย
 - **D20** สามารถ โปรแกรมหน้าที่เป็น SDA ของ I2C Bus สำหรับใช้สื่อสารกับ I2C ได้ด้วย
 - **D21** สามารถ โปรแกรมหน้าที่เป็น SCL ของ I2C Bus สำหรับใช้สื่อสารกับ I2C ได้ด้วย
- **AREF** เป็นสัญญาณ Analog Reference จากภายนอกที่ต้องการป้อนให้กับ MCU ซึ่งตามปกติแล้ว ATMEGA1280 สามารถโปรแกรมให้เลือกใช้แรงดันอ้างอิงจากภายในได้อยู่แล้วโดยสามารถเลือกเป็น 1.1V หรือ 2.56V หรือ AVCC(+5V) โดยไม่จำเป็นต้องป้อนแรงดันอ้างอิงจากภายนอกให้กับบอร์ดอีก แต่ถ้าต้องการแรงดันอ้างอิงที่มีความแตกต่างจากที่กล่าวมาแล้วก็สามารถป้อนเป็นแรงดันอ้างอิงจากภายนอกผ่านทางขา AREF นี้เข้าไปเองได้ระหว่าง 0-5V

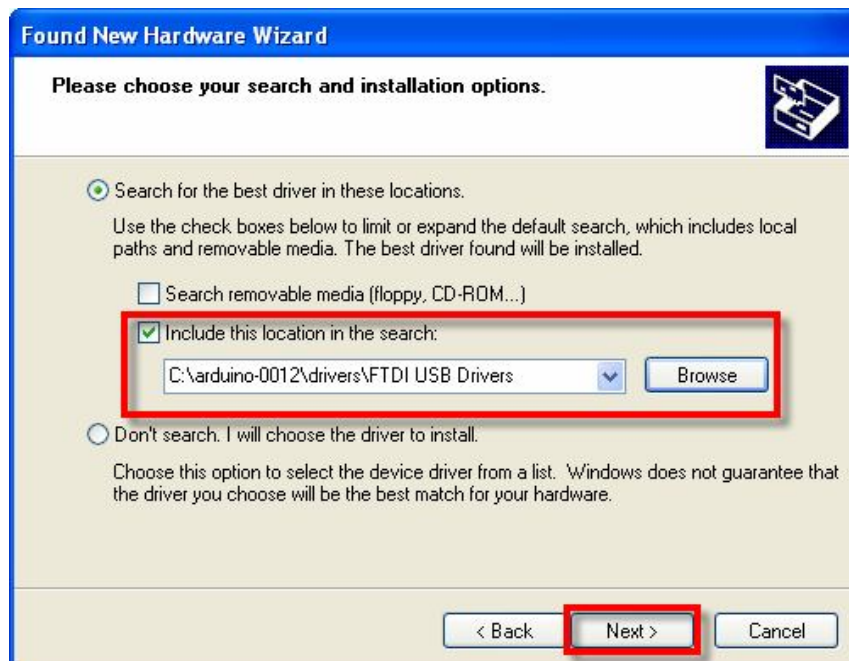
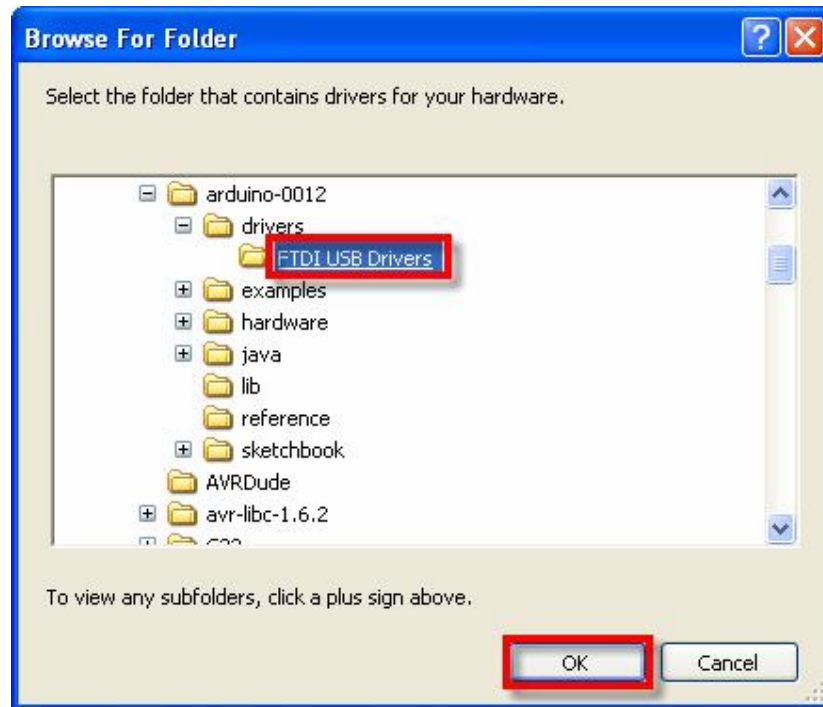
การติดตั้ง Driver ของ USB Bridge ของบอร์ด ET-EASY MEGA1280 (Duino Mega)

บอร์ด ET-EASY MEGA1280 จะใช้ชิพ USB Bridge ของ FTDI เป็นตัวกลางในการเชื่อมต่อกับคอมพิวเตอร์ PC โดย USB Bridge ของ FTDI จะทำหน้าที่เป็นตัวกลางในการเชื่อมต่อและติดต่อสื่อสารระหว่างเครื่องคอมพิวเตอร์ PC กับ MCU ATmega1280 ของบอร์ด ET-EASY MEGA1280 ในรูปแบบของพอร์ตอนุกรม (Visual Com Port) โดยโปรแกรม Application ต่างๆที่ทำงานอยู่บนคอมพิวเตอร์ PC รวมทั้งโปรแกรม Arduino จะมองเห็น พอร์ต USB ที่เชื่อมต่อกับบอร์ด ET-EASY MEGA1280 เป็นพอร์ตสื่อสารอนุกรม (Com Port) ช่องหนึ่งเท่านั้น ซึ่งถ้าเครื่องคอมพิวเตอร์ของผู้ใช้เคยทำการติดตั้ง Driver สำหรับ USB Bridge ของ FTDI ไว้ก่อนแล้ว เมื่อทำการเชื่อมต่อสาย USB ของบอร์ด ET-EASY MEGA1280 เข้ากับ USB HUB ของเครื่องคอมพิวเตอร์ PC แล้ว Windows จะทำการติดตั้ง Driver ให้เองโดยอัตโนมัติ แต่ถ้าเครื่องคอมพิวเตอร์ PC ยังไม่เคยติดตั้ง Driver ของ FTDI ไว้ก่อนก็จะต้องทำการติดตั้ง Driver ให้กับบอร์ดให้เรียบร้อยเสียก่อนซึ่งมีลำดับขั้นตอนดังนี้

1. เตรียมแผ่น CD ROM ที่บรรจุ Driver ของ FTDI ไว้ให้พร้อม หรือ ในกรณีที่ผู้ใช้ได้ทำการติดตั้งโปรแกรมของ Arduino ไว้เรียบร้อยแล้ว ภายในโฟลเดอร์ของโปรแกรม Arduino ก็จะมี Driver ของ FTDI จัดเตรียมไว้ให้เรียบร้อยแล้ว โดยจะอยู่ที่ **"C:\arduino-0012\drivers\FTDI USB Drivers\"**
2. ทำการเสียบสาย USB ของบอร์ด ET-EASY MEGA1280 เข้ากับพอร์ต USB HUB ของเครื่องคอมพิวเตอร์ PC ซึ่ง Windows จะตรวจพบอุปกรณ์ใหม่ โดยเป็น **"FT232R USB UART"** และแจ้งให้ผู้ใช้ทำการติดตั้ง Driver ให้กับอุปกรณ์ ดังรูป



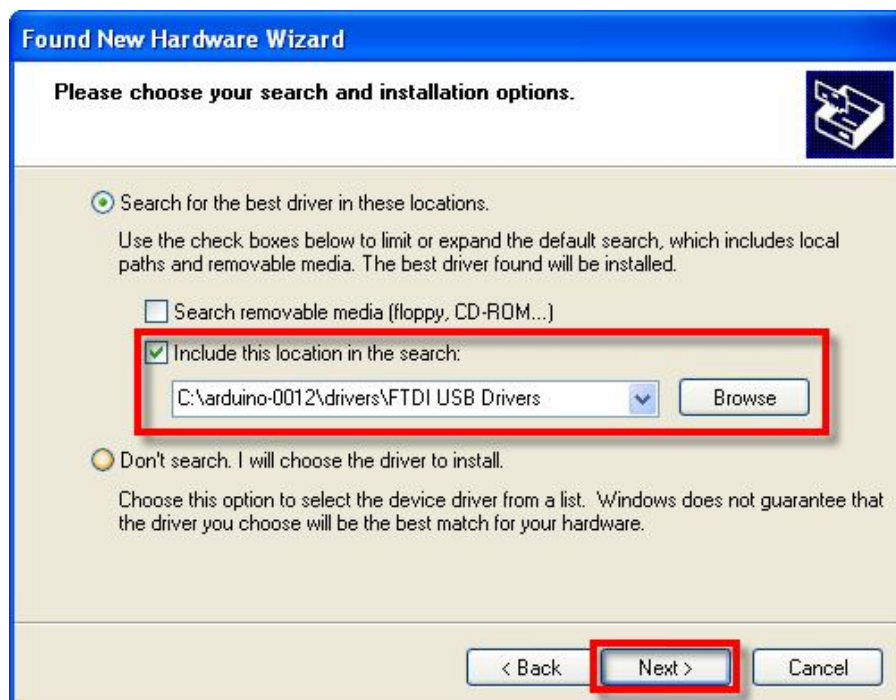
3. ให้เลือก **Install from list or specific location(Advanced)** แล้วเลือก **Next** ซึ่ง Windows ก็จะแจ้งให้ผู้ระบุตำแหน่งไฟล์ไดรเวอร์ที่บรรจุไฟล์ Driver ของ FTDI ไว้ ก็ให้เลือกที่ **Browse** และเลือกไปยัง Drive และ ไฟล์ไดรเวอร์ที่เก็บไฟล์ Driver ไว้ ซึ่งถ้าผู้ใช้ได้ทำการติดตั้งโปรแกรมของ Arduino ไว้แล้ว ก็ให้เลือกไปที่ **"C:\arduino-0012\drivers\FTDI USB Drivers"** แล้วเลือก **Next** ดังรูป



4. ในขั้นตอนนี้โปรแกรม Windows จะทำการค้นหาและติดตั้ง Driver ให้กับอุปกรณ์ ให้รอสักครู่จนการทำงานเสร็จเรียบร้อยแล้ว แล้วเลือก Finish ดังรูป



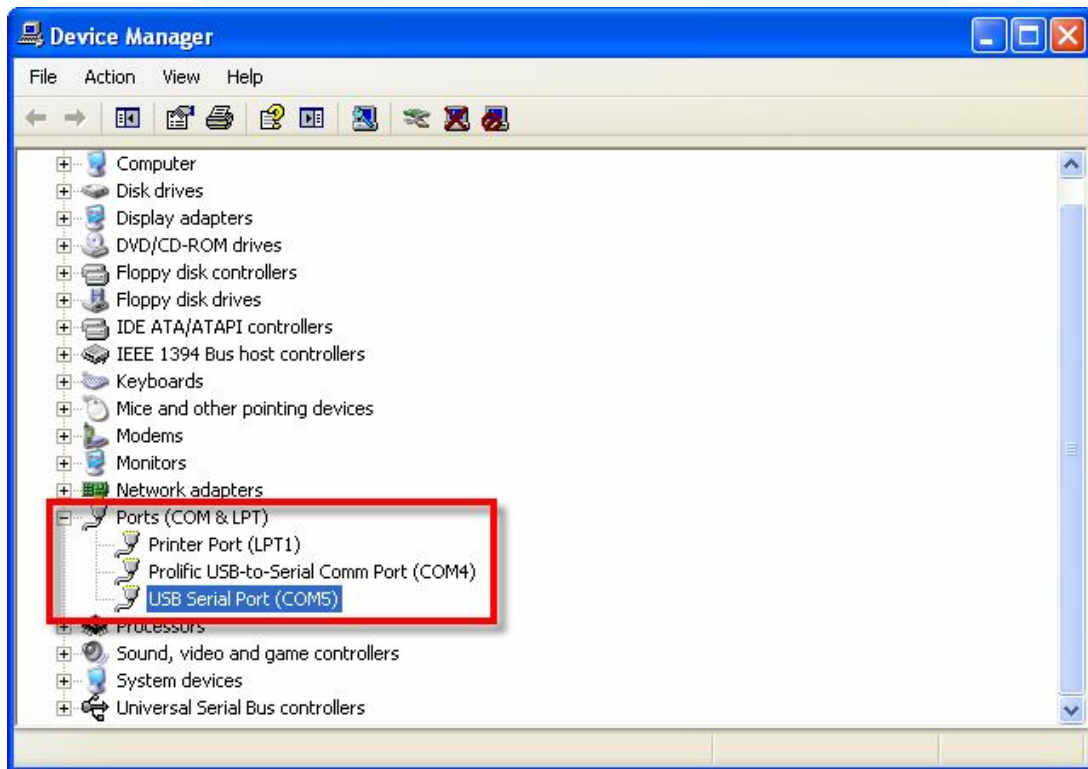
5. หลังจากทำการติดตั้ง Driver ของฮาร์ดแวร์เรียบร้อยแล้ว Windows ก็จะตรวจพบว่ามีอุปกรณ์ใหม่ถูกเชื่อมต่ออยู่ โดยเป็นอุปกรณ์ประเภท **“USB Serial Port”** และแจ้งให้ผู้ใช้ทำการติดตั้ง Driver ให้กับอุปกรณ์ใหม่ที่ระบุเป็น **“USB Serial Port”** อีกครั้งหนึ่ง ซึ่งก็ให้เลือกระบุตำแหน่งไฟล์เดอไรท์เก็บไฟล์ Driver ไว้ ซึ่งให้เลือกเหมือนขั้นตอนในหัวข้อที่ 3 ดังรูป



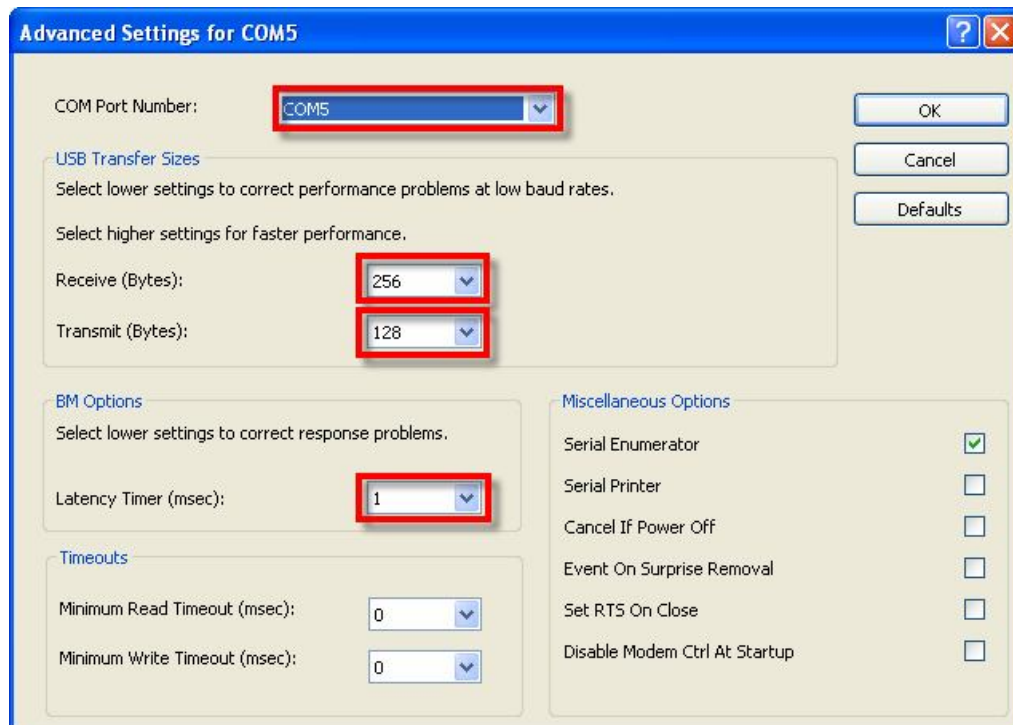
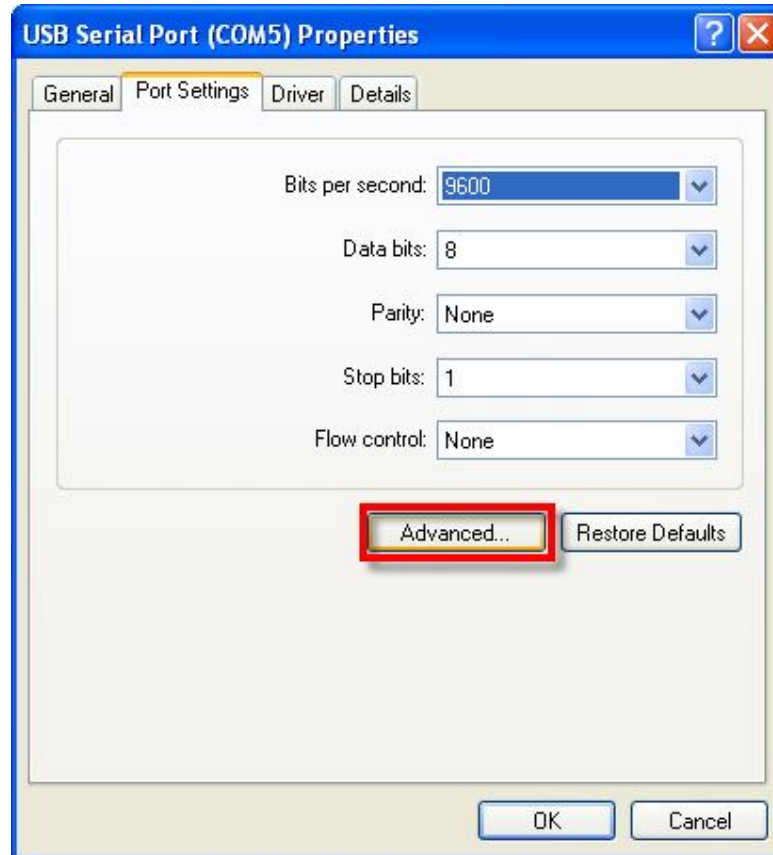
6. ในขั้นตอนนี้โปรแกรม Windows จะทำการค้นหาและติดตั้ง Driver ให้กับอุปกรณ์ ให้รอสักครู่จนการทำงานเสร็จเรียบร้อย แล้วเลือก Finish ดังรูป



7. หลังจากทำการติดตั้ง Driver เรียบร้อยแล้ว ก็สามารถใช้งานอุปกรณ์ได้แล้ว แต่เพื่อความถูกต้องในครั้งแรกนี้ควรต้องเข้าไปทำการตรวจสอบและปรับแต่งค่าให้กับอุปกรณ์ก่อน โดยในขั้นตอนนี้นำไปที่ "My Computer → Control Panel → System → Hardware → Device Manager" แล้วทำการตรวจสอบที่ Ports (COM&LPT) แล้วดูที่ชื่อของ **"USB Serial Port"** ซึ่งให้ผู้ใช้จดจำหมายเลขของ Com Port ของอุปกรณ์ดังกล่าวไว้ เพื่อใช้อ้างอิงถึงในการเรียกใช้งาน ดังรูป



8. ในขั้นตอนนี้นำเมาส์ที่เครื่องหมาย (+) หน้าหัวข้อ Ports(COM&LPT) แล้วมองหาอุปกรณ์ที่ชื่อ **"USB Serial Port"** ตามที่เราได้ทำการติดตั้ง Driver ไว้เรียบร้อยแล้ว หรือ ถ้าไม่แน่ใจว่าอุปกรณ์ดังกล่าวใช่อุปกรณ์ที่เป็นของบอร์ด **"ET-EASY MEGA1280"** หรือไม่ ให้ทดสอบด้วยการถอดสาย USB ออก รายชื่ออุปกรณ์ดังกล่าวจะต้องหายไป แต่เมื่อเสียบสาย USB กลับเข้ามาใหม่ รายชื่อของอุปกรณ์ดังกล่าวก็จะต้องปรากฏให้เห็นอีกครั้ง ถ้าทุกอย่างถูกต้อง ก็ให้ทำการคลิกเมาส์ที่ Tab รายชื่อของอุปกรณ์ดังกล่าว เมื่อปรากฏหน้าต่าง **USB Serial Port Properties** ขึ้นมาแล้วให้เลือกที่ **Port Setting** แล้วเลือก **Advance** เพื่อเข้าไปกำหนดค่าให้กับอุปกรณ์ให้เรียบร้อย ดังนี้
- USB Transfer Size → Receive (Bytes) ให้กำหนดเป็น 256
 - USB Transfer Size → Transmit (Bytes) ให้กำหนดเป็น 128
 - BM Option → Latency Timer (mSec) ให้กำหนดเป็น 1



การโปรแกรม Bootloader ให้กับบอร์ด ET-EASY MEGA1280

ตามปกติแล้วบอร์ด ET-EASY MEGA1280 ได้ทำการโปรแกรม Bootloader ไว้ให้เป็นที่เรียบร้อยแล้ว ผู้ใช้สามารถใช้งานได้ทันที แต่อย่างไรก็ตามในกรณีที่ต้องการเปลี่ยนแปลง Bootloader หรือ เกิดความผิดพลาดในการใช้งาน จนทำให้ Bootloader เสียหายไป ผู้ใช้ยังสามารถทำการ โปรแกรม Bootloader ให้กับบอร์ดได้ใหม่ โดยบอร์ด ET-EASY MEGA1280 ได้ออกแบบ ให้มีขั้วต่อ AVRISP สำหรับใช้เป็นช่องทางในการโปรแกรม Code ให้กับ MCU ได้โดยตรง ด้วยเครื่องโปรแกรมทุกระดับที่มีขั้วต่อ ตรงตามมาตรฐาน AVRISP ของ ATMEL ได้ทันที แต่ในกรณีนี้ผู้ใช้จะต้องมีเครื่องมือในการโปรแกรม AVR ด้วย

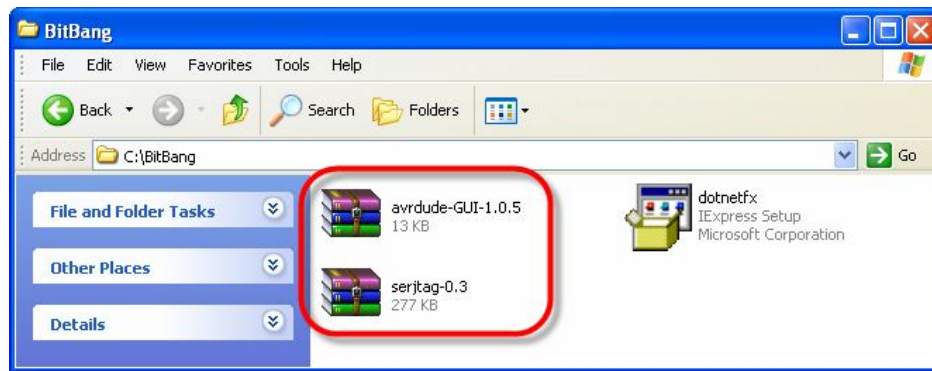
แต่อย่างไรก็ตามบอร์ด ET-EASY MEGA1280 ได้รับการออกแบบ ให้สามารถโปรแกรม Hex File รวมทั้ง Bootloader ให้กับบอร์ดโดยไม่ต้องใช้เครื่องโปรแกรมภายนอกได้ โดยการใช้ความสามารถในโหมด BitBang ของชิพ USB Bridge ของ FTDI(FT232RL) มาดัดแปลงทดแทนได้ ซึ่งวิธีการนี้ได้รับการคิดค้นและพัฒนาขึ้นโดยชาวญี่ปุ่นชื่อ "Kimio Kosoka" (kim.kosmac@gmail.com) ซึ่งมีการเผยแพร่แนะนำไว้ใน website ชื่อ http://www.geocities.jp/arduino_diecimila/bootloader/index_en.html ซึ่งวิธีการดังกล่าวจะอ้างอิงกับบอร์ด Arduino รุ่น Diecimila ซึ่งเราสามารถนำวิธีการดังกล่าวมาประยุกต์ใช้กับบอร์ดรุ่นอื่นๆที่ใช้ MCU ของ AVR ได้เป็นอย่างดี รวมทั้งบอร์ด ET-EASY MEGA1280 ด้วย โดยวิธีการ BitBang จะใช้การนำสัญญาณของ FT232RL มาเชื่อมต่อกับสัญญาณ ISP ของ AVR แล้วใช้โปรแกรมส่งงานสัญญาณ FT232RL ในโหมด BitBang ให้สร้างสัญญาณเพื่อส่งโปรแกรม AVR ด้วยวิธีการแบบ ISP อีกต่อหนึ่ง โดยวิธีนี้จะต้องทำการเชื่อมต่อสัญญาณระหว่าง FT232RL กับ AVR ดังนี้

- ต่อสัญญาณ CTS ของ FT232RL เข้ากับ MISO ของ AVR
- ต่อสัญญาณ DSR ของ FT232RL เข้ากับ SCK ของ AVR
- ต่อสัญญาณ DCD ของ FT232RL เข้ากับ MOSI ของ AVR
- ต่อสัญญาณ RI ของ FT232RL เข้ากับ RES# ของ AVR

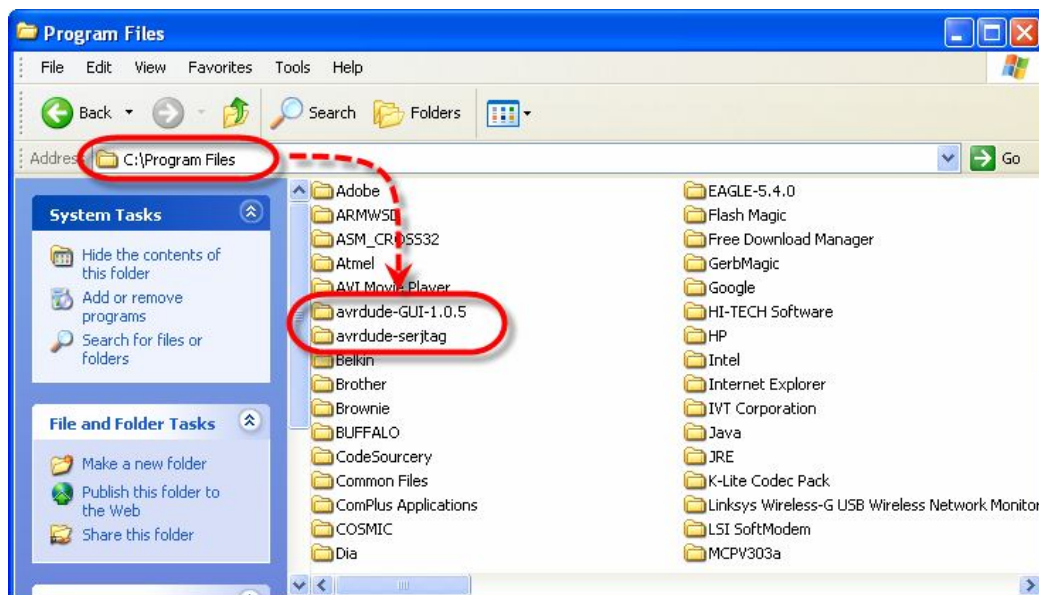
ซึ่งในกรณีของบอร์ด ET-EASY MEGA1280 นั้นสามารถทำได้สะดวกมากกว่าด้วย เนื่องจากระบบฮาร์ดแวร์ของบอร์ดได้จัดเตรียม Jumper สำหรับเลือกตัดต่อสัญญาณของชิพ USB Bridge FT232RL กับ ISP ของ ATMEGA1280 ไว้ให้เป็นที่เรียบร้อยแล้ว ไม่ต้องใช้การ Jump สายให้ยุ่งยากเหมือนกับบอร์ด Diecimila ของ Arduino โดยผู้ใช้เพียงแต่เลือก Jumper(BOOT PROG) ไว้ทางด้าน Enable ก็สามารถใช้ในการโปรแกรม Bootloader ให้กับ MCU ATMEGA1280 ด้วย BitBang ของ FT232RL ได้แล้ว

การโปรแกรม Bootloader แบบ BitBang ด้วย avrduide GUI

1. ทำการ Download โปรแกรม "avrduide-serjtag" และ "avrduide-GUI-1.0.5" ซึ่ง Download ได้จาก http://www.geocities.jp/arduino_diecimila/bootloader/index_en.html และถ้าเครื่องคอมพิวเตอร์ที่ใช้งานอยู่ยังไม่ได้ติดตั้งโปรแกรม Microsoft.Net Framework V2.0 ไว้ก่อนก็ให้ทำการ Download โปรแกรมดังกล่าวมาด้วย เมื่อได้โปรแกรมทั้ง 3 ชุดมาแล้วให้ทำการติดตั้งโปรแกรม Microsoft.Net Framework V2.0 ให้เรียบร้อย จากนั้นให้ทำการ Unzip ไฟล์ที่ Download มา คือ "avrduide-GUI-1.0.5.zip" และ "serjtag-0.3.zip" ดังนี้คือ
 - ไฟล์ "avrduide-GUI-1.0.5.zip" ให้ Unzip ไว้ในโฟลเดอร์ชื่อ "avrduide-GUI-1.0.5"
 - ไฟล์ "serjtag-0.3.zip" ให้ Unzip ไว้ในโฟลเดอร์ชื่อ "avrduide-serjtag"

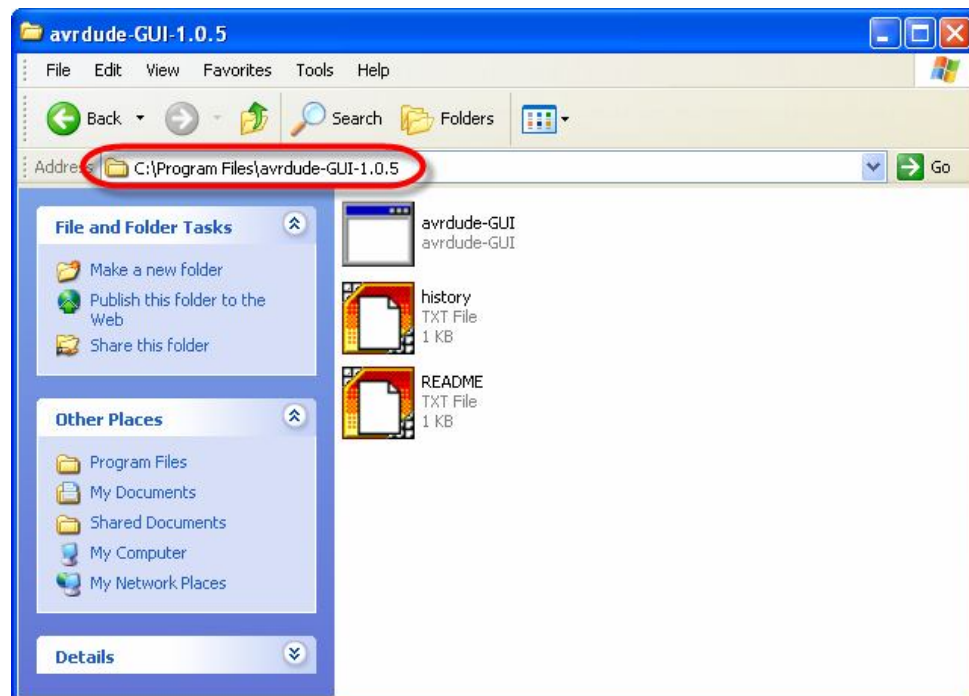
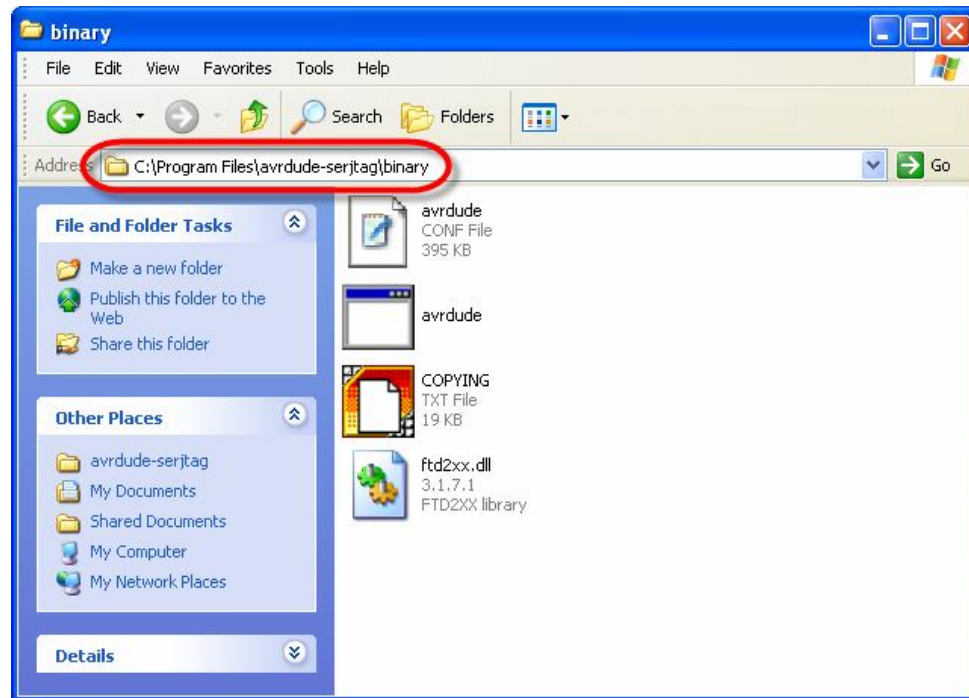


แล้ว Copy ทั้งโฟลเดอร์ของ "avrduide-serjtag" และ "avrduide-GUI-1.0.5" ไปไว้ในโฟลเดอร์ที่ชื่อ "C:\Program Files" ดังรูป

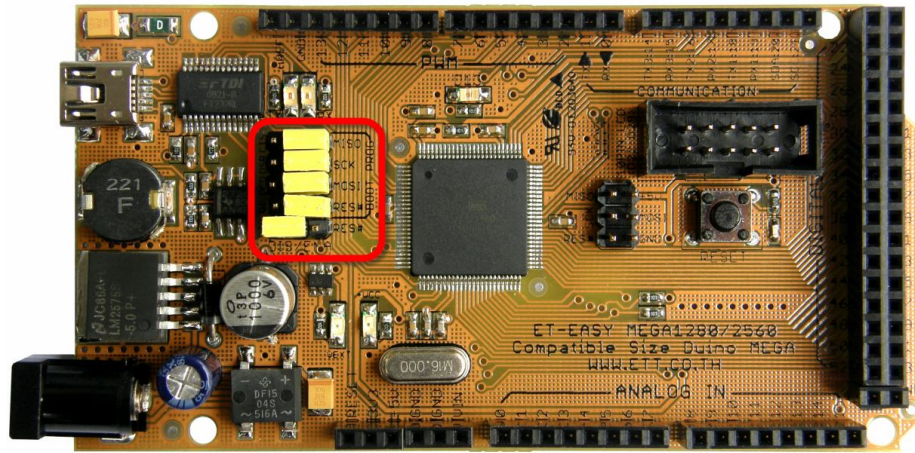


โดยในแต่ละโฟลเดอร์ควรประกอบด้วยไฟล์ต่างๆดังนี้

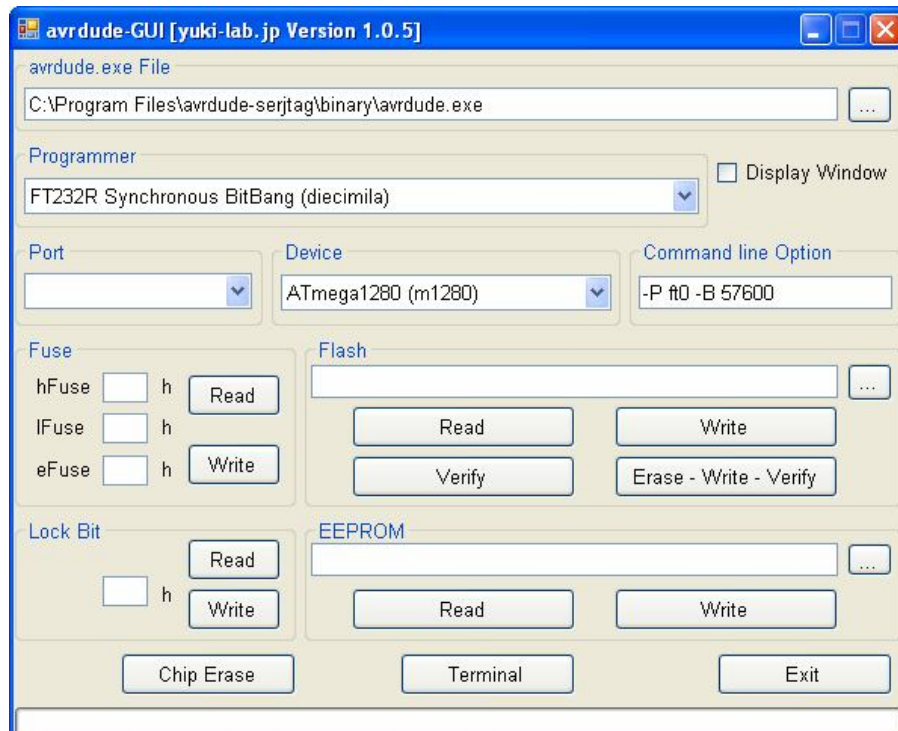
- ในโฟลเดอร์ `../Avrdude-serjtag/bin/` จะมีไฟล์ที่จำเป็นอยู่ 3 ไฟล์ คือ `Avrdude.exe`, `Avrdude.conf` และ `Ftd2xx.dll`
- ในโฟลเดอร์ `../avrdude-GUI-1.0.5/` จะมีไฟล์ที่จำเป็นอยู่ 1 ไฟล์ คือ `Avrdude-GUI.exe`



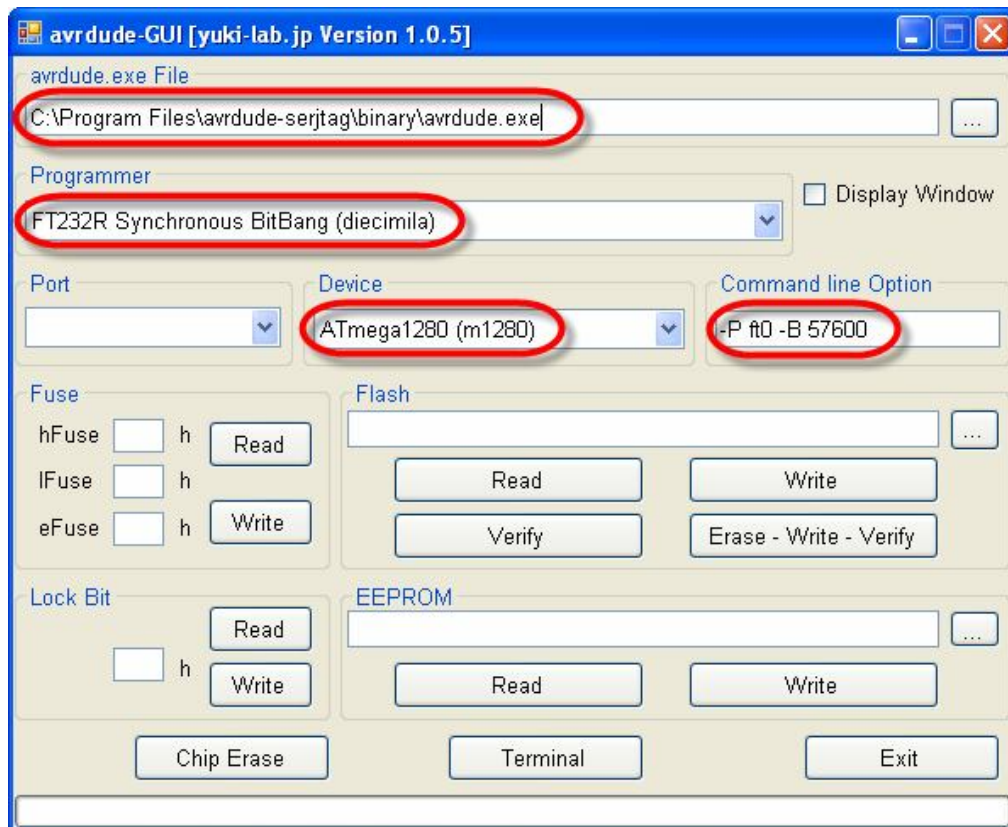
- เลือก Jumper สำหรับ กำหนดการทำงานของ FTDI BitBang ในบอร์ด ET-EASY MEGA1280 ใต้ทางด้าน BOOT PROG ดังรูป



- จ่ายไฟให้กับบอร์ด และทำการต่อสาย USB ระหว่างเครื่องคอมพิวเตอร์ PC เข้ากับบอร์ดให้เรียบร้อย ซึ่งในขั้นตอนนี้ ถ้าหากเป็นการใช้งานครั้งแรก โดยยังไม่เคยทำการติดตั้ง Driver ให้กับชิพ USB Bridge เบอร์ FT232RL ของ FTDI ไว้ก่อนเลย เครื่องคอมพิวเตอร์จะแสดงผลการตรวจพบอุปกรณ์ USB ขึ้นมาใหม่เป็น "FT232R USB UART" ให้ทำการสั่งติดตั้ง Driver ให้เรียบร้อยเสียก่อน ซึ่งรายละเอียดและวิธีการขอให้ไปศึกษาจากหัวข้อการติดตั้ง Driver ในคู่มือ
- สั่ง Run โปรแกรม "avrdude-GUI-1.0.5" ดังรูป

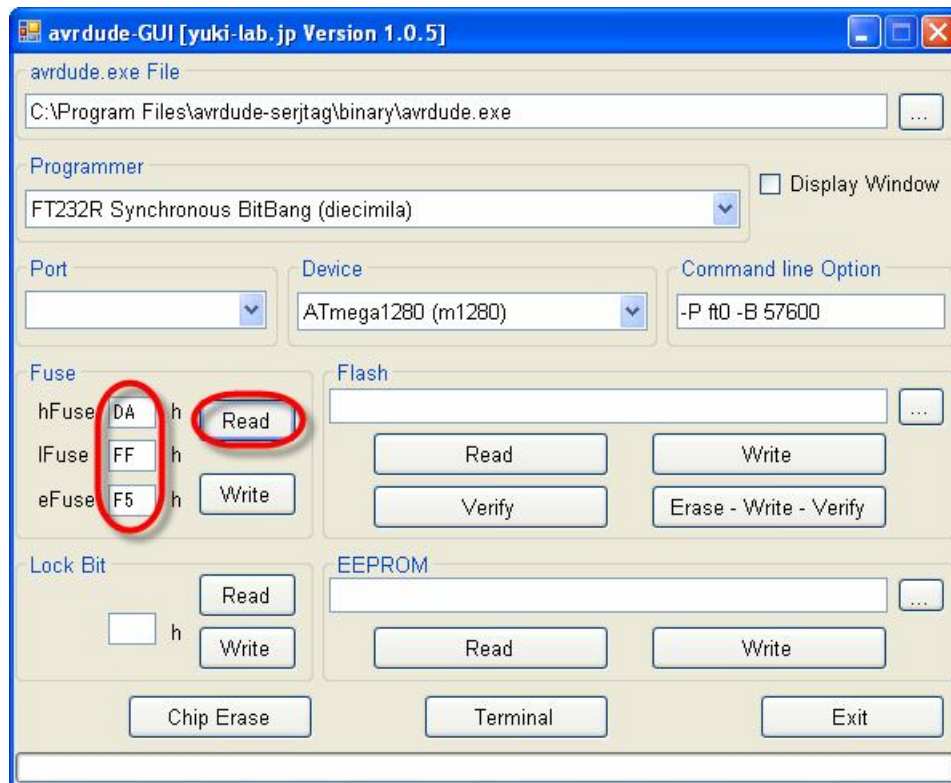


5. ให้ทำการกำหนดค่าการทำงานให้กับโปรแกรม avrdude GUI ดังนี้
 - Avrdude.exe File ให้ทำการกำหนดไฟล์เดอร์ที่เป็นที่อยู่ของโปรแกรม "avrdude.exe" โดยให้ไปที่ไฟล์เดอร์ที่ได้ทำการ Copy ไฟล์ของ avrdude.exe ไปติดตั้งไว้ก่อนหน้านี้แล้ว ในที่นี้คือ "C:\Program Files\avrdude-serjtag\binary\avrdude.exe"
 - Programmer ให้เลือกเป็น FT232R Synchronous BitBang(decimila)
 - Device ให้เลือกเป็น ATmega1280(m1280)
 - Command line Option ให้เลือกเป็น -P ft0 -B 57600

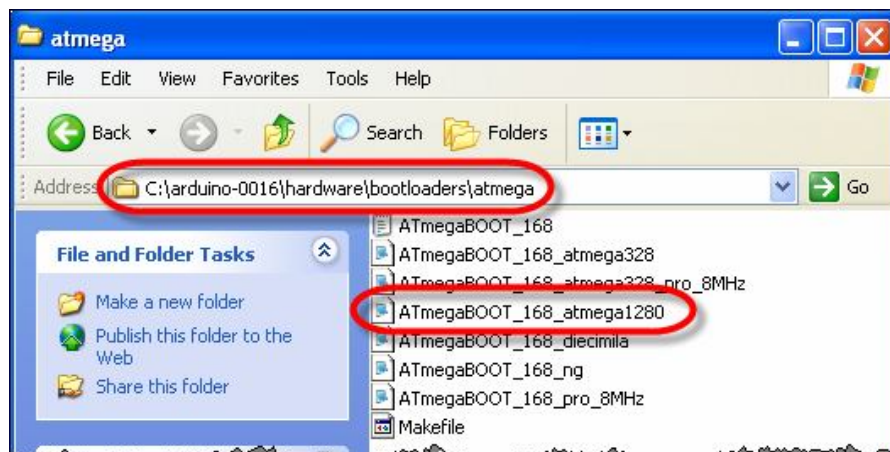


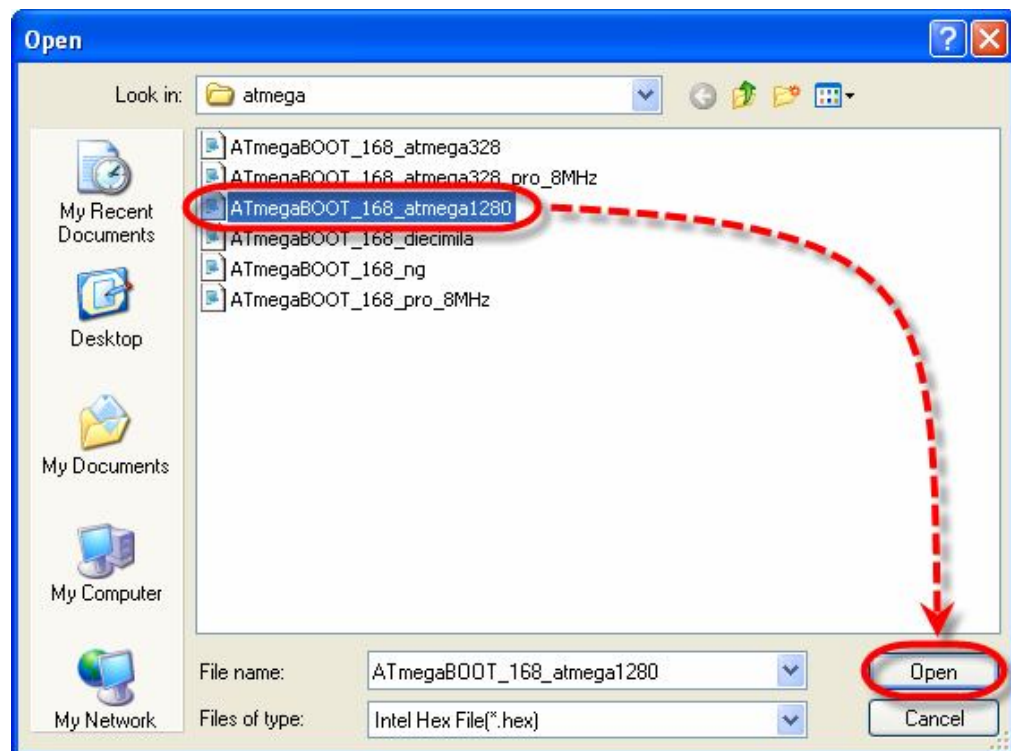
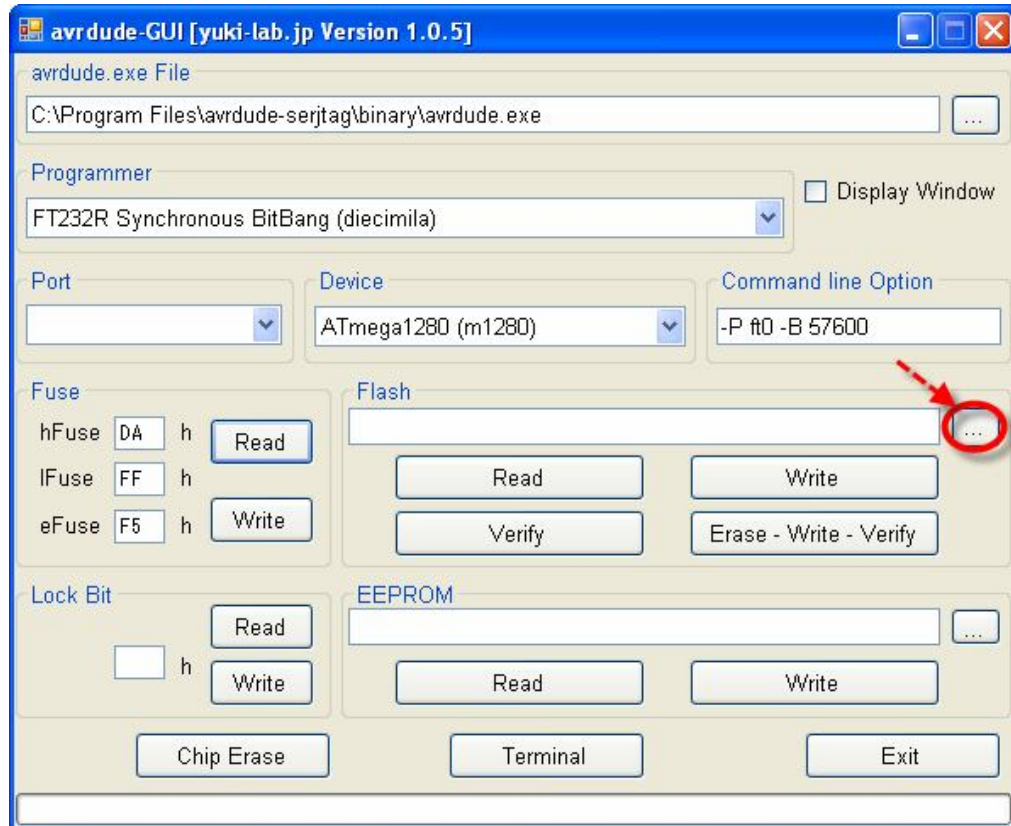
6. ให้ทดสอบการติดต่อสั่งงานโปรแกรม avrdude GUI กับ ATMEGA1280 ของบอร์ดว่าทำงานได้ถูกต้องหรือไม่ โดยให้ลองคลิกเมาส์ที่ปุ่มคำสั่ง Read ของ Fuse แล้วดูผล ซึ่งถ้าทุกอย่างถูกต้องจะได้ค่าของ Fuse Bit แสดงให้เห็นในช่องแสดงค่า แต่ถ้าเกิดความผิดพลาด โปรแกรม avrdude GUI จะรายงานผลเพื่อแสดงความผิดพลาดนั้นให้ทราบ โดยถ้าเกิดความผิดพลาดขึ้นให้ลองตรวจสอบดูว่าได้ทำการเลือก Jumper ของ "BOOT PROG" ภายในบอร์ดไว้ทางด้าน Enable ทั้งหมดแล้วหรือยัง ซึ่งทิศทางที่ถูกต้องของ Jumper จะอยู่ฝั่งทางด้านตัว MCU ATMEGA1280 ทั้งหมด แต่ค่าของ Fuse Bit อาจแตกต่างไปจากตัวอย่างได้ ถ้าค่า Fuse Bit แตกต่างไปจากนี้ ให้ลองแก้ไขแล้วสั่ง

Write เพื่อแก้ไขค่า Fuse ใหม่ให้ถูกต้อง แล้วลองสั่งอ่านค่า Fuse ใหม่อีกครั้งเพื่อตรวจสอบความถูกต้อง ซึ่งถ้าทุกอย่างถูกต้องควรได้ผลการทำงานของโปรแกรกดังตัวอย่าง

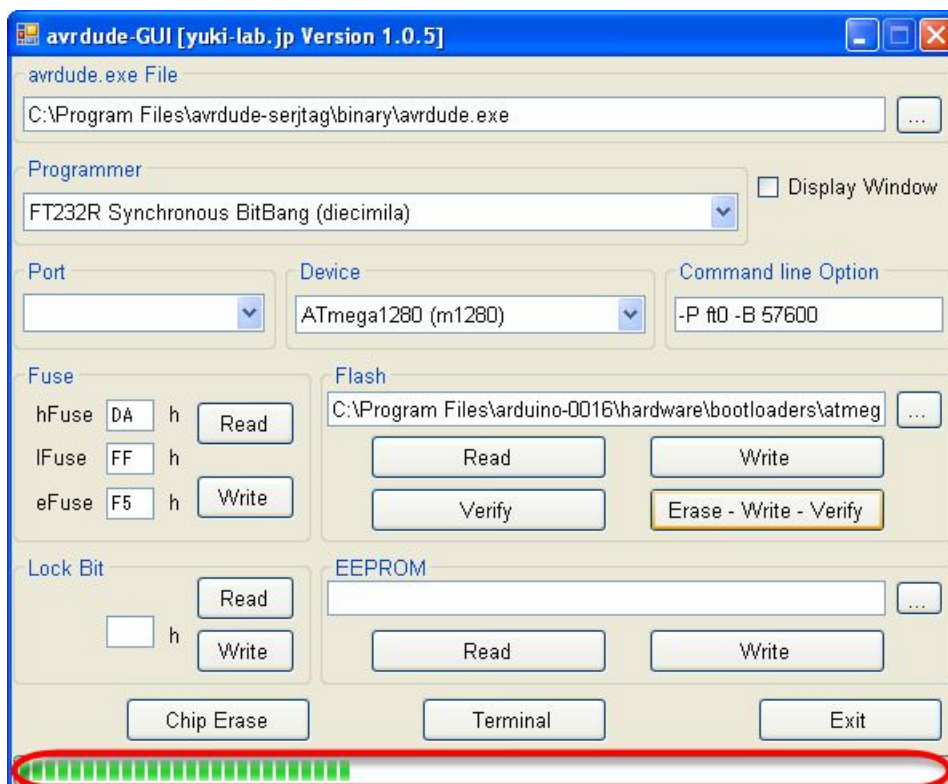
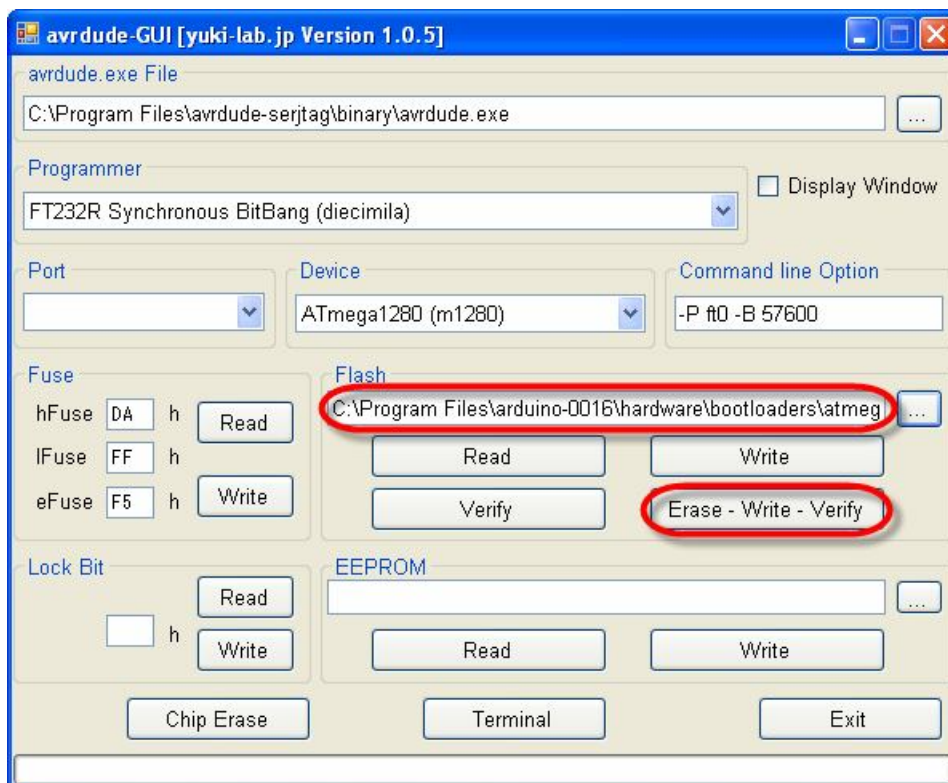


- ที่หัวข้อ **Flash** ให้คลิกเมาส์เพื่อกำหนดตำแหน่งที่จัดเก็บไฟ **Bootloader** ไว้ ซึ่งให้ชี้ไปที่ตำแหน่งของไฟล์เดอร์ที่ได้ทำการติดตั้งโปรแกรมของ **Arduino** ไว้ โดยไฟล์ของ **Bootloader** จะถูกจัดเก็บไว้ใน **"..hardware\bootloaders\atmega"** โดยในกรณีของบอร์ด **ET-EASY MEGA1280** จะเลือกใช้ **Bootloader** ชุดเดียวกันกับบอร์ด **Arduino Mega** ดังนั้นในที่นี้ให้เลือกไฟล์ของ **Bootloader** เป็น **"ATmegaBOOT_168_atmega1280.hex"** ดังรูป

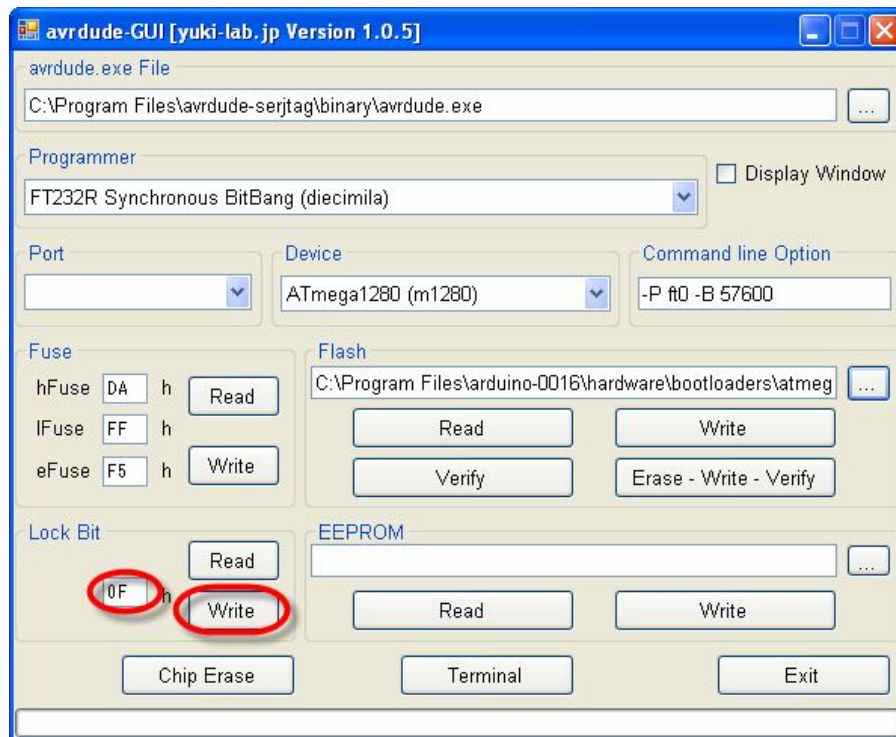




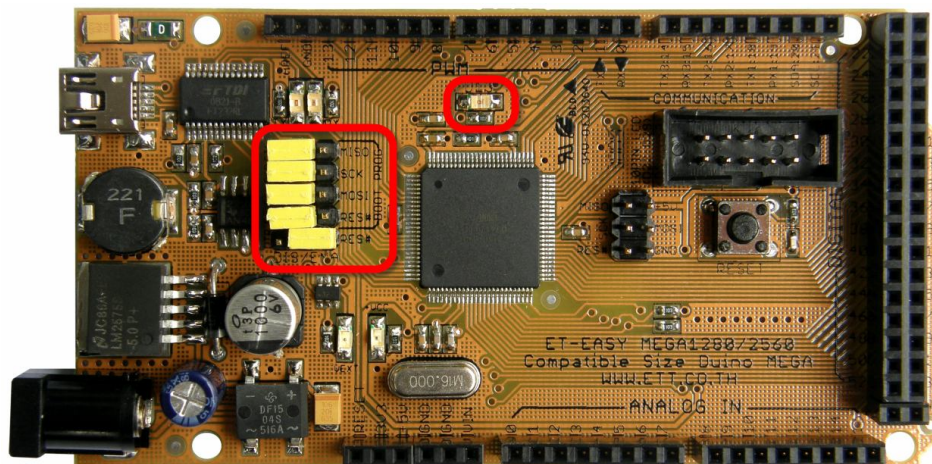
8. ให้เริ่มต้นสั่ง **Burn Bootloader** โดยให้คลิกเมาส์ที่ปุ่มคำสั่ง **erase-Write-Verify** แล้วรอจนการทำงานโปรแกรมเสร็จเรียบร้อย ซึ่งขั้นตอนนี้อาจใช้เวลานานพอสมควร



9. เมื่อการโปรแกรม **Bootloader** เสร็จเรียบร้อยแล้วให้ทำการ ตั้ง **Lock Bit** โดยกำหนดค่าของการ **Lock** เป็น **0F** แล้วเลือกที่ **Write** เมื่อเสร็จให้ลองสั่งอ่านค่า **Lock** กลับดูว่าถูกต้องหรือไม่



10. ซึ่งเมื่อเสร็จเรียบร้อยแล้วให้ ถอดสาย **USB** ออกจากเครื่องคอมพิวเตอร์ **PC** และ หายุดจ่ายไฟให้กับ บอร์ด เพื่อให้ **FT232RL** หยุดการทำงานใน **BitBang** โหมด แล้วเลือก **Jumper** สำหรับ "**BOOT PROG**" ของบอร์ด กลับมาไว้ทางด้าน **Disable** โดยให้เหลือ **Enable** การทำงานของ **Jumper** สำหรับ **DTR** สำหรับใช้ควบคุมสัญญาณ **RES#** ไว้ด้วย แล้วจึงจ่ายไฟให้กับบอร์ดเพื่อให้อบอร์ด เริ่มต้นทำงานตามปกติ ซึ่งควรเห็น **LED** ของ **D13** กระพริบให้เห็น ดังตัวอย่าง

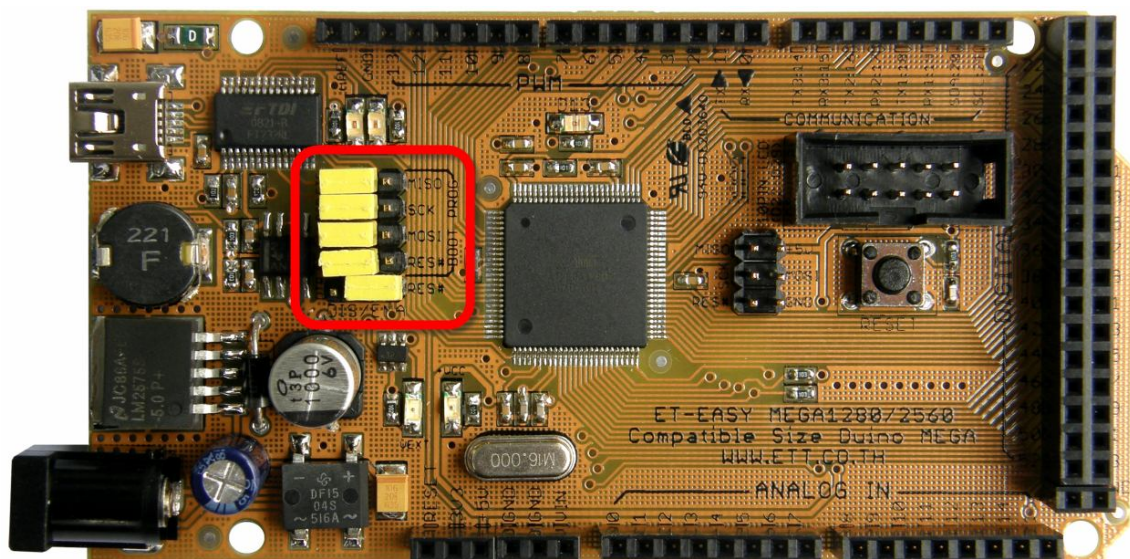


การพัฒนาโปรแกรมของ ET-EASY MEGA1280 ด้วย Arduino

ตามปกติแล้วบอร์ด ET-EASY MEGA1280 จะทำการ ติดตั้งโปรแกรม Bootloader ไว้ให้กับ MCU เป็นที่เรียบร้อยแล้ว โดยใช้ Bootloader ชื่อ "ATmegaBOOT_168_atmega1280.hex" ซึ่งเป็น Bootloader มาตรฐานจาก Arduino โดยโปรแกรม Bootloader นี้จะใช้สำหรับติดต่อสื่อสารเพื่อส่ง Upload Code จากคอมพิวเตอร์ PC ให้กับ MCU ในบอร์ดทำงาน โดยไม่ต้องใช้เครื่องโปรแกรมภายนอกให้ยุ่งยาก ซึ่ง คุณสมบัติของ Bootloader รุ่น Arduino-0016 มีคุณสมบัติการทำงานเป็นดังนี้

- สื่อสารกับโปรแกรมภายนอกด้วย Protocol แบบ STK500 (STK500V1)
- ใช้ความเร็ว Baudrate 57600 โดยใช้ความถี่ XTAL 16 MHz
- โปรแกรม Bootloader มีขนาด 4KByte ทำงานที่ตำแหน่ง 0x1F000-0x1FFFF
- ใช้ LED ที่ต่อกับขา Digital-13 เป็นตัวแสดงสถานะในขณะที่ Bootloader ทำงาน
- โปรแกรมใน Bootloader จะทำงานโดยอัตโนมัติทุกครั้งหลังการรีเซ็ต โดย MCU จะเริ่มต้นทำงานใน Bootloader นี้ก่อนเสมอ เพื่อรอการติดต่อสื่อสารจากโปรแกรมสำหรับสั่งให้ทำการ Upload Code ให้กับ MCU แต่ถ้าไม่มีการติดต่อสื่อสารเข้ามาภายในเวลาที่กำหนดไว้ ก็จะกระโดดไปทำงานตามโปรแกรมที่ผู้ใช้โหลดไว้ให้ทันที

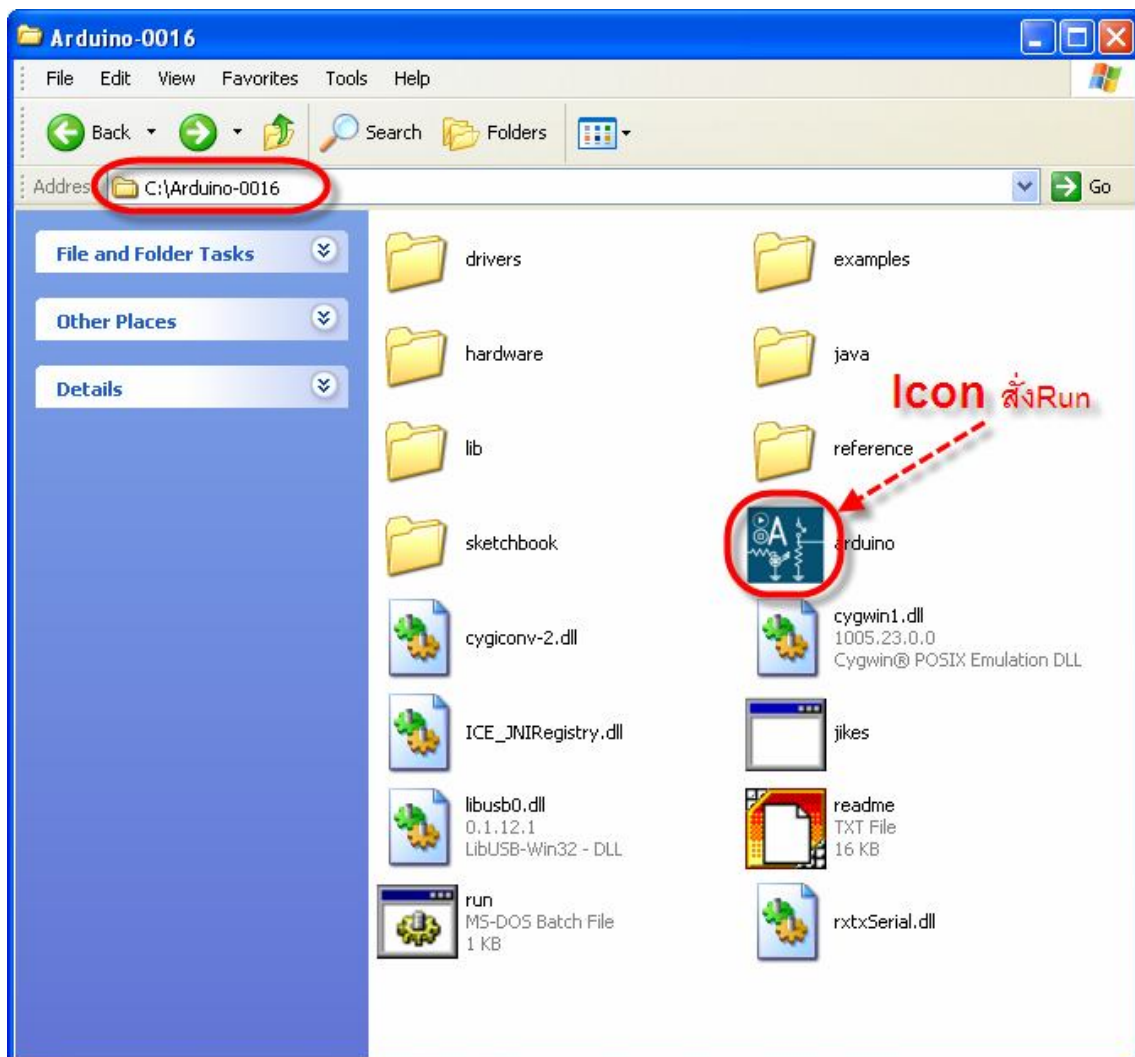
สำหรับบอร์ด ET-EASY MEGA1280 นั้น จะรองรับการ Reset MCU แบบอัตโนมัติจาก USB Bridge (FT232RL) โดยใช้ขา DTR จาก FT232RL เป็นขาควบคุมการรีเซ็ต MCU โดยให้ทำการกำหนด Jumper ไว้ทางด้าน Run Mode ดังรูป



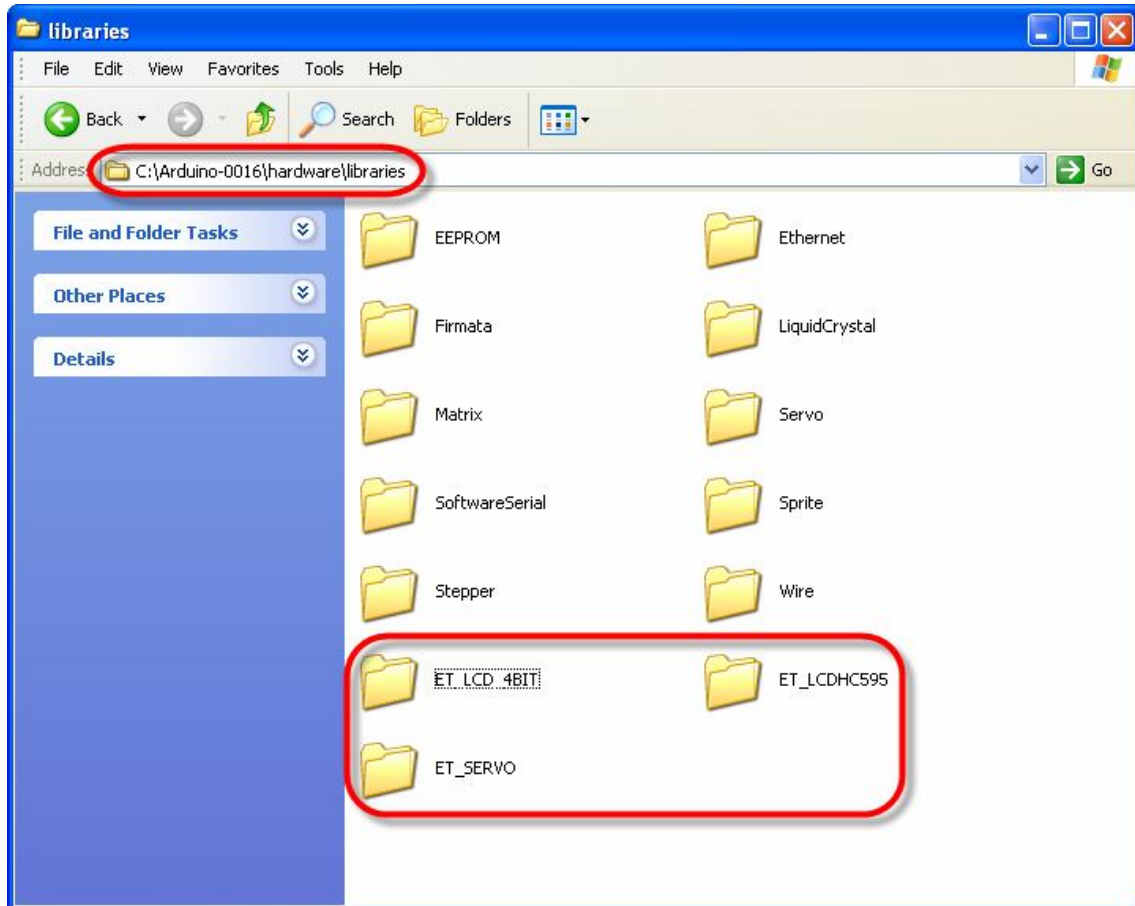
การติดตั้งโปรแกรม Arduino

สำหรับโปรแกรม Arduino นั้น ได้รับการพัฒนาขึ้นมาให้สามารถใช้งานกับระบบปฏิบัติการแบบต่างๆ ได้หลาย Platform ซึ่งปัจจุบัน (เดือน สิงหาคม พ.ศ.2552) โปรแกรมของ Arduino ได้รับการปรับปรุงเป็นรุ่น เวอร์ชัน "Arduino-0016" แล้ว โดยมีโปรแกรมให้เลือกใช้งาน 4 Platform ทั้ง Windows, Mac OSx และ Linux โดยผู้อ่านสามารถเข้าไป ตรวจสอบ หรือ Download โปรแกรมรุ่นใหม่ๆ ของ Arduino มาใช้งาน ได้ฟรีโดยไม่เสียค่าใช้จ่ายใดๆ จาก "<http://arduino.cc/>" หรือ "<http://arduino.cc/en/Main/Software>" ซึ่งเป็นเว็บไซต์ที่ได้รวบรวมรายละเอียดและข่าวคราวความเคลื่อนไหวต่างๆ เกี่ยวกับ Arduino มากมาย ซึ่งข้อมูลต่างๆ จะได้รับการปรับปรุงอย่างต่อเนื่องเป็นประจำ

โดยในการติดตั้งโปรแกรมของ Arduino นั้นให้ทำการ Unzip แล้ว Copy ไปติดตั้งไว้ในตำแหน่งโฟลเดอร์ "c:\arduino-0016" ดังตัวอย่าง



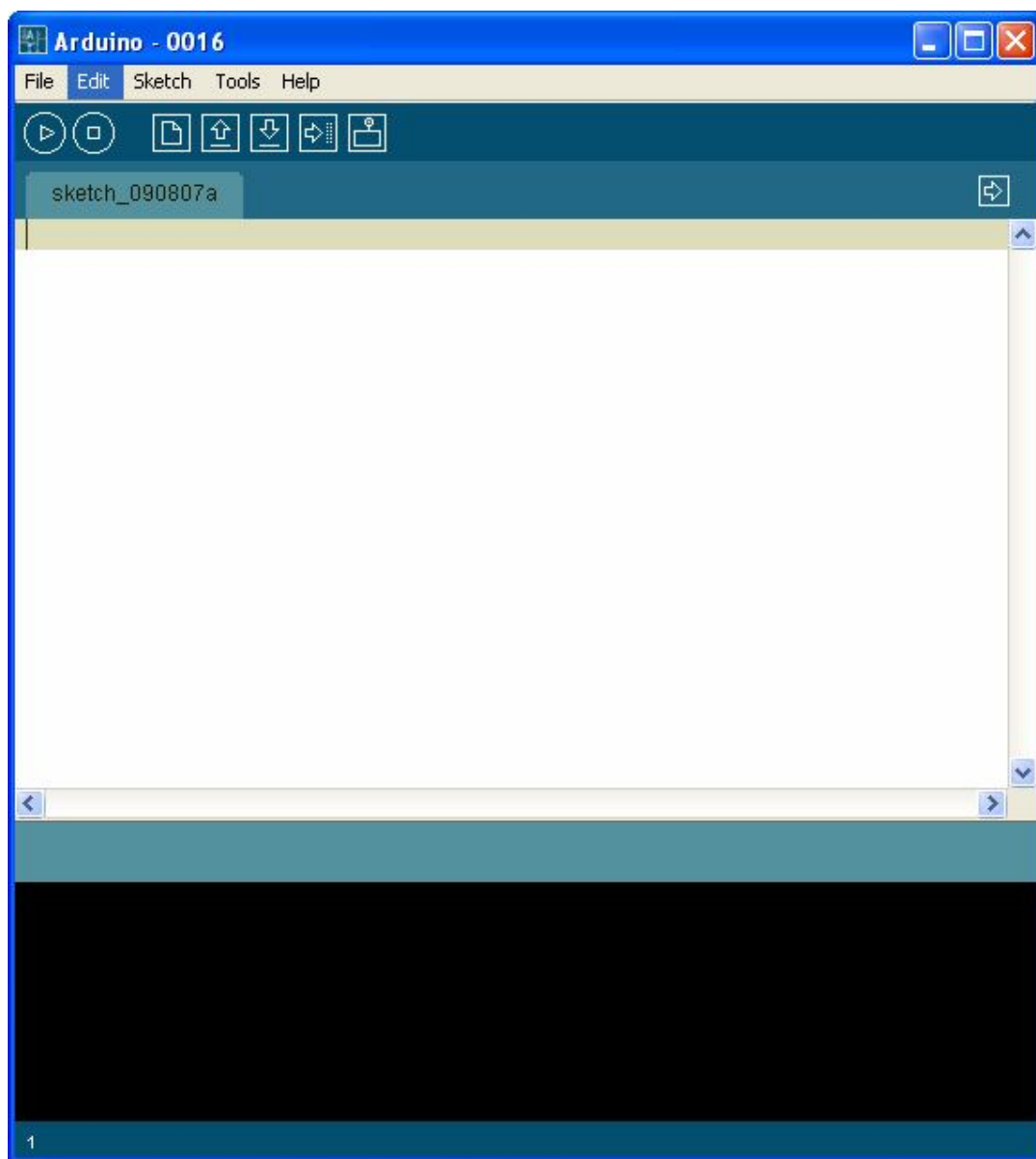
ให้ทำการ Copy โฟลเดอร์ Library ต่างๆ ของ อีทีที ในแผ่น CD-ROM เพิ่มเข้าไปในโฟลเดอร์ที่ได้ทำการติดตั้งโปรแกรม Arduino ไว้คือ "c:\Arduino-0016\hardware\Library" ดังรูป



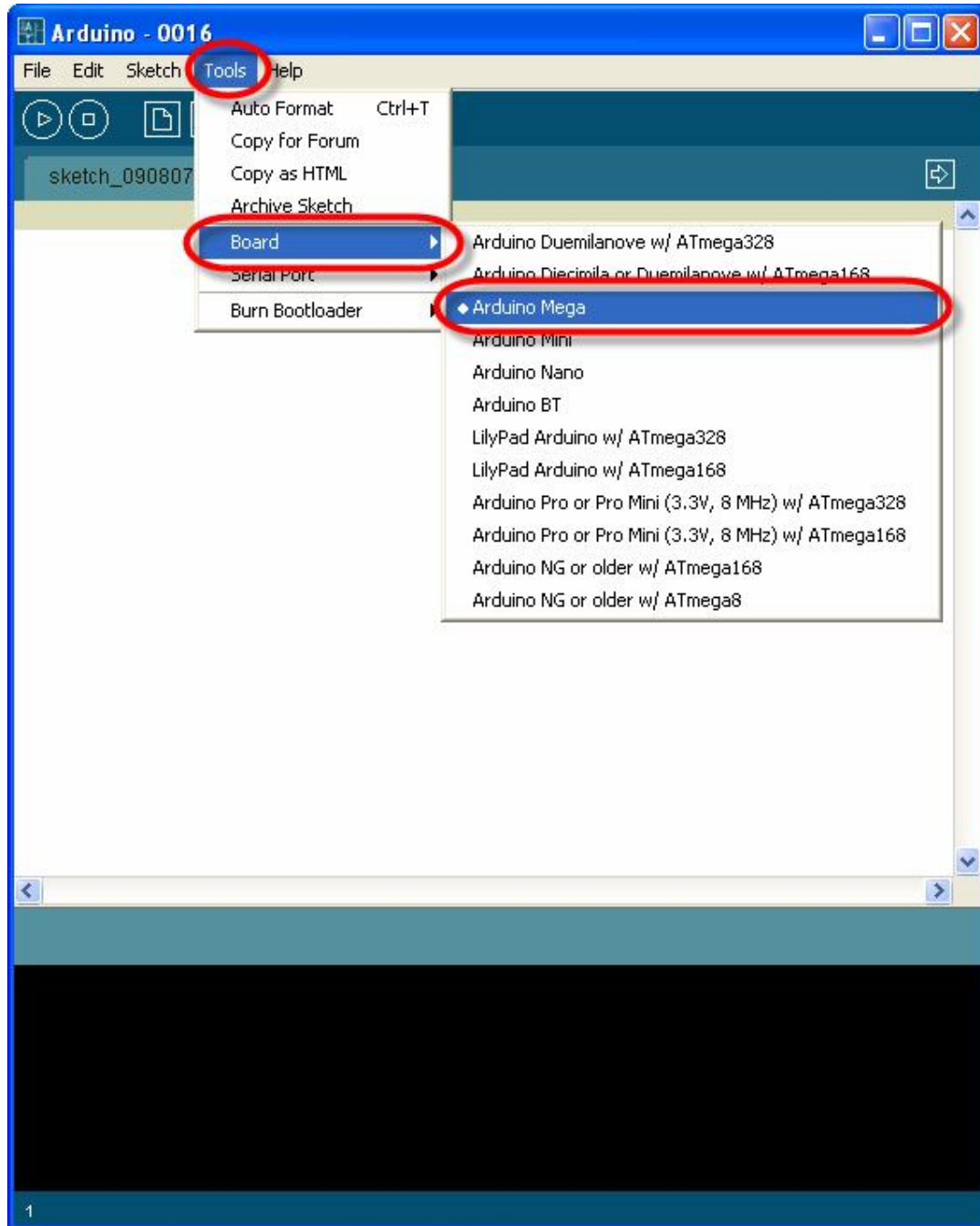
ทดสอบเขียนโปรแกรมใช้งานด้วย Arduino

หลังจากที่เราได้ทำการติดตั้งโปรแกรม Arduino เป็นที่เรียบร้อยแล้ว ก็เป็นอันเสร็จสิ้นขั้นตอนของการเตรียมการแล้ว ลำดับขั้นตอนต่อจากนี้เป็นต้นไป ก็เป็นเรื่องของการใช้งาน การเขียนโปรแกรม และการศึกษาเรียนรู้ต่างๆตามความต้องการแล้ว แต่ก่อนอื่นเราจะต้องทำการติดตั้งโปรแกรมของ Arduino เพื่อให้เป็นโปรแกรมสำหรับศึกษาเรียนรู้ ซึ่งมีลำดับขั้นตอนดังต่อไปนี้

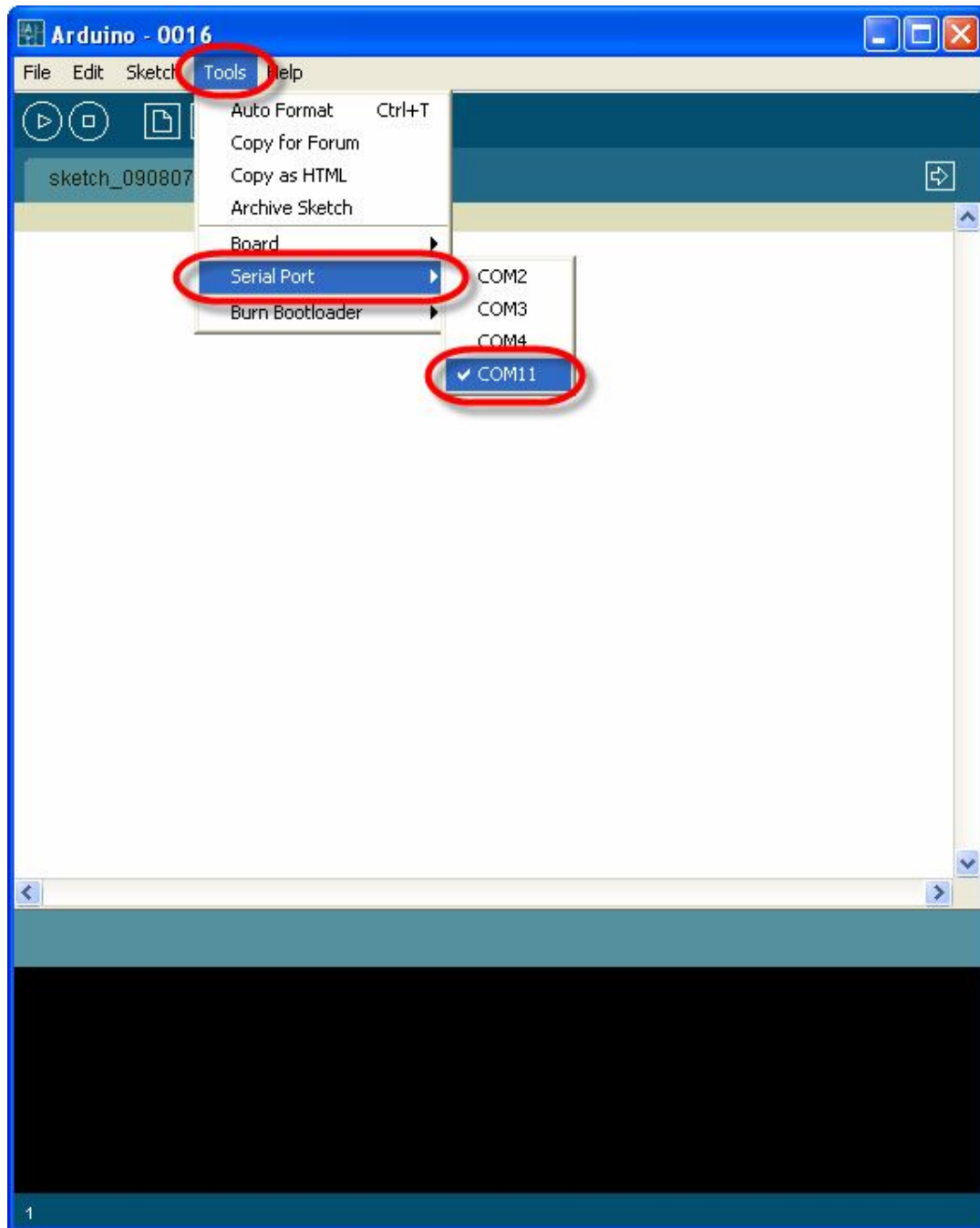
1. ทำการสั่ง Run โปรแกรม "arduino.exe" จะได้ผลดังรูป



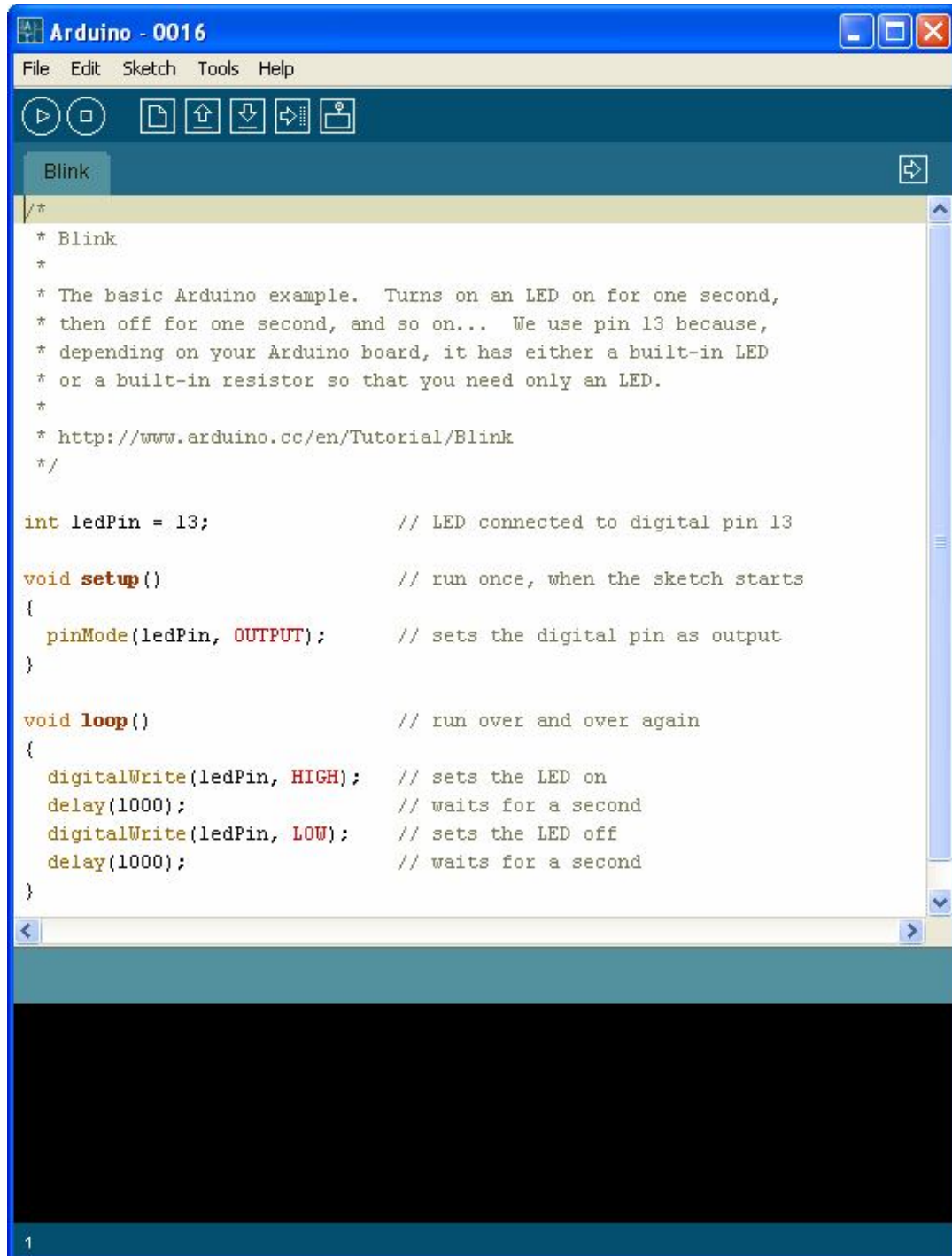
2. ในครั้งแรกของการเรียกใช้งานโปรแกรม ให้ทำการกำหนดระบบฮาร์ดแวร์ที่จะใช้งานกับโปรแกรมของ **Arduino** ให้เรียบร้อยเสียก่อน เนื่องจากในปัจจุบันนี้ มีการออกแบบวงจรและสร้างฮาร์ดแวร์บอร์ดแบบต่างๆสำหรับนำมาใช้งานร่วมกับโปรแกรมพัฒนาของ **Arduino** อยู่มากมายหลายรุ่น โดยในกรณีของบอร์ด ET-EASY MEGA1280 ให้ทำการเลือกกำหนดชื่อบอร์ดเป็น **"Arduino Mega"** โดยคลิกเมาส์ที่ **"Tools → Board → "Arduino Mega"** ดังรูป



3. เลือกกำหนดหมายเลขพอร์ต สำหรับติดต่อสื่อสารกับบอร์ด ให้ตรงกับหมายเลข Comport ที่ต่อใช้งานไว้จริงในเครื่องคอมพิวเตอร์ PC เช่น ถ้าหมายเลข Comport ของเครื่องคอมพิวเตอร์ PC เป็น COM11 ให้คลิกเมาส์ที่ Tools → Serial Port → COM11 ดังรูป



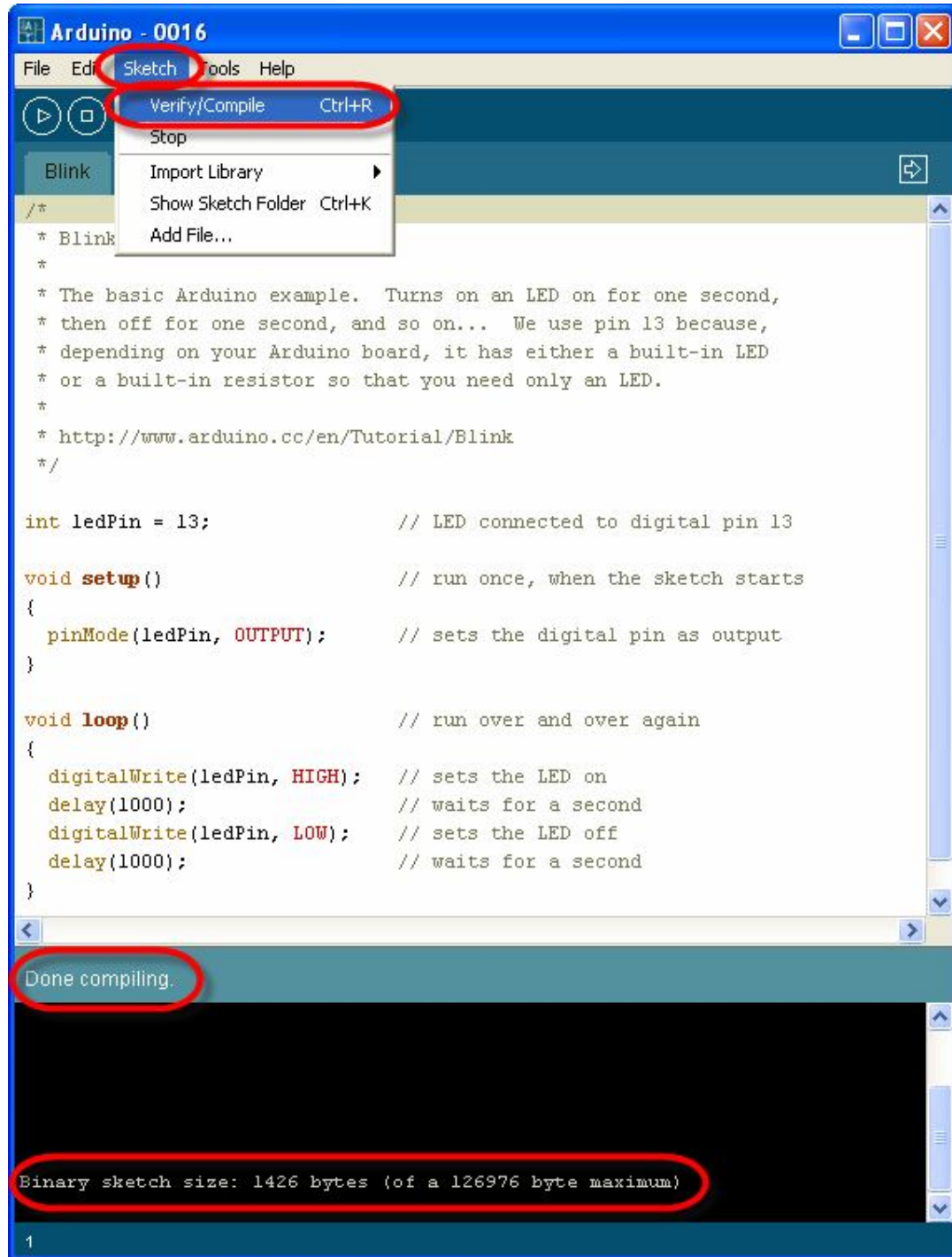
4. ทดสอบเขียนโปรแกรม โดยคลิกเมาส์ที่ File → New แล้วพิมพ์โปรแกรมทดสอบ หรืออาจใช้การ
สั่งเปิดไฟล์ตัวอย่างที่สร้างไว้แล้วขึ้นมาแทนก็ได้ โดยในที่นี้ขอแนะนำให้ทดสอบด้วยโปรแกรมไฟ
กระพริบ โดยให้เลือก “File → sketchbook → Examples → Digital → Blink” ซึ่งจะได้ดังรูป

The image shows the Arduino IDE interface. The title bar reads "Arduino - 0016". The menu bar includes "File", "Edit", "Sketch", "Tools", and "Help". Below the menu bar is a toolbar with icons for running, stopping, saving, opening, and uploading. A "Blink" tab is selected. The code editor contains the following text:

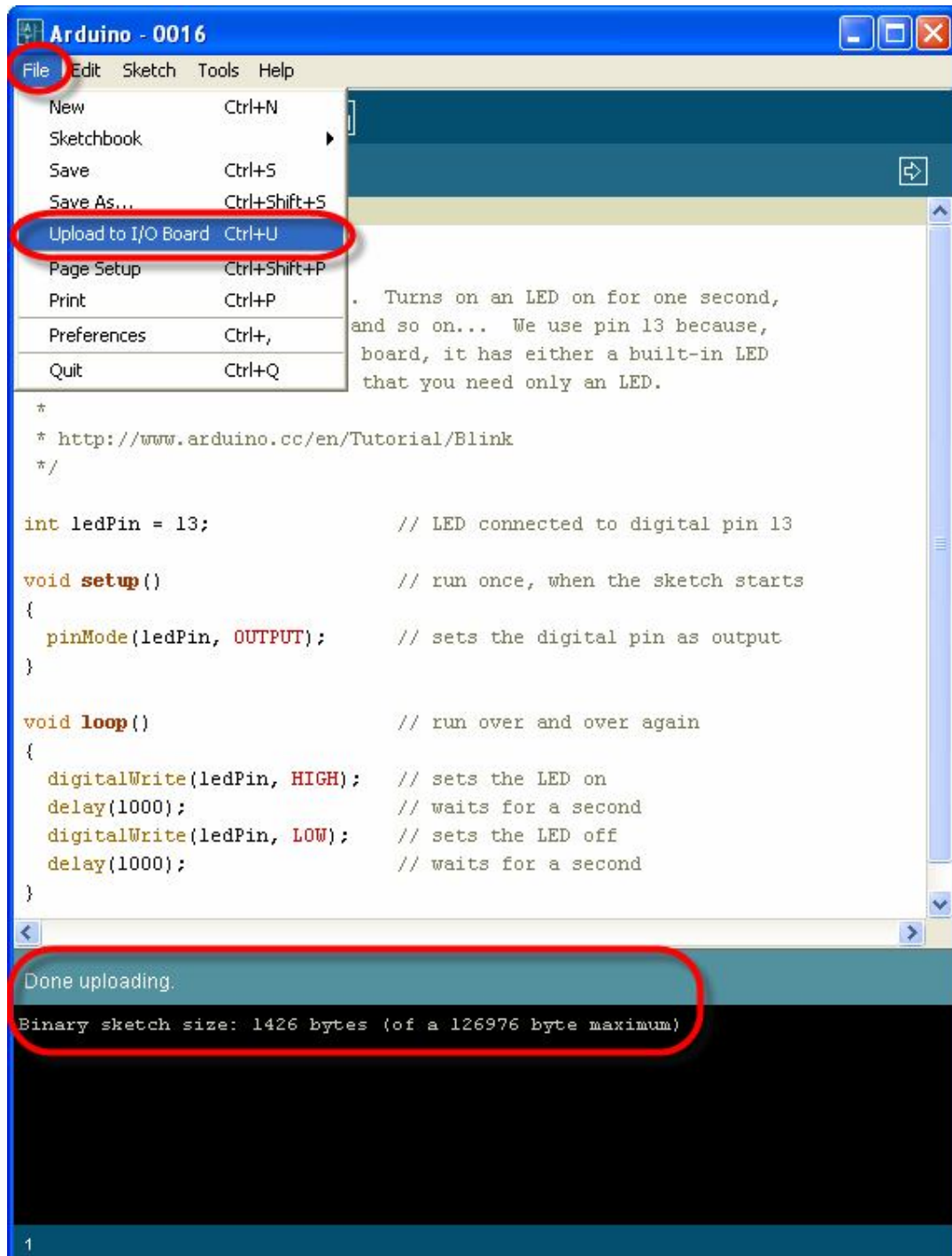
```
/*  
 * Blink  
 *  
 * The basic Arduino example. Turns on an LED on for one second,  
 * then off for one second, and so on... We use pin 13 because,  
 * depending on your Arduino board, it has either a built-in LED  
 * or a built-in resistor so that you need only an LED.  
 *  
 * http://www.arduino.cc/en/Tutorial/Blink  
 */  
  
int ledPin = 13;           // LED connected to digital pin 13  
  
void setup()               // run once, when the sketch starts  
{  
  pinMode(ledPin, OUTPUT); // sets the digital pin as output  
}  
  
void loop()                // run over and over again  
{  
  digitalWrite(ledPin, HIGH); // sets the LED on  
  delay(1000);                // waits for a second  
  digitalWrite(ledPin, LOW);  // sets the LED off  
  delay(1000);                // waits for a second  
}
```

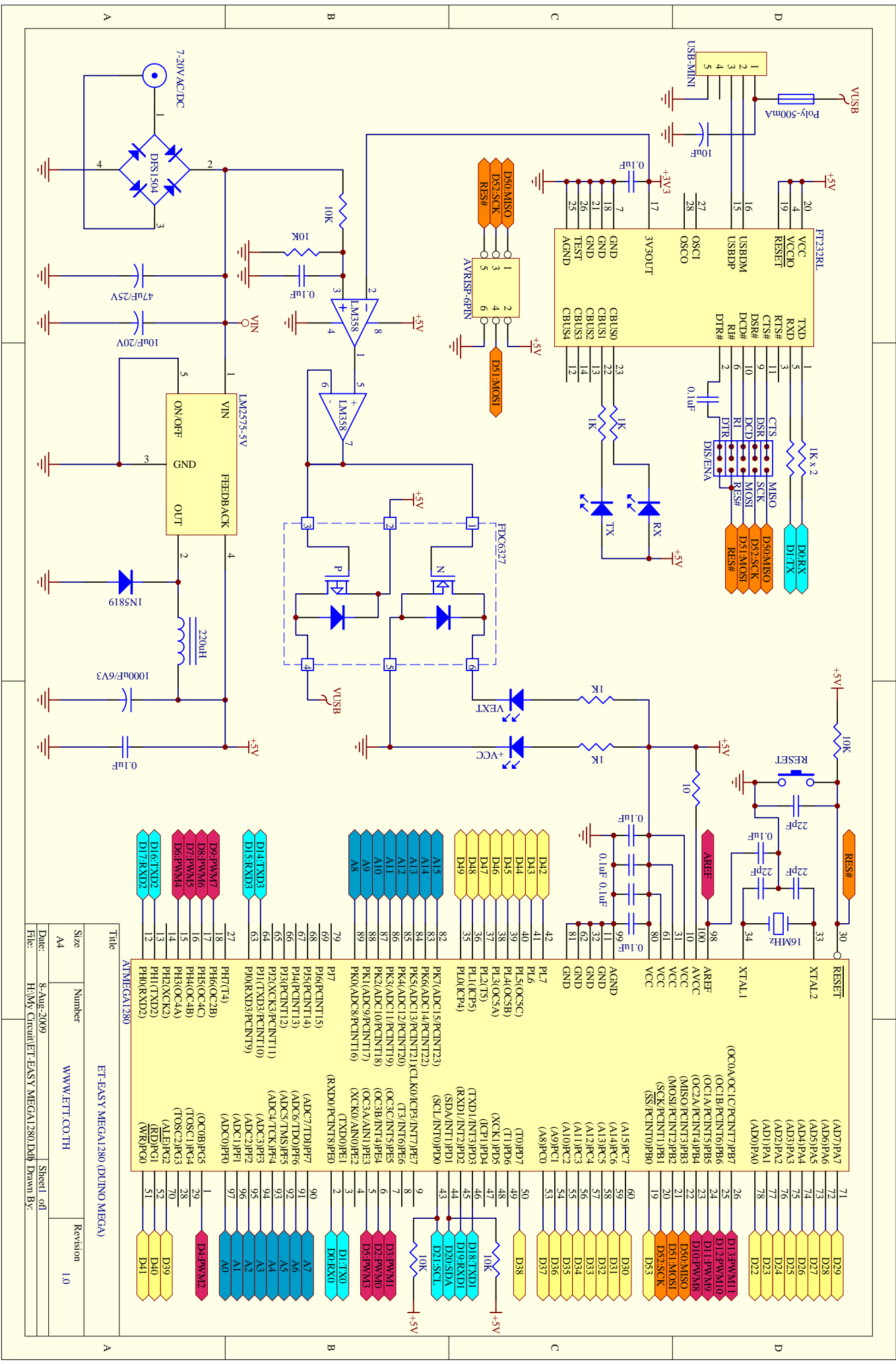
The status bar at the bottom left shows the line number "1".

5. สั่งแปลโปรแกรมโดยคลิกเมาส์ที่ "Sketch → Verify/Compile" เพื่อตรวจสอบคำสั่งต่างๆในโปรแกรมว่าถูกต้องหรือไม่ ดังตัวอย่าง



6. สั่ง Download Code ให้กับบอร์ด โดยคลิกเมาส์เลือกที่ "File → Upload to I/O Board" แล้วรอ
สักครู่จนโปรแกรมทำงานเสร็จ หลังจากที่ทำกร Upload Code ให้กับบอร์ดเป็นที่เรียบร้อยแล้ว
บอร์ดก็จะเริ่มต้นทำงานตามคำสั่งที่เขียนไว้ในโปรแกรมทันที โดยจะสังเกตเห็น LED กระพริบ ติด
และดับ สลับกันไปมา ด้วยความเร็วประมาณ 1 วินาที ตลอดเวลา ซึ่งควรได้ผลดังรูป





Title	
ET-EASY MEGA1280 (DUINO MEGA)	
Size	Number
A4	7
Date:	8-Aug-2009
File:	H:\My Circuit\ET-EASY MEGA1280.D06 Drawn By:

Revision	
1.0	

