

ET-BASE AVR ATmega64/128 r3

ET-BASE AVR ATmega64/128 r3 เป็นบอร์ดไมโครคอนโทรลเลอร์ในตระกูล AVR ของบริษัท Atmel ซึ่งบอร์ดนี้เลือกใช้ MCU เบอร์ ATmega64 และ เบอร์ ATmega128 ขนาด 64 Pin โดยในบอร์ด ET-BASE AVR ATmega64/128 r3 นี้จะเน้นจะเน้นการใช้งานทรัพยากรของตัว MCU เองเป็นหลัก ซึ่งจะมีการต่อขาสัญญาณ I/O ออกมาจัดเรียงให้เป็นพอร์ต PA,PB,PC,PD,PE,PF และพอร์ต ET-CLCD เพื่อสะดวกต่อการใช้งาน พร้อมทั้งพอร์ตสำหรับดาวน์โหลดโปรแกรม นอกจากนี้ยังได้เพิ่มวงจร Line Driver RS-232 เข้าไปด้วยเพื่อให้สามารถใช้งานทางด้านพอร์ตอนุกรม RS-232 ได้ง่ายและสะดวกยิ่งขึ้น

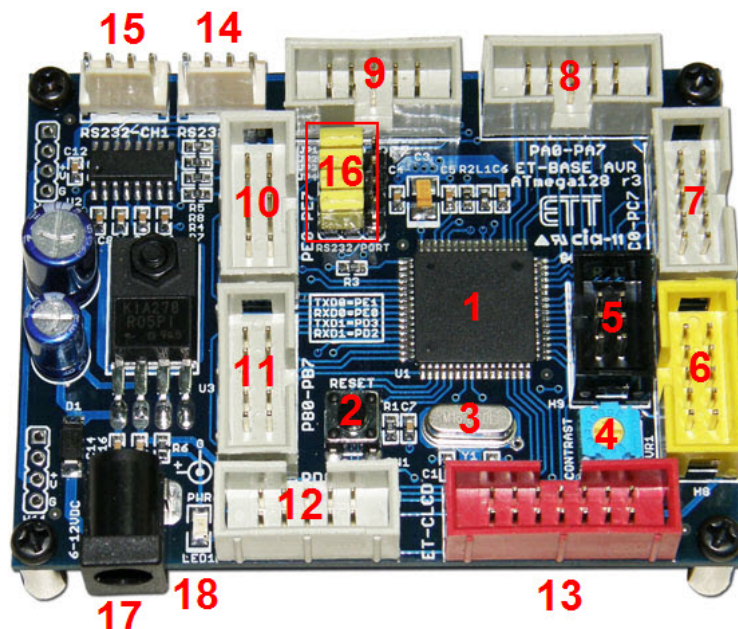
คุณสมบัติของบอร์ด

- เลือกใช้ MCU ตระกูล AVR เบอร์ ATmega64 , ATmega128 ของ Atmel ซึ่งเป็น MCU ขนาด 8-Bit โดยเลือกใช้แหล่งกำเนิดสัญญาณนาฬิกาแบบ XTAL ค่า 16 MHz ซึ่งคุณสมบัติเด่น ๆ ของ MCU ได้แก่
 - มีหน่วยความจำ Flash สำหรับเขียนโปรแกรม 64 KBytes สำหรับ ATmega64 และ 128K Bytes สำหรับ ATmega128 และมี RAM 4 KBytes
 - มีหน่วยความจำข้อมูลถาวรแบบ EEPROM ขนาด 2K Bytes สำหรับ ATmega64 และ 4 K Byte สำหรับ ATmega128 ซึ่งสามารถลบและเขียนซ้ำได้กว่า 100,000 ครั้ง
 - จำนวน I/O สูงสุดถึง 53 I/O Pins
 - มีวงจรสื่อสาร SPI จำนวน 1 ช่อง , I2C จำนวน 1 ช่อง , Programmable Serial USARTs จำนวน 2 ช่อง
 - มี ADC ขนาด 10-Bit จำนวน 8 ช่อง
 - มี Timers/Counters 8-Bit จำนวน 2 ช่อง , Timers/Counters 16-Bit จำนวน 2 ช่อง , 8-Bit PWM 2 ช่อง , Watchdog Timer , Real Time Counter
- I/O PORT 10 PIN จำนวน 6 PORT ดังนี้ PA,PB,PC,PD,PE,PF
- พอร์ต ISP LOAD สำหรับโปรแกรม MCU (ต้องใช้ร่วมกับ ET-AVR ISP หรือเครื่องโปรแกรม ISP อื่นที่มีการจัดเรียงขาสัญญาณเหมือนกัน)
- วงจร Line Driver สำหรับพอร์ตสื่อสารอนุกรม RS232 จำนวน 2 ช่อง โดยเชื่อมต่อกับสัญญาณ PE0(RXD0) และ PE1(TXD0) จำนวน 1 ช่อง ส่วนที่เหลืออีก 1 ช่อง จะต่อกับ

สัญญาณ PD2(RXD1) และ PD3(TXD1) เพื่อให้ผู้ใช้สามารถต่อทดลองการติดต่อสื่อสาร RS232

- วงจรเชื่อมต่อจอแสดงผล LCD แบบ Character (ET-CLCD) พร้อม VR ปรับความเข้มของ LCD ซึ่งใช้การเชื่อมต่อวงจรกับ LCD แบบ 4 Bit Interface
- วงจร Regulate ขนาด +5V / 2A สำหรับใช้งานเป็นแหล่งจ่ายไฟเลี้ยงวงจรให้กับจอแสดงผล LCD และอุปกรณ์ I/O ต่างๆที่ใช้กับแหล่งจ่ายขนาด +5V พร้อม LED แสดงสถานะสีแดง
- ขนาด PCB Size เล็กเพียง 8 X 6 cm

โครงสร้างของบอร์ด



- หมายเลข 1 คือ MCU เบอร์ ATmega64 หรือ ATmega128 ซึ่งเป็น MCU ตระกูล AVR จาก ATMEL
- หมายเลข 2 คือ Switch RESET ใช้สำหรับ Reset การทำงานของ MCU
- หมายเลข 3 คือ Crystal ค่า 16 MHz
- หมายเลข 4 คือ ตัวต้านทานสำหรับปรับค่าความเข้มให้ LCD
- หมายเลข 5 พอร์ต AVR ISP (6 PIN) ใช้สำหรับดาวน์โหลด Hex File ให้กับ MCU
- หมายเลข 6 พอร์ต AVR ISP (10 PIN) ใช้สำหรับดาวน์โหลด Hex File ให้กับ MCU
- หมายเลข 7 คือ PORTC มีขนาด 8 Bit คือ PC0-PC7

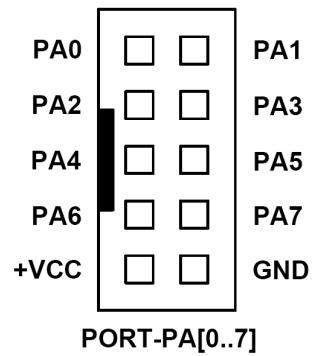
- หมายเลข 8 คือ PORTA มีขนาด 8 Bit คือ PA0-PA7
- หมายเลข 9 คือ PORTF มีขนาด 8 Bit คือ PF0-PF7
- หมายเลข 10 คือ PORTE มีขนาด 8 Bit คือ PE0-PE7
- หมายเลข 11 คือ PORTB มีขนาด 8 Bit คือ PB0-PB7
- หมายเลข 12 คือ PORTD มีขนาด 8 Bit คือ PD0-PD7
- หมายเลข 13 คือ พอร์ต ET-CLCD สำหรับเชื่อมต่อกับ LCD ชนิด Character Type ซึ่งใช้การเชื่อมต่อแบบ 4 Bit
- หมายเลข 14 และ 15 คือ ขั้วต่อ RS232 สำหรับใช้งานทั่วไป
- หมายเลข 16 คือ จัมเปอร์ สำหรับเลือกใช้งาน RS232 หรือ พอร์ต IO
- หมายเลข 17 คือ ขั้วต่อแหล่งจ่ายไฟสำหรับเลี้ยงวงจรของบอร์ด
- หมายเลข 18 คือ LED Power ใช้สำหรับแสดงสถานะของแหล่งจ่ายไฟ +5VDC

ขั้วต่อสัญญาณต่าง ๆ

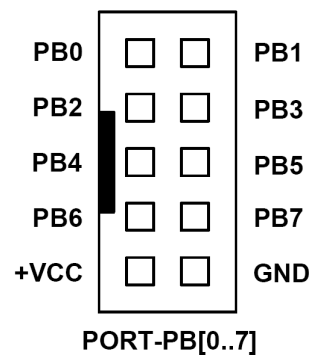
สำหรับขั้วต่อสัญญาณของพอร์ต I/O จาก MCU นั้นจะถูกออกแบบและจัดเตรียมไว้ผ่านทางขั้วต่อแบบ IDC-Header ขนาด 10 Pin (2X5) จำนวน 6 ชุด คือ PA,PB,PC,PD,PE,PF ตามลำดับ โดยที่ขั้วต่อสัญญาณแต่ละชุด จะประกอบไปด้วยสัญญาณของ I/O ที่เชื่อมต่อมาจากขาสัญญาณของ MCU โดยตรงทั้งหมด โดยจุดเชื่อมต่อกับสัญญาณภายนอกบอร์ดมีดังนี้

- ขั้วต่อแหล่งจ่ายไฟสำหรับเลี้ยงวงจรของบอร์ด
- ขั้วต่อ PORTA มีขนาด 8 Bit คือ PA0-PA7
- ขั้วต่อ PORTB มีขนาด 8 Bit คือ PB0-PB7
- ขั้วต่อ PORTC มีขนาด 8 Bit คือ PC0-PC7
- ขั้วต่อ PORTD มีขนาด 8 Bit คือ PD0-PD7
- ขั้วต่อ PORTE มีขนาด 8 Bit คือ PE0-PE7
- ขั้วต่อ PORTF มีขนาด 8 Bit คือ PF0-PF7
- ขั้วต่อ ET-CLCD สำหรับเชื่อมต่อกับ LCD ชนิด Character Type
- ขั้วต่อ RS232 จำนวน 2 ช่อง โดยเชื่อมต่อกับสัญญาณ PE0(RXD0) และ PE1(TXD0) จำนวน 1 ช่อง ส่วนที่เหลืออีก 1 ช่อง จะต่อกับสัญญาณ PD2(RXD1) และ PD3(TXD1) เพื่อให้ผู้ใช้สามารถทดสอบการติดต่อสื่อสาร RS232
- ขั้วต่อ AVR ISP ใช้สำหรับดาวน์โหลด Hex File ให้กับ MCU

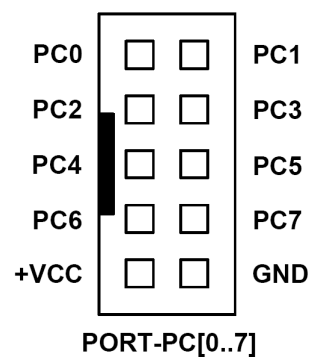
พอร์ต PA มีขนาด 8 บิต



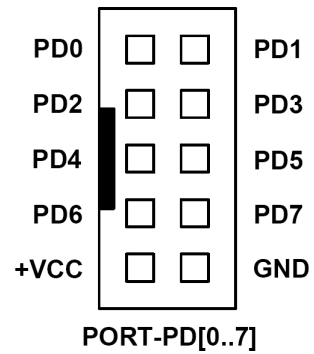
พอร์ต PB มีขนาด 8 บิต



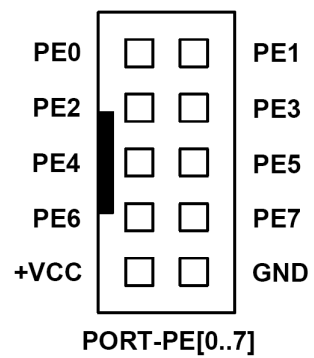
พอร์ต PC มีขนาด 8 บิต



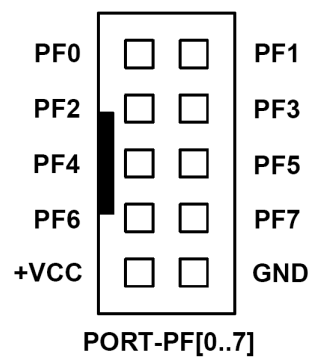
พอร์ต PD มีขนาด 8 บิต



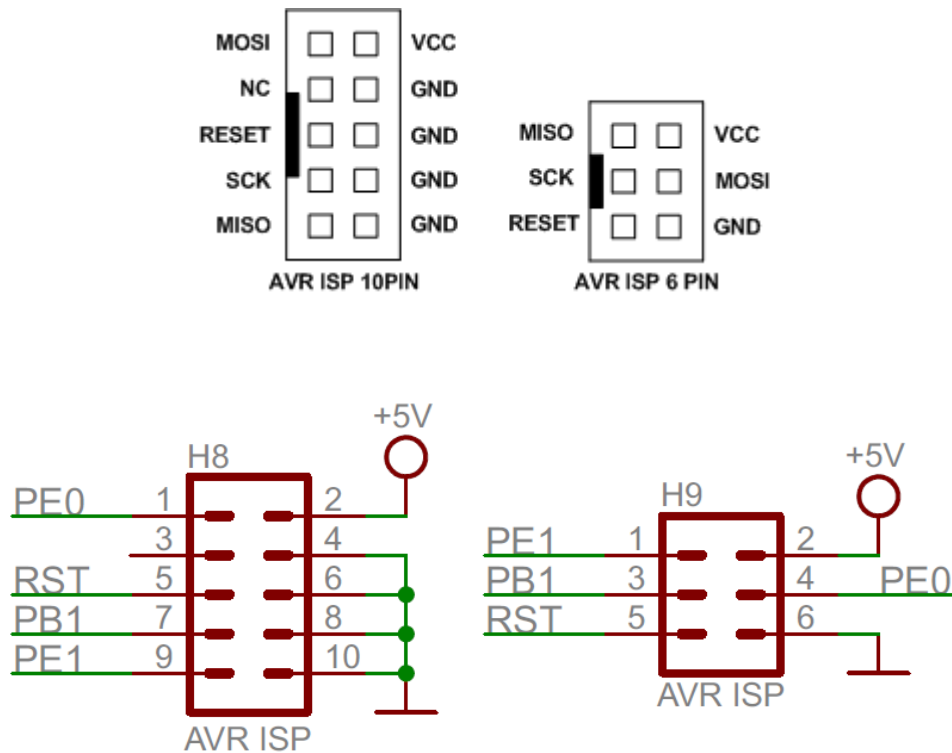
พอร์ต PE มีขนาด 8 บิต



พอร์ต PF มีขนาด 8 บิต

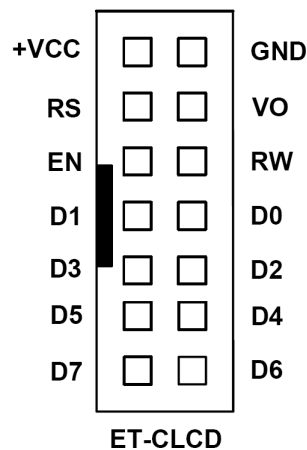


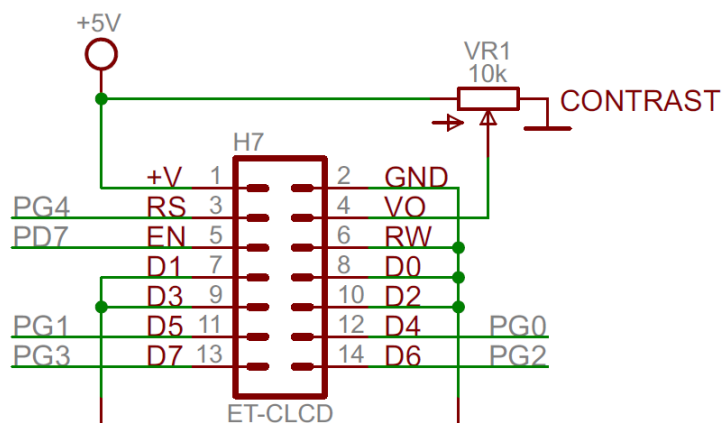
พอร์ต AVR ISP



รูปแสดง วงจรส่วนที่เชื่อมต่อกับ AVR ISP

พอร์ต ET-CLCD ใช้กับ Character Type LCD โดยใช้การเชื่อมต่อแบบ 4 บิต โดยสัญญาณที่ใช้เชื่อมต่อกับ LCD จะเป็นสัญญาณจากพอร์ต PG และ PD (PD7) โดยในการเชื่อมต่อสายสัญญาณจากขั้วต่อของพอร์ต LCD ไปยังจอแสดงผล LCD นั้นให้ยึดชื่อขาสัญญาณเป็นจุดอ้างอิง โดยให้ต่อสัญญาณที่มีชื่อตรงกันเข้าด้วยกันให้ครบทั้ง 14 เส้น

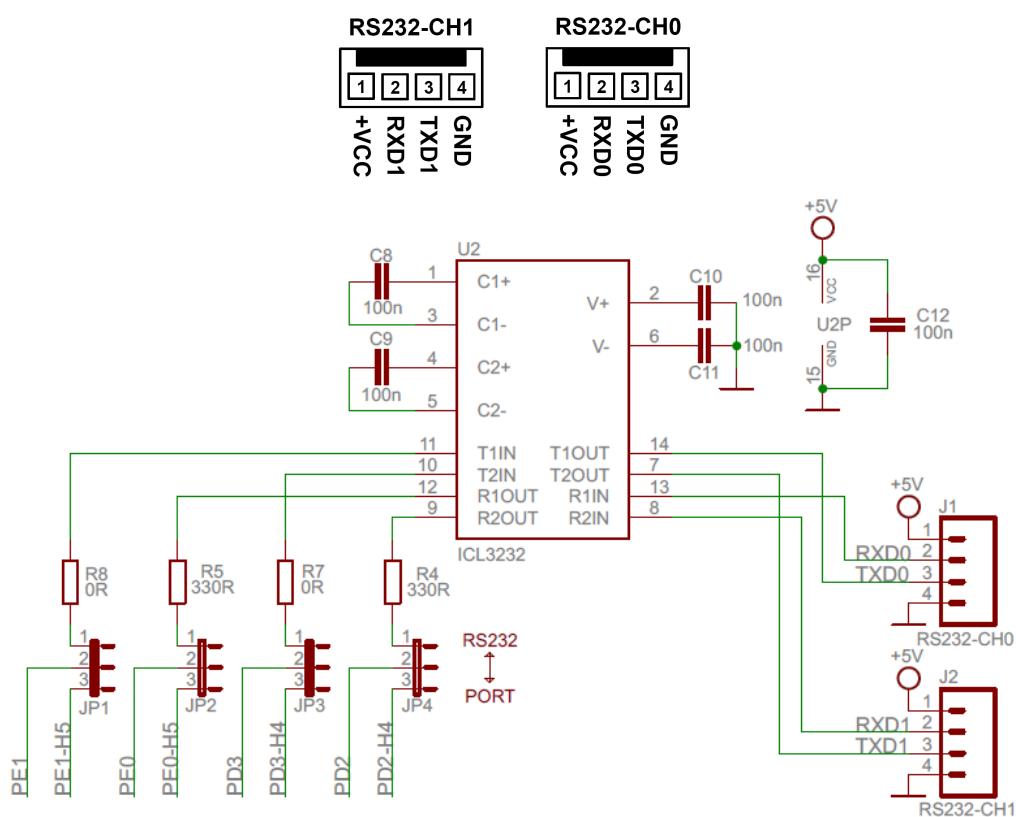




1	2	3	4	5	6	7	8	9	10	11	12	13	14
GND	+VCC	VO	RS	RW	EN	D0	D1	D2	D3	D4	D5	D6	D7

แสดงการจัดเรียงขาสัญญานของ Character LCD มาตรฐาน

พอร์ต RS232 จำนวน 2 ช่อง โดยเชื่อมต่อกับสัญญาณ PE0(RXD0) และ PE1(TXD0) จำนวน 1 ช่อง ส่วนที่เหลืออีก 1 ช่อง จะต่อกับสัญญาณ PD2(RXD1) และ PD3(TXD1)

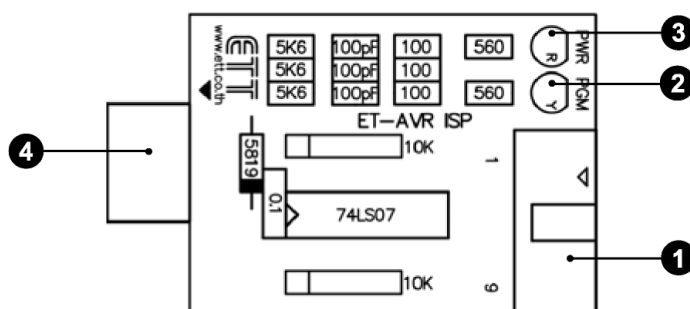


รูปแสดง วงจรส่วนที่เชื่อมต่อกับ RS232

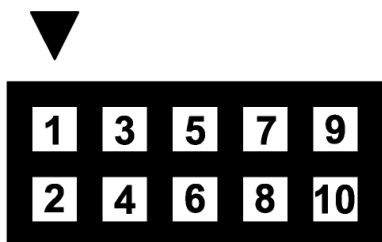
การดาวน์โหลด Hex File ให้กับ MCU

การดาวน์โหลด Hex File ให้กับ MCU นั้นจำเป็นจะต้องใช้ ET-AVR ISP หรือเครื่องโปรแกรมแบบ ISP อื่นๆ เช่น AVRISP ของ ATMEL เพื่อใช้ในการดาวน์โหลด Hex File ให้กับ MCU ตระกูล AVR ของ Atmel โดยใช้วิธีการแบบ Serial Programming ซึ่งการดาวน์โหลด Hex File ในกรณีที่ใช้ ET-AVR ISP จะกระทำผ่านทางพอร์ตขนานของคอมพิวเตอร์ โดยที่จะต้องใช้งานร่วมกับ ET-CAP10P ของอีทีที และ Software ที่ใช้ร่วมกับ ET-AVR ISP ก็คือ PonyProg2000 ซึ่ง PonyProg2000 เป็นโปรแกรม Download ข้อมูลแบบ HEX File ให้กับ CPU ตระกูล AVR โดยใช้วิธีการแบบ Serial Programming ซึ่งสามารถใช้งานกับบอร์ดตระกูล AVR ของ อีทีที ได้เป็นอย่างดี ซึ่งวิธีการใช้งานโปรแกรมโดยทั่วไปนั้น สามารถศึกษาได้จาก Help ของโปรแกรมได้เอง โดยในที่นี้จะขอแนะนำให้ทราบถึงวิธีการ Setup โปรแกรม PonyProg2000 เพื่อใช้งานกับบอร์ดตระกูล AVR ของ อีทีที ซึ่งสามารถใช้งานได้กับบอร์ดตระกูล AVR ทุกรุ่นของ อีทีที

โครงสร้างของบอร์ด ET-AVR ISP



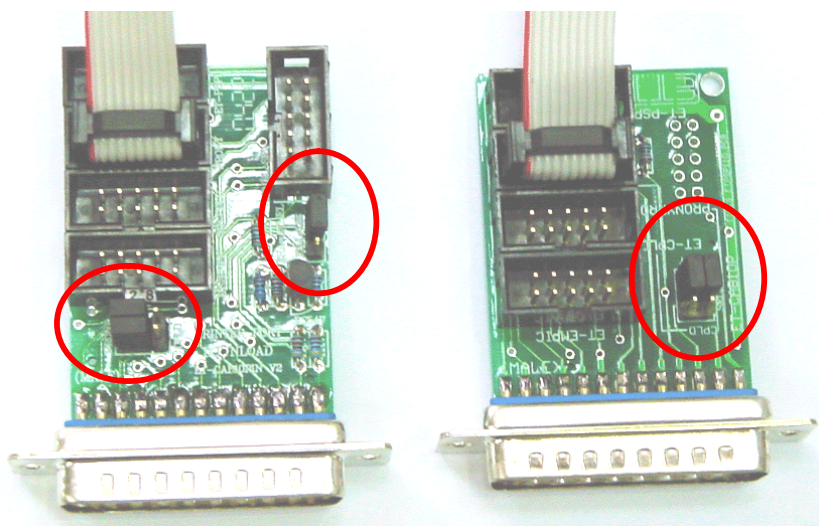
- **หมายเลข 1** คือ พอร์ตสำหรับเชื่อมต่อกับ ET-CAP10P ของอีทีที เพื่อโปรแกรม Hex File ให้กับ MCU
- **หมายเลข 2** คือ LED PGM (สีเขียว) แสดงสถานะของการโปรแกรมหรือดาวน์โหลด Hex File ลง MCU
- **หมายเลข 3** คือ LED PWR (สีแดง) แสดงสถานะของไฟเลี้ยงบอร์ด
- **หมายเลข 4** คือ พอร์ตสำหรับเชื่อมต่อกับบอร์ด Target ซึ่งสามารถใช้โปรแกรม Hex File ให้กับบอร์ด ET-BASE AVR ATmega64/128 r3โดยเสียบบอร์ด ET-AVR ISP เข้าที่ พอร์ต AVR ISP ซึ่งมีการจัดเรียงขาสัญญาณดังรูป



ตำแหน่งขา	ชื่อสัญญาณ
1	MOSI
2	VCC
3	ไม่ได้ใช้งาน
4,6,8,10	GND
5	RESET
7	SCK
9	MISO

การเชื่อมต่ออุปกรณ์สำหรับโปรแกรม Hex File

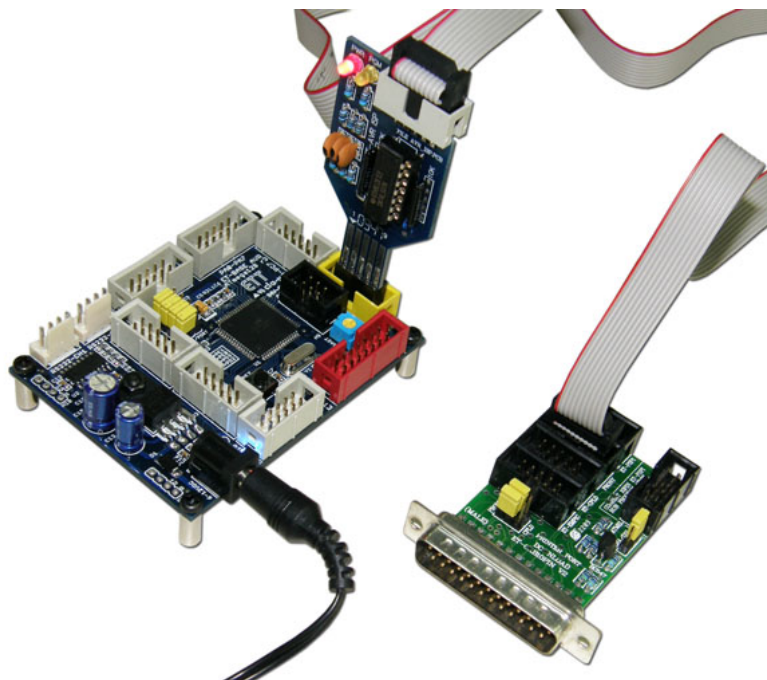
การโปรแกรมไค้ด (Hex File) ให้กับ AVR MCU ต้องใช้งานร่วมกับ ET-CAB10PIN และ โปรแกรม PonyProg2000 โดยต่อ ET-CAP10PIN เข้ากับพอร์ต Printer พร้อมทั้งเลือก Jumper สำหรับใช้งานกับโปรแกรม PonyProg2000 แล้วต่อสาย Download ที่ขั้วต่อ AVR ISP Download ของบอร์ด พร้อมทั้งจ่ายไฟเข้าบอร์ดให้เรียบร้อย ถ้ามีการต่ออุปกรณ์ภายนอกที่พอร์ต PB ให้ปลดออกก่อน โดยการเชื่อมต่อจะมีลักษณะดังรูปต่อไปนี้



(ซ้าย) ET-CAP10P V2.0

(ขวา) ET-CAP10P V1.0

รูปแสดง การเลือก Jumper และการต่อสาย Download ของ ET-CAP10P เพื่อใช้กับ AVR



รูปแสดงการต่อ ET-AVR ISP เข้ากับ ET-BASE AVR ATmega64/128 r3 โดยการต่อบอร์ดทั้งสองเข้าด้วยกันนั้นจะให้สังเกตที่ตำแหน่งขา 1 จะต้องตรงกัน

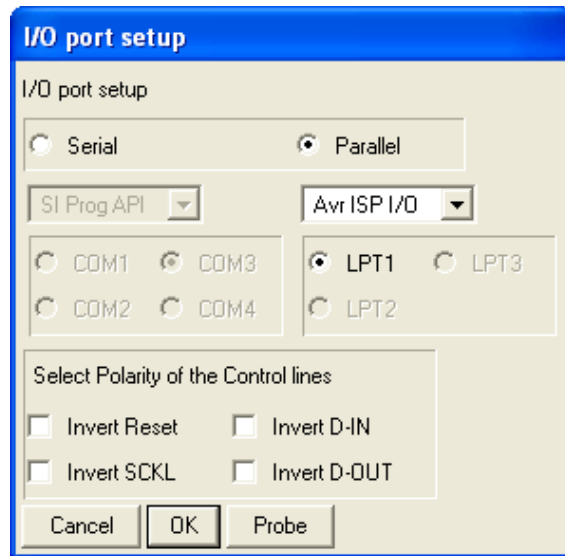
การ Program ให้ Board ET-BASE AVR ATmega64/128 r3 ด้วยโปรแกรม PonyProg2000

โปรแกรม PonyProg2000 เป็นโปรแกรม Download ข้อมูลแบบ HEX File ให้กับ CPU ตระกูล AVR โดยใช้วิธีการแบบ Serial Programming ซึ่งสามารถใช้งานกับบอร์ดตระกูล AVR ของ อีทีที ได้เป็นอย่างดี ซึ่งวิธีการใช้งานโปรแกรมโดยทั่วไปนั้น สามารถศึกษาได้จาก Help ของโปรแกรมได้เอง โดยในที่นี้จะขอแนะนำให้ทราบถึงวิธีการ Setup โปรแกรม PonyProg2000 เพื่อใช้งานกับบอร์ดตระกูล AVR ของ อีทีที ซึ่งสามารถใช้งานได้กับบอร์ดตระกูล AVR ทุกรุ่นของ อีทีที

สำหรับกรณีที่ใช้ CPU ตระกูล AVR เบอร์ ATmega64/128 นั้น จะมีข้อควรระวังอยู่อย่างหนึ่ง เนื่องจากโครงสร้างภายในของ ATmega64/128 นั้นจะมี Fuse Bit สำหรับกำหนดเงื่อนไขการทำงานของ CPU รวมอยู่ด้วยหลายบิต ซึ่ง Fuse Bit ต่างๆเหล่านี้ บางบิตจะมีผลต่อการ Download แบบ Serial Programming ด้วย เนื่องจากถ้าเลือกกำหนดคุณสมบัติของ Fuse Bit ไม่ถูกต้องอาจทำให้ไม่สามารถสั่งโปรแกรม CPU ตัวนั้นด้วยวิธีการ Serial Programming ได้อีก นอกจากจะนำ CPU ตัวนั้นไปแก้ไข Fuse Bit ด้วยเครื่องโปรแกรมแบบ Parallel ให้ได้ค่าที่ถูกต้องเสียก่อน

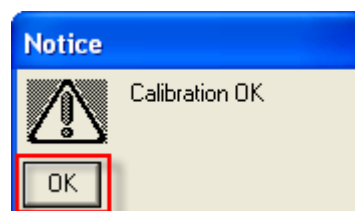
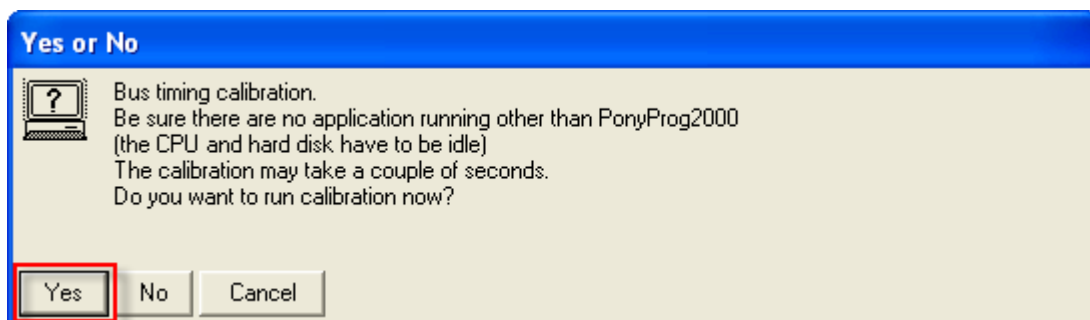
โดยในการสั่งโปรแกรม CPU ตระกูล AVR ที่ใช้งานกับบอร์ดของ อีทีที นั้น ถ้าใช้การโปรแกรมด้วยโปรแกรมของ “PonyProg2000” จะต้องกำหนด Option ของโปรแกรมเพื่อให้สามารถใช้งานกับบอร์ดของ อีทีที ดังนี้

1. กำหนด Setup → Interface Setup... เป็นดังนี้



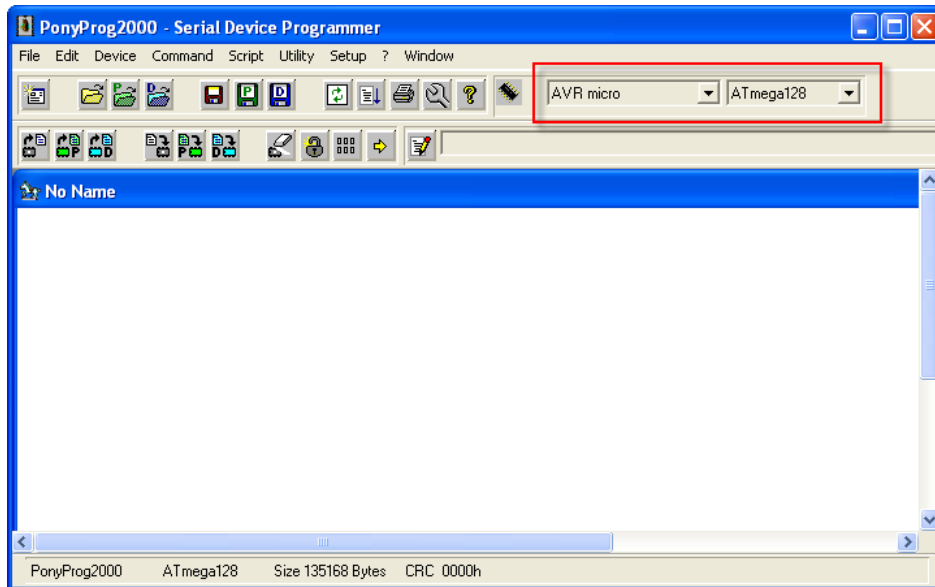
- ให้เลือก I/O Port เป็น Parallel และเลือกรูปแบบการโปรแกรมเป็น Avr ISP I/O
- ให้เลือก Printer Port ตามที่ต่อจริง เช่น LPT1 ในกรณีที่ใช้กับ Printer Port LPT1
- ส่วนของ Polarity Control Line ไม่ต้องเลือก
- การ Setup นี้ทำเพียงครั้งเดียวตอนเริ่มใช้งานโปรแกรมในครั้งแรกเท่านั้น

2. สั่งให้โปรแกรม PonyProg2000 ทำการคำนวณหาค่าความเร็วที่เหมาะสมสำหรับการส่งสัญญาณไปโปรแกรม CPU โดยเลือกจาก Setup → Calibration

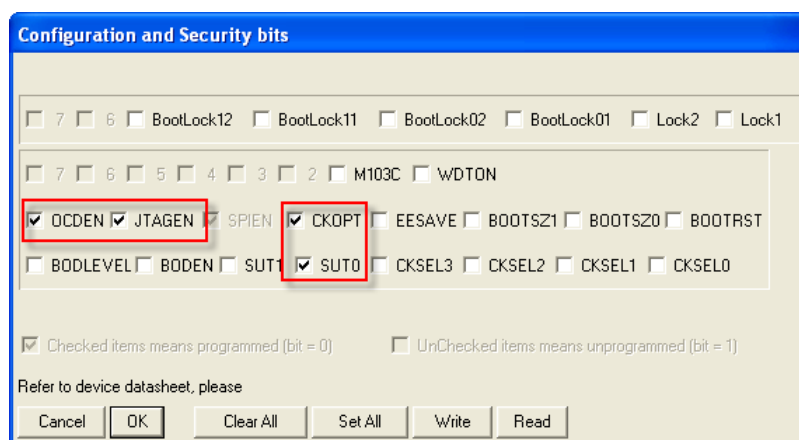


- การสั่ง Calibration จะกระทำเพียงครั้งเดียวในตอนเรียกใช้งานโปรแกรมครั้งแรกเท่านั้น

3. เลือกกำหนดเบอร์ CPU จาก Device → AVR Micro → Atmega64 หรือ ATmega128



4. เลือกกำหนด Command → Security and Configuration Bits โดยถ้าเป็น AVR เบอร์อื่นๆ สามารถกำหนดการทำงานของ Fuse Bit ได้ตามต้องการ โดยสามารถศึกษารายละเอียดของ Fuse Bit ต่างๆได้จาก Data Sheet ของ CPU ที่ใช้ได้เอง แต่ในกรณีที่ใช้งานกับ Atmega64/128 นั้นต้องระมัดระวังในการเลือกกำหนด Fuse Bit ให้ถูกต้องด้วย ซึ่งถ้ากำหนดผิดพลาดส่งผลให้ไม่สามารถสั่งโปรแกรม CPU ด้วยวิธีการ Serial Programming ได้อีก เมื่อเลือกดังรูปแล้วกดปุ่ม Write (ขั้นตอนนี้ทำเพียงครั้งเดียวเท่านั้น ครั้งต่อไปก็สามารถข้ามไปได้เลย ในกรณีที่ใช้บอร์ด ET-BASE AVR ATmega64/128 r3 ไม่จำเป็นต้องทำขั้นตอนนี้เพราะทางบริษัทได้ตั้งค่าไว้เรียบร้อยแล้ว ยกเว้นผู้ใช้งานต้องการเปลี่ยนค่า)



รูปแสดง การเลือกกำหนด Fuse Bit เพื่อใช้กับ CPU เบอร์ ATmega64/128

ความหมายของ Fuse Bit ต่างๆ ของ ATmega64/128

- ในกรณีที่เลือก [✓] ที่หน้า Fuse Bit ตัวใด หมายถึงการกำหนดให้ Fuse Bit นั้นๆ มีค่าเป็น “0” หรือการสั่งโปรแกรม Fuse Bit นั้นๆ
- ในกรณีที่ไม่เลือก [✓] ที่หน้า Fuse Bit ตัวใด หมายถึงการกำหนดให้ Fuse Bit นั้นๆ มีค่าเป็น “1” หรือสั่งไม่โปรแกรม Fuse Bit นั้นๆ

ความหมายของ Fuse Bit ของ ATmega64/128 ที่มีผลต่อ Serial Programming

- SPIEN เป็น Serial Programming Enable Bit ซึ่งจะต้องสั่งโปรแกรม Fuse Bit นี้ไว้เสมอ เพื่อให้สามารถสั่ง Download โปรแกรมให้กับ CPU ด้วยวิธีการ In-System Serial Programming ได้ ซึ่งตามปกติแล้ว Fuse Bit นี้จะถูกสั่งโปรแกรมมาจากโรงงานอยู่แล้ว และไม่สามารถสั่งลบหรือแก้ไข Fuse Bit นี้ได้ด้วยโหมด Serial Programming แต่ถ้ามีการนำ CPU ไปโปรแกรมด้วยเครื่องแบบ Parallel Programming จะต้องไม่ลืมสั่งโปรแกรม Fuse Bit นี้ไว้ด้วยเสมอทุกครั้ง
- OCDEN และ JTAGEN ทั้งสองบิตนี้จะใช้ในกรณีที่ต้องการ Debug การทำงานของ MCU และโปรแกรมผ่านทาง JTAG Interface ซึ่งต้องร่วมกับ AVR JTAG Debugger ซึ่งถ้าไม่ได้ใช้งานก็ไม่จำเป็นต้องเลือกทั้งสองบิตนี้
- CKOPT เป็น Oscillator Option Bit ถ้าสั่งโปรแกรม Fuse Bit นี้จะเป็นการกำหนดให้ CPU ทำงานที่ย่านความถี่ 16MHz แต่ถ้าไม่ได้สั่งโปรแกรม Fuse Bit นี้จะเป็นการกำหนดให้ CPU ทำงานที่ย่านความถี่ไม่เกิน 8MHz ซึ่งถ้าใช้กับบอร์ดมาตรฐานของอีทีที จะใช้ XTAL เป็นแหล่งกำเนิดความถี่ ดังนั้นควรสั่งโปรแกรมค่า Fuse Bit นี้ไว้ เพื่อให้ CPU สามารถทำงานได้ที่ย่านความถี่ของ XTAL ตั้งแต่ 1.0MHz-16.0MHz
- CKSEL3...0 เป็น Select Clock Source Bit ใช้ร่วมกันสำหรับเลือกแหล่งกำเนิดและย่านของความถี่ที่จะให้กับ CPU ซึ่งในกรณีใช้งานกับบอร์ดมาตรฐานของอีทีที ต้องเลือกเป็น External Crystal ค่า 1.0 MHz - 16.0 MHz ซึ่งถ้าเลือกเป็นอย่างอื่นจะทำให้การทำงานของโปรแกรมผิดพลาด และที่สำคัญถ้าเลือกแหล่งกำเนิดความถี่ผิด เช่น เลือกเป็น External Clock หรือ External RC Oscillator จะทำให้ CPU ไม่สามารถทำงานได้ เนื่องจากไม่มีการต่อสัญญาณพิกจากภายนอกไว้ให้ และจะทำให้ไม่สามารถสั่งโปรแกรม CPU ตัวนั้นด้วยวิธีการแบบ Serial Programming ได้อีก จนกว่าจะมี

การนำ CPU ไปแก้ไขค่า Fuse Bit เพื่อเลือกแหล่งกำเนิดสัญญาณนาฬิกาเป็น External Crystal ให้ถูกต้องเสียก่อน

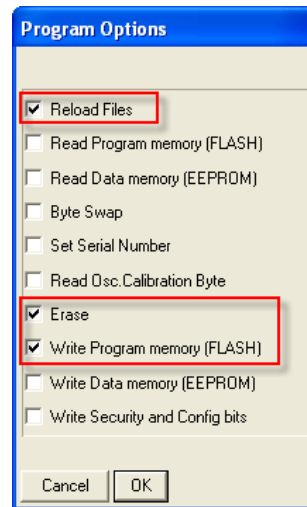
แหล่งกำเนิดสัญญาณนาฬิกาของ AVR Atmega128	การกำหนด Fuse Bit ของ CKSEL[3...0] (0=Program,1=Un-Program)
External Crystal/Ceramic Resonator	1111-1010
External Low Frequency Crystal	1001
External RC Oscillator	1000-0101
Calibrated Internal RC Oscillator	0100-0001
External Clock	0000

ตารางแสดง การเลือกแหล่งกำเนิดความถี่จาก Fuse Bit CKSEL [3...0]

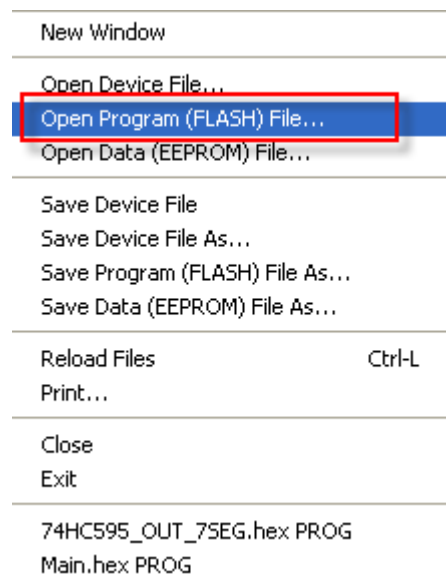
หมายเหตุ

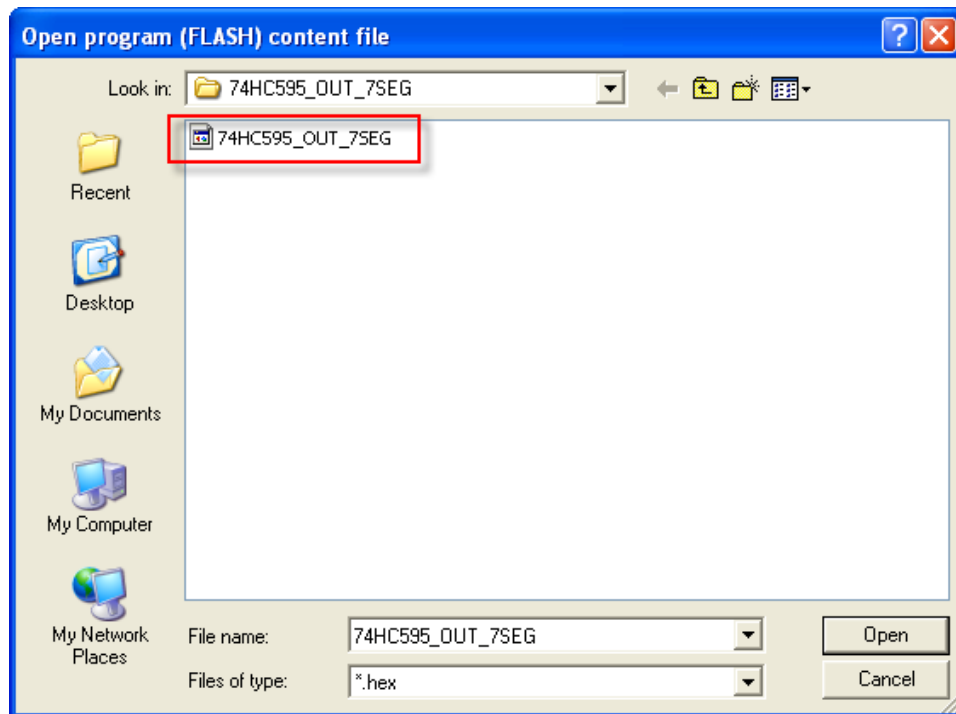
- ค่า 1 หมายถึง การสั่งไม่โปรแกรม Fuse Bit นั้นๆ โดยไม่ต้องใส่เครื่องหมาย [✓] หน้า Fuse Bit
- ค่า 0 หมายถึง การสั่งโปรแกรม Fuse Bit นั้นๆ โดยการเลือกเครื่องหมาย [✓] หน้าชื่อ Fuse Bit
- ควรสั่งโปรแกรม Fuse Bit ของ CKOPT เพื่อให้ใช้งานที่ย่านความถี่ 1.0MHz-16.00MHz
- ห้ามสั่งโปรแกรม Fuse Bit ของ CKSEL[3..0] เพราะจะทำให้การทำงานไม่ถูกต้อง ตัวอย่างเช่น ถ้าเลือกสั่งโปรแกรม Fuse Bit ของ CKSEL[3..0] ให้มีค่าเป็น 0 ทั้งหมด ซึ่งหลังจากโปรแกรม PonyProg2000 ทำการเขียนค่า Fuse Bit นี้ให้กับ CPU เรียบร้อยแล้ว จะทำให้ CPU ไม่สามารถใช้งานได้กับบอร์ดของ อีทีที อีก และจะไม่สามารถสั่งโปรแกรมแก้ไขค่า Fuse Bit ใหม่ให้กับ CPU ด้วยวิธีการแบบ Serial Programming ได้อีก เนื่องจาก CPU ไม่สามารถทำงานได้อีก เพราะว่าการสั่งโปรแกรม Fuse Bit ของ CKSEL[3..0] ให้เป็น 0 ทั้งหมด จะเป็นการสั่งให้ CPU ทำงานด้วยความถี่ของสัญญาณนาฬิกาจากภายนอก (External Clock) ซึ่งจะทำให้วงจรกำเนิดความถี่ของ External Crystal หยุดทำงาน โดย CPU จะรอรับสัญญาณนาฬิกาจากภายนอกเพียงอย่างเดียวเท่านั้น แต่บอร์ดของ อีทีที จะใช้สัญญาณนาฬิกาจาก วงจร Crystal (External Crystal) เท่านั้น ดังนั้นเมื่อ CPU ไม่สามารถเริ่มต้นทำงานได้ ก็จะทำให้เราไม่สามารถสั่งโปรแกรมแก้ไขค่า Fuse Bit ที่ถูกต้องให้กับ CPU ด้วยวิธีการแบบ Serial Programming ได้อีก ซึ่งจะต้องนำ CPU ตัวนั้น ไปทำการแก้ไขค่า Fuse Bit ด้วยเครื่องโปรแกรมแบบ Parallel เสียก่อนจึงจะสามารถนำมาใช้งานกับวิธีการโปรแกรมแบบ Serial Programming ได้เหมือนเดิม
- ตำแหน่ง Fuse Bit ของ Lock[2..1] สามารถกำหนดได้ตามต้องการ

5. เลือกกำหนด Command → Program Option เป็นดังนี้

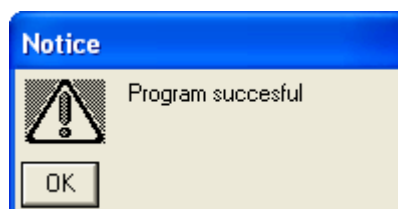
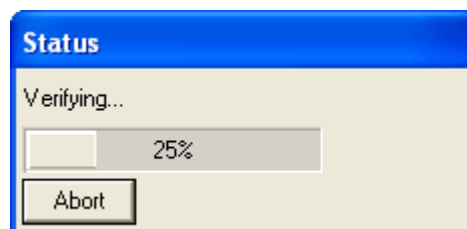


6. สั่งเปิดไฟล์สำหรับที่จะใช้โปรแกรมให้กับ CPU โดยเลือกจาก File → Open Program (FLASH) File... → พร้อมทั้งระบุชื่อและที่อยู่ของ HEX File ที่จะใช้โปรแกรมให้เรียบร้อย





7. สั่งเริ่มต้นโปรแกรมข้อมูลให้กับ CPU โดยเลือก Command → Program จากนั้นโปรแกรมจะเริ่มทำงานตามคำสั่งที่เราเลือกกำหนดไว้ในข้อ 5 คือ Load File → Erase → Write Program memory (FLASH) ตามลำดับ ซึ่งให้รอจนการทำงานของโปรแกรมเสร็จสมบูรณ์



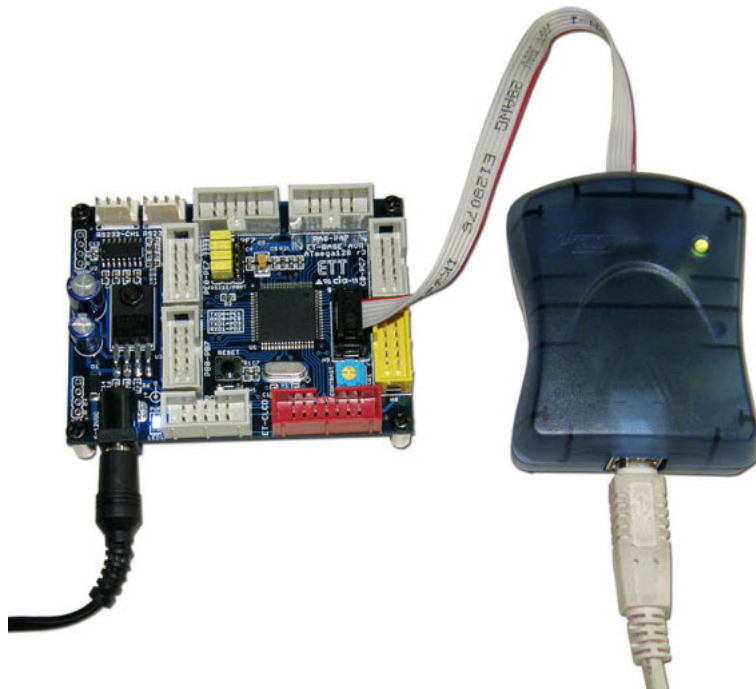
ซึ่งหลังจากการโปรแกรมเสร็จเรียบร้อยแล้ว CPU จะเริ่มต้นทำงานตามข้อมูลในโปรแกรมที่สั่ง Download ให้ทันที

การตรวจสอบเบื้องต้นเมื่อไม่สามารถดาวน์โหลดโปรแกรมได้

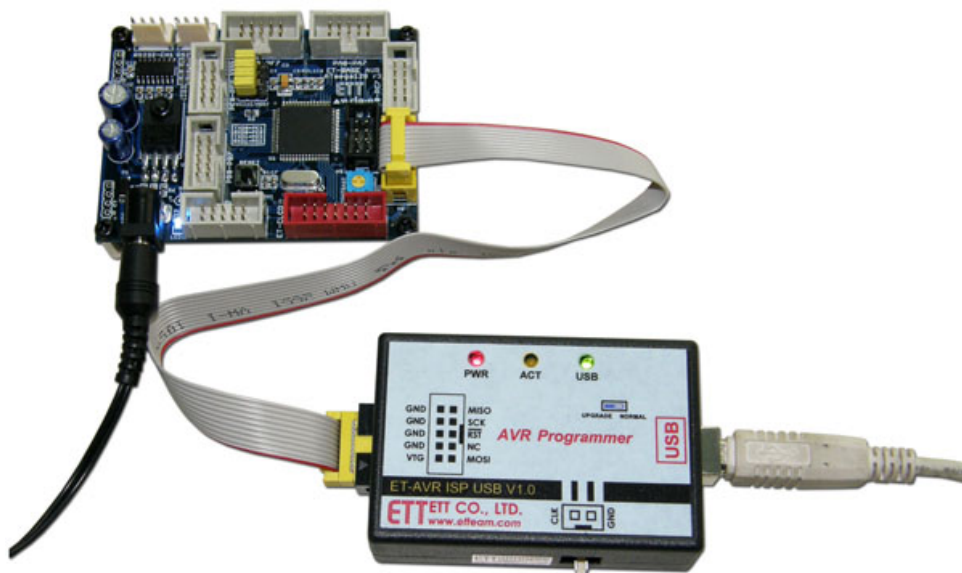
ถ้าเกิดการ Error ในขั้นตอนของการโปรแกรมให้ตรวจสอบปัญหาดังนี้

- อ่านคู่มือการใช้งานบอร์ด และคู่มือวิธีการ Download โปรแกรม AVR ด้วย PonyProg2000 ให้ละเอียด
- ตรวจสอบการเชื่อมต่อของสายสัญญาณต่างๆ และในการ Download โปรแกรมโดยใช้ PonyProg2000 นั้น จะต้องใช้งานร่วมกับชุด Cable Download รุ่น ET-CAP10PIN ของ ETT ด้วย ซึ่งต้องมีการกำหนด JUMPER ให้เป็น PonyProg ให้ถูกต้องด้วย (รายละเอียดหน้า 9)
- ตรวจสอบการจ่ายไฟเลี้ยงให้กับบอร์ด
- ตรวจสอบการตั้งค่าต่างๆของโปรแกรมสำหรับดาวน์โหลด PonyProg2000
- ตรวจสอบว่ามีการนำสัญญาณจากพอร์ต PORT-PB ของ CPU ไปต่อไว้กับอุปกรณ์ภายนอก ในขณะที่สั่ง Download หรือไม่ ตัวอย่างเช่น ต่อกับ LED หรือ นำสัญญาณจากพอร์ต PB ไปต่อไว้กับวงจรอื่นๆในขณะที่สั่ง Download ข้อมูลอยู่

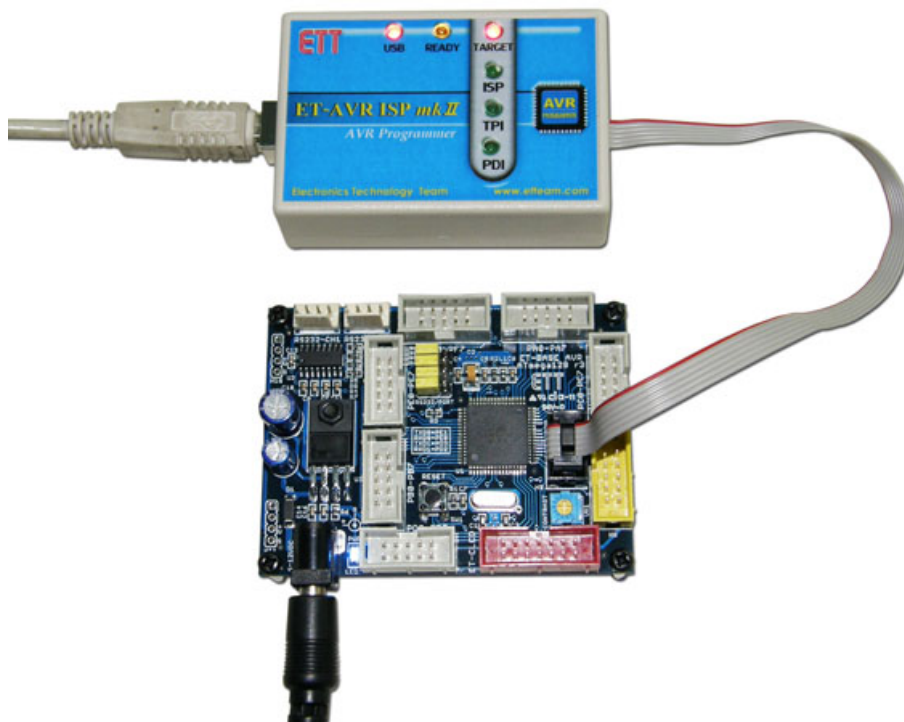
การใช้ร่วมกับเครื่องโปรแกรมอื่นๆ



ตัวอย่างการใช้งานร่วมกับ AVRISP mkII ของบริษัท ATMEL



การใช้งานร่วมกับ ET-AVR ISP USB V1.0 ของบริษัท ETT



การใช้งานร่วมกับ ET-AVR ISP mkII ของบริษัท ETT