

คู่มือการใช้งาน User's manual

ET-dsPIC33WEB TRAINING VI BOX

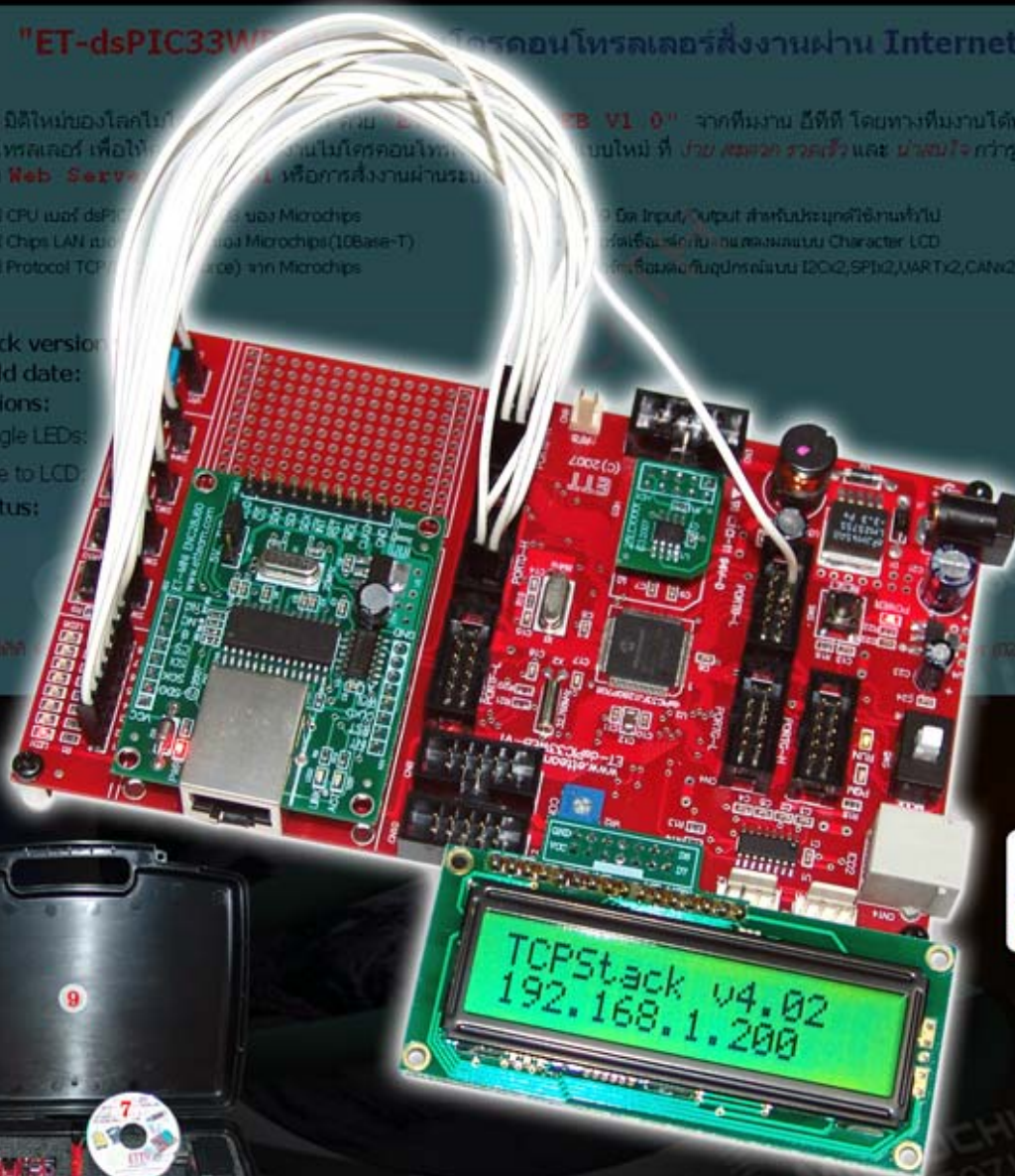


"ET-dsPIC33WEB" ไมโครคอนโทรลเลอร์สั่งงานผ่าน Internet

ขอต้อนรับสู่ มิติใหม่ของโลกไมโครคอนโทรลเลอร์ ด้วย "ET-dsPIC33WEB V1.0" จากทีมงาน อีทีที โดยทางทีมงานได้พัฒนาบอร์ด ไมโครคอนโทรลเลอร์ เพื่อให้งานไมโครคอนโทรลเลอร์ สามารถสั่งงานผ่านระบบใหม่ ที่ **ง่าย สะดวก รวดเร็ว** และ **น่าสนใจ** กว่ารูปแบบเดิมๆ ใน ลักษณะของ **Web Server** สำหรับการสั่งงานผ่านระบบ Internet

- 16 CPU เอน์ dsPIC33FJ128GP708 ของ Microchips
- 16 Chips LAN module ของ Microchips (10Base-T)
- 16 Protocol TCP/IP (Software) จาก Microchips

Stack version:
Build date:
Actions:
Toggle LEDs:
write to LCD:
Status:



dsPIC33FJ128GP708
128 KBYTE
16 KBYTE RAM



บริษัท อีทีที จำกัด ETT CO., LTD.

1112/96-98 ถนนสุขุมวิท แขวงพระโขนง เขตคลองเตย กรุงเทพฯ 10110 <http://www.etteam.com>

1112/96-98 Sukhumvit Rd., Phraknong Klongtoey Bangkok 10110 <http://www.ett.co.th>

Tel : 02-7121120 Fax : 02-3917216

email : sale@etteam.com

ET-dsPIC33WEB V1.0

ET-dsPIC33WEB V1.0 เป็นบอร์ดไมโครคอนโทรลเลอร์ในตระกูล dsPIC ของบริษัท Microchip โดยได้นำเอาไมโครคอนโทรลเลอร์ที่ประมวลผลข้อมูลแบบ 16 บิต เบอร์ dsPIC33FJ128GP708 มาพัฒนาเป็นบอร์ดใช้งาน ซึ่งคุณสมบัติเด่นของ dsPIC33FJ128GP708 คือ หน่วยประมวลผลสัญญาณดิจิทัล (Digital Signal Processing) และ ทรัพยากรต่างๆ ดังต่อไปนี้

หน่วยประมวลผล (CPU)

- ความเร็วในการประมวลผล 40 MIPS (16 Bit Data / 24 Bit Instruction Code)
- ฮาร์ดแวร์รองรับการคูณข้อมูล 16 x 16 บิต โดยใช้เวลาเพียง 1 ไชเคิลคำสั่ง
- ฮาร์ดแวร์รองรับการหารข้อมูล 32-bit x 16 บิต
- C Compiler ถูกออกแบบให้มีความกะทัดรัด Optimized Instruction Set
- รองรับการ Interrupt มากถึง 118 Vector Interrupt จาก 63 แหล่ง 7 Priority Level Program
- รองรับการ DMA กับ Peripheral Hardware ได้ 8 ช่อง พร้อม DMA Buffer 2KByte

ระบบ (System)

- แหล่งกำเนิดสัญญาณนาฬิกาสามารถเลือกได้ ทั้งจากภายในและภายนอก
- มีวงจร Power-Up Timer และ Oscillator Start-Up
- มีระบบตรวจสอบสัญญาณนาฬิกา (Fail-Safe Clock Monitor)
- ระบบ Watchdog Timer ที่ใช้แหล่งสัญญาณนาฬิกาแบบ RC oscillator ที่แยกจากส่วนอื่นๆ
- ทำงานที่แรงดันระดับ 3.0 ถึง 3.6 โวลต์
- I/O Pin 4mA Sink สามารถเชื่อมต่อกับสัญญาณ 5VTTL ได้ (5V Tolerant)
- รองรับโหมดการทำงานแบบ Run, Idle และ Sleep modes
- สามารถปรับเปลี่ยนโหมดการทำงานของสัญญาณนาฬิกาได้หลากหลายเพื่อประสิทธิภาพ และ ให้สอดคล้องกับการดูแลจัดการในเรื่องของพลังงาน

คุณสมบัติทางด้านสัญญาณอนาล็อก (Analog Features)

- โมดูลแปลงสัญญาณ Analog to Digital ความละเอียด 10-bit จำนวน 24 ช่อง และสามารถโปรแกรมเป็น 12Bit ได้ 2 ช่อง ความเร็วในการ Sampling สัญญาณสูงสุด 1.1 MSPS

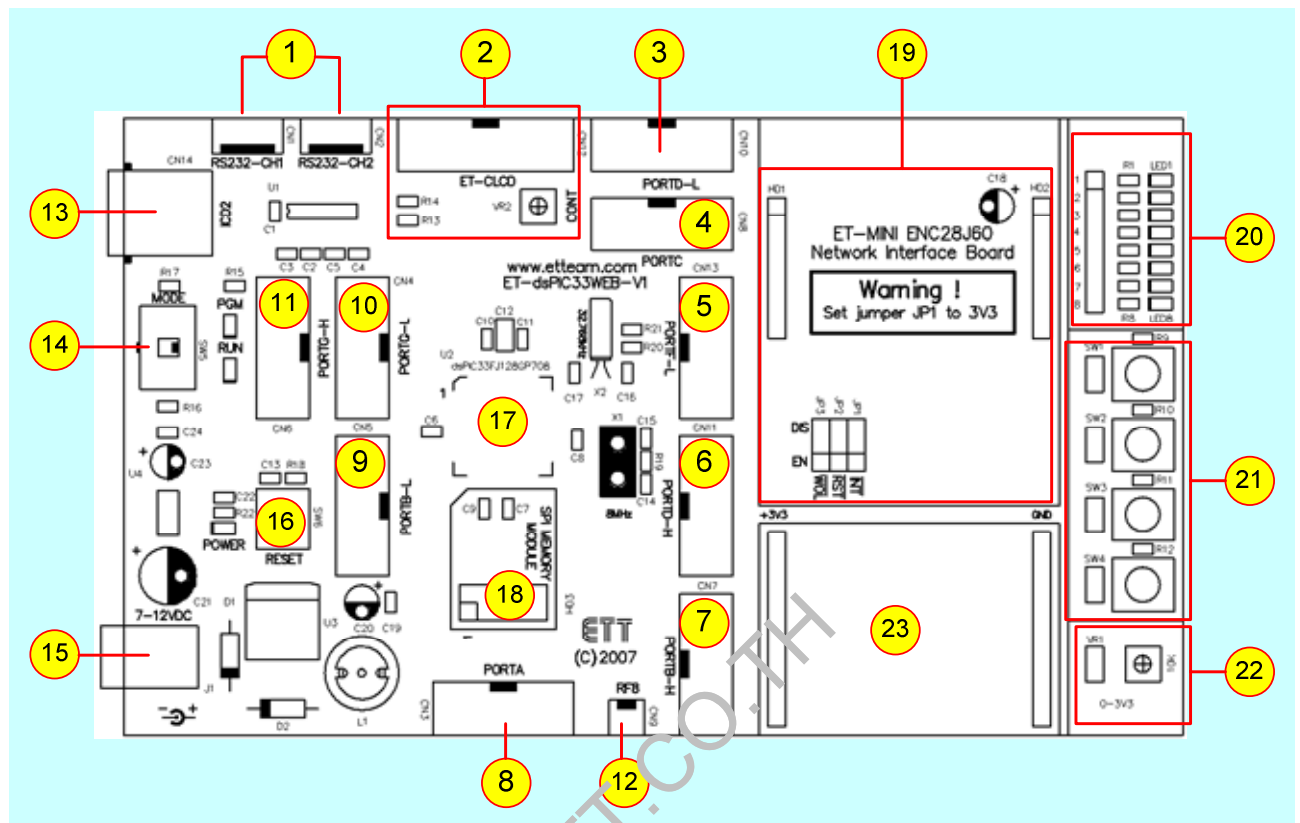
คุณสมบัติโดยทั่วไปของ MCU เบอร์ dsPIC33FJ128GP708

- หน่วยความจำโปรแกรมแบบ Flash Memory ขนาด 128 K Byte
- หน่วยความจำข้อมูล SRAM ขนาด 16 K Byte
- I/O Ports ใช้งานจำนวน 69 บิต(รวม Peripheral Function ต่างๆ)
 - โมดูลการสื่อสาร UART จำนวน 2 ช่อง
 - โมดูลการสื่อสารแบบ SPI จำนวน 2 ช่องรองรับทั้ง Master และ Slave Modes
 - โมดูลการสื่อสารแบบ I2C จำนวน 2 ช่องรองรับทั้ง Master และ Slave Modes
 - โมดูลการสื่อสารแบบ CAN จำนวน 2 ช่อง
 - โมดูล Timer ขนาด 16-bit จำนวน 9 ช่อง และสามารถจับคู่ใช้งานเป็น Timer ขนาด 32 Bit ได้ พร้อมกันจำนวน 4 ช่อง
 - โมดูล Capture , Compare / PWM จำนวน 8 ชุด
 - ระบบฮาร์ดแวร์ RTCC, Real-Time Clock Calendar with Alarms ภายใน
 - โมดูล ADC ขนาด 10Bit จำนวน 24 ช่อง และสามารถโปรแกรมค่าเป็น 12Bit ได้ 2 ช่อง
 - ระบบการสื่อสารแบบขนาน DCI(Data Converter Interface) จำนวน 1 ช่อง

คุณสมบัติโดยทั่วไปของบอร์ด ET-dsPIC33WEB V1.0

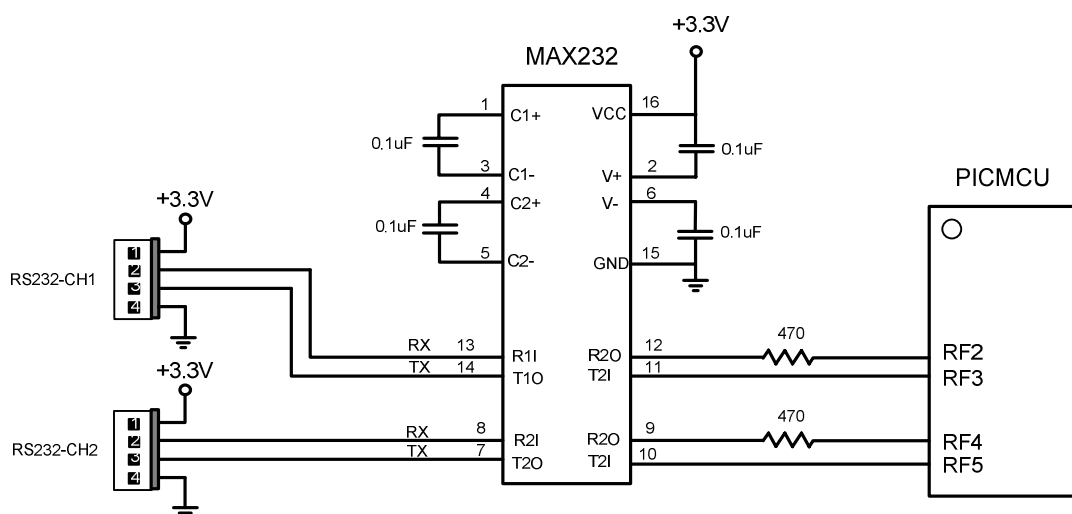
- ใช้ไมโครคอนโทรลเลอร์ dsPIC33FJ128GP708 ขนาด 80 PIN
- สัญญาณนาฬิกาคริสตอลออสซิลเลเตอร์ขนาด 8 MHz (สามารถใช้ PLL รันได้ถึง 40 MHz)
- สัญญาณนาฬิกาคริสตอลออสซิลเลเตอร์ขนาด 32.768KHz สำหรับ RTC
- I/O Port ขนาด 10 PIN (จัดเรียงตามมาตรฐานของ อีทีที) จำนวน 9 พอร์ต
- ชุดวงจร Line Driver RS232 จำนวน 2 พอร์ต
- พอร์ตสำหรับต่อ LCD เรียงตามมาตรฐานของ อีทีที (14Pin ET-CLCD) จำนวน 1 พอร์ต
- ขั้วต่อสัญญาณดาวน์โหลดโปรแกรมแบบ ICD2 และ สวิตช์ตัดต่อสัญญาณ Run / Program
- วงจร LED สำหรับใช้ทดลองเอาต์พุตแบบ Digital จำนวน 8 ช่อง
- วงจรสวิตช์ Push-Button สำหรับใช้ทดลองอินพุตแบบ Digital จำนวน 4 ช่อง
- วงจรสร้างแรงดัน 0-3.3V จากตัวต้านทานปรับค่าได้ สำหรับทดลองโมดูล A/D จำนวน 1 ช่อง
- พอร์ตสำหรับเชื่อมต่อกับโมดูล Ethernet รุ่น ET-MINI ENC28J60(ใช้ SPI1)
- พอร์ตเชื่อมต่อกับหน่วยความจำ EEPROM 25LCxxx จำนวน 1 ช่อง(ใช้ SPI2)
- ชุด Regulate แบบ Switching สำหรับแปลงไฟ DC Input ให้เป็น 5V และ 3.3 V
- LED สถานะสำหรับ Power(แดง),Program(แดง) และ Run(เขียว)
- ขั้วต่อแรงดันไฟ VCC และ GND ใช้ได้กับไฟ 7-12 VDC

โครงสร้างบอร์ด ET-dsPIC33WEB V1.0

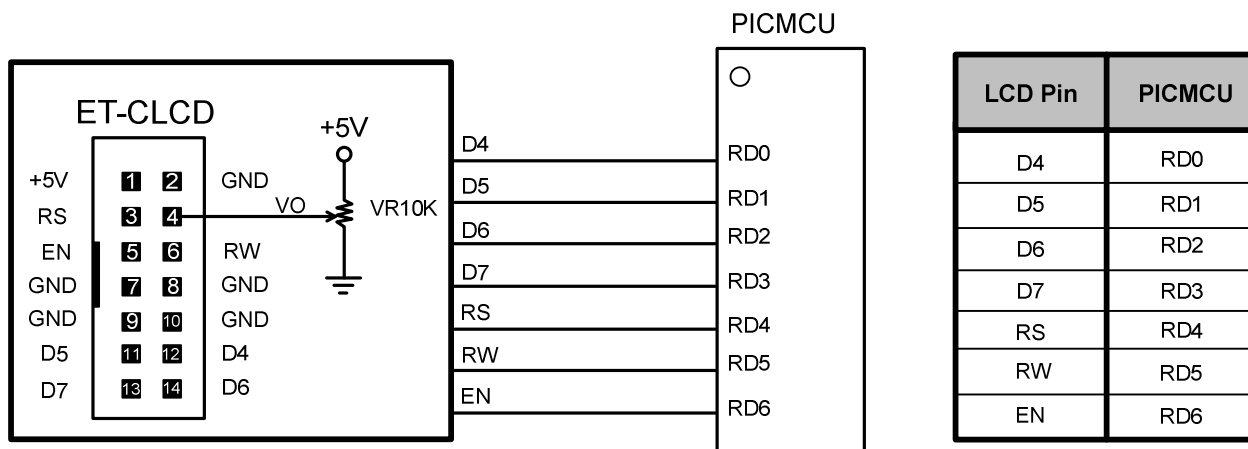


อธิบายรายละเอียดตามหมายเลขได้ดังต่อไปนี้

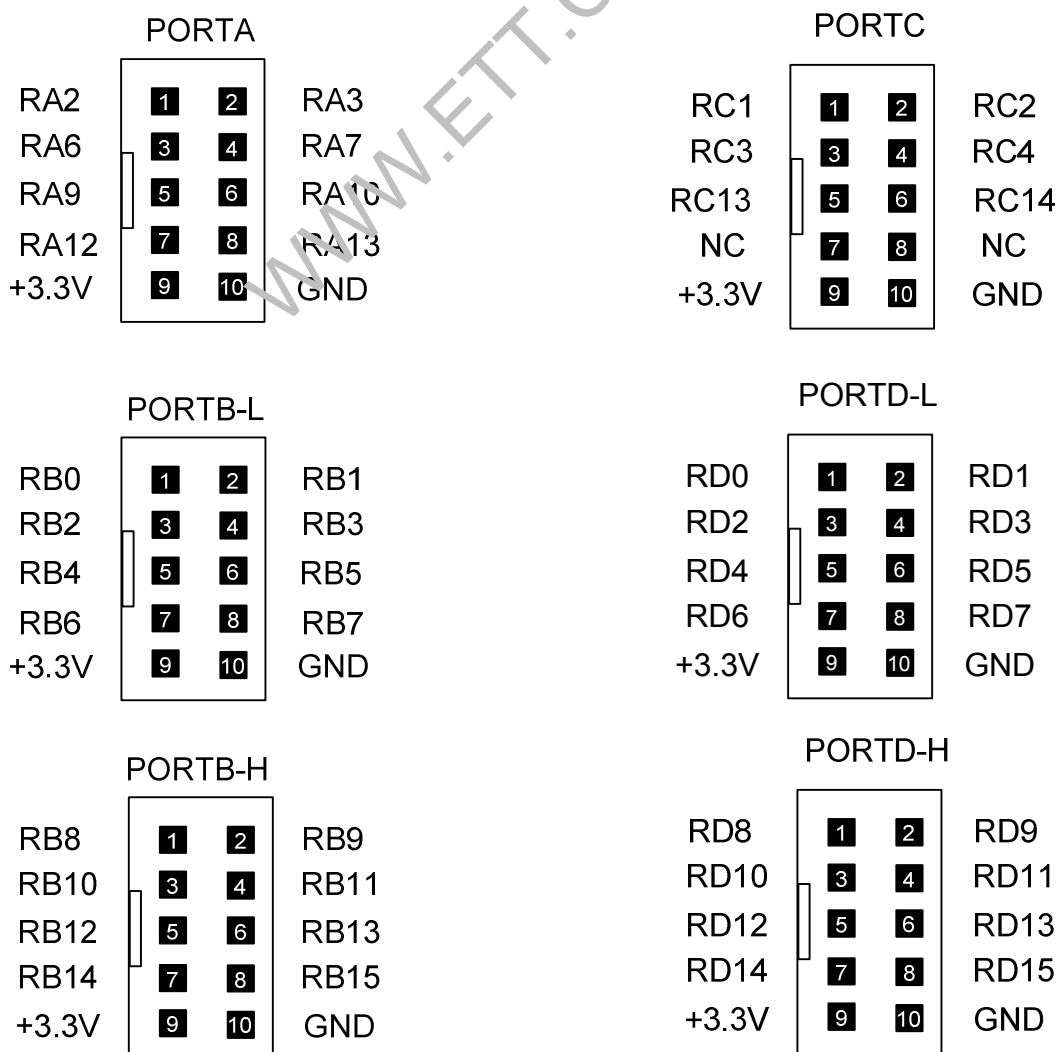
- หมายเลข 1 พอร์ตเชื่อมต่อสัญญาณแบบ RS-232 จำนวน 2 พอร์ต มีวงจรการเชื่อมต่อดังต่อไปนี้

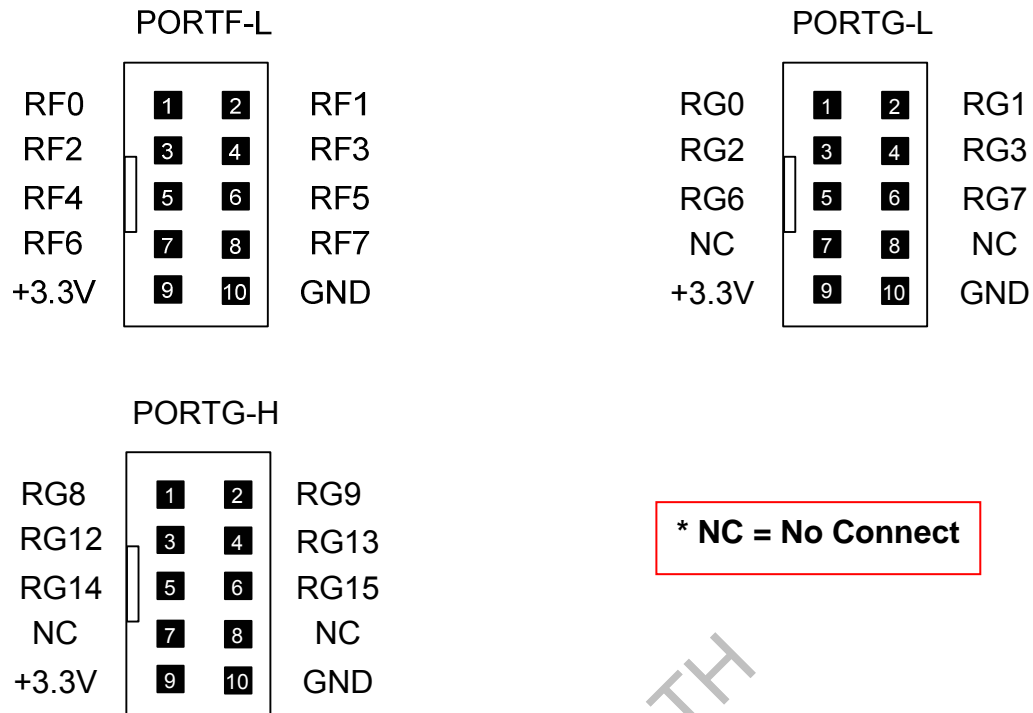


- **หมายเลข 2** พอร์ต ET-CLCD สำหรับเชื่อมต่อกับจอแสดงผล LCD แบบตัวอักษร (Character LCD) โดยมีการจัดวางขาสัญญาณต่างๆ ดังต่อไปนี้

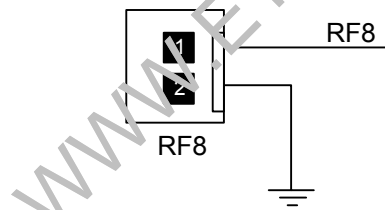


- **หมายเลข 3, 4, 5, 6, 7, 8, 9, 10 และ 11** คือ พอร์ต VO ของไมโครคอนโทรลเลอร์ ที่ถูกออกแบบให้อยู่ในรูปแบบของพอร์ตมาตรฐาน 10-PIN ETT โดยในแต่ละพอร์ตมีการจัดเรียงสัญญาณดังต่อไปนี้

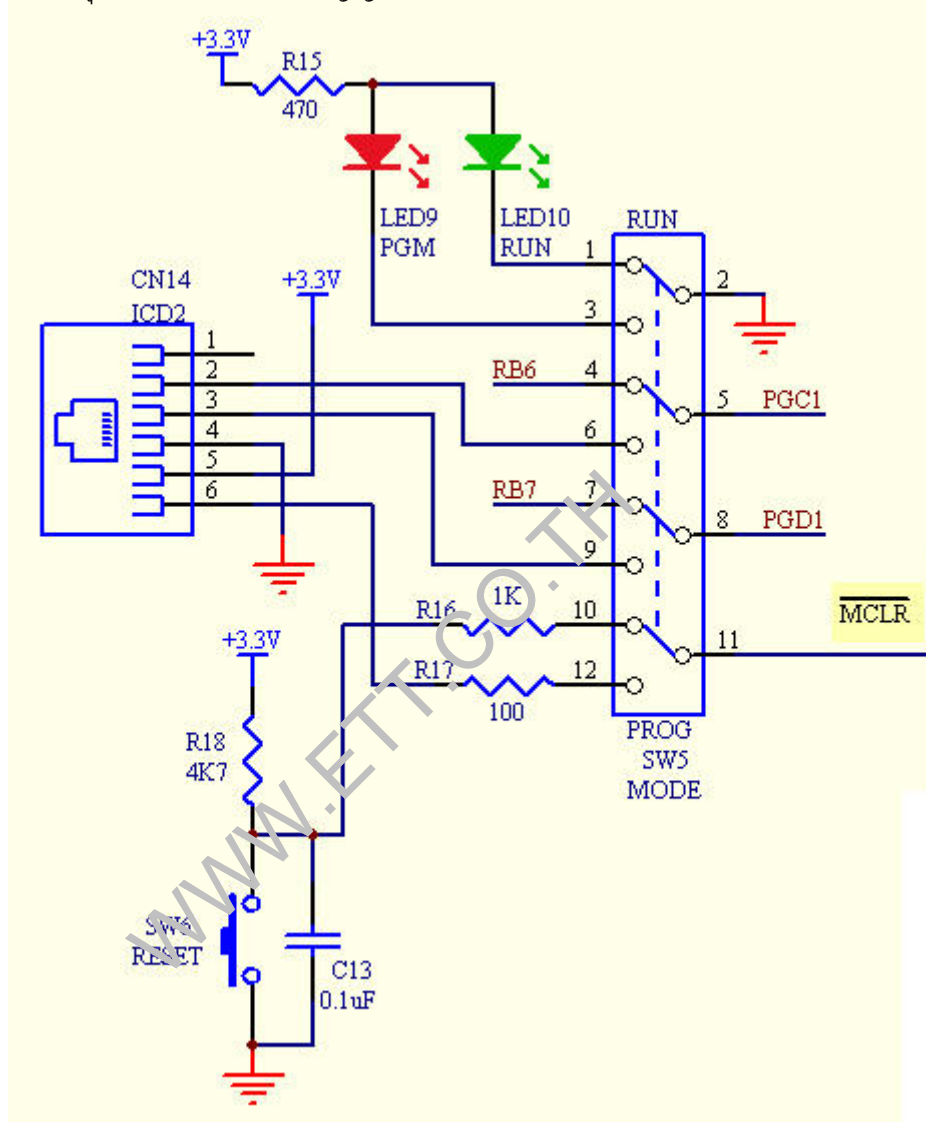




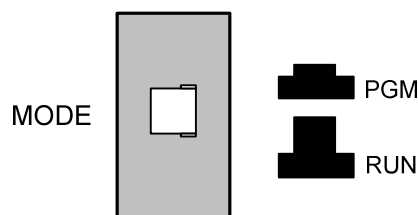
- **หมายเลข 12** พอร์ตสัญญาณ I/O ขนาด 2 PIN คือ สัญญาณ RF8 และ GND ดังต่อไปนี้



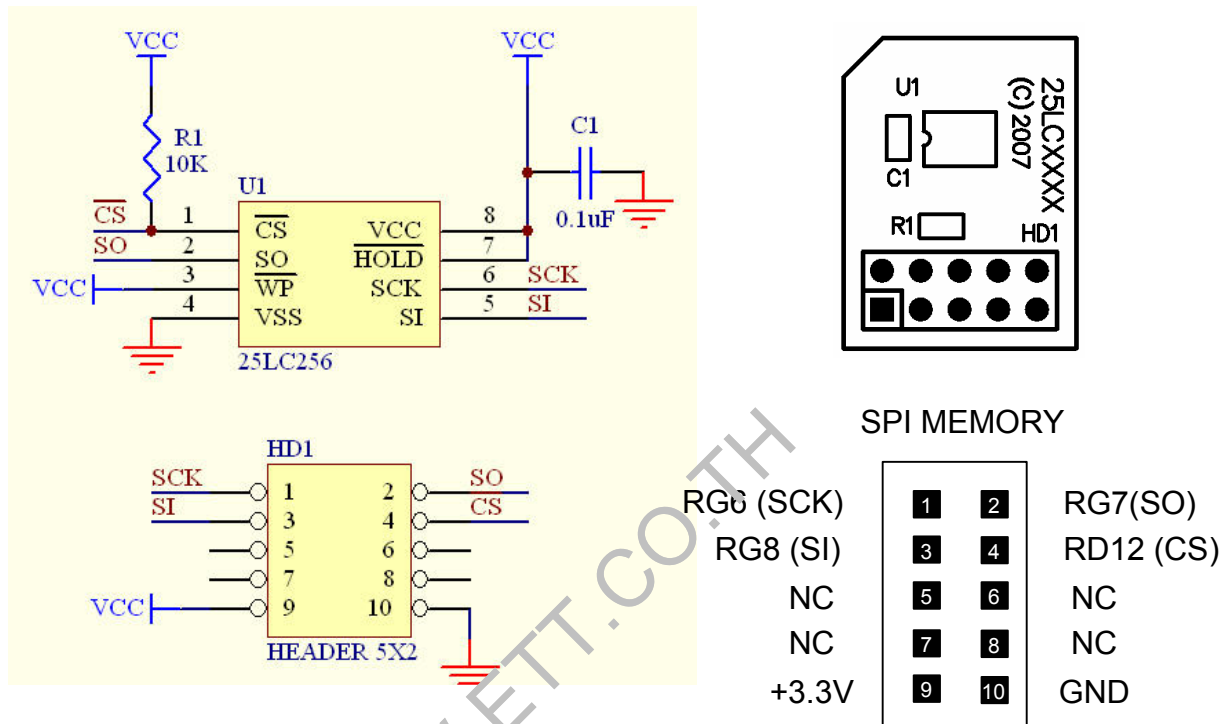
- **หมายเลข 13** ขั้วต่อสำหรับดาวน์โหลดโปรแกรม เป็นขั้วที่จัดเรียงตามมาตรฐานของ ICD2 รองรับเครื่องโปรแกรมที่มีการเชื่อมต่อตามมาตรฐานของ ICD2 เช่น PICKit2 , ICD2 และ เครื่องโปรแกรมของทางบริษัท อีทีที คือ ET-PGMPIC USB โดยก่อนทำการโปรแกรมทุกครั้งต้องกดสวิตช์ MODE ให้มาอยู่ที่ตำแหน่ง PGM ทุกครั้ง เพื่อตัดต่อขาสัญญาณมาเข้ากับเครื่องโปรแกรมจากภายนอก



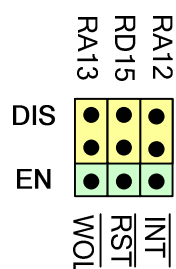
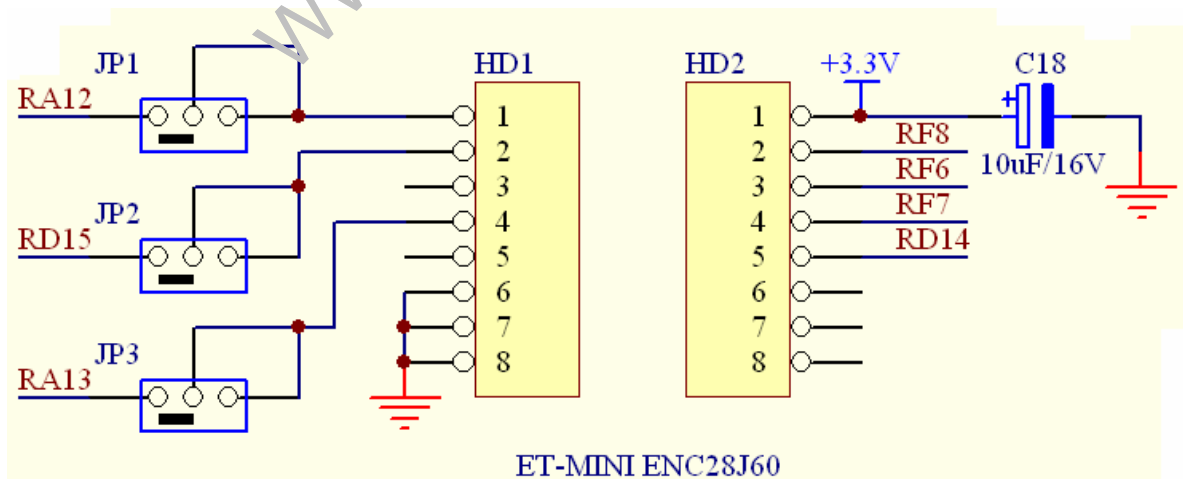
- **หมายเลข 14** สวิตช์เลือกโหมด RUN และ PGM สวิตช์นี้ เมื่อกดมาที่ตำแหน่ง PGM จะทำหน้าที่ตัดต่อขาสัญญาณที่ใช้ในการโปรแกรมได้ข้อมูลเข้ากับเครื่องโปรแกรม เพื่อทำการโปรแกรมข้อมูลโปรแกรมที่เราออกแบบ และ เมื่อกดปล่อยกลับมาที่ตำแหน่ง RUN ขาสัญญาณต่างๆ จะกลับมาเป็น I/O ใช้งานได้ตามปกติ



- หมายเลข 15 ขั้วต่อ DC-JACK สัญญาณไฟเลี้ยงบอร์ด รองรับแรงดันไฟจากภายนอก 7-12 VDC
- หมายเลข 16 สวิตช์รีเซ็ต (Reset Switch)
- หมายเลข 17 ไอซีไมโครคอนโทรลเลอร์ dsPIC33FJ128GP708
- หมายเลข 18 หน่วยความจำ EEPROM เบอร์ 25LCxxx ของ บริษัท Microchip เชื่อมต่อแบบ SPI



- หมายเลข 19 ขั้วสัญญาณเชื่อมต่อกับโมดูลสื่อสาร Ethernet รุ่น ET-MINI ENC28J60

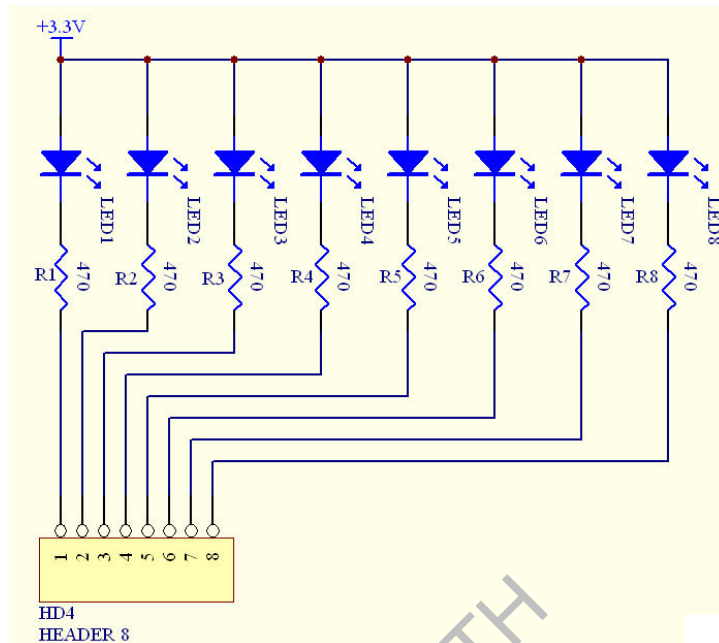


จัมเปอร์สำหรับการเลือกการเชื่อมต่อสัญญาณ RA12, RD15 และ RA13 ของไมโครคอนโทรลเลอร์ กับขาสัญญาณ INT, RST และ WOL ของโมดูล ET-MINI ENC28J60

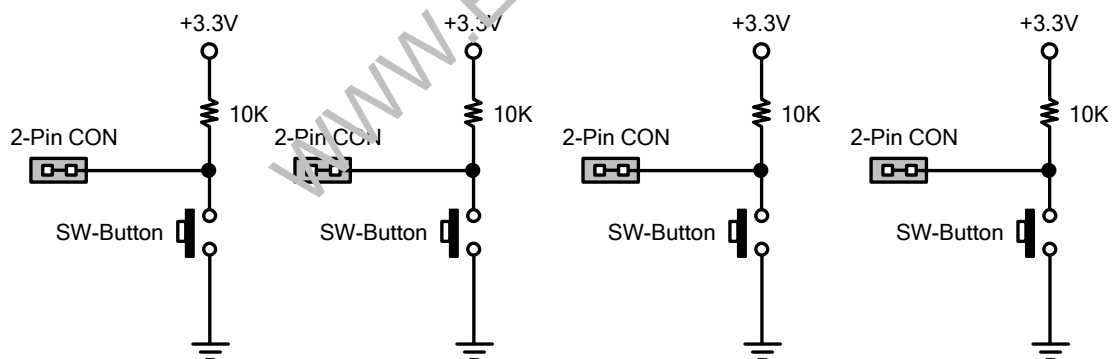
DIS = Disable คือ ไม่เชื่อมต่อสัญญาณ

EN = Enable คือ เชื่อมต่อสัญญาณ

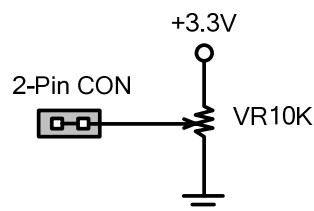
- หมายเลข 20 ชุด Test I/O LED ประกอบด้วยหลอดไฟ LED จำนวน 8 ชุด ดังวงจรต่อไปนี้



- หมายเลข 21 ชุดทดลองสัญญาณอินพุต จากสวิตช์ 4 ชุด สามารถสร้างสัญญาณลอจิก 0 (0 โวลท์) และ ลอจิก 1 (+3.3 โวลท์) ดังวงจรต่อไปนี้



- หมายเลข 22 ชุดทดลองแรงดันอนาล็อก 4 ชุด สามารถปรับระดับแรงดันไฟได้ตั้งแต่ 0 – 3.3 โวลท์ โดย มีการต่อวงจรดังต่อไปนี้



- หมายเลข 23 พื้นที่ออกแบบประสงค์ สำหรับต่อวงจรเพิ่มเติม

โมดูล ET-MINI ENC28J60

ET-MINI ENC28J60 เป็นโมดูลที่ออกแบบมาเพื่อเป็นตัวกลางในการเชื่อมต่อระบบการสื่อสารระหว่างไมโครคอนโทรลเลอร์ กับโครงข่าย Ethernet รองรับการทำงานของโปรโตคอล TCP/IP โดยใช้ไอซี ENC28J60 ซึ่งเป็นไอซี Ethernet Controller รองรับมาตรฐานการสื่อสาร IEEE 802.3 เชื่อมต่อสัญญาณควบคุมผ่าน SPI Bus ความเร็วสูงสุด 10 Mb/s ในการพัฒนาโปรแกรมทางบริษัท Microchip จะสนับสนุนตัว Microchip TCP/IP Stack ซึ่งสามารถดาวน์โหลด และ นำไปใช้ได้ฟรีที่เว็บไซต์ของ Microchip (www.microchip.com) โดยคุณสมบัติต่างๆ ของ ENC28J60 จะเป็นดังนี้

คุณสมบัติของ IC ENC28J60

General:

- IEEE 802.3 compatible Ethernet Controller
- Integrated MAC and 10BASE-T PHY
- 8 Kbyte Transmit/Receive Packet Dual Port Buffer SRAM
- Programmable Automatic Retransmit on Collision
- Programmable Padding and CRC Generation
- Programmable Automatic Rejection of Erroneous Packets
- SPI™ Interface with speeds up to 10 Mb/s
- Supports Full and Half-Duplex modes

Buffer:

- Configurable transmit/receive buffer size
- Hardware managed circular receive FIFO
- Byte-wide random and sequential access
- Internal DMA for fast memory copying
- Hardware assisted IP checksum calculation

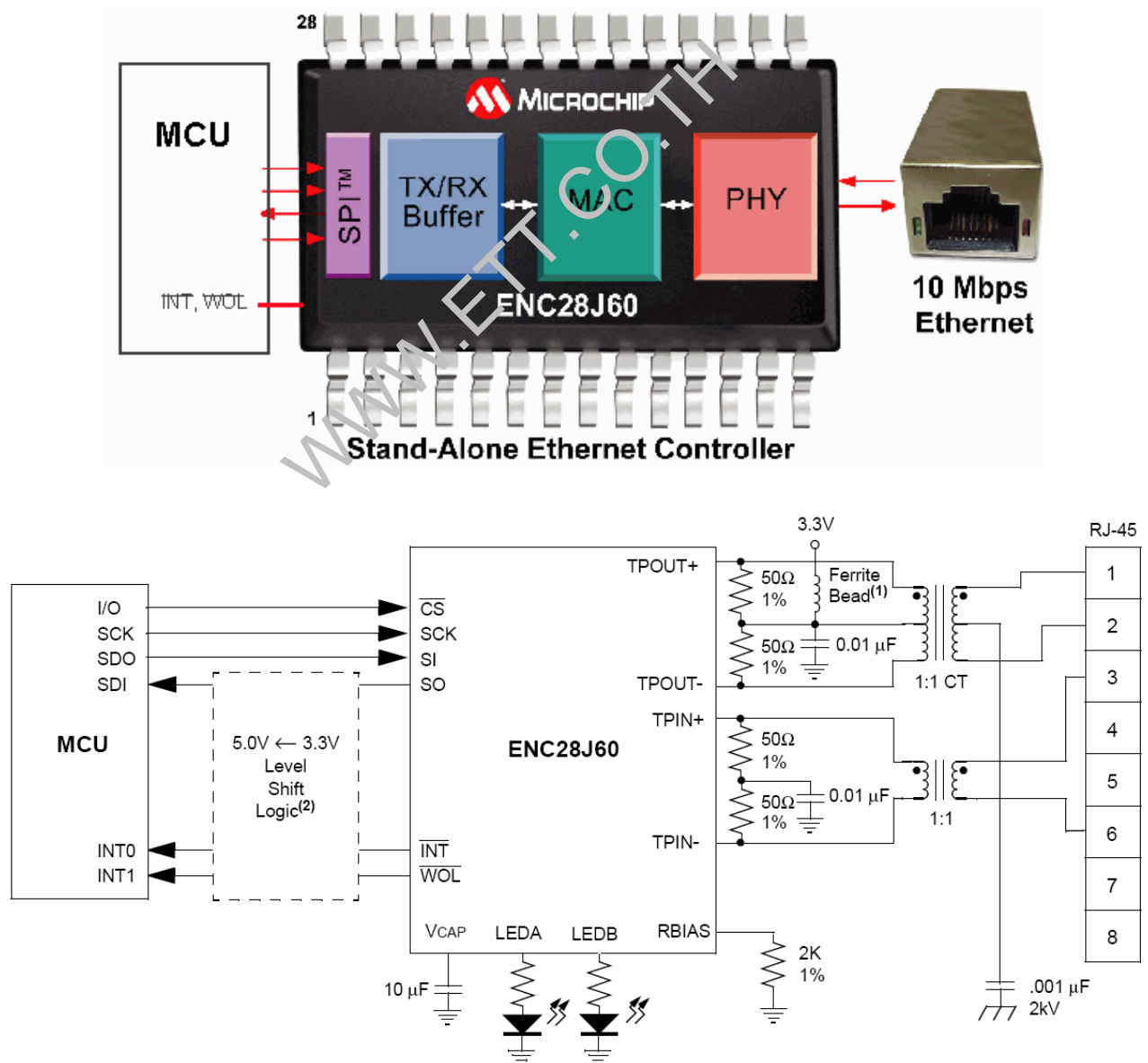
PHY:

- Wave shaping output filter
- Loopback mode

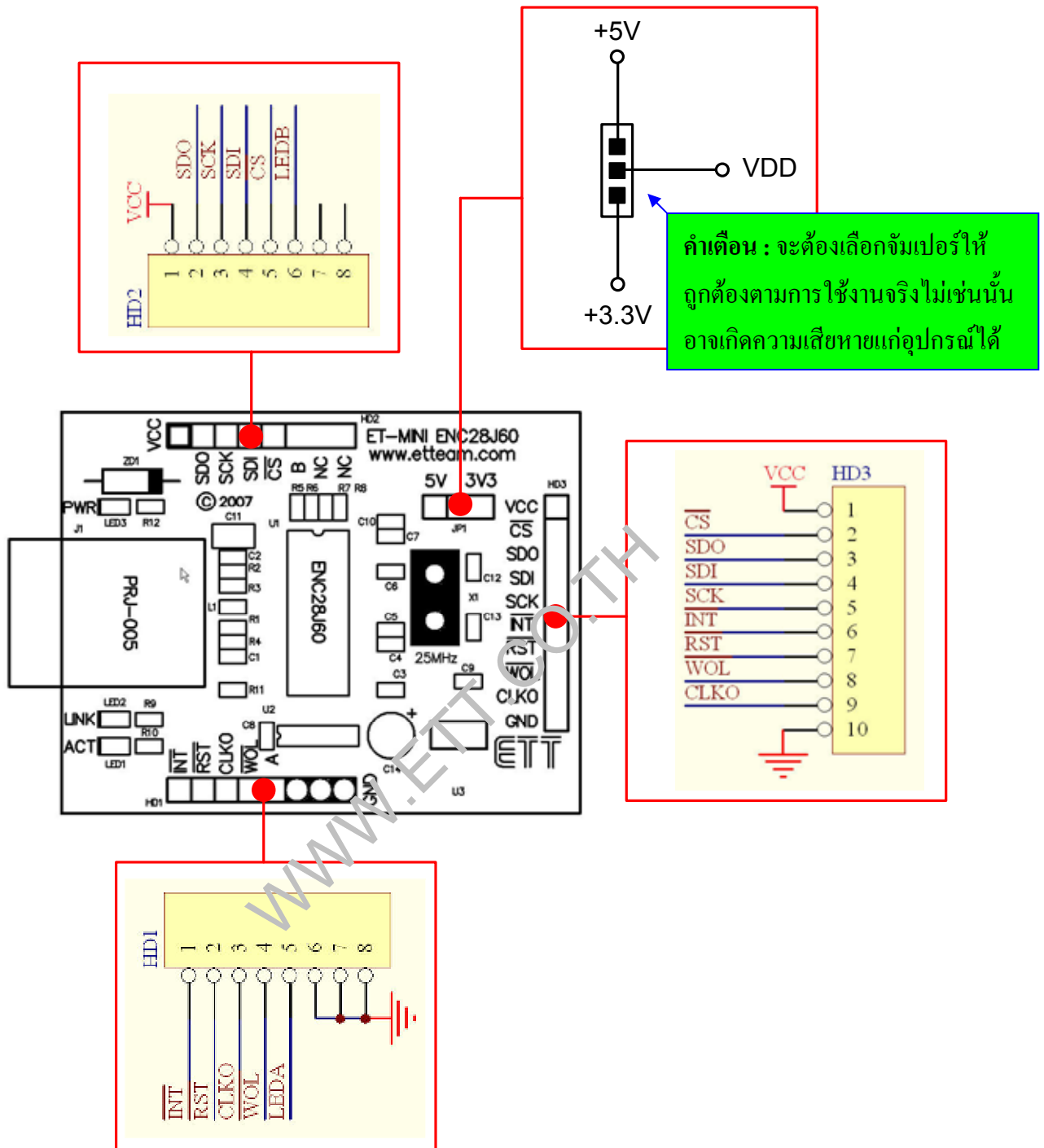
MAC:

- Support for Unicast, Multicast and Broadcast packets
- Programmable pattern matching of up to 64 bytes within packet at user defined offset
- Programmable wake-up on multiple packet formats, including Magic Packet®, Unicast, Multicast, Broadcast, specific packet match or any packet

การเชื่อมต่อกับไมโครคอนโทรลเลอร์ สามารถทำได้โดยง่าย โดยจะใช้การเชื่อมต่อแบบ SPI Bus ซึ่งจะใช้ขาสัญญาณเพียงไม่กี่ขา และ ในส่วนของระบบไฟ เนื่องจาก ENC28J60 เป็นไอซีที่ทำงานที่แรงดัน 3 โวลต์ ดังนั้นเพื่อให้สามารถใช้งานร่วมกับไมโครคอนโทรลเลอร์ที่ทำงาน 5 โวลต์ ทางทีมงานจึงได้ออกแบบ วงจรบัฟเฟอร์ สำหรับรองรับการเชื่อมต่อบนระบบไฟ ระหว่าง 3 โวลต์ กับ 5 โวลต์ เอาไว้ภายในบอร์ด ET-MINI ENC28J60 ซึ่งสามารถเลือกระบบไฟได้โดยการเลือก จัมป์เปอร์ 5V/3V3



รูปแสดงบล็อกไดอะแกรมการเชื่อมต่อ ENC28J60 กับ ไมโครคอนโทรลเลอร์



รูปแสดงโครงสร้างของบอร์ด ET-MINI ENC28J60

จากรูป HD1 และ HD2 ออกแบบไว้สำหรับการเชื่อมต่อกับบอร์ด ET-dsPIC33WEB V1.0 โดยสามารถเสียบบอร์ด ET-MINI ENC28J60 ขั้วบนบนบอร์ด ET-dsPIC33WEB V1.0 เข้ากับขั้ว Connector ตัวเมียที่จัดเตรียมไว้ให้ได้ทันที ส่วน HD3 ออกแบบไว้สำหรับนำไปใช้เชื่อมต่อกับไมโครคอนโทรเลอร์อื่นๆตามต้องการ

ชื่อสัญญาณ	ชนิดของสัญญาณ	หน้าที่การทำงาน
CS	INPUT	สัญญาณ Enable/Disable การเชื่อมต่อ SPI Bus ของ ENC28J60 CS = 0 คือ Enable การเชื่อมต่อ SPI ของ ENC28J60 CS = 1 คือ Disable การเชื่อมต่อ SPI ของ ENC28J60
SDO	OUTPUT	สัญญาณ Serial Data Output
SCK	INPUT	สัญญาณนาฬิกา Serial Clock
SDI	INPUT	สัญญาณ Serial Data Input
INT	OUTPUT	สัญญาณอินเตอร์รัพท์ แอคทีฟ ลอจิก 0
RST	INPUT	สัญญาณรีเซ็ต แอคทีฟ ลอจิก 0
WOL	OUTPUT	สัญญาณ Wake-up on LAN interrupt แอคทีฟ ลอจิก 0
CLKO	OUTPUT	สัญญาณ Programmable clock output
LEDA	OUTPUT	แสดงสถานะของสัญญาณ LINK
LEDB	OUTPUT	แสดงสถานะของสัญญาณ ACT

ตาราง แสดงชื่อและหน้าที่ขาสัญญาณต่างๆ ของ ENC28J60

ENC28J60	ET-dsPIC33WEB V1.0 (dsPIC33FJ128GP708)
CS	RD14
SDO	RF8
SCK	RF6
SDI	RF7
INT	RA12 (เลือกใช้โดยจัมเปอร์)
RST	RD15 (เลือกใช้โดยจัมเปอร์)
WOL	RA13 (เลือกใช้โดยจัมเปอร์)
CLKO	-
LEDA	-
LEDB	-

ตารางแสดง สัญญาณการเชื่อมต่อระหว่างบอร์ด ENC28J60 กับ ET-dsPIC33WEB V1.0

สรุปการจัดสรรและใช้งานทรัพยากรของบอร์ด ET-dsPIC33WEB V1.0

ตามปกติแล้ว MCU เบอร์ dsPIC33FJ128GP708 ที่ใช้กับบอร์ด ET-dsPIC33WEB V1.0 นั้น จะมี ขาสัญญาณให้ใช้งานได้โดยอิสระมากถึง 69 I/O แต่จะมีขาสัญญาณบางส่วน ถูกออกแบบและเชื่อมต่อไว้กับ อุปกรณ์ I/O เป็นการเฉพาะไว้เรียบร้อยแล้ว ไม่สามารถนำมาใช้เป็น I/O โดยทั่วไปได้ ซึ่งพอสรุปได้ดังนี้

MCU Oscillator

- RC12 ใช้เป็น OSC1 ต่อกับ Crystal ค่า 8.00MHz สำหรับใช้เป็นสัญญาณนาฬิกาของ MCU
- RC15 ใช้เป็น OSC2 ต่อกับ Crystal ค่า 8.00MHz สำหรับใช้เป็นสัญญาณนาฬิกาของ MCU

RTC Oscillator

- RC13 ใช้เป็น OSC1 ต่อกับ Crystal ค่า 32.768KHz สำหรับใช้เป็นสัญญาณนาฬิกาของ RTC
- RC14 ใช้เป็น OSC2 ต่อกับ Crystal ค่า 32.768KHz สำหรับใช้เป็นสัญญาณนาฬิกาของ RTC

พอร์ตสื่อสารอนุกรม(UART) RS232-CH1

- RF2 ใช้เป็นขา RXD สำหรับรับข้อมูลจาก RS232 ช่อง-1
- RF3 ใช้เป็นขา TXD สำหรับส่งข้อมูลให้ RS232 ช่อง-1

พอร์ตสื่อสารอนุกรม(UART) RS232-CH2

- RF4 ใช้เป็นขา RXD สำหรับรับข้อมูลจาก RS232 ช่อง-2
- RF5 ใช้เป็นขา TXD สำหรับส่งข้อมูลให้ RS232 ช่อง-2

Ethernet Module (SPI-1)

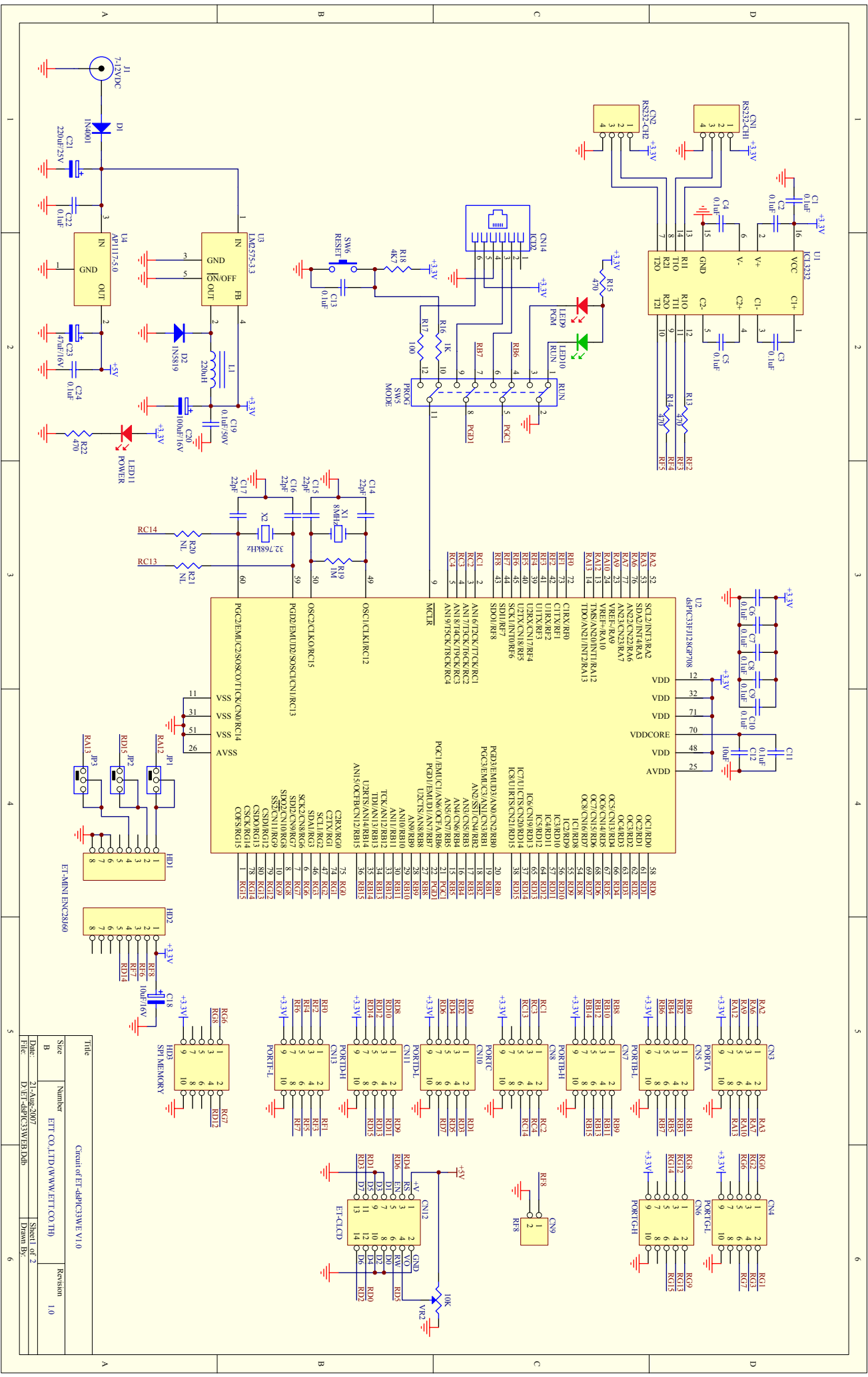
- RF6 ใช้เป็น SCK ในการเชื่อมต่อกับ Ethernet Module (ET-MINI ENC28J60)
- RF7 ใช้เป็น SDI ในการเชื่อมต่อกับ Ethernet Module (ET-MINI ENC28J60)
- RF8 ใช้เป็น SDO ในการเชื่อมต่อกับ Ethernet Module (ET-MINI ENC28J60)
- RD14 ใช้เป็น CS ในการเชื่อมต่อกับ Ethernet Module (ET-MINI ENC28J60)
- RA12 ใช้เป็น INT1 ในการเชื่อมต่อกับ Ethernet Module (ET-MINI ENC28J60) โดยสามารถเลือกใช้หรือไม่ใช้ได้ โดยการกำหนดที่ Jumper INT(EN/DS) ซึ่งตามปกติเลือกเป็น DS(Disable:ไม่ใช้งาน)
- RA13 ใช้เป็น WOL ในการเชื่อมต่อกับ Ethernet Module (ET-MINI ENC28J60) โดยสามารถเลือกใช้หรือไม่ใช้ได้ โดยการกำหนดที่ Jumper WOL(EN/DS) ซึ่งตามปกติเลือกเป็น DS(Disable:ไม่ใช้งาน)
- RD15 ใช้เป็น RST ในการเชื่อมต่อกับ Ethernet Module (ET-MINI ENC28J60) โดยสามารถเลือกใช้หรือไม่ใช้ได้ โดยการกำหนดที่ Jumper RST(EN/DS) ซึ่งตามปกติเลือกเป็น DS(Disable:ไม่ใช้งาน)

SPI Memory Module (SPI-2)

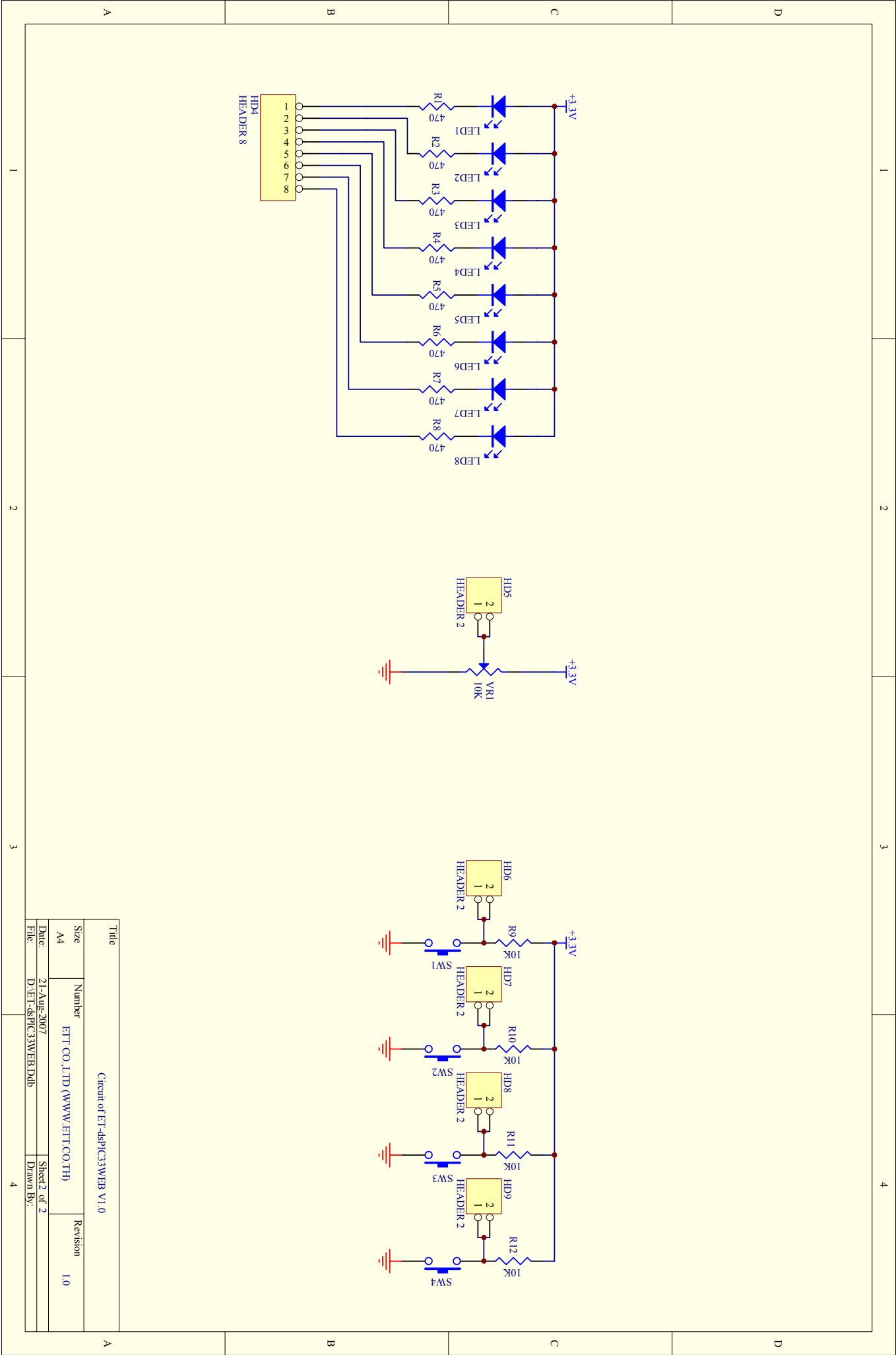
- RG6 ใช้เป็น SCK ในการเชื่อมต่อกับ SPI Memory
- RG7 ใช้เป็น SDI ในการเชื่อมต่อกับ SPI Memory
- RG8 ใช้เป็น SDO ในการเชื่อมต่อกับ SPI Memory
- RD12 ใช้เป็น CS ในการเชื่อมต่อกับ SPI Memory

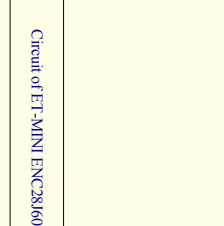
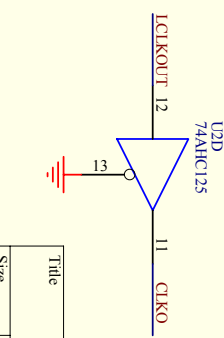
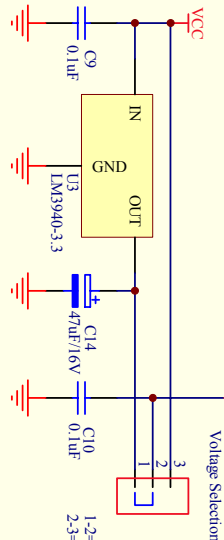
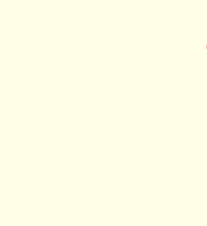
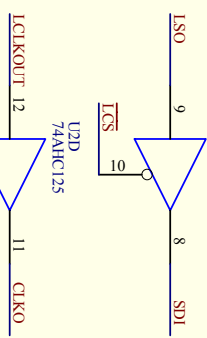
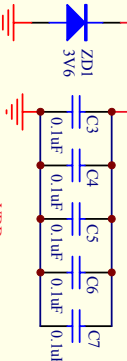
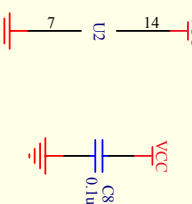
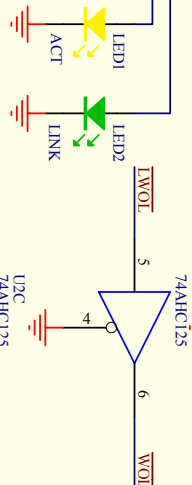
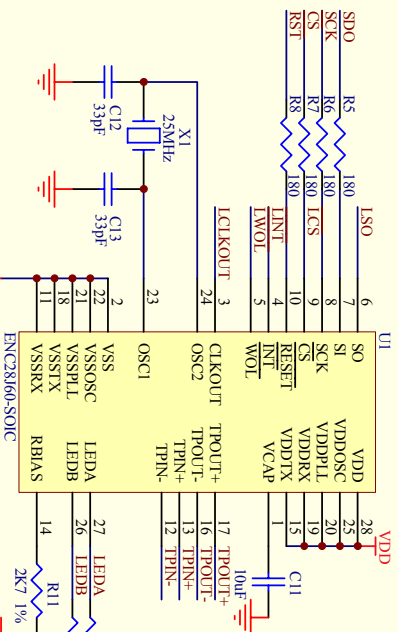
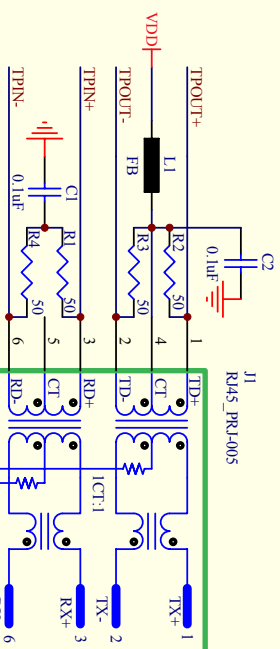
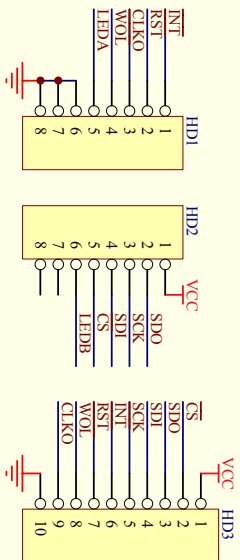
Character LCD Display

- RD0 ใช้เป็น LCD D4 ในการเชื่อมต่อกับ Character LCD แบบ 4 Bit
- RD1 ใช้เป็น LCD D5 ในการเชื่อมต่อกับ Character LCD แบบ 4 Bit
- RD2 ใช้เป็น LCD D6 ในการเชื่อมต่อกับ Character LCD แบบ 4 Bit
- RD3 ใช้เป็น LCD D7 ในการเชื่อมต่อกับ Character LCD แบบ 4 Bit
- RD4 ใช้เป็น LCD RS ในการเชื่อมต่อกับ Character LCD แบบ 4 Bit
- RD5 ใช้เป็น LCD RW ในการเชื่อมต่อกับ Character LCD แบบ 4 Bit
- RD6 ใช้เป็น LCD EN ในการเชื่อมต่อกับ Character LCD แบบ 4 Bit



Title			Circuit of ET-486C3WE V1.0		
Size			Number		
B			ETT CO.,LTD.(WWW.ETT.CO.TH)		
Date:			Revision		
21-Aug-2007			1.0		
Title:			Sheet of 2		
D:\ET-486C3WE\B.DDB			Drawn By:		



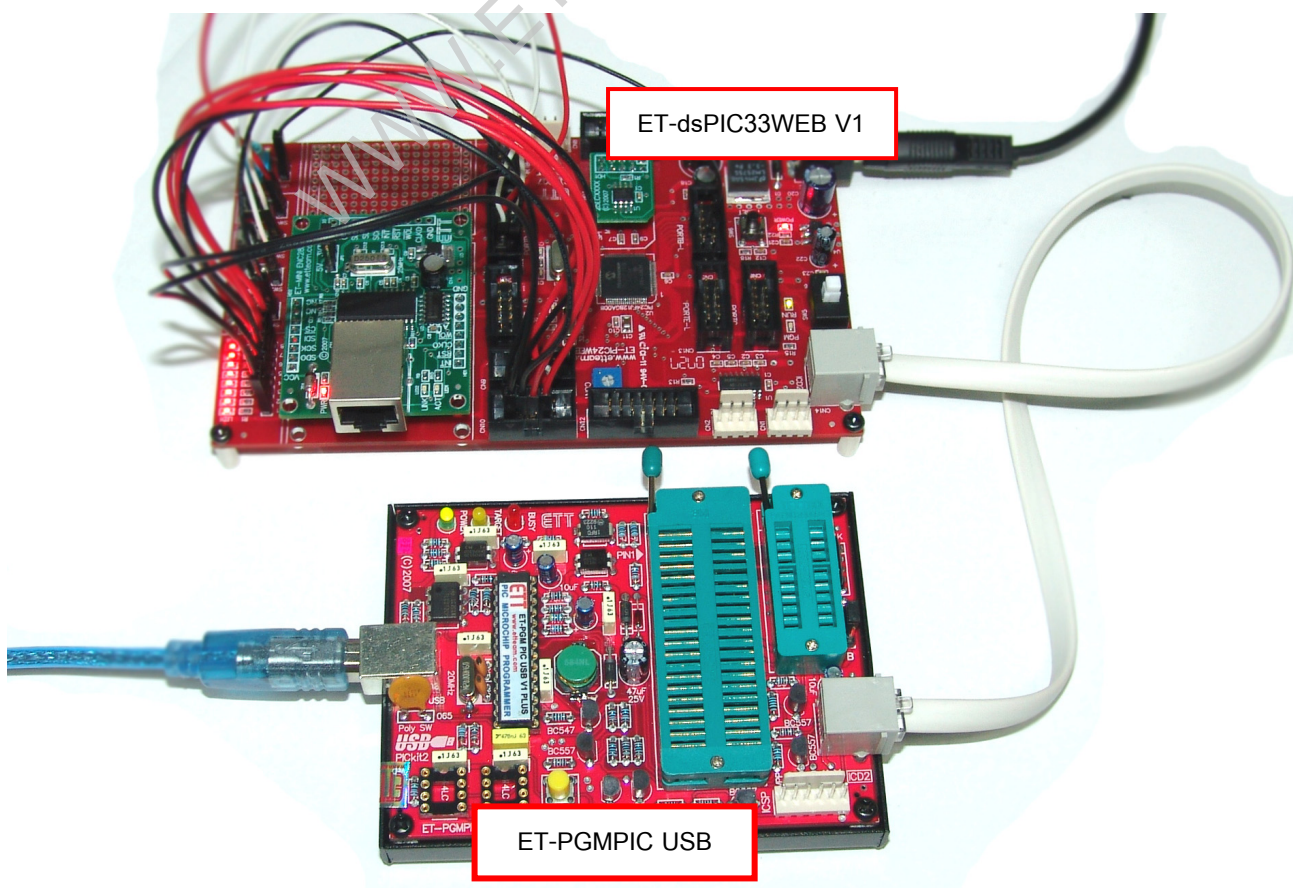


Title			
Circuit of ET-MINI ENC28J60			
Size	Number	Revision	
A4	1	1.0	
Date:	3-Sep-2007	Sheet 1 of 1	
File:	D:\ET-MINI ENC28J60.ddb	Drawn By:	

การพัฒนาโปรแกรมของบอร์ด

สำหรับการพัฒนาโปรแกรมของบอร์ด “ET-dsPIC33WEB V1.0” นั้น จะแบ่งออกเป็น 2 ส่วนด้วยกัน คือ ส่วนของการพัฒนาโปรแกรม และการ Download โปรแกรม โดยในส่วนของการพัฒนาโปรแกรมของบอร์ดนั้น จะเป็นการเขียนโปรแกรมเพื่อควบคุมและสั่งงานให้ MCU สามารถทำงานตามจุดประสงค์ที่ได้ออกแบบไว้ ซึ่งในส่วนนี้เป็นหน้าที่ของผู้พัฒนาโปรแกรม ที่จะเลือกใช้ภาษาใดในการพัฒนาโปรแกรม รวมไปถึงการเลือกใช้โปรแกรมสำหรับทำหน้าที่แปลคำสั่งของภาษานั้นๆ ให้เป็นรหัสคำสั่งในรูปแบบของ HEX File สำหรับใช้ Download ให้กับหน่วยความจำโปรแกรม (Flash Memory) ของ MCU เพื่อสั่งงานให้ MCU ปฏิบัติตามคำสั่งในโปรแกรมที่ผู้พัฒนาโปรแกรมเขียนขึ้นมา

สำหรับส่วนของการ Download โปรแกรม หรือการ Download HEX File ที่ได้จากการพัฒนาโปรแกรมให้กับหน่วยความจำโปรแกรม (Flash Memory) ของ MCU นั้น จะต้องอาศัยเครื่องมือทางฮาร์ดแวร์เป็นส่วนประกอบในการพัฒนาโปรแกรมด้วย ซึ่งในกรณีของบอร์ด “ET-dsPIC33WEB V1.0” นั้น จะต้องใช้ เครื่องมือสำหรับโปรแกรมข้อมูลให้กับ MCU ภายในบอร์ดด้วยวิธีการแบบ In-Circuit โดยต้องใช้เครื่อง Programmer หรือใช้ชุดพัฒนาโปรแกรมจำพวก In-Circuit Debugger ที่สนับสนุนการใช้งานร่วมกับ MCU เบอร์ dsPIC33FJ128GP708 ซึ่งใช้ข้อต่อสัญญาณตรงตามมาตรฐาน “ICD2” ของ Microchips เช่น ICD2, Pickit2 ซึ่งทางบริษัท อีทีที ก็มีสนับสนุน คือ “ET-PGMPIC USB” ดังตัวอย่าง



การเขียนโปรแกรมใช้งานกับบอร์ดโดยใช้ MPLAB C30

MPLAB C30 หรือ C30 Tools เป็นโปรแกรมภาษาซี สำหรับใช้แปลคำสั่งของ MCU ตระกูล dsPIC ซึ่งได้รับการพัฒนาขึ้นโดย Microchips เอง โดยข้อกำหนดและรายละเอียดของการเขียนโปรแกรมภาษาซี นั้น จะไม่กล่าวถึงในที่นี้ด้วย โดยถ้าผู้ใช้งานต้องการพัฒนาโปรแกรมให้กับ dsPIC ด้วยภาษาซี แต่ยังไม่มีความรู้เรื่องการเขียนโปรแกรมภาษาซีเลยนั้นขอแนะนำให้ หาหนังสือที่อธิบายเกี่ยวกับการเขียนโปรแกรมภาษาซีในส่วนที่เป็นมาตรฐานตามข้อกำหนดของ “ANSI C” มาศึกษาให้เข้าใจเสียก่อน และสำหรับส่วนของข้อกำหนดปลีกย่อยอื่น ๆ ที่เป็นของ MPLAB C30 เองก็สามารถอ่านเพิ่มเติมได้จากเอกสารและคู่มือการใช้งานของ MPLAB C30 ที่ทาง Microchips จัดทำไว้ได้ โดยสามารถ Download จาก Website ของ Microchips หรือจาก Folder ของ “C:\Program Files\Microchip\MPLAB C30\docs” ที่ทำการติดตั้งโปรแกรม MPLAB C30 ไว้ก็ได้ โดยในที่นี้ จะขอกล่าวแนะนำถึงเฉพาะส่วนของการกำหนดค่าตัวเลือกในโปรแกรมเพื่อใช้งานร่วมกับบอร์ด “ET-dsPIC33WEB V1.0” เท่านั้น โดยในการที่จะใช้งานโปรแกรม MPLAB C30 ในการเขียนโปรแกรมนั้น ผู้ใช้จำเป็นต้องทำการติดตั้งโปรแกรมของ Microchips จำนวน 2 โปรแกรมดังนี้คือ

- MPLAB IDE ซึ่งเป็นโปรแกรม Text Editor ของ Microchips ซึ่งในปัจจุบัน (ตุลาคม 2548) จะเป็นรุ่น 7.21 แล้วสามารถ Download มาใช้งานได้ฟรีจาก Web ของ Microchips
- MPLAB C30 ซึ่งเป็นตัวแปลภาษาซี (C Compiler) ให้เป็นรหัสคำสั่งของ dsPIC ซึ่งตามปกติแล้วโปรแกรมชุดนี้จะต้องซื้อมาใช้งานเอง แต่อย่างไรก็ตามทาง Microchips เองมีรุ่นทดลองใช้งานให้ผู้ใช้งานสามารถ Download มาใช้งานได้เช่นเดียวกันกับ MPLAB IDE

โดยโปรแกรมทั้ง 2 ชุดนี้ ทางอีทีที ได้ทำการ Download มาจัดเตรียมไว้ให้ในแผ่น CD-ROM ที่แถมไปกับบอร์ดของ “ET-dsPIC33WEB V1.0” ด้วยอยู่แล้ว โดยในการติดตั้งโปรแกรมนั้นขอแนะนำให้ผู้ใช้ทำการติดตั้งโปรแกรมในชุดของ MPLAB IDE ก่อนเป็นอันดับแรก โดยขอแนะนำให้ติดตั้งโปรแกรมของ MPLAB IDE ไว้ตามค่า Default ของโปรแกรมติดตั้งเลย คือ “C:\Program Files\Microchip\MPLAB IDE\” จะสะดวกต่อการใช้งานมากกว่า ซึ่งหลังจากทำการติดตั้งโปรแกรม MPLAB IDE เสร็จเรียบร้อยแล้วในครั้งแรกก่อนการใช้นั้นต้องสั่ง Restart เครื่องคอมพิวเตอร์ก่อน หลังจากนั้นแล้ว MPLAB IDE จึงจะสามารถทำงานได้โดยไม่เกิดปัญหา จากนั้นจึงทำการติดตั้งโปรแกรม MPLAB C30 เป็นลำดับถัดไป โดยขอแนะนำให้ทำการติดตั้งโปรแกรมชุดนี้ไว้ตามค่า Default ของการติดตั้งโปรแกรม จะเกิดความสะดวกต่อการใช้งานมากที่สุด โดยเฉพาะในขั้นตอนของการกำหนดการเชื่อมโยงการทำงานระหว่าง MPLAB IDE และ MPLAB C30 โดยในที่นี้จะขออธิบายโดยอ้างถึงตำแหน่งการติดตั้งโปรแกรมหดที่กล่าวไว้แล้วข้างต้นเท่านั้น ซึ่งถ้าผู้ใช้ทำการสั่งติดตั้งโปรแกรมไว้ยังตำแหน่ง Folder ที่แตกต่างไปจากนี้แล้วขอให้ทำความเข้าใจและดัดแปลงวิธีการกำหนดค่าเองตามที่ติดตั้งโปรแกรมไว้จริงๆด้วย

ตัวอย่างการสร้างโปรแกรมภาษาซีของ MPLAB C30

เมื่อทำการติดตั้งโปรแกรมกำหนดการเชื่อมโยงคำสั่งระหว่าง MPLAB IDE และ MPLAB C30 เป็นที่เรียบร้อยแล้ว ต่อจากนี้ไป ผู้ใช้ก็สามารถทำการเรียกใช้งานโปรแกรม MPLAB C30 ผ่านทางโปรแกรม MPLAB IDE ได้แล้ว โดยค่าตัวเลือกต่างๆที่ได้กำหนดไว้แล้วนั้นจะถูกเก็บไว้ใน Configuration ของโปรแกรมตลอดไป จนกว่าจะมีการเปลี่ยนแปลงใหม่ ซึ่งในที่นี้จะขอยกตัวอย่างการเขียนโปรแกรมภาษาซี ของ MPLAB C30 ดัง 1 ตัวอย่างพอเป็นแนวทางให้ผู้ใช้งานทำความเข้าใจ โดยมีลำดับขั้นตอนดังนี้

1. สั่ง Run Program ของ MPLAB IDE ขึ้นมา จากนั้นก็สร้างไฟล์ใหม่ขึ้นมา 1 ไฟล์ สำหรับเขียน Source Code ภาษาซี โดยใช้คำสั่ง “File → New” จากนั้นให้ทำการพิมพ์ Source Code ภาษาซี ใน Work Sheet ของโปรแกรม MPLAB IDE ดังตัวอย่าง

```

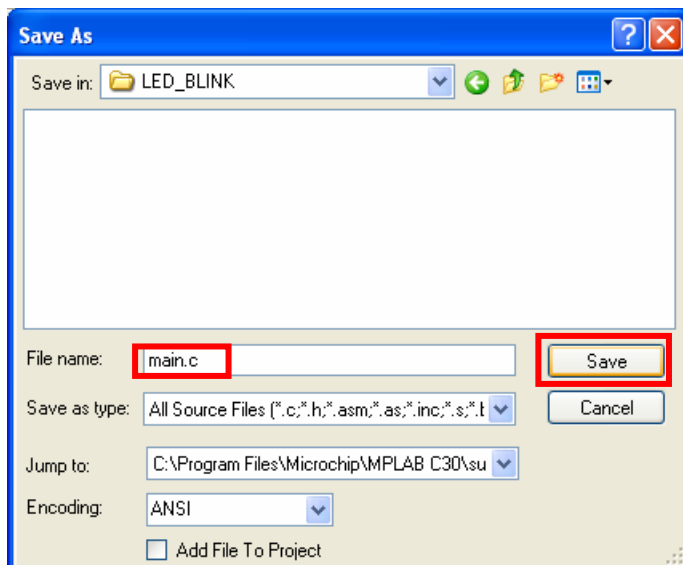
*****
/* Examples Program For "ET-dsPIC33WEB-V1" Board */
/* Hardware      : ET-dsPIC33WEB-V1.0 */
/* Target MCU    : dsPIC33FJ128GP708 */
/*               : X-TAL : 8.00 MHz */
/* Config Fosc   : XTAL = 8MHz (XT+PLL) */
/*               : PLLPRE[4..0] = 00000 (N1=2) */
/*               : VCO Input = 8MHz / 2 = 4MHz */
/*               : PLLDIV[8..0] = 0x26 (M=40) */
/*               : VCO Output = 4 x 40MHz = 160MHz */
/*               : PLLPOST[1:0] = 0:0 (M2=2) */
/*               : Fosc = 160MHz / 2 = 80MHz */
/*               : Fcy = Fosc/2 = 80MHz / 2 = 40MHz */
/* Compiler      : MPLAB + C30 V3.01 */
/* Write By      : Sakachai Makarn(ETT CO.,LTD.) */
/* Last Update   : 16/August/2007 */
/* Function      : Example LED Blink on Port-RE8 */
*****
// ET-dsPIC33WEB-V1 Hardware Board
// Fcy = 40MHz

#include "p33FJ128GP708.h"                // dsPIC33FJ128GP708 MPU Register

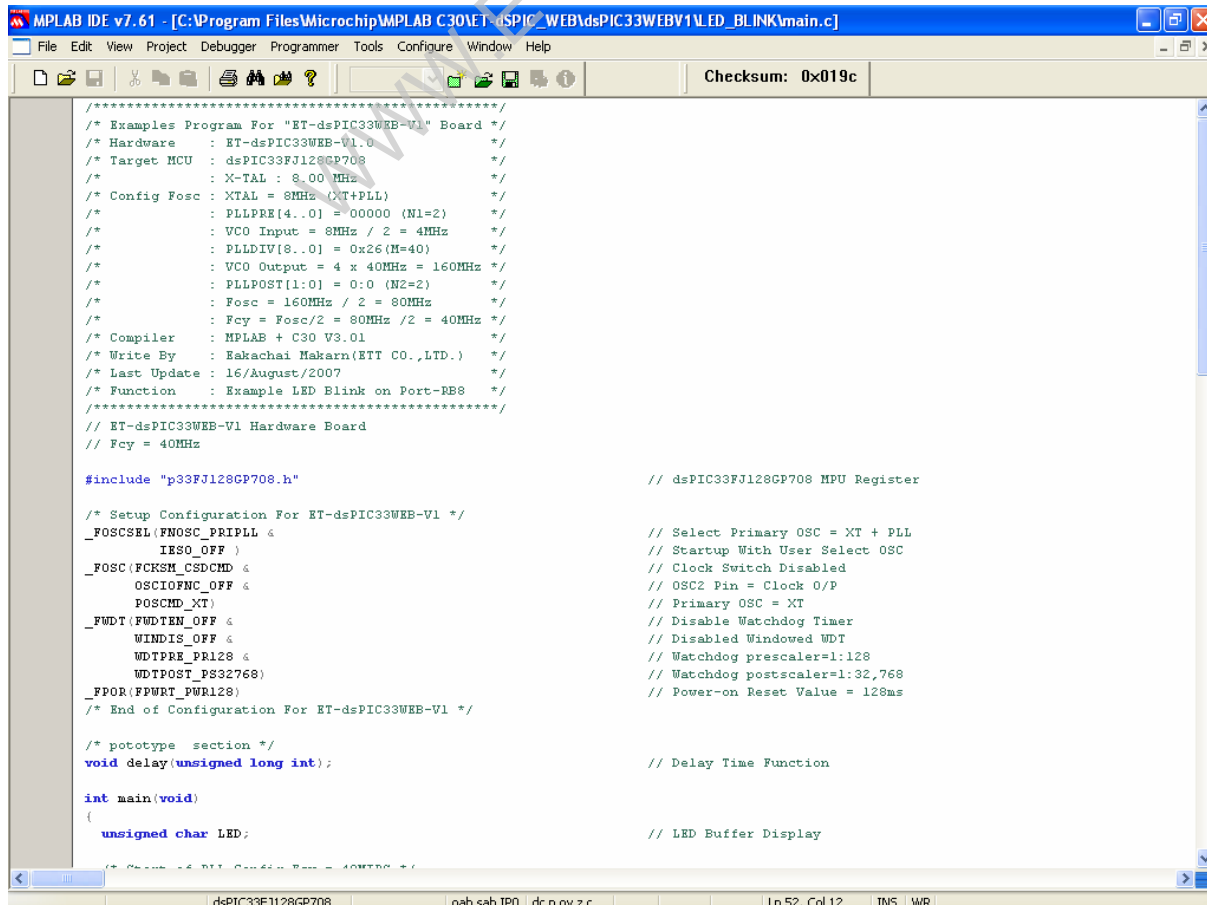
/* Setup Configuration For ET-dsPIC33WEB-V1 */
_FOSCSEL(FNOSC_PRIPLL &                // Select Primary OSC = XT + PLL
        IESO_OFF)                      // Startup With User Select OSC
_FOSC(FCKSM_CSDCMD &                   // Clock Switch Disabled
        OSCIOFNC_OFF &                 // OSC2 Pin = Clock O/P
        POSCMD_XT)                     // Primary OSC = XT
_FWDT(FWDTEN_OFF &                     // Disable Watchdog Timer
        WINDIS_OFF &                  // Disabled Windowed WDT
        WDTPRE_PR128 &                 // Watchdog prescaler=1:128
        WDTPRST_PS32768)                // Watchdog postscaler=1:32,768
_FPOR(FPWRT_FWR128)                     // Power-on Reset Value = 128ms
/* End of Configuration For ET-dsPIC33WEB-V1 */

/* pototype section */
void delay(unsigned long int);           // Delay Time Function
  
```

2. ทำการสั่งบันทึกไฟล์ที่เขียนขึ้นให้เป็น Text File ภาษาซี โดยให้ทดลองกำหนดชื่อเป็น “main.c” แล้วสั่งบันทึกไว้ใน Folder ชื่อ “...Microchip\MPLAB C30\examples\led_blink\main.c” โดยในที่นี้ให้ใช้คำสั่ง “File → Save As...” แล้วสร้าง Folder ชื่อ led_blink ไว้ภายใต้ Folder ของ examples อีกชั้นหนึ่ง แล้วกำหนดชื่อเป็น “main.c” แล้วเลือก “Save” ดังรูป



ซึ่งจะเห็นว่าเมื่อทำการสั่งบันทึกไฟล์เป็น “main.c” ไปแล้ว กลุ่มของตัวอักษรต่างๆที่ได้พิมพ์ไว้ จะถูกจัดแบ่งกลุ่ม โดยใช้สีในการแสดงผลที่แตกต่างกันไปตามหน้าที่ของกลุ่มตัวอักษร เช่น กลุ่มตัวอักษรที่ใช้เป็นคำอธิบาย (Comment) กลุ่มตัวอักษรที่เป็นคำสั่ง และกลุ่มตัวอักษรที่เป็นตัวแปรต่างๆ ซึ่งจุดนี้เป็นข้อดีของ MPLAB IDE ที่สามารถแยกการแสดงผลกลุ่มตัวอักษรตามหน้าที่การใช้งานได้ ทำให้เราสามารถอ่านโปรแกรมได้ง่ายและสะดวกมากยิ่งขึ้นดังรูป



```

#include "p33FJ128GP708.h" // dsPIC33FJ128GP708 MPU Register

/* Setup Configuration For ET-dsPIC33WEB-V1 */
_FOSCSEL(FNOSC_PRIPLL & // Select Primary OSC = XT + PLL
        IESO_OFF )      // Startup With User Select OSC
_FOSC(FCKSM_CSDCMD &    // Clock Switch Disabled
        OSCIOFNC_OFF & // OSC2 Pin = Clock O/P
        POSCMD_XT)      // Primary OSC = XT
_FWDT(FWDTEN_OFF &     // Disable Watchdog Timer
        WINDIS_OFF &   // Disabled Windowed WDT
        WDTPRE_PR128 & // Watchdog prescaler=1:128
        WDTPOST_PS32768) // Watchdog postscaler=1:32,768
_FPOR(FPWRT_PWR128)     // Power-on Reset Value = 128ms
_CONFIG3(JTAGEN_OFF);   // Disable JTAG Interface
/* End of Configuration For ET-dsPIC33WEB-V1 */

/* Pototype Section */
void delay(unsigned long int); // Delay Time Function

int main(void)
{
    /* Start of PLL Config Fcy = 40MIPS */
    CLKDIV &= 0xFFFE0; // PLLPRE[4..0] = 00000 (N1=2)
    CLKDIV &= 0xFF3F; // PLLPOST[1:0] = 00 (N2=2)
    PLLFBD = 0x26;    // PLLDIV[8..0] = 000100110 (M=40)
    while(!OSCCONbits.LOCK); // Wait PLL to Lock
    OSCCONbits.CLKLOCK = 1; // Disable PLL Modify
    /* End of PLL Config Fly = 40MIPS */

    /* Start Config RB8 = Output */
    AD1PCFGLbits.PCFG8 = 1; // AN8(RB8) = Digital Mode
    TRISBbits.TRISB8 = 0; // RB8 = Output
    LATBbits.LATB8 = 0; // Start = ON LED

    while(1) // Loop Continue
    {
        LATBbits.LATB8 = !LATBbits.LATB8; // Toggle Output RB8
        delay(2000000); // Display LED Delay
    }
}

/*****
/* Delay Time Function */
/* 1-4294967296 */
*****/
void delay(unsigned long int count1)
{
    while(count1 > 0) {count1--;} // Loop Decrease Counter
}

```

แสดง ตัวอย่าง Source Code สำหรับใช้ทดลองการทำงาน

สำหรับตัวอย่างนี้จะเป็นการสั่งให้ใช้พอร์ต RB8 ทำหน้าที่เป็น Output ขับ LED ให้ติดและดับสลับกันไปไม่รู้จักในลักษณะของไฟกระพริบ ซึ่งวิธีการทดสอบการทำงานของโปรแกรมนี้นี้ โดยใช้กับบอร์ด

ET-dsPIC33WEB V1.0 นั้นทำได้โดยต่อสัญญาณจาก RB8(PORTB-H) เข้ากับ LED1 Output ของบอร์ด โดยจะเห็นผลการทำงานของโปรแกรมแกรม คือ LED1 จะกะพริบ ติดและดับอย่างต่อเนื่องตลอดเวลา

โดยส่วนที่สำคัญและมีความจำเป็นมากที่สุดของโปรแกรมก็คือส่วนของคำสั่งที่ใช้ในการกำหนดค่าของ Configuration Bit ให้กับ MCU โดยค่านี้จะถูกแปลและใช้กำหนดค่าการทำงานของ Fuse Bit ในตัว MCU ในขั้นตอนของการ Download Code ให้กับ MCU โดยในตัวอย่างเป็นการกำหนดให้ MCU ทำงานด้วยสัญญาณนาฬิกา 8MHz จากภายนอก ร่วมกับวงจรคูณความถี่ (Phase Lock Loop :PLL) ซึ่งตัวอย่างโปรแกรมต่างๆที่ทางอีทีที สร้างขึ้น จะอ้างอิงจาก Configuration ดังต่อไปนี้

- กำหนดให้ MCU ทำงานจากสัญญาณนาฬิกา : XTAL = 8MHz (XT+PLL)
- กำหนดค่าการทำงานของวงจรคูณความถี่ Phase Lock Loop ดังนี้
 - PLLPRE[4..0] = 00000 (N1=2)
 - VCO Input = 8MHz / 2 = 4MHz
 - PLLDIV[8..0] = 0x26(M=40)
 - VCO Output = 4 x 40MHz = 160MHz
 - PLLPOST[1:0] = 0:0 (N2=2)
 - Fosc = 160MHz / 2 = 80MHz
 - Fcy = Fosc/2 = 80MHz / 2 = 40MHz

```

/* Setup Configuration For ET-dsPIC33WEB-V1 */
_FOSCSEL(FNOSC_PRIPLL & IESO_OFF ) // Select Primary OSC = XT + PLL
_FOSC(FCKSM_CSDCMD & OSCIOFNC_OFF & POSCMD_XT) // Startup With User Select OSC
// Clock Switch Disabled
// OSC2 Pin = Clock O/P
// Primary OSC = XT
_FWDT(FWDTEN_OFF & WINDIS_OFF & WDTPRE_PR128 & WDTPOST_PS32768) // Disable Watchdog Timer
// Disabled Windowed WDT
// Watchdog prescaler=1:128
// Watchdog postscaler=1:32,768
_FPOR(FPWRT_PWR128) // Power-on Reset Value = 128ms
_CONFIG3(JTAGEN_OFF); // Disable JTAG Interface
/* End of Configuration For ET-dsPIC33WEB-V1 */

```

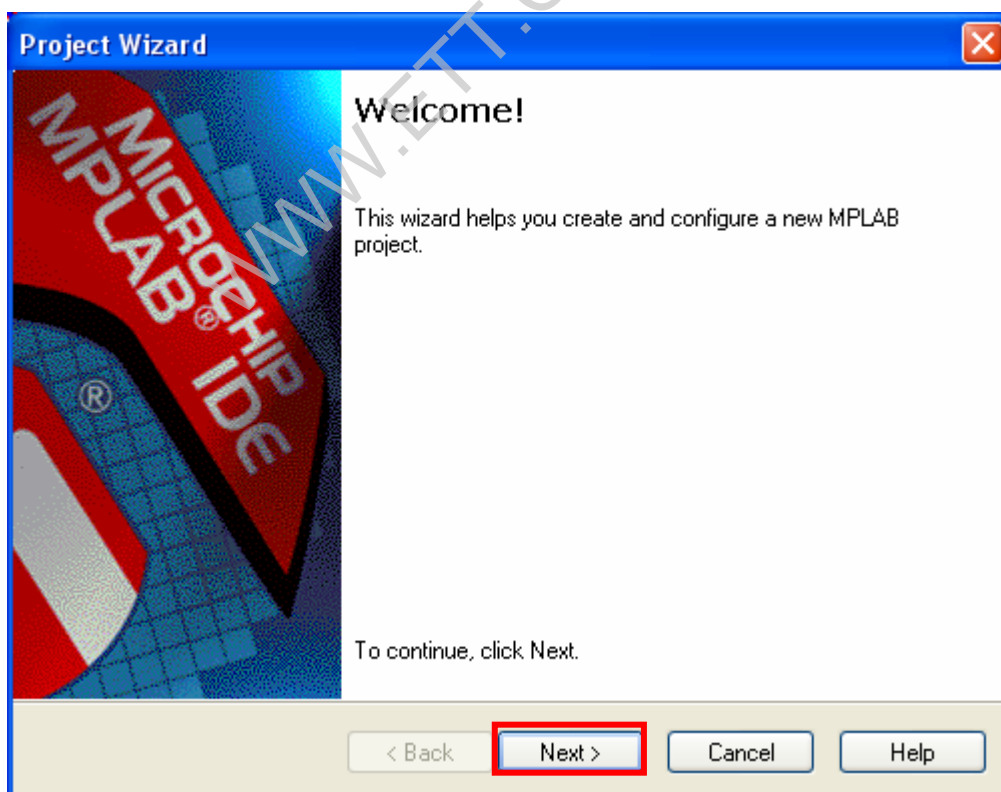
ส่วนของคำสั่งในการกำหนดค่าการทำงานของ Phase Lock Loop เพื่อทำการคูณความถี่จากสัญญาณนาฬิกา 8.00MHz ที่ต่อไว้ภายนอก เพื่อให้ได้ความถี่ Fosc เป็น 80 MHz เพื่อให้ CPU สามารถประมวลผลด้วยความเร็วสูงสุดที่ 40MIPS (Fcy = 40MHz) ดังตัวอย่าง

```

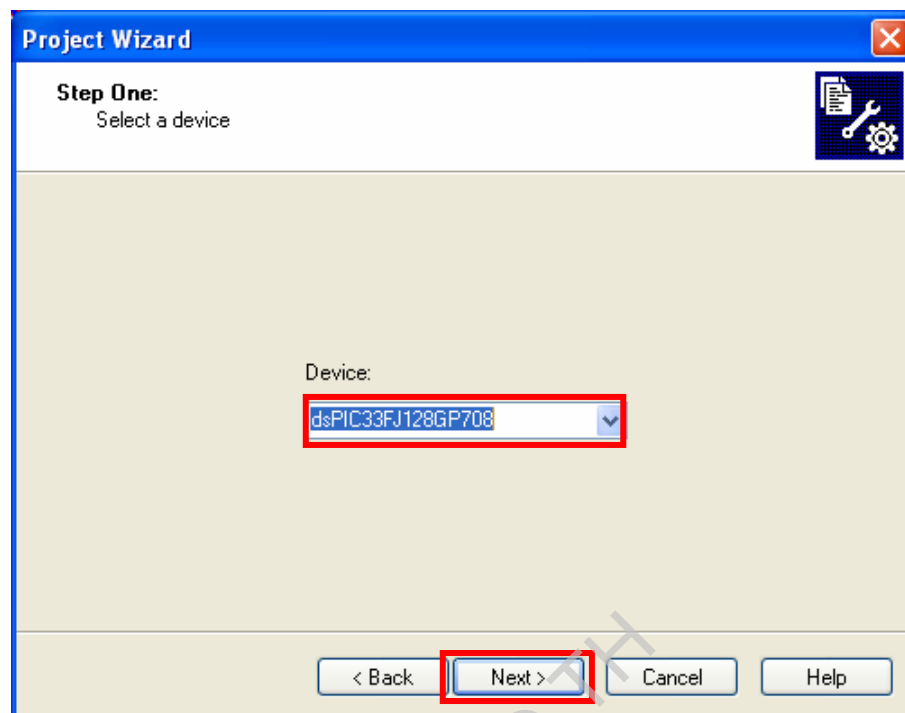
/* Start of PLL Config Fcy = 40MIPS */
CLKDIV &= 0xFFE0; // PLLPRE[4..0] = 00000 (N1=2)
CLKDIV &= 0xFF3F; // PLLPOST[1:0] = 00 (N2=2)
PLLFBD = 0x26; // PLLDIV[8..0] = 000100110 (M=40)
while(!OSCCONbits.LOCK); // Wait PLL to Lock
OSCCONbits.CLKLOCK = 1; // Disable PLL Modify
/* End of PLL Config Fly = 40MIPS */

```

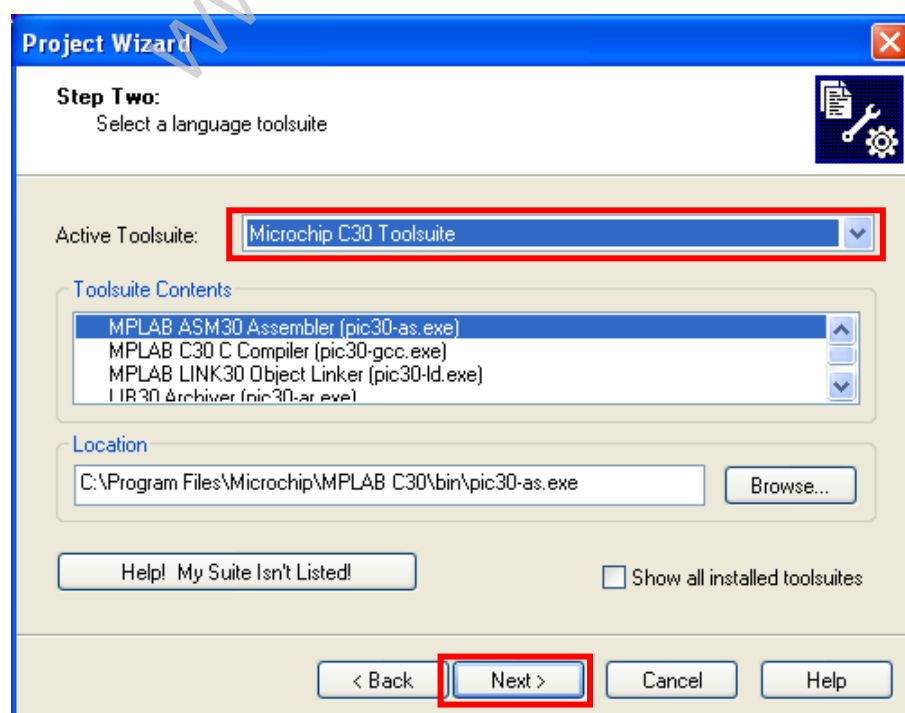
3. ทำการสร้าง Project File เพื่อใช้ส่งผนวกไฟล์ต่างๆ ที่เกี่ยวข้องเข้าด้วยกัน ทั้งนี้เนื่องมาจากว่า ภาษาซีของ MPLAB C30 นั้น ถูกออกแบบให้มีความอ่อนตัวในการทำงาน ดังนั้นจึงมีการจัดสร้าง และแบ่งแยกไฟล์ออกเป็นหลายๆไฟล์ตามหน้าที่การใช้งาน เพื่อให้ผู้ใช้สามารถเรียกไฟล์ต่างๆ เหล่านั้นเข้ามาใช้งานร่วมกับ Source Code ที่เขียนขึ้นมาได้ได้ง่าย โดยไม่ต้องเสียเวลาเขียน Source Code เองทั้งหมด ซึ่งจะทำให้ผู้ใช้สามารถลดเวลาในการเขียนโปรแกรมไปได้เป็นอย่างมากเนื่องจากเพียงแค่ทำการส่งผนวกไฟล์ที่ทาง MPLAB C30 สร้างเตรียมไว้ให้ เข้ากับ Source Code ที่ผู้ใช้เขียนขึ้นใหม่แล้วส่งแปลโปรแกรมก็จะได้ไฟล์ที่มีความสมบูรณ์ต่อการใช้งานแล้ว โดย จากตัวอย่าง Source Code ที่ได้ทดลองเขียนไปแล้วในข้างต้น ก็เช่นเดียวกัน จะเห็นได้ว่าการส่ง ผนวกไฟล์ชื่อ “p33FJ128GP708.h” เข้ามาใช้งานด้วย ซึ่งทำให้ไม่ต้องเสียเวลาไปส่งประกาศชื่อ และตำแหน่งรีจิสเตอร์ต่างๆของ “dsPIC33FJ128GP708” ให้เสียเวลา แต่สามารถอ้างถึงชื่อของ รีจิสเตอร์ต่างๆในโปรแกรมได้ทันที โดยวิธีการกำหนดคุณสมบัติของ Project File มีดังนี้
- สั่งกำหนดคุณสมบัติของ project File โดยใช้คำสั่ง “Project → Project Wizard...” ซึ่ง จะได้ผลดังรูป จากนั้นให้เลือก “Next >” เพื่อไปยังขั้นตอนต่อไป



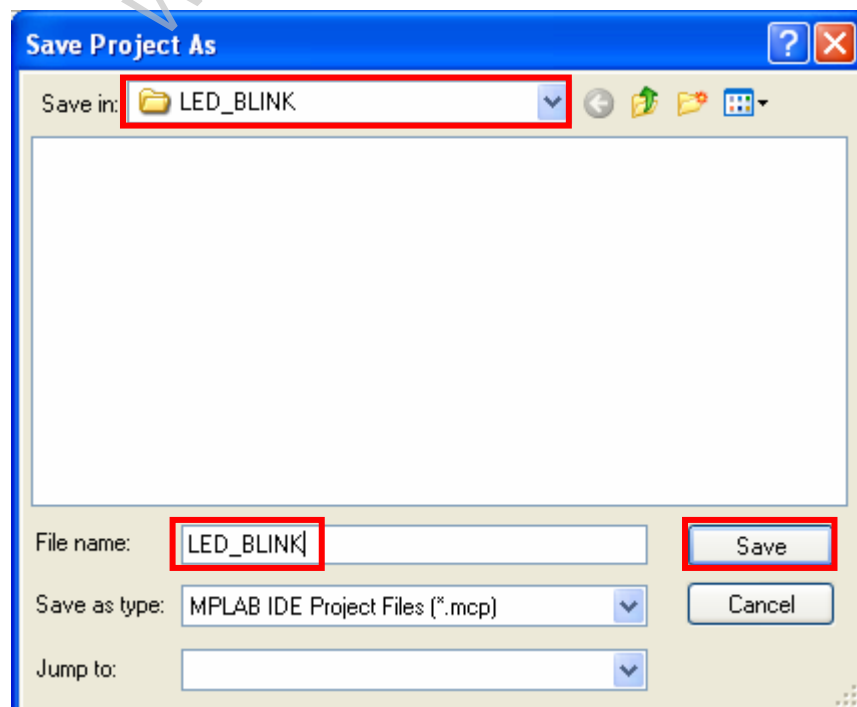
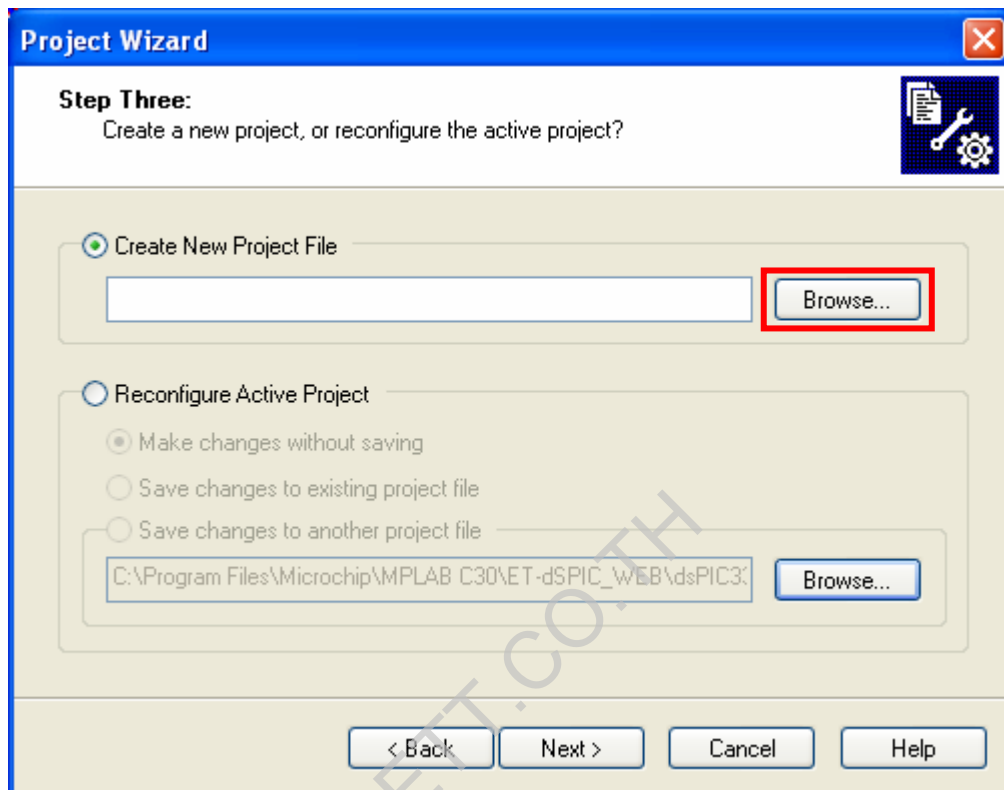
หลังจากเลือก “Next >” แล้ว โปรแกรมจะขอให้กำหนดเบอร์ของ MCU ที่จะใช้งานร่วมกับ โปรแกรมที่เขียนขึ้น ซึ่งให้เลือกกำหนดเป็น “dsPIC33FJ128GP708” จากนั้นเลือก “Next >” เพื่อข้ามไป ทำงานยังขั้นตอนต่อไปดังรูป



ในขั้นตอนนี้จะเป็นการเลือกที่จะใช้โปรแกรมชุดใดในการแปลคำสั่ง เนื่องจาก MPLAB IDE สามารถใช้งานได้กับชุดโปรแกรมต่างๆมากมายหลายโปรแกรม ซึ่งในที่นี้ให้เลือกกำหนดใช้โปรแกรม ของ MPLAB C30 โดยการเลือกกำหนดตัวเลือกของ “Active Toolsuite” ให้เป็นของ MPLAB C30 โดยกำหนด ตัวเลือกเป็น “Microchip C30 Toolsuite” ดังรูป แล้วเลือก “Next >”



ในขั้นตอนนี้จะเป็นการ กำหนดชื่อ Project และตำแหน่ง Folder ที่จะใช้เก็บไฟล์ต่างๆที่ได้จากการทำงานของ Project โดยให้กำหนดชื่อเป็น “LED_BLINK” ดังรูป แล้วเลือก “Next >”

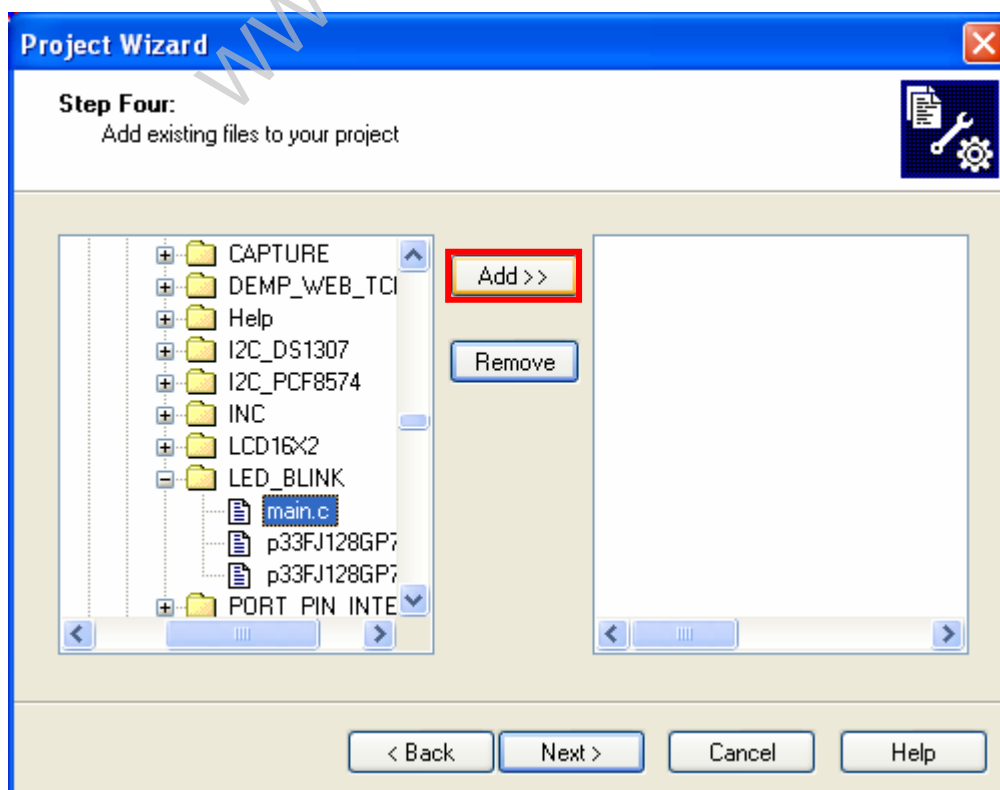


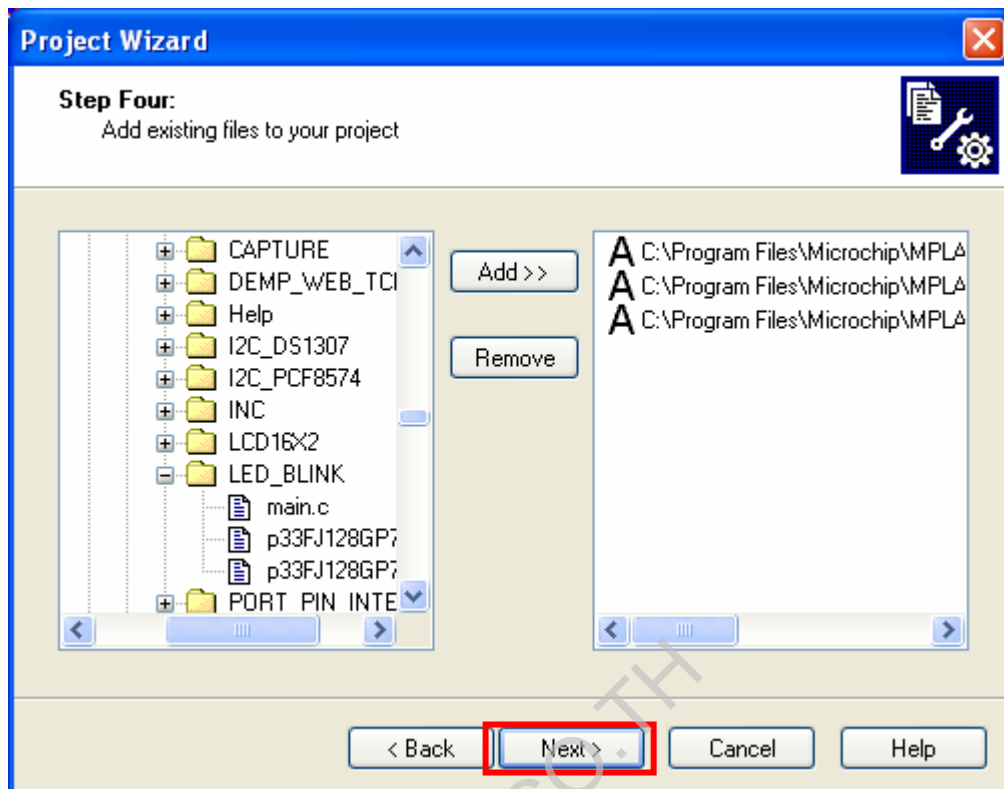
ในขั้นตอนนี้จะเป็นการส่งผนวกไฟล์ต่างๆเข้าไว้ด้วยกันภายใต้ชื่อ Project ของ LED_BLINK โดยให้ทำการส่งผนวกไฟล์ทั้งหมด 3 ไฟล์เข้าไว้ใน Project ดังนี้

- ส่งผนวกไฟล์ชื่อ “main.c” ซึ่งเป็น Source Code ที่เราได้เขียนและสั่งบันทึกไว้แล้วในขั้นตอนที่ผ่านมาโดยเก็บอยู่ใน ...\\LED_BLINK\\main.c”
- ส่งผนวกไฟล์ชื่อ “p33FJ128GP708.h” ซึ่งเป็น Header ของภาษาซี ในการอ้างถึงรีจิสเตอร์ต่างๆของ MCU เบอร์ “dsPIC33FJ128GP708” โดยถ้าติดตั้งโปรแกรม C30 ตามค่า Default ไฟล์ดังกล่าวจะเก็บอยู่ใน “...\\MPLAB C30\\support\\h\\”
- ส่งผนวกไฟล์ชื่อ “p33FJ128GP708.gld” ซึ่งเป็น Script File ของ dsPIC33FJ128GP708 ที่ทาง MPLAB C30 สร้างเตรียมไว้ให้ โดยถ้าติดตั้งโปรแกรมตาม C30 ตามค่า Default ไฟล์ดังกล่าวจะเก็บอยู่ใน “...\\MPLAB C30\\support\\gld\\”

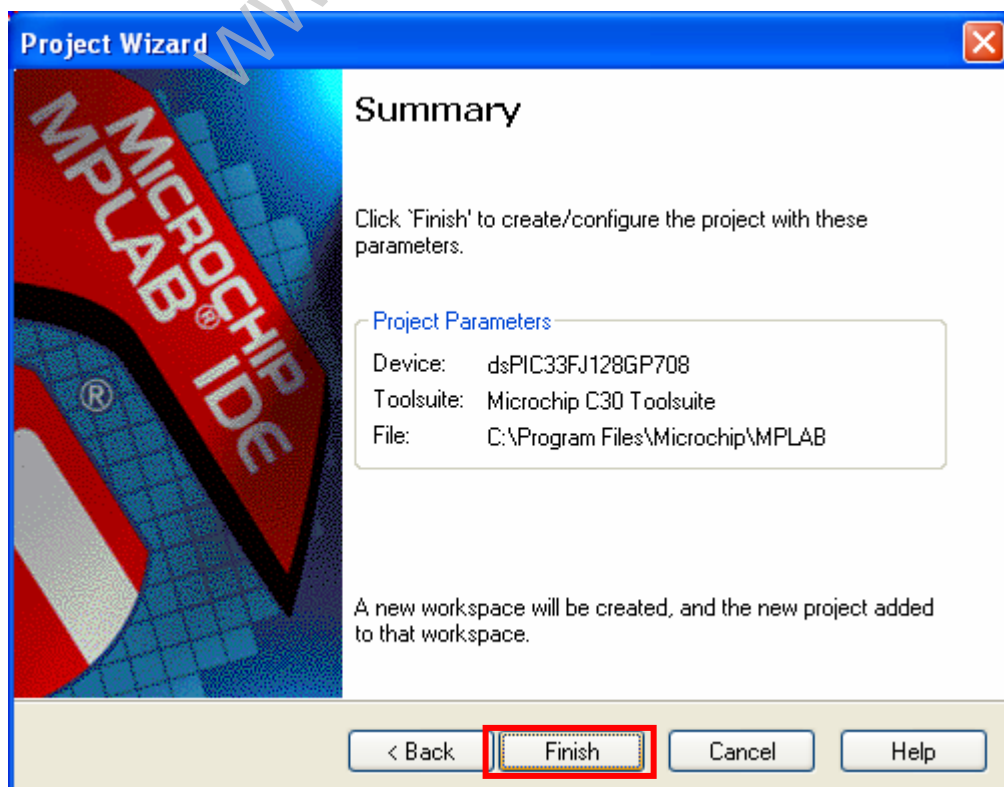
หมายเหตุ ค่า Default มาตรฐานของการติดตั้งโปรแกรมคอมไพเลอร์ C30 โดยปกติแล้วโปรแกรมจะถูกติดตั้งไว้ที่ “C:\\Program Files\\Microchip\\MPLAB C30”

โดยในการส่งผนวกไฟล์ทั้ง 3 ดังกล่าวให้ทำการคลิกเมาส์ไปยัง “ICON” ของไฟล์จากตำแหน่ง Folder ที่กล่าวไว้ในข้างต้นที่ผ่านมาแล้วเลือก “Add >>” จนชื่อไฟล์ดังกล่าวไปปรากฏอยู่ที่กรอบหน้าต่างด้านขวาของโปรแกรม ซึ่งให้เลือกทำตามวิธีการนี้จนสามารถส่ง “Add” ไฟล์ได้ครบทั้ง 3 ไฟล์ แล้วเลือก “Next >” เพื่อไปยังขั้นตอนต่อไปดังรูป

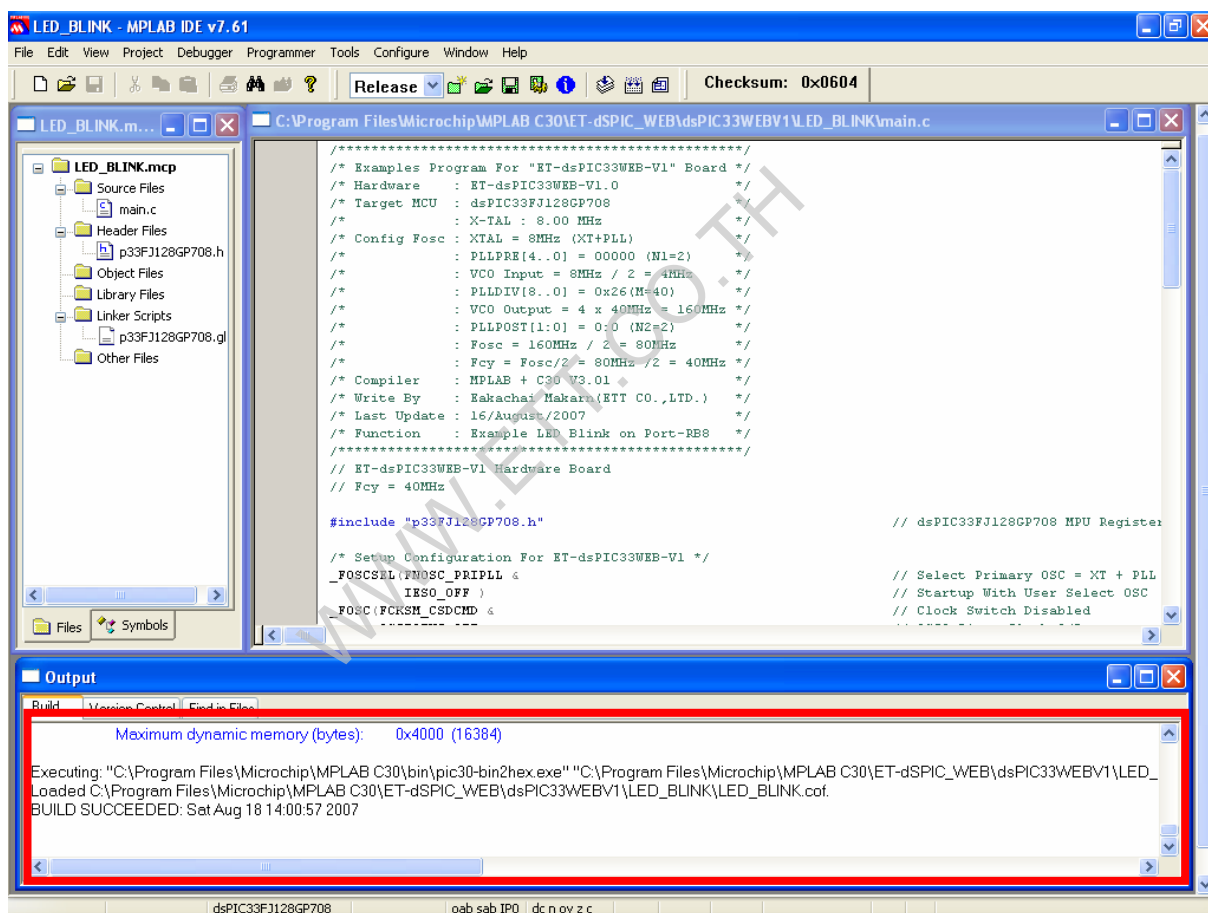




เมื่อสั่งผนวกไฟล์ทั้งหมดเข้ากับ Project ไฟล์ที่สร้างขึ้นเป็นที่เรียบร้อยแล้ว โปรแกรมจะรายงานผล โดยแสดงค่าพารามิเตอร์ต่างๆที่ได้กำหนดไว้แล้วให้ทราบดังรูป ให้เลือก “Finish” เป็นอันเสร็จขั้นตอนของการสร้าง Project File ของ BLINK_LED



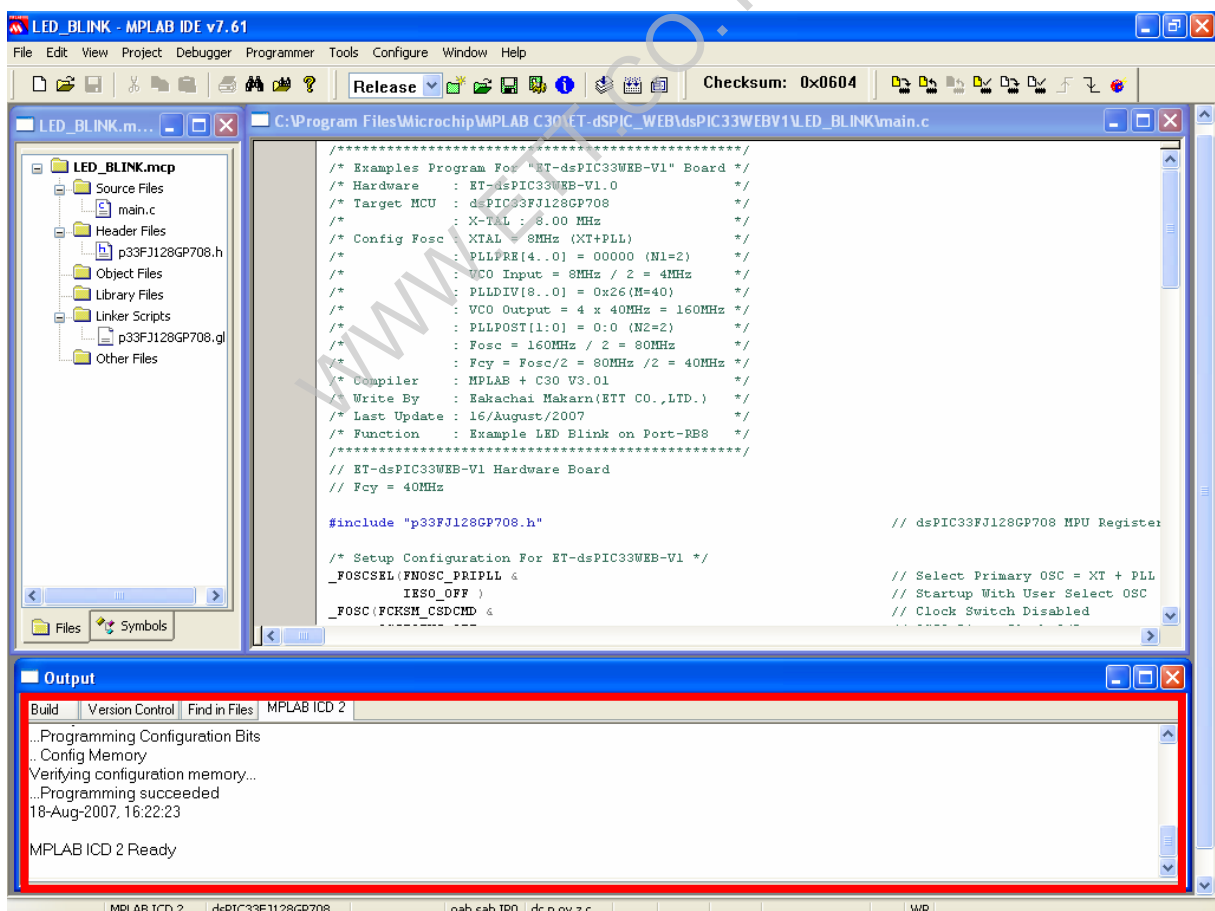
ซึ่งหลังจากกำหนดค่าต่างๆให้กับ Project File เป็นที่เรียบร้อยแล้ว ผู้ใช้ก็สามารถทำการสั่งแปลคำสั่งของโปรแกรม “main.c” ที่เขียนขึ้นได้ทันที โดยใช้คำสั่ง “Project → Build All” ซึ่งจะทำให้ได้ไฟล์ Output มีชื่อเดียวกับ Project File ที่สร้างไว้แต่มีนามสกุลเป็น HEX ซึ่งจากตัวอย่าง Project นี้เมื่อสั่งแปลโปรแกรมแล้วถ้าไม่เกิดข้อผิดพลาดใดๆจะได้ Output ไฟล์ชื่อ “LED_BLINK.HEX” โดยไฟล์ดังกล่าวจะถูกสร้างและเก็บไว้ในตำแหน่ง Folder ของ Project คือ “...\led_blink\led_blink.hex” โดยผู้ใช้สามารถสั่ง Download Hex File ชื่อ “LED_BLINK.HEX” นี้ให้กับบอร์ด “ET-dsPIC33WEB V1.0” เพื่อทดสอบการทำงานได้ทันที



การ Download Code ให้กับบอร์ด

หลังจากทำการเขียนโปรแกรมและสั่งคอมไพล์จนได้ HEX เรียบร้อยแล้ว ขั้นตอนการพัฒนาโปรแกรมของบอร์ดก็จะเหลือเพียงการ Download Code ให้กับ MCU ซึ่งในขั้นตอนนี้จะต้องใช้เครื่องมือในการ Download Code ร่วมด้วย ซึ่งอาจเป็น ICD2 หรือ Pickit2 ก็ได้

สำหรับในกรณีที่ใช้ ICD2 นั้นจะสามารถสั่ง Download Code ได้จากโปรแกรม MPLAB โดยตรงเลย โดยให้ทำการเชื่อมต่อสายสัญญาณจาก "ICD2" เข้ากับบอร์ด "ET-dsPIC33WEB V1.0" แล้วเลือกสวิตช์ Mode ให้อยู่ในตำแหน่ง PGM (สวิตช์อยู่ในตำแหน่งถูกกด LED PGM สีแดงติดสว่าง) จากนั้นเลือกกำหนดการเชื่อมต่อกับ ICD2 โดยการคลิกเมาส์ที่เมนูคำสั่ง "Programmer → Select Program → MPLAB ICD 2" จากนั้นให้สั่ง Download Code โดยคลิกเมาส์ที่เมนูคำสั่ง "Programmer → Program" ซึ่งเครื่องก็จะทำการ Download Code ให้เองโดยอัตโนมัติ จนเมื่อเสร็จเรียบร้อยแล้ว จะมีข้อความแจ้งให้ทราบคือ "MPLAB ICD2 Ready" ดังรูป

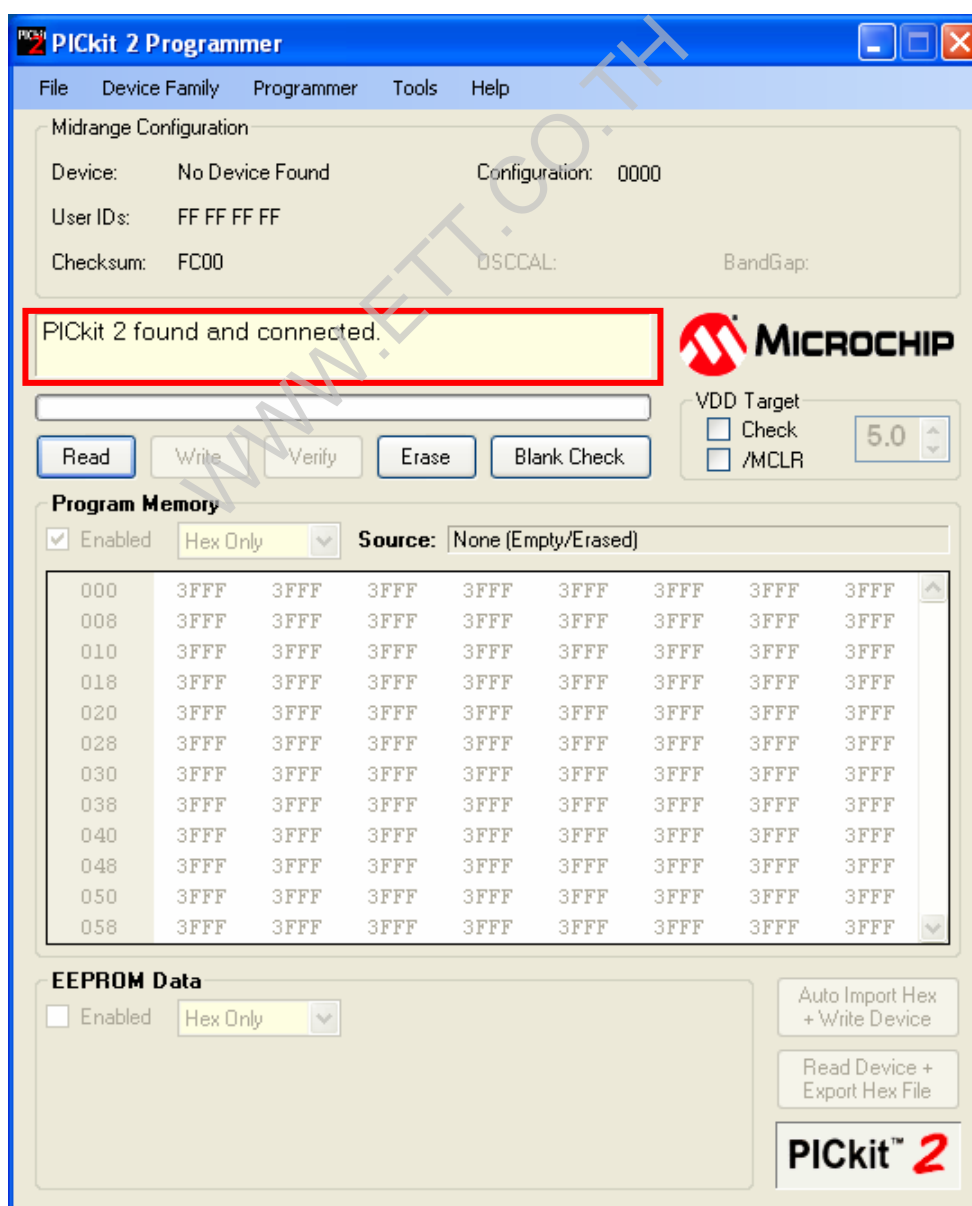


ซึ่งหลังจากทำการ Download Code เรียบร้อยแล้วให้เลือกสวิตช์ Mode ไปไว้ในตำแหน่ง Run โดยให้ตำแหน่งของสวิตช์ Mode อยู่ในตำแหน่งปล่อย และ LED Run สีเขียวติดสว่าง จากนั้นให้กดสวิตช์ Reset 1 ครั้ง บอร์ดก็จะเริ่มต้นทำงานตามคำสั่งที่ Download ให้แล้วในทันที

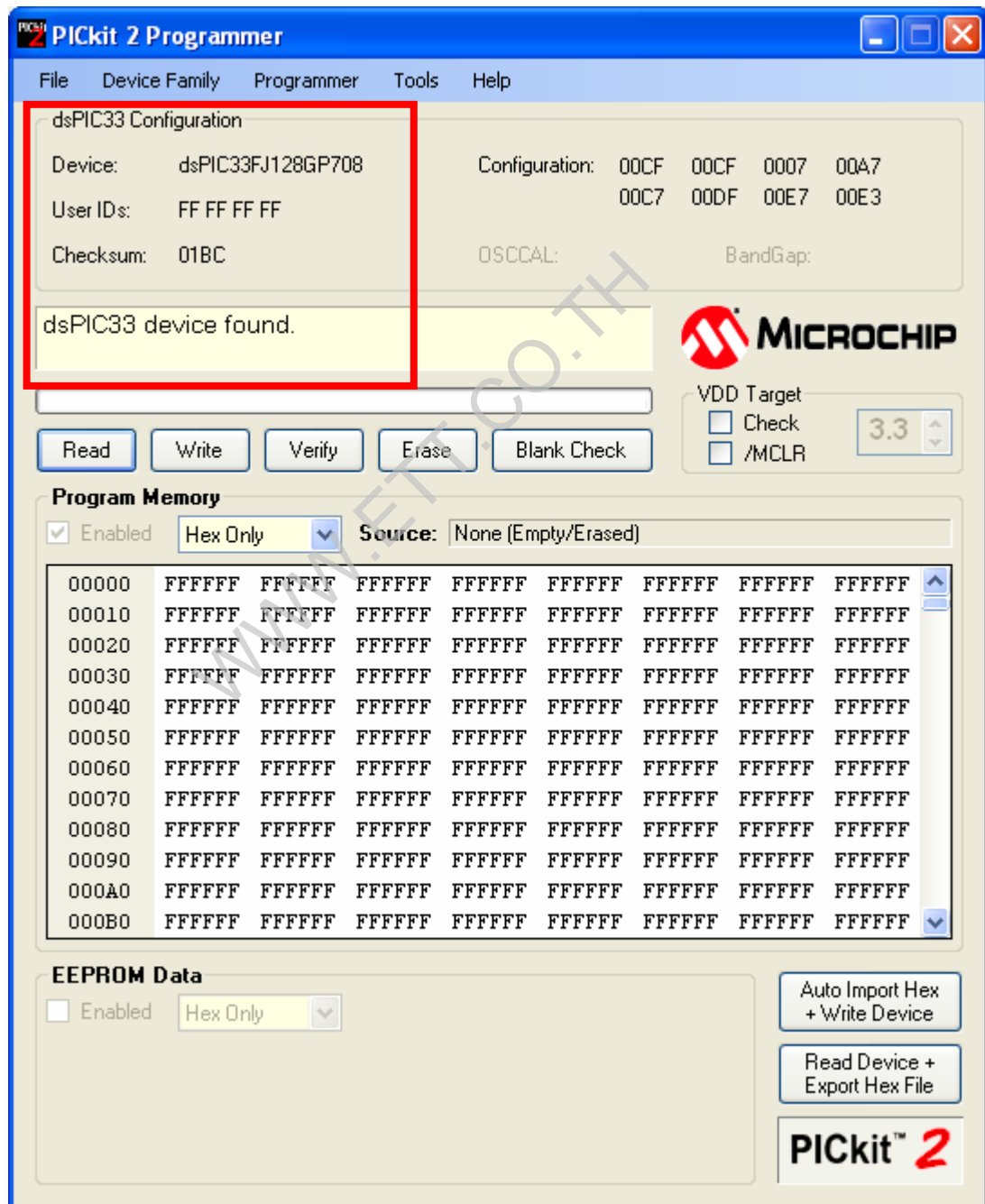
การ Download Code ด้วย “ET-PGMPIC USB”

สำหรับในกรณีที่ผู้ใช้เลือกใช้เครื่องโปรแกรม Pickit2 หรือ ET-PGMPIC USB นั้น ในปัจจุบัน โปรแกรม MPLAB ยังไม่ได้สร้างการเชื่อมต่อคำสั่งกับเครื่องโปรแกรมห้ไว้ให้ ดังนั้นผู้ใช้จึงไม่สามารถสั่ง Download Code ผ่านทางโปรแกรม MPLAB ได้เหมือน ICD2 แต่จะต้องสลับไปให้โปรแกรม Pickit2 เพื่อสั่ง Download Code เองดังนี้

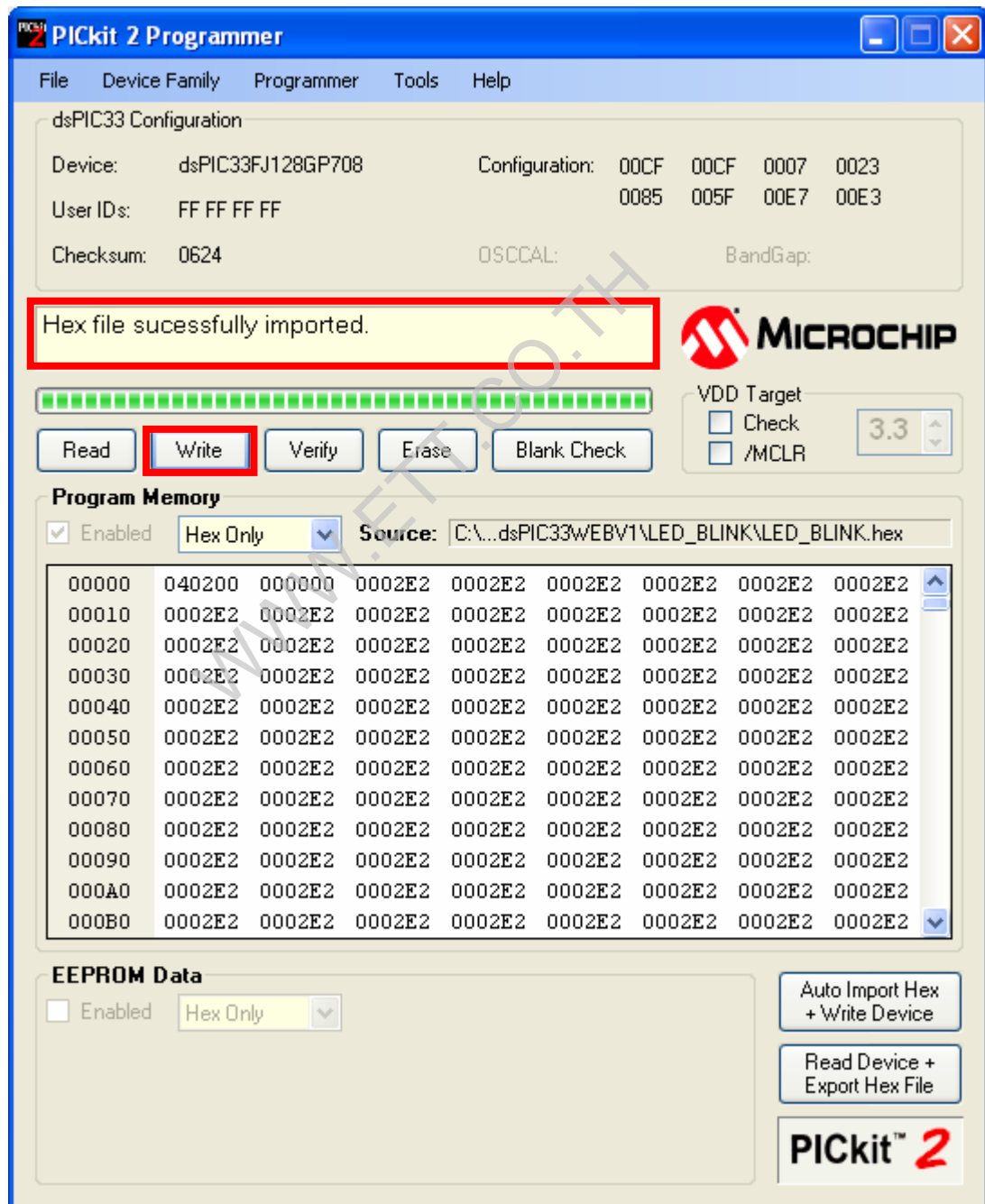
1. ต่อสายสัญญาณ ICD2 จาก ET-PGMPIC USB เข้ากับบอร์ด ET-dsPIC33WEB V1.0 โดยให้เลือกกำหนด Jumper “B/T” ของ ET-PGMPIC USB ไว้ในตำแหน่ง “B” (Target Board) พร้อมกับกำหนดสวิตซ์เลือก Mode ของบอร์ด “ET-dsPIC33WEB V1.0” ไว้ในตำแหน่ง PGM (LED PGM สีแดงติดสว่าง)
2. สั่ง Run โปรแกรม Pickit2 ถ้าทุกอย่างถูกต้องจะได้ผลดังรูป



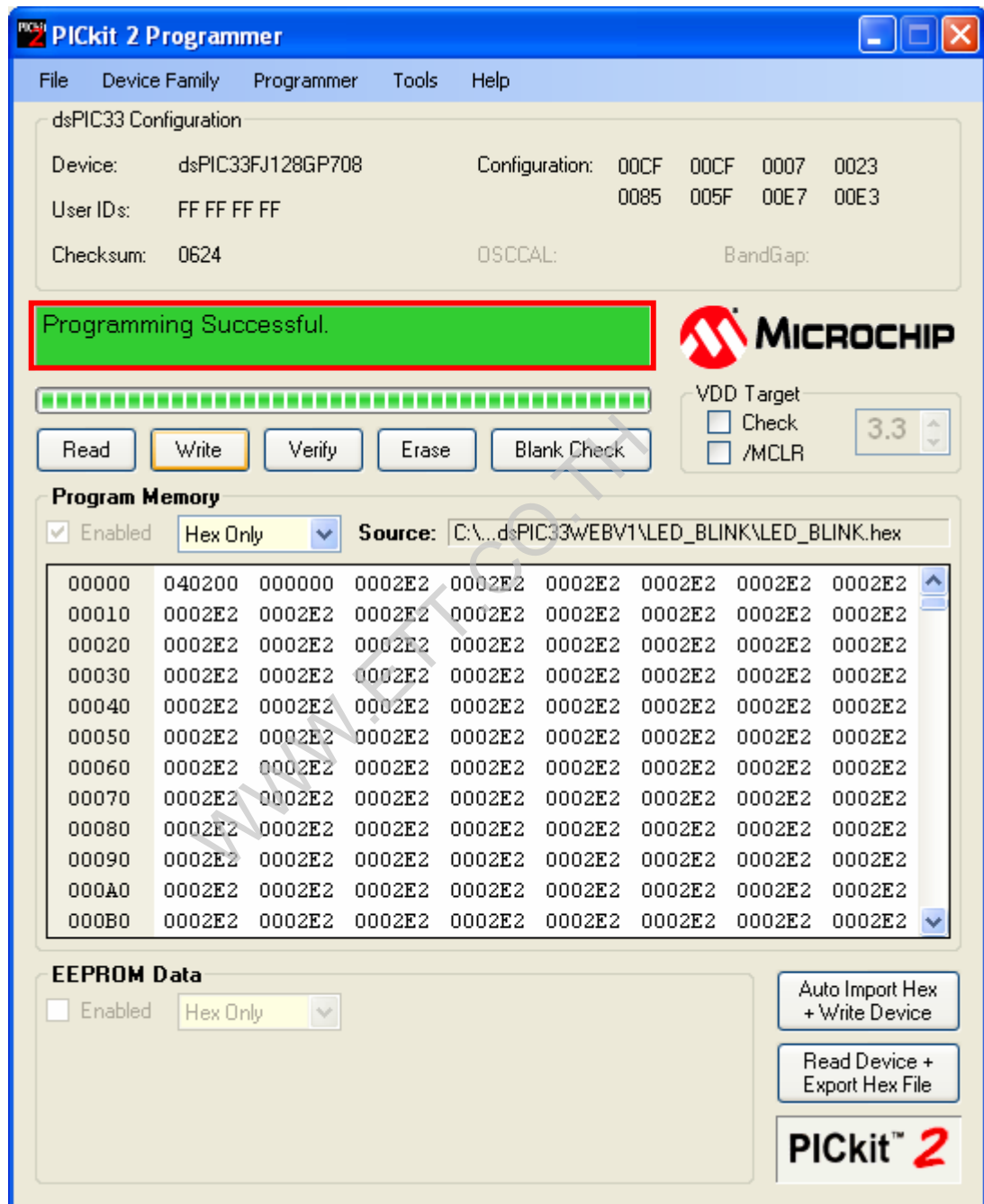
3. ทำการคลิกเมาส์ที่เมนูคำสั่ง “Device Family → dsPIC33” ซึ่งเครื่องโปรแกรมจะทำการเชื่อมต่อกับ MCU พร้อมกับอ่านค่า Configuration ต่างๆของ MCU ขึ้นมาแสดงผลให้เห็นที่หน้าจอของโปรแกรม ให้เห็นในทันที ซึ่งในกรณีของบอร์ด “ET-dsPIC33WEB V1.0” นั้น ถ้าทุกอย่างถูกต้องที่หน้าจอของโปรแกรมจะแสดงข้อความ “dsPIC33 device found” พร้อมกับมีการแสดงเบอร์ของ MCU (Device) เป็น “dsPIC33FJ128GP708” ดังรูป



4. ให้ทำการสั่ง Load HEX File ที่ได้จากการแปลโปรแกรมของ C30 โดยให้ทำการคลิกเมาส์ที่เมนูคำสั่ง “File → Import Hex” แล้วเลือก HEX File ที่ได้จากจากแปลคำสั่งของ C30 ซึ่งถ้าทุกอย่างถูกต้อง จะปรากฏข้อความ “Hex file successfully imported” ซึ่งหมายถึงการสั่ง Load Hex ไฟล์เรียบร้อยแล้ว ซึ่งผู้ใช้สามารถสั่ง Download Code จาก Hex File ให้กับหน่วยความจำของ MCU ได้ทันที โดยการคลิกเมาส์ที่ปุ่มคำสั่ง “Write” ดังรูป



5. เครื่อง “ET-PGM PIC USB” จะเริ่มต้นทำการ Download Code ให้กับ MCU ในทันที เมื่อการ Download Code เสร็จเรียบร้อยแล้ว จะมีข้อความ “Programming Successful” ดังรูป



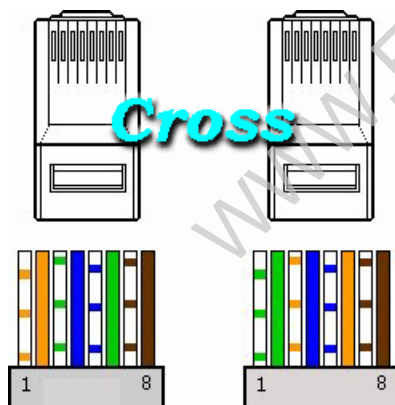
6. หลังจากทำการ Download Code เรียบร้อยแล้ว ให้ทำการเลือกกำหนดการทำงานของสวิตช์ Mode ของบอร์ด ET-dsPIC33WEB V1.0 ไปไว้ในตำแหน่ง Run โดยให้ตำแหน่งของสวิตช์ Mode อยู่ในตำแหน่งปล่อย และ LED Run สีเขียวติดสว่าง จากนั้นให้กดสวิตช์ Reset 1 ครั้ง บอร์ดก็จะเริ่มต้นทำงานตามคำสั่งที่ Download ให้แล้วในทันที

ตัวอย่างการพัฒนาโปรแกรม Web Server Control ของ ET-dsPIC33WEB V1.0

ความต้องการทางด้านฮาร์ดแวร์ (Hardware Requirements)

อุปกรณ์ที่ใช้

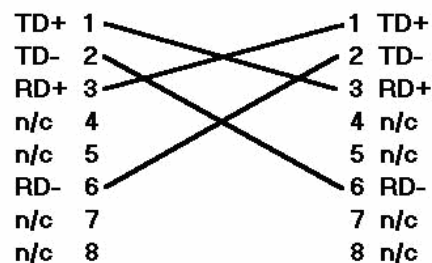
- บอร์ดไมโครคอนโทรลเลอร์ ET-dsPIC33WEB V1.0
- โมดูล ET-MINI ENC28J60
- สายเชื่อมต่อวงจร อย่างน้อย 13 เส้น
- จอแสดงผล Character LCD ขนาด 16 ตัวอักษร 2 บรรทัด
- โมดูล EEPROM 25LC256
- เครื่องโปรแกรม PIC/dsPIC รุ่น ET-PGM PIC USB หรือรุ่นอื่นๆ พร้อมสายเชื่อมต่อ USB
- แหล่งจ่ายไฟสำหรับบอร์ด ET-dsPIC33WEB V1.0 (7-12V/850mA)
- เครื่องคอมพิวเตอร์ ซึ่งมีพอร์ตเชื่อมต่อ Ethernet LAN
- สายเคเบิลของระบบแลนค์ ซึ่งมีอยู่สองกรณี คือ
 - กรณีที่ 1 คือ การเชื่อมต่อเข้ากับคอมพิวเตอร์โดยตรง สาย LAN จะต้องเข้าสายแบบ Cross



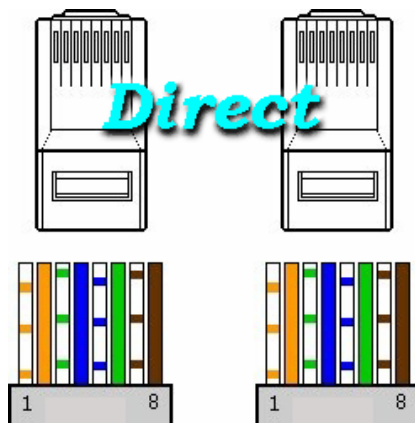
10BaseT cross-cable diagram

RJ-45 plug

RJ-45 jack



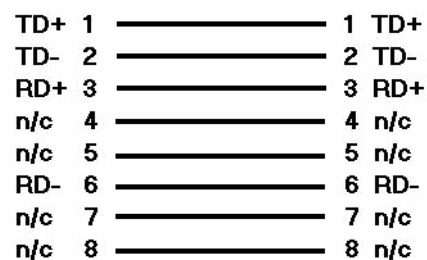
- กรณีที่ 2 คือ การเชื่อมต่อผ่าน Hub ของเครื่องคอมพิวเตอร์ Server จะต้องเข้าสายแบบ Direct



10BaseT cross-cable diagram

RJ-45 plug

RJ-45 jack



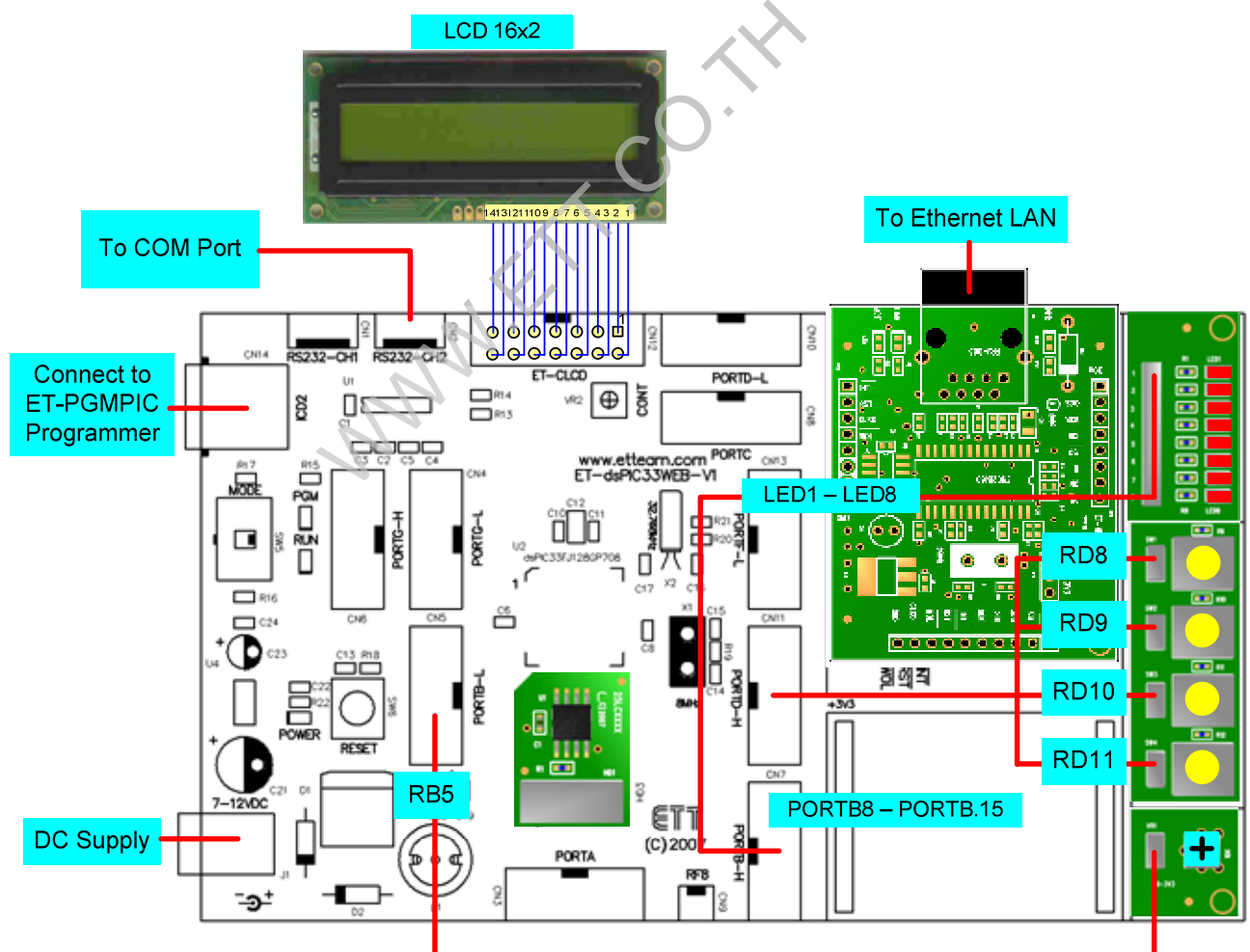
ความต้องการทางด้านซอฟต์แวร์ (Software Requirements)

- ติดตั้งโปรแกรม MPLAB IDE v7.61 (เวอร์ชันล่าสุด ณ ปัจจุบัน กรกฎาคม 2550)
- โปรแกรมคอมไพเลอร์ C30 เวอร์ชัน 3.01 Student Edition
- โปรแกรม TCPIP Stack 4.02 (เวอร์ชันล่าสุด ณ ปัจจุบัน กรกฎาคม 2550)
- โปรแกรม รับส่งข้อมูลอนุกรม RS232 เช่น Procomm หรือ HyperTerminal ของวินโดวส์
- โปรแกรม Internet Explorer หรือ โปรแกรมที่เปิด Webpage ทั่วไป

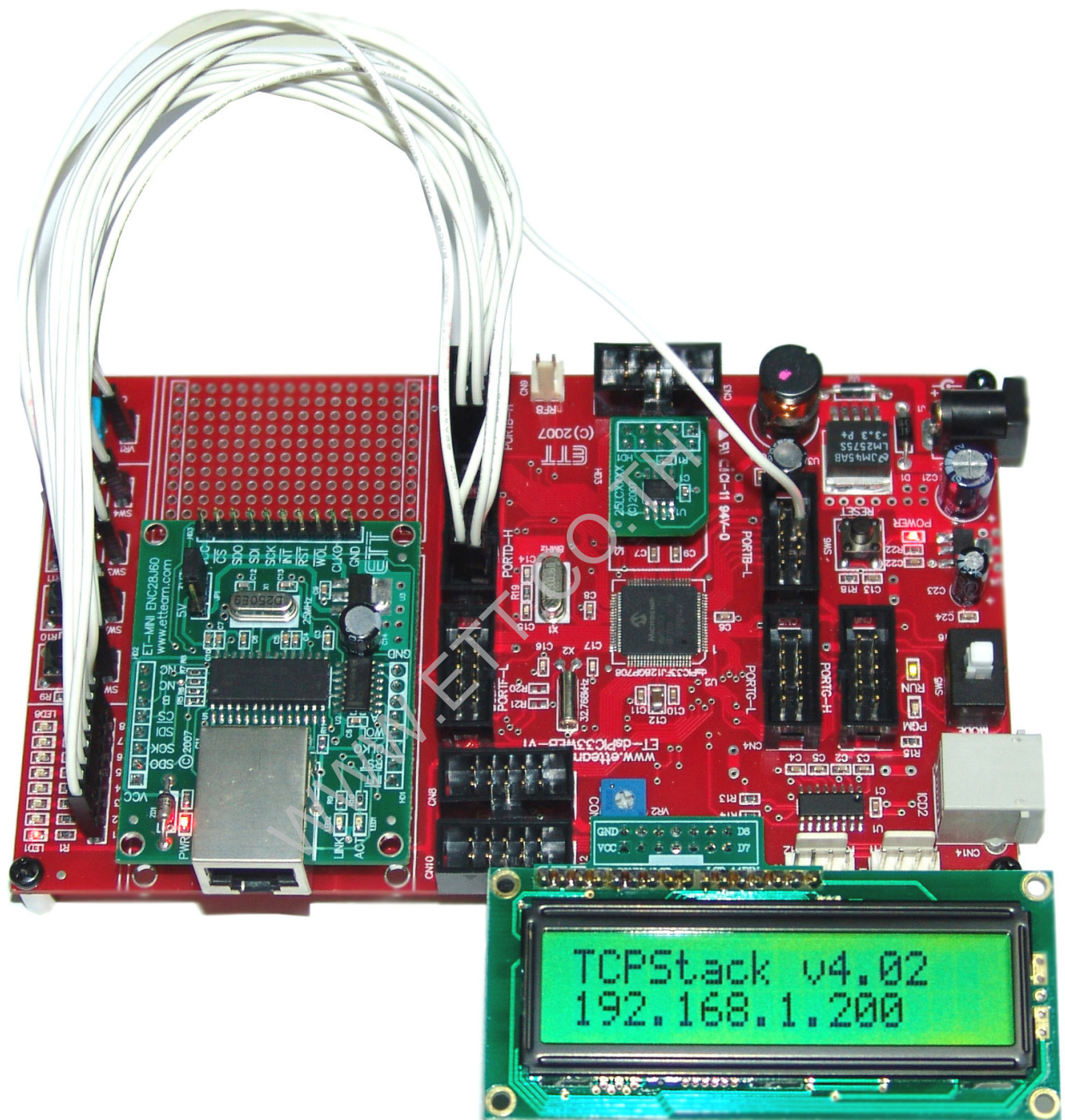
*** ให้ทำการติดตั้งโปรแกรมเหล่านี้ให้เรียบร้อย

ขั้นตอนการทดลอง

1. ต่อวงจรให้เรียบร้อยดังนี้



รูปแสดงการเชื่อมต่อสัญญาณต่างๆ เพื่อทำการทดลอง



รูปแสดงบอร์ดทดลองหลังจากเชื่อมต่อสัญญาณต่างๆ

สัญญาณการเชื่อมต่อ

สำหรับตัวอย่าง Code Program ที่เป็น Web Server Control ที่ทาง อีทีที จัดทำขึ้น เพื่อเป็นแนวทาง และ ตัวอย่างการใช้งาน เพื่อให้ผู้ใช้นำไปทดสอบการทำงาน และแก้ไขพัฒนาตามความต้องการนั้น จะเป็นการปรับปรุง มาจากตัวอย่าง Code ของ Microchips โดยทางอีทีที ได้นำ Source Code ของ TCP/IP Stack V4.02 จากทาง Microchips มาปรับปรุงแก้ไขให้สอดคล้องกับ I/O ต่างๆของบอร์ด ET-dsPIC33WEB V1.0 ซึ่งได้มีการปรับปรุง และแก้ไขดัดแปลง Code เฉพาะในส่วนของการ Webpage และ HTML ส่วนของการ Process I/O Function เท่านั้น ส่วนรายละเอียดการใช้งานฟังก์ชันต่างๆที่เป็นของ TCP/IP Stack นั้นจะยึดตามต้นฉบับของ Microchips เป็นหลัก ซึ่งผู้ใช้สามารถศึกษารายละเอียดต่างๆเกี่ยวกับการใช้งาน ฟังก์ชันต่างๆของ TCP/IP Stack ได้จากเอกสารที่ทาง Microchips จัดทำไว้ให้ได้เลย โดยจะไม่ขอกล่าวถึงในที่นี้ด้วย

โดยในตัวอย่างทาง อีทีที จะจัดทำเป็นลักษณะของการประยุกต์ใช้งาน Web Server Control ขนาดเล็ก โดยแสดงให้เห็นวิธีการรับส่งคำสั่งจากหน้า Webpage สำหรับควบคุมการทำงานของอุปกรณ์ I/O ต่างๆ ที่อยู่ใน บอร์ด ET-dsPIC33WEB V1.0 ทั้ง Digital Output (LED), Digital Input (SW), Analog Input (ADC) และ LCD Display โดยในการทดลองตามตัวอย่างโปรแกรม ให้ทำการเชื่อมต่อสัญญาณ I/O ของ MCU กับวงจร I/O ต่างๆ ตามรายละเอียดดังต่อไปนี้

- **Ethernet Module** สำหรับใช้ในการเชื่อมต่อกับ Ethernet LAN จะใช้โมดูล ET-MINI ENC28J60 ซึ่งสามารถรับส่งข้อมูลได้ด้วยความเร็วสูงสุด 10MB โดยฟังก์ชันในการเชื่อมต่อจะใช้สัญญาณ SPI1 ของ MCU โดยใช้สัญญาณต่างๆดังนี้
 - RF6 ใช้เป็น SCK ในการเชื่อมต่อกับ Ethernet Module (ET-MINI ENC28J60)
 - RF7 ใช้เป็น SDI ในการเชื่อมต่อกับ Ethernet Module (ET-MINI ENC28J60)
 - RF8 ใช้เป็น SDO ในการเชื่อมต่อกับ Ethernet Module (ET-MINI ENC28J60)
 - RD14 ใช้เป็น CS ในการเชื่อมต่อกับ Ethernet Module (ET-MINI ENC28J60)
 - RA12 ใช้เป็น INT1 ในการเชื่อมต่อกับ Ethernet Module (ET-MINI ENC28J60) ซึ่งในตัวอย่างนี้จะไม่ได้ใช้งานให้กำหนด Jumper INT(EN/DS) เป็น DS(Disable:ไม่ใช้งาน)
 - RA13 ใช้เป็น WOL ในการเชื่อมต่อกับ Ethernet Module (ET-MINI ENC28J60) ซึ่งในตัวอย่างนี้จะไม่ได้ใช้งานให้กำหนด Jumper WOL(EN/DS) เป็น DS(Disable:ไม่ใช้งาน)
 - RD15 ใช้เป็น RST ในการเชื่อมต่อกับ Ethernet Module (ET-MINI ENC28J60) ซึ่งในตัวอย่างนี้จะไม่ได้ใช้งานให้กำหนด Jumper RST(EN/DS) เป็น DS(Disable:ไม่ใช้งาน)
- **SPI Memory Module** สำหรับใช้เก็บ Webpage File และ CGI Scrip ต่างๆ จะใช้โมดูล 25LC256 โดยฟังก์ชันการเชื่อมต่อจะใช้สัญญาณ SPI2 ของ MCU โดยใช้สัญญาณต่างๆดังนี้
 - RG6 ใช้เป็น SCK ในการเชื่อมต่อกับ SPI Memory
 - RG7 ใช้เป็น SDI ในการเชื่อมต่อกับ SPI Memory
 - RG8 ใช้เป็น SDO ในการเชื่อมต่อกับ SPI Memory
 - RD12 ใช้เป็น CS ในการเชื่อมต่อกับ SPI Memory

- I/O Control Signal สำหรับใช้ควบคุมการทำงานและรับค่า Input มาแสดงผลที่หน้า Webpage โดยจะมีอยู่ด้วยกัน 4 กลุ่ม คือ
 - 8 Bit Digital Output โดยใช้ LED เป็นตัวแสดงสถานะของการทำงาน
 - 4 Bit Digital Input โดยใช้ Push-Button Switch เป็น Input
 - 1 ช่อง Analog Input ขนาด 12 Bit โดยใช้ตัวต้านทานปรับค่าได้เป็น Input
 - Character Display LCD 16x2 สำหรับแสดงผลข้อความต่างๆ

dsPIC33FJ128GP708	LED
RB8	LED1
RB9	LED2
RB10	LED3
RB11	LED4
RB12	LED5
RB13	LED6
RB14	LED7
RB15	LED8

dsPIC33FJ128GP708	SW
RD8	SW1
RD9	SW2
RD10	SW3
RD11	SW4

dsPIC33FJ128GP708	VR1 (0-3V3)
RB5	VR1

dsPIC33FJ128GP708	LCD
RD0	D4
RD1	D5
RD2	D6
RD3	D7
RD4	RS
RD5	RW
RD6	EN

การกำหนดค่า Configuration ให้บอร์ด ET-dsPIC33WEB V1.0

ตามปกติแล้วค่า Configuration ต่างๆของบอร์ด จะถูกกำหนดไว้แล้วในส่วน Source Code ไม่ว่าจะเป็นค่า IP Address, DHCP Name หรืออื่นๆ ซึ่งค่า Configuration ตามค่า Default ที่กำหนดไว้ภายใน Code นั้นตามตัวอย่างที่ทาง อีทีที จัดทำขึ้นจะเป็นดังนี้

- Device IP Address คือ 192.168.1.200
- Gateway IP Address คือ 192.168.1.255
- Subnet mask คือ 255.255.255.0
- DNS Server IP Address คือ 192.168.1.255
- DHCP Protocol = Enable ,DHCP name = dspic33web

แต่อย่างไรก็ตามผู้ใช้สามารถเข้าไปแก้ไขค่า Configuration ต่างๆได้เอง โดยไม่จำเป็นต้องกลับไปแก้ไขที่ Source Code ก็ได้ ซึ่งการแก้ไขจะทำผ่านพอร์ตสื่อสารอนุกรม RS232 ช่อง 2 (RS232-CH2) โดยใช้โปรแกรมจำลอง Serial Terminal ต่างๆ เช่น Procomm หรือ Hyper Terminal ก็ได้ โดยให้ส่งเชื่อมต่อด้วยเงื่อนไขดังนี้

Baudrate = 19200BPS, Data Bit = 8 Bit, Parity Bit = None, Stop Bit = 1 Bit, Flow Control = None

ซึ่งหลังจากทำการเปิดโปรแกรม Hyper Terminal และกำหนดการเชื่อมต่อเรียบร้อยแล้ว พร้อมกับต่อสายสัญญาณจาก RS232-CH2 ของบอร์ด ET-dsPIC33WEB V1.0 กับ Comport PC เป็นที่เรียบร้อยแล้ว ให้ทำการกดสวิตช์ SW1 ที่ต่อกับ Port-RD8 ค้างไว้ พร้อมกับกดสวิตช์ RESET ของบอร์ดแล้วปล่อย โดยที่ SW1 ยังกดค้างอยู่ โดยโปรแกรมมอนิเตอร์จะตรวจสอบว่า มีการกด SW1 ในตอนเริ่มต้นการทำงานหรือไม่ ซึ่งถ้าพบว่ามี การกดสวิตช์ค้างไว้เป็นเวลานานอย่างน้อย 4 วินาที ก็จะเข้าสู่โหมดการ Setup Configuration ของบอร์ด แต่ถ้าไม่พบการกดสวิตช์ใดๆก็จะข้ามไปทำงานตามปกติ โดยถ้ามีการกด SW1 ค้างไว้ตอนเริ่มต้นการทำงาน(หลังรีเซ็ต) จะมีการส่งข้อความและเมนูคำสั่งกลับมาทางพอร์ต RS232-CH2 ซึ่งผู้ใช้สามารถกดคีย์ เลือกตัวเลข ตามหัวข้อต่างๆ เพื่อทำการเปลี่ยนแปลงแก้ไขค่า Configuration ต่างๆได้ตามต้องการ ดังตัวอย่าง

BUTTON0 held for more than 4 seconds. EEPROM contents erased.

Microchip TCP/IP Config Application (v4.02, Aug 22 2007)

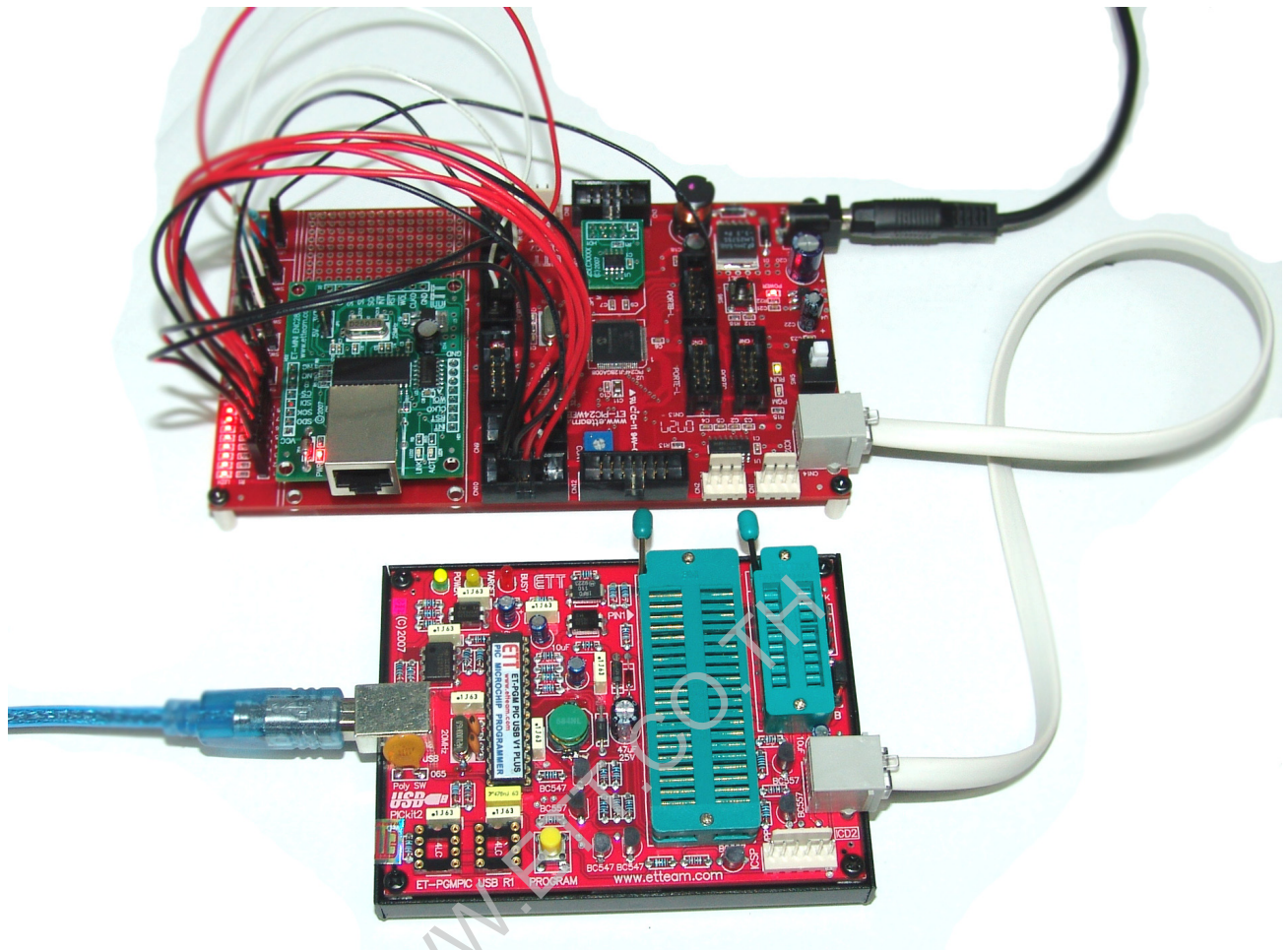
```

1: Change serial number:          0
2: Change host name:              DSPIC33WEB
3: Change static IP address:      192.168.1.200
4: Change static gateway address: 192.168.1.255
5: Change static subnet mask:     255.255.255.0
6: Change static primary DNS server: 192.168.1.255
7: Change static secondary DNS server: 192.168.1.255
8: Disable DHCP & IP Gleaning:    DHCP is currently enabled
9: Download MPFS image.
0: Save & Quit.
```

Enter a menu choice:

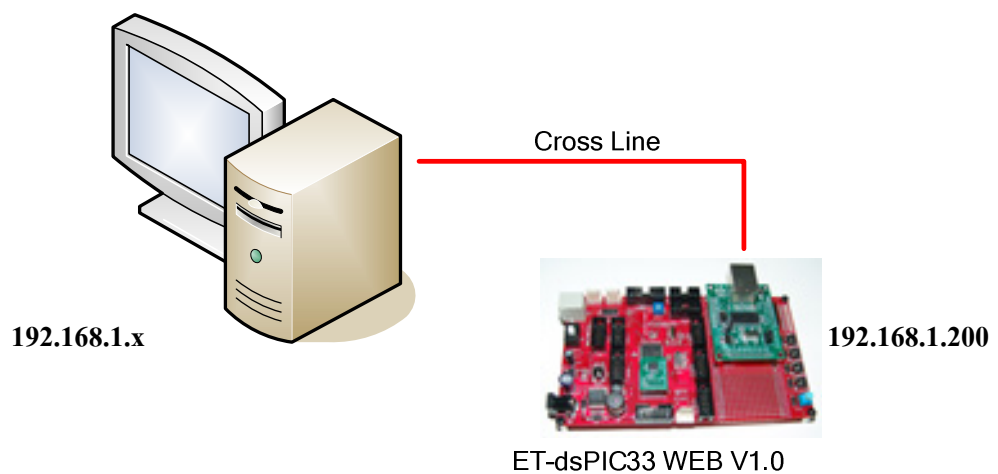
ขั้นตอนการทดสอบ Web Server Control

1. ทำการติดตั้ง โมดูล และ เชื่อมต่อสัญญาณให้กับอุปกรณ์ I/O ต่างๆดังกล่าวข้างต้นให้เรียบร้อย ดังตัวอย่าง

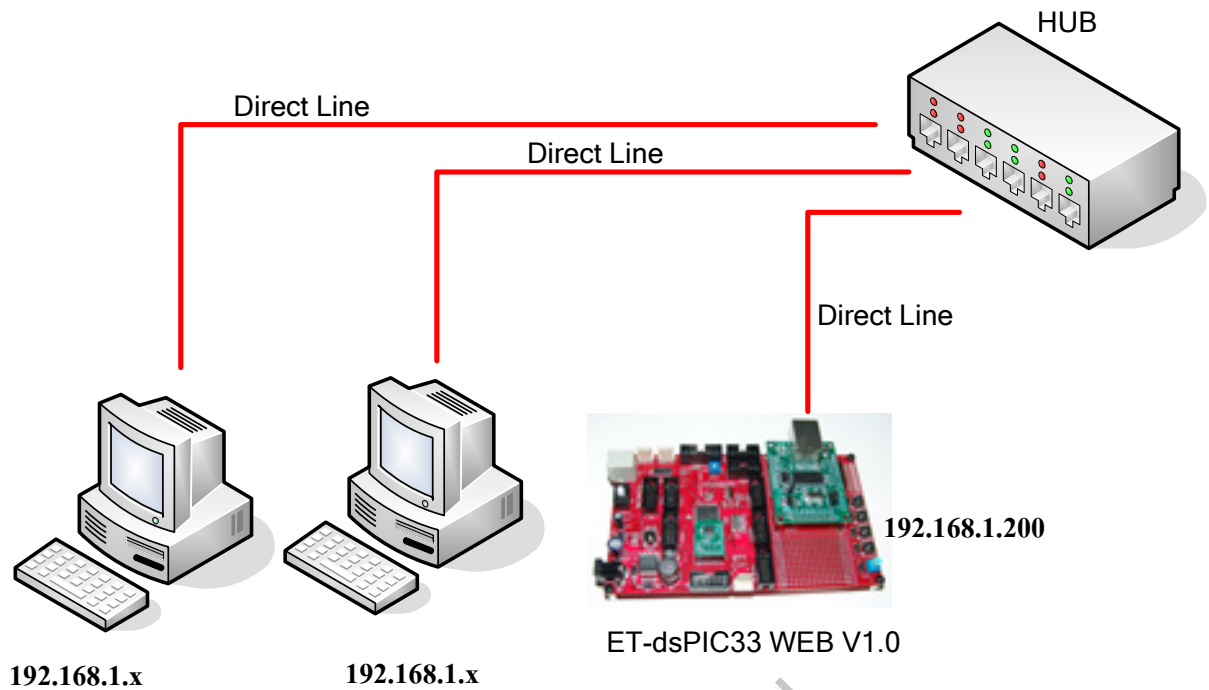


รูปแสดงตัวอย่างการเชื่อมต่อฮาร์ดแวร์

2. เชื่อมต่อสาย LAN ระหว่างบอร์ด เข้ากับระบบเครือข่าย LAN โดย กรณีที่เชื่อมต่อผ่าน HUB จะต้องใช้สาย แบบ Direct และ กรณีที่เชื่อมต่อเข้ากับการ์ด LAN ของเครื่องคอมพิวเตอร์จะต้องใช้สายแบบ Cross ดังรูปต่อไปนี้

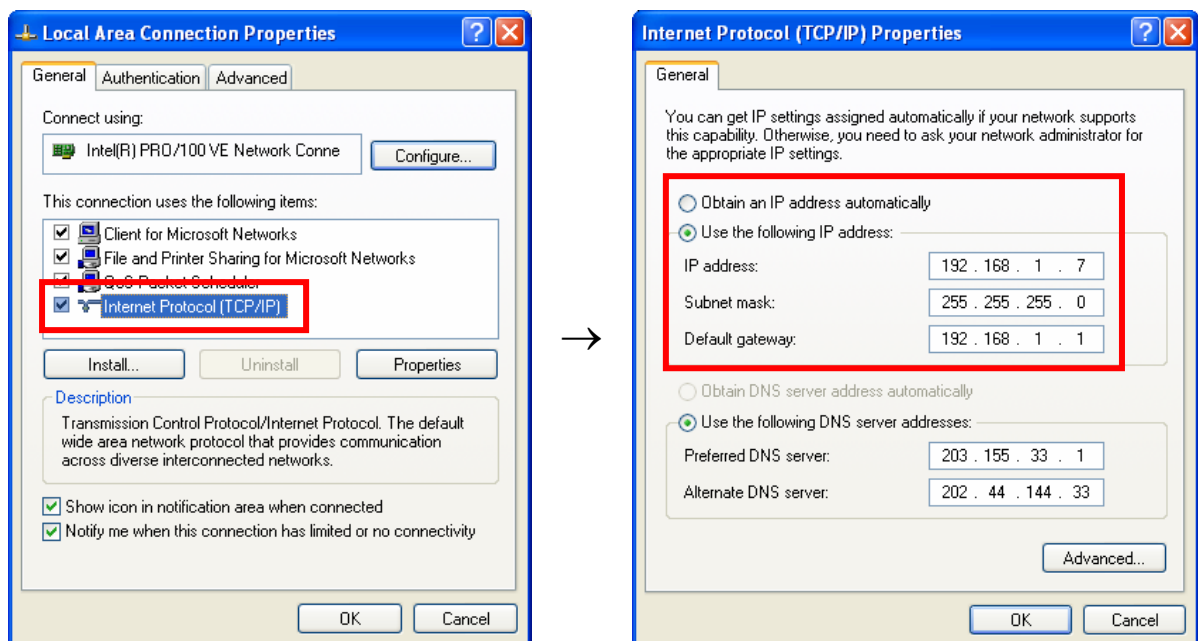


รูปแสดงการเชื่อมต่อ ET-dsPIC33WEB เข้ากับการ์ด LAN ของคอมพิวเตอร์ใช้สายแบบ Cross



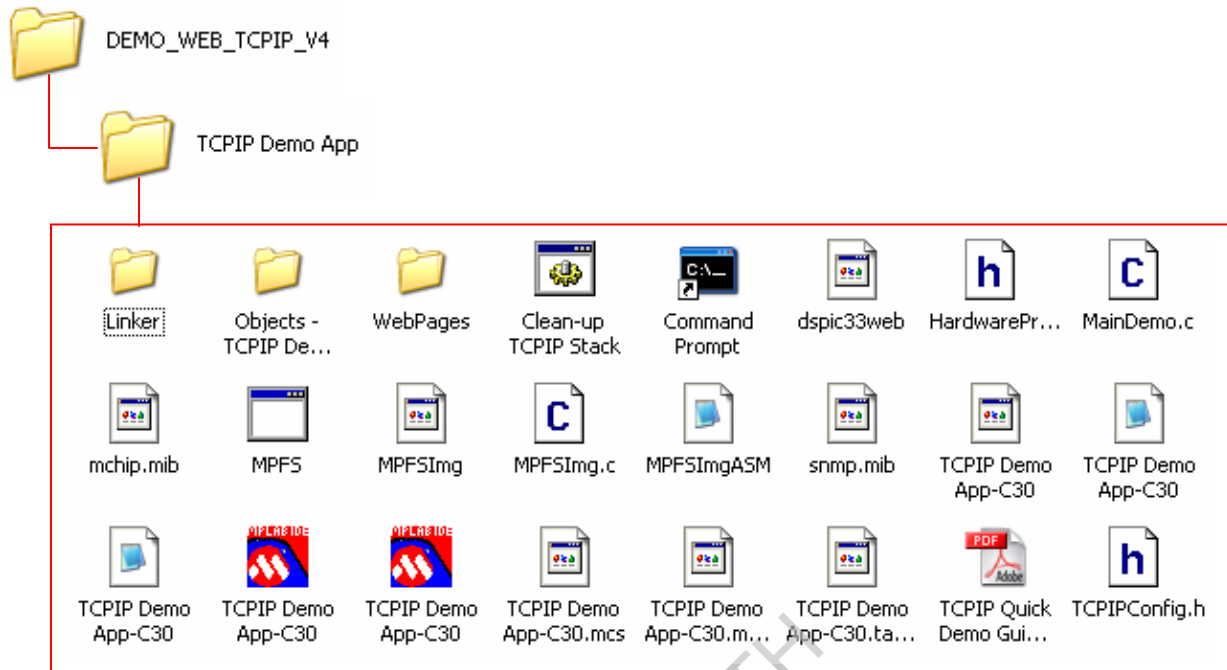
รูปแสดงการเชื่อมต่อ ET-dsPIC33WEB เข้ากับ HUB ของระบบ LAN ใช้สายแบบ Direct

3. ทำการดาวน์โหลด Code ซึ่งอยู่ใน “\DEMO_WEB_TCPIP_V4\TCPIP Demo App\TCPIP Demo App.HEX” ให้กับบอร์ด โดยสามารถศึกษาขั้นตอนการ Download Code ได้จากหัวข้อ “การ Download Code ให้กับบอร์ด” ในส่วนของคู่มือการใช้งาน
4. ทำการเซตค่า IP Address ของคอมพิวเตอร์ให้ Subnet mask: และ IP address ให้เป็นคลาสเดียวกันกับ บอร์ด ET-dsPIC33WEB V1.0 ในตัวอย่างนี้ IP ของบอร์ด คือ 192.168.1.200 เราจะต้องเซต IP ของเครื่องคอมพิวเตอร์ให้เป็น 192.168.1.xxx (xxx = 0 ถึง 255 แต่ต้องไม่ซ้ำกับ 200 ซึ่งเป็น IP ของบอร์ด) ส่วน Subnet mask: ตั้งเป็น 255.255.255.0 ดังรูปต่อไปนี้



รูปแสดงตัวอย่างการตั้งค่า IP Address ของคอมพิวเตอร์

5. เปิดโปรแกรม Command Prompt ซึ่งจะอยู่ในโฟลเดอร์ "...\DEMO_WEB_TCPIP_V4\TCPIP Demo App"



เมื่อดับเบิลคลิก ICON ดังกล่าวจะปรากฏหน้าต่าง Command Prompt ดังต่อไปนี้

```
C:\...\ DEMO_WEB_TCPIP_V4\TCPIP Demo App>
```

6. ลองทดสอบการเชื่อมต่อโดยใช้วิธีการ Ping ไปที่หมายเลข IP ของบอร์ดในตัวอย่างนี้ คือ 192.168.1.200 โดยให้พิมพ์คำสั่ง ping 192.168.1.200 แล้ว Enter ดังรูปต่อไปนี้

```
C:\...\ DEMO_WEB_TCPIP_V4\TCPIP Demo App>ping 192.168.1.200
```

โดยผลจากการ Ping หากการเชื่อมต่อสมบูรณ์ ก็จะได้แสดงข้อความ ดังตัวอย่าง

```
C:\...\DEMO_WEB_TCPIP_V4\TCPIP Demo App>ping 192.168.1.200
```

```
Pinging 192.168.1.200 with 32 bytes of data:
```

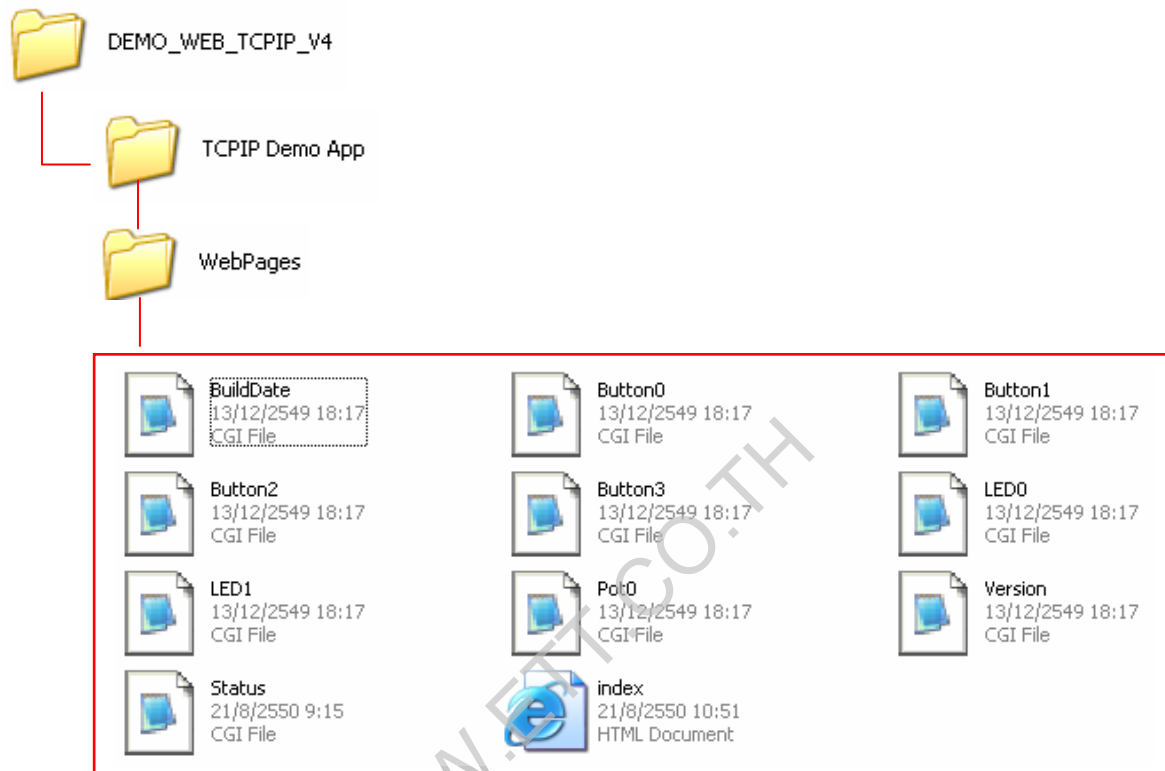
```
Reply from 192.168.1.200: bytes=32 time<1ms TTL=100
Reply from 192.168.1.200: bytes=32 time<1ms TTL=100
Reply from 192.168.1.200: bytes=32 time<1ms TTL=100
Reply from 192.168.1.200: bytes=32 time<1ms TTL=100
```

```
Ping statistics for 192.168.1.200:
```

```
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

การคอมไพล์ Webpage

ในเบื้องต้นผู้ใช้สามารถทำการแก้ไขเปลี่ยนแปลงรายละเอียดของหน้า Webpage ใหม่ได้ โดยเข้าไปในโฟลเดอร์ “...\Examples\DEMO_WEB_TCPIP_V4\TCPIP Demo App\Webpages” โดยอาจใช้โปรแกรม Notepad หรือ Text Editor ใดๆ ในการแก้ไขโค้ดเหล่านี้ก็ได้



โดยไฟล์ที่เป็นหน้า Webpage จะชื่อ index.html ส่วนอีก 10 ไฟล์ที่เหลือเป็น CGI Scrip สำหรับใช้รับส่งข้อมูลกันระหว่างหน้า Webpage และ บอร์ด ET-dsPIC33WEB V1.0 ซึ่งหลังจากทำการแก้ไขโค้ดต่างๆตามต้องการเป็นที่เรียบร้อยแล้ว ผู้ใช้จะต้องทำการคอมไพล์โค้ดเหล่านี้ใหม่ ให้เป็นไฟล์แบบ Binary (.bin) เพื่อที่จะได้สั่ง Download ไฟล์ดังกล่าวไปเก็บไว้ในหน่วยความจำ SPI Memory Module (25LC256) ที่อยู่ในบอร์ด ซึ่งทาง Microchips เองได้สร้างเครื่องมือสำหรับอำนวยความสะดวกในการพัฒนาโปรแกรมส่วนนี้เตรียมไว้ให้เราได้ใช้งานด้วย คือไฟล์ “MPFS.EXE” โดยไฟล์ตัวนี้เป็นโปรแกรมที่ทำงานแบบ Command Line ใช้สำหรับแปลงไฟล์ต่างๆที่อยู่ใน Folder ให้เป็น Binary File โดยขั้นตอนในการคอมไพล์ Webpage File ให้ทำดังนี้

- เปิดหน้าต่าง Command Prompt โดยคลิกเมาส์ที่ Shortcut ของ Command Prompt ที่เตรียมไว้ให้แล้วใน Folder “...\ET-dSPIC33WEB\Examples\DEMO_WEB_TCPIP_V4\TCPIP Demo App”
- พิมพ์คำสั่ง MPFS Input Output
 - Input หมายถึง ชื่อของ Folder ที่ใช้เก็บไฟล์ทั้งหมดที่ต้องการแปลงเป็น Binary File ซึ่งในกรณีนี้คือ Folder ชื่อ “Webpages”
 - Output หมายถึง ชื่อ ไฟล์ Output ที่ได้จากการแปลง ซึ่งให้กำหนดนามสกุลเป็น bin โดยในที่นี้จะกำหนดชื่อของ Output File เป็น “dspic33web.bin”

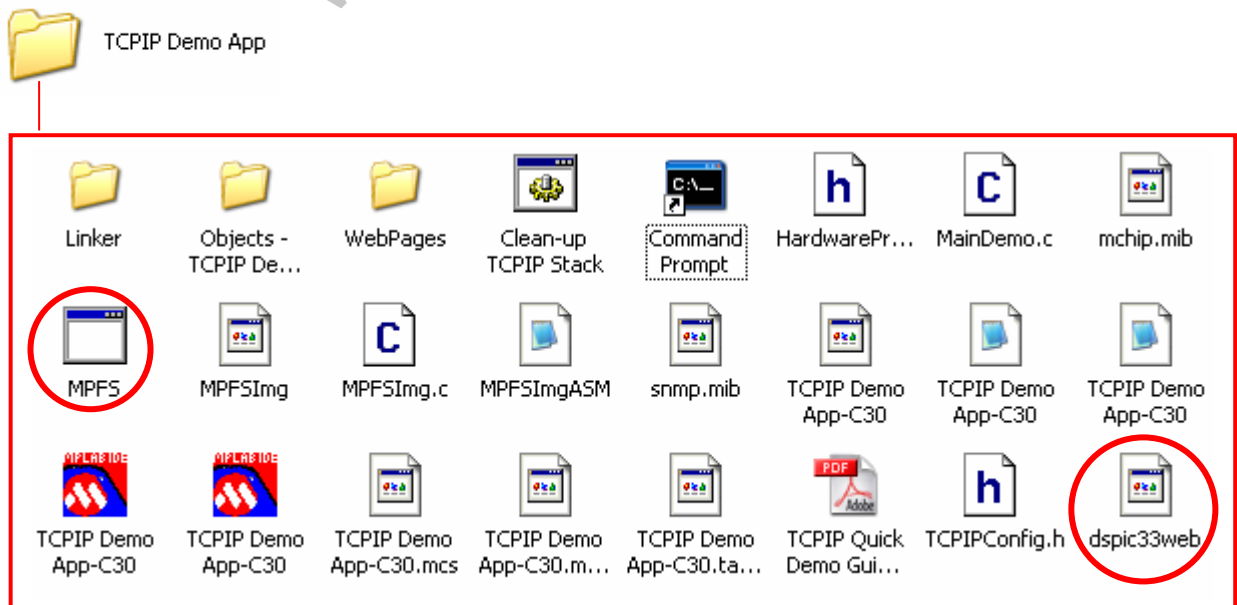

```

C:\...\TCPIP Demo App>mpfs Webpages dspic33web.bin
Adding 'Webpages\BUILDDATE.CGI'...
MPFS Size so far 247...
Adding 'Webpages\BUTTON0.CGI'...
MPFS Size so far 254...
Adding 'Webpages\BUTTON1.CGI'...
MPFS Size so far 261...
Adding 'Webpages\BUTTON2.CGI'...
MPFS Size so far 268...
Adding 'Webpages\BUTTON3.CGI'...
MPFS Size so far 275...
Adding 'Webpages\INDEX.HTML'...
MPFS Size so far 7056...
Adding 'Webpages\LED0.CGI'...
MPFS Size so far 7063...
Adding 'Webpages\LED1.CGI'...
MPFS Size so far 7070...
Adding 'Webpages\POT0.CGI'...
MPFS Size so far 7077...
Adding 'Webpages\STATUS.CGI'...
MPFS Size so far 7387...
Adding 'Webpages\VERSION.CGI'...
MPFS Size so far 7394...

```

รูปแสดง ผลการคอมไพล์ Webpage File

โดยไฟล์ “dspic33web.bin” ที่ถูกสร้างจะอยู่ในโฟลเดอร์เดียวกันกับ โปรแกรม MPFS ซึ่งในกรณีนี้จะอยู่ที่ “...\ET-dSPIC33WEB\Examples\DEMO_WEB_TCPIP_V4\TCPIP Demo App” ดังต่อไปนี้



การดาวน์โหลด Webpage File ให้บอร์ด ET-dsPIC33WEB V1.0

ในตัวอย่างการทดลอง Web Server Control นี้ ไฟล์ต่างๆที่เป็นของ Webpage นั้น จะต้องนำไปเก็บไว้ในหน่วยความจำแบบภายนอก ซึ่งก็คือ EEPROM ที่เป็น SPI Memory Module เบอร์ 25LC256 ซึ่งได้ทำการติดตั้งไว้ในบอร์ดเป็นที่เรียบร้อยแล้ว โดยใน Code Program ที่สร้างขึ้นตามตัวอย่างนี้ จะทำให้เราสามารถใช้อยู่ของ TCP/IP ในการส่งไฟล์ไปยังบอร์ดผ่านทางสายสัญญาณ Ethernet LAN ได้ทันที โดยให้ FTP Protocol โดยวิธีการนี้เราสามารถเรียกใช้ คำสั่ง FTP ใน Command Line เพื่อทำการ Login และส่งไฟล์ไปให้กับบอร์ดได้โดยตรง โดยมีลำดับขั้นตอนดังนี้

- เปิดหน้าต่าง Command Prompt โดยคลิกเมาส์ที่ Shortcut ของ Command Prompt ที่เตรียมไว้ให้แล้วใน Folder "...\ET-dSPIC33WEB\Examples\DEMO_WEB_TCPIP_V4\TCPIP Demo App"
- ใช้คำสั่ง FTP เพื่อเชื่อมต่อกับ IP Address ของบอร์ด ET-dsPIC33WEB ซึ่งในที่นี้ ก็คือ 192.168.1.200
- เมื่อผลการ FTP ได้รับการเชื่อมต่อ Connected เป็นที่เรียบร้อยแล้วให้ทำการ Login โดยใช้ชื่อ "ftp" และใช้รหัสผ่าน Password เป็น "ett" (จะไม่มีการ Echo ตัวอักษรของ Password ให้เห็น)
- เมื่อสามารถทำการ Login ได้สำเร็จให้ใช้คำสั่ง PUT ในการส่งไฟล์ชื่อ "dspic33web.bin" โดยพิมพ์คำสั่งเป็น put dspic33web.bin แล้ว Enter
- เพื่อการส่งไฟล์เสร็จเรียบร้อยแล้ว ให้ทำการพิมพ์คำสั่ง quit เพื่อออกจากการเชื่อมต่อกับ FTP Protocol ซึ่งหลังจากนี้ก็สามารถใช้งาน Web Server Control ของบอร์ด ET-dsPIC33WEB V1.0 ได้แล้ว
- เปิดโปรแกรม Internet Explorer แล้ว พิมพ์หมายเลข IP Address ของบอร์ด ซึ่งในที่นี้ คือ 192.168.1.200 หรือชื่อ DHCP ของบอร์ด ซึ่งก็คือ dspic33web ลงในช่อง Address แล้ว Enter เพื่อทดสอบ

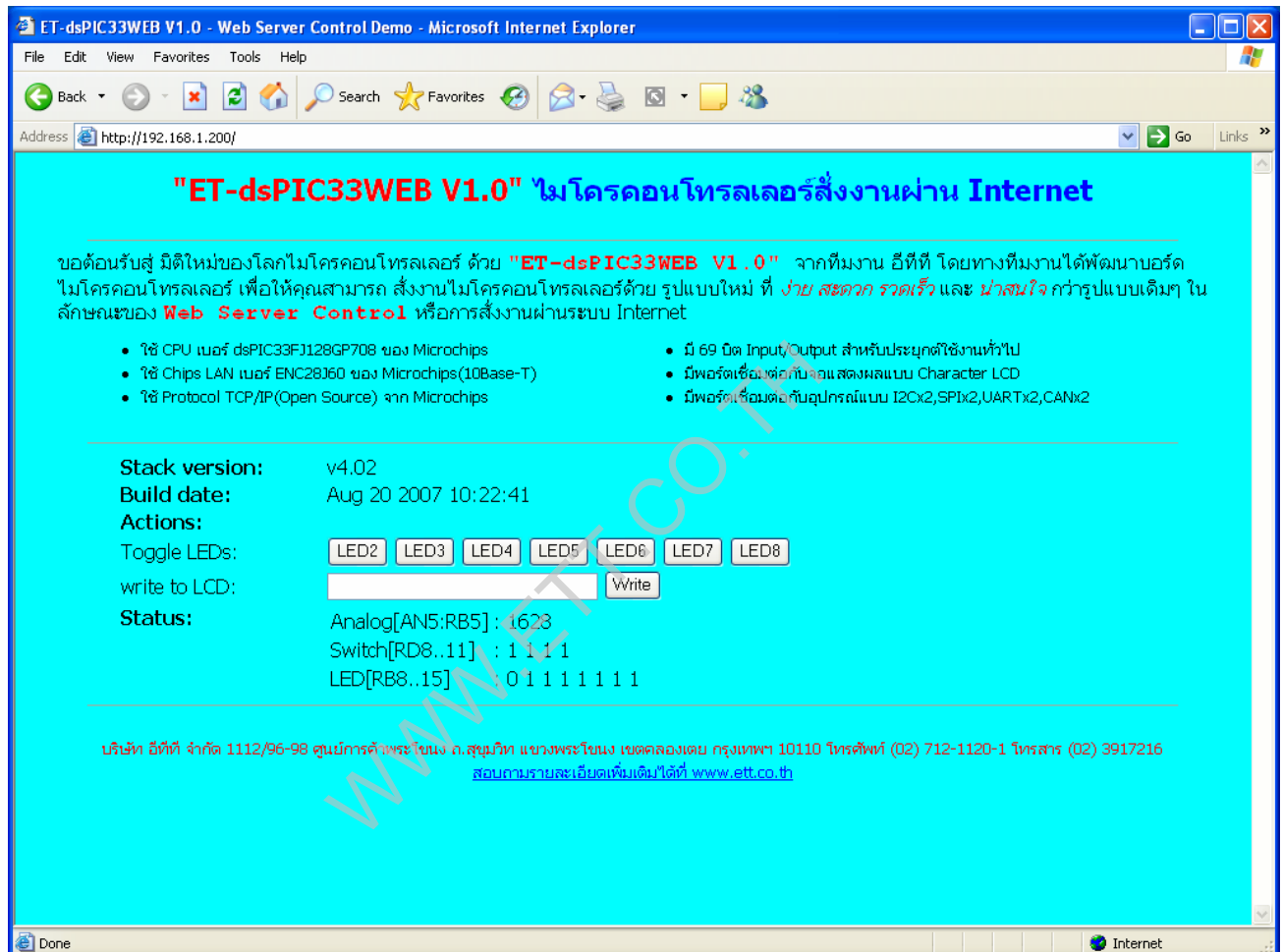
```
\...\Examples\DEMO_WEB_TCPIP_V4\TCPIP Demo App>ftp 192.168.1.200
Connected to 192.168.1.200.
220 Ready
User (192.168.1.200:(none)): ftp
331 Password required
Password:ett
230 Logged in
ftp> put dspic33web.bin
200 Ok
150 Transferring data...
#####
226 Transfer Complete
ftp: 7595 bytes sent in 0.00Seconds 7595000.00Kbytes/sec.
ftp> quit
221 Bye
```

รูปแสดง ตัวอย่างการใช้คำสั่ง FTP เพื่อ Download ไฟล์ dspic33web.bin ให้บอร์ด

หมายเหตุ รหัส Password ที่ป้อนให้กับคำสั่ง จะไม่มีการ Echo กลับมาให้เห็นด้วย ซึ่งผู้ใช้งานต้องพิมพ์ให้ถูกต้องตามลำดับทั้ง 3 หลัก (ett) ไม่เช่นนั้นจะไม่สามารถ Login ได้ ซึ่งถ้าพิมพ์ผิด และไม่สามารถทำการ Login ได้ให้ใช้คำสั่ง quit กลับไปก่อนแล้วเริ่มต้นกระบวนการใหม่

การทดสอบการทำงานของ Web Server Control

หลังจากทำการ Download Code ให้กับบอร์ด และ Download Webpage Files ต่างๆให้กับบอร์ดเป็นที่เรียบร้อยแล้ว ผู้ใช้สามารถทำการทดสอบการทำงานของโปรแกรมตัวอย่างของ Web Server Control ได้ทันที โดยให้เปิดโปรแกรม Internet Explorer แล้วพิมพ์หมายเลข IP ของบอร์ดลงไปในช่วง Address ในตัวอย่างนี้คือ 192.168.1.200 ซึ่งจะปรากฏหน้าต่าง Webpage ดังแสดงในรูป



รูปแสดง ลักษณะ Web Server Control ของ ET-dsPIC33WEB V1.0

โดยจากตัวอย่างโปรแกรมนี้ ผู้ใช้สามารถทำการทดสอบการส่งคำสั่งเพื่อควบคุมการทำงานของ I/O ภายในบอร์ด ET-dsPIC33WEB V1.0 จากหน้า Webpage ได้โดยตรง ดังนี้

- Toggle LEDs สามารถทำการทดสอบได้โดยคลิกเมาส์ที่ ปุ่ม LED2 - LED8 ซึ่งผลการคลิกเมาส์แต่ละครั้งจะส่งผลให้ LED กลับสถานะเป็นตรงกันข้าม คือ ดับ และ ติด สลับกันไป
- Write to LCD: สามารถพิมพ์ข้อความตามต้องการแต่ไม่เกิน 32 อักขระ แล้วคลิกเมาส์ที่ปุ่ม "Write" เพื่อส่งข้อความดังกล่าวไปแสดงผลที่หน้าจอ LCD บนบอร์ดได้

- Status จะแสดงผล อยู่ 3 ส่วน คือ
 - Analog[AN5:RB5]: จะแสดงผลจากการแปลงสัญญาณ Analog to Digital ที่ขา RB5 ซึ่งต่ออยู่กับ VR 10K บนบอร์ด เมื่อทำการปรับค่าของ VR1 ก็จะทำให้ค่า Status ของ A/D เปลี่ยนแปลงตามค่าการปรับ โดยมีค่าระหว่าง 0..4095
 - Switch[RD8..11]: จะแสดงค่าสถานะของการกดสวิตช์ ของ SW1-SW4 ที่อยู่บนบอร์ด โดยเมื่อกดจะได้ค่าเป็น “0” แต่เมื่อปล่อยจะได้ค่าเป็น “1”
 - LED[RB8..15]: จะแสดงค่าสถานะของพอร์ต RB8..15 ที่ต่อกับ LED บนบอร์ดไว้ โดยค่าจะเปลี่ยนแปลงไปตามการคลิกเมาส์ที่ปุ่ม LED2..LED8 โดยถ้ามีค่าเป็น “1” จะทำให้ LED ดับ แต่ถ้าเป็น “0” จะทำให้ LED ติดสว่าง ส่วน LED1 จะไม่สามารถสั่งควบคุมการทำงานจากปุ่มได้ เนื่องจาก LED1 จะสงวนไว้ให้แสดงการทำงานของ TCP/IP โดยจะกะพริบติดและดับสลับกันตลอดเวลา

การพัฒนาและปรับปรุง Code Program

โดยรวมแล้ว Code ต่างๆทั้งหมดที่อยู่ใน Project นี้จะเป็นส่วนของ Protocol TCP/IP มาตรฐานทั้งหมด ที่ทาง Microchips ได้พัฒนาขึ้น ทำให้เราสามารถนำมาเรียกใช้งานฟังก์ชันต่างๆที่อยู่ในไฟล์ต่างๆเหล่านั้นได้ทันที โดยไม่จำเป็นต้องไปแก้ไข Code ต่างๆที่เป็นส่วนของ Protocol TCP/IP ให้เสียเวลา ซึ่ง Code ในส่วนที่เราจำเป็นต้องแก้ไขนั้น จะเป็นในส่วนของ Application Layer ของ TCP/IP Stack เท่านั้น ซึ่งใน Project นี้ไฟล์ต่างๆที่เป็นของ Application นั้นจะถูกแยกเก็บไว้ใน Folder ชื่อ “..\DEMO_WEB_TCPIP_V4\TCPIP Demo App”

- HardwareProfile.h เป็นส่วนของการกำหนดรายละเอียดของสัญญาณ I/O ต่างๆ ที่จะใช้ในการเชื่อมต่อกับอุปกรณ์ภายนอก โดยต้องกำหนดให้สอดคล้องกับระบบฮาร์ดแวร์ของบอร์ด ET-dsPIC33WEB V1.0 อันได้แก่ Ethernet Driver (ET-MINI ENC28J60) ,SPI Memory Module(25LC256),LCD,LED,SW,ADC
- TCPIPConfig.h เป็นส่วนของการกำหนดค่า Default ต่างๆเกี่ยวกับ TCP/IP Stack อันได้แก่ DHCP Name, IP Address ของบอร์ด,MAC Address ของบอร์ด
- MainDemo.c เป็นส่วนของ Main Program ซึ่งจะมีการเรียกใช้ TCP/IP Stack และนำข้อมูลต่างๆที่ได้จาก Stack มาประมวลผล และตัดสินใจ อันได้แก่การแปลคำสั่งจาก CGI Scrip เพื่อนำรหัสคำสั่งไปสั่งงาน LED Output และ LCD Display รวมทั้งการอ่านค่าสถานะของ I/O ต่างๆเพื่อส่งออกไป Refresh Status ของหน้า Webpage เป็นต้น
- Wabpage File ต่างๆ เช่น HTML และ CGI Scrip ใน Folder ชื่อ “Webpages” ซึ่งเป็นส่วนที่ผู้ใช้สามารถทำการดัดแปลงแก้ไขได้เองตามต้องการ แต่ถ้ามีการแก้ไขค่าใน CGI Scrip ให้มีค่าต่างจากที่กำหนดไว้แล้ว ผู้ใช้จำเป็นต้องเข้าไปแก้ไข Code ที่อยู่ในไฟล์ “MainDemo.c” ให้สัมพันธ์และสอดคล้องกันด้วย