

พื้นฐานของดิจิทัลและไมโครคอนโทรลเลอร์

รหัสวิชา 30127-2004 (2-3-3) ดิจิทัลและไมโครคอนโทรลเลอร์

Digital And Microcontroller

1

พื้นฐานไมโครคอนโทรลเลอร์

พื้นฐานไมโครคอนโทรลเลอร์

1. ความหมายและคุณสมบัติของไมโครคอนโทรลเลอร์
 - 1.1 ความหมายของไมโครคอนโทรลเลอร์
 - 1.2 สถาปัตยกรรมของไมโครคอนโทรลเลอร์
 - 1.3 ประเภทของไมโครคอนโทรลเลอร์
 - 1.4 คุณสมบัติของไมโครคอนโทรลเลอร์ AT89C51ED2 , PIC16F887 และ ATMEGA32
 - 1.5 การใช้งานสัญญาณของไมโครคอนโทรลเลอร์

Digital And Microcontroller

2

พื้นฐานไมโครคอนโทรลเลอร์

2. ชุดคำสั่งภาษาแอสเซมบลีของไมโครคอนโทรลเลอร์
 - 2.1 ชุดคำสั่งภาษาแอสเซมบลีของไมโครคอนโทรลเลอร์ตระกูล MCS51
 - 2.2 ชุดคำสั่งภาษาแอสเซมบลีของไมโครคอนโทรลเลอร์ตระกูล PIC16F
 - 2.3 ชุดคำสั่งภาษาแอสเซมบลีของไมโครคอนโทรลเลอร์ตระกูล AVR
 - 2.4 ชุดคำสั่งภาษาแอสเซมบลีของไมโครคอนโทรลเลอร์ที่ใช้สำหรับการเขียนโปรแกรมเพื่อจำลองการทำงานเป็นลอจิกเกต
 - 2.5 การเขียนโปรแกรมภาษาแอสเซมบลี

Digital And Microcontroller

3

พื้นฐานไมโครคอนโทรลเลอร์

3. การใช้งานซอฟต์แวร์เพื่อเขียนโปรแกรมภาษาแอสเซมบลีของไมโครคอนโทรลเลอร์
 - 3.1 การใช้งานโปรแกรม Keil uVision3 สำหรับเขียนโปรแกรมภาษาแอสเซมบลีของไมโครคอนโทรลเลอร์ตระกูล MCS51
 - 3.2 การใช้งานโปรแกรม MPLAB X สำหรับเขียนโปรแกรมภาษาแอสเซมบลีของไมโครคอนโทรลเลอร์ตระกูล PIC16F
 - 3.3 การใช้งานโปรแกรม AVR Studio 6.2 สำหรับเขียนโปรแกรมภาษาแอสเซมบลีของไมโครคอนโทรลเลอร์ตระกูล AVR

Digital And Microcontroller

4

พื้นฐานไมโครคอนโทรลเลอร์

4. การใช้งานโปรแกรม Proteus เพื่อจำลองการทำงานของวงจรไมโครคอนโทรลเลอร์
 - 4.1 การใช้งานโปรแกรม Proteus เพื่อจำลองการทำงานของวงจรไมโครคอนโทรลเลอร์ตระกูล MCS51
 - 4.2 การใช้งานโปรแกรม Proteus เพื่อจำลองการทำงานของวงจรไมโครคอนโทรลเลอร์ตระกูล PIC16F
 - 4.3 การใช้งานโปรแกรม Proteus เพื่อจำลองการทำงานของวงจรไมโครคอนโทรลเลอร์ตระกูล AVR

Digital And Microcontroller

5

พื้นฐานไมโครคอนโทรลเลอร์

1. ความหมายและคุณสมบัติของไมโครคอนโทรลเลอร์
 - 1.1 ความหมายของไมโครคอนโทรลเลอร์

ไมโครคอนโทรลเลอร์ (Microcontroller เรียกกย่อ ๆ ว่า μC , μC หรือ MCU) หมายถึงอุปกรณ์ไอซีประเภทการประมวลผล ที่รวมเอาความสามารถที่คล้ายคลึงกับไมโครคอมพิวเตอร์บรรจุเข้าไว้ในตัวเดียวกัน แล้วสามารถนำไปต่อใช้งานได้เลย เพราะภายในของไมโครคอนโทรลเลอร์จะประกอบด้วย 5 องค์ประกอบสำคัญ ได้แก่ ส่วนที่เป็นขาสัญญาณที่ใช้ในการติดต่อกับอุปกรณ์อินพุต , ส่วนที่เป็นขาสัญญาณที่ใช้ในการติดต่อกับอุปกรณ์เอาต์พุต , ส่วนที่ใช้ในการประมวลผลข้อมูลในรูปแบบคณิตศาสตร์ และลอจิก ที่นิยมเรียกว่า CPU , ส่วนที่เป็นหน่วยความจำ และส่วนสุดท้ายคือส่วนโมดูลที่ทำหน้าที่พิเศษอื่น ๆ ซึ่งไมโครคอนโทรลเลอร์เป็นอุปกรณ์ที่สามารถเขียนโปรแกรมควบคุมการทำงานไว้ภายในไอซีได้

Digital And Microcontroller

6

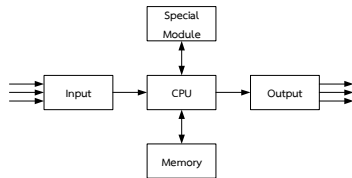
พื้นฐานไมโครคอนโทรลเลอร์

1.2 สถาปัตยกรรมของไมโครคอนโทรลเลอร์

สถาปัตยกรรมของไมโครคอนโทรลเลอร์ จะกล่าวถึงส่วนสำคัญทั้งหมด

3 ส่วน

1.2.1 โครงสร้างของไมโครคอนโทรลเลอร์



รูปที่ 1.2 โครงสร้างภายในของไมโครคอนโทรลเลอร์

Digital And Microcontroller

7

พื้นฐานไมโครคอนโทรลเลอร์

1.2.2 สถาปัตยกรรมของไมโครคอนโทรลเลอร์ที่เกี่ยวข้องกับการเชื่อมต่อหน่วยความจำ

สถาปัตยกรรมของไมโครคอนโทรลเลอร์ถ้าแบ่งตามลักษณะของการเชื่อมต่อหน่วยความจำ หรือการแบ่งพื้นที่หน่วยความจำของไมโครคอนโทรลเลอร์ จะสามารถแบ่งออกได้เป็น 2 รูปแบบ คือ

- สถาปัตยกรรมแบบฟอนนอยมันน์ (Von Neumann Architecture) มีหลักการคือการประมวลผลทั้งหมดจะกระทำที่หน่วยประมวลผลกลาง คำสั่งและข้อมูลจะถูกเก็บที่หน่วยความจำเดียวกัน เช่น ไมโครคอนโทรลเลอร์ตระกูล MCS-51

- สถาปัตยกรรมแบบฮาร์วาร์ด (Harvard Architecture) เป็นสถาปัตยกรรมที่ออกแบบเพื่อแก้ปัญหาจุดอ่อนของ Von Neumann โดยแยกหน่วยความจำโปรแกรมและหน่วยความจำข้อมูลออกจากกันรวมทั้งแยกบัสข้อมูลด้วย ซึ่งเป็นสถาปัตยกรรมของไมโครคอนโทรลเลอร์ในยุคปัจจุบัน เช่น ไมโครคอนโทรลเลอร์ตระกูล PIC และ AVR

Digital And Microcontroller

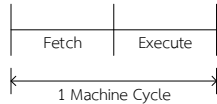
8

พื้นฐานไมโครคอนโทรลเลอร์

1.2.3 สถาปัตยกรรมของไมโครคอนโทรลเลอร์ที่กล่าวถึงลักษณะของการประมวลผล

สถาปัตยกรรมของไมโครคอนโทรลเลอร์ที่แบ่งแยกตามลักษณะของการประมวลผลสามารถแบ่งได้เป็น 2 กลุ่ม ได้แก่

- สถาปัตยกรรมการประมวลผลแบบ CISC (Complex Instruction Set Computing)



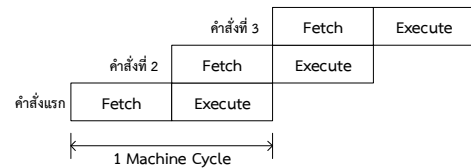
รูปที่ 1.3 ลักษณะการประมวลผลแบบ CISC

Digital And Microcontroller

9

พื้นฐานไมโครคอนโทรลเลอร์

- สถาปัตยกรรมการประมวลผลแบบ RISC (Reduced Instruction Set Computing)



รูปที่ 1.4 ลักษณะการประมวลผลแบบ RISC

Digital And Microcontroller

10

พื้นฐานไมโครคอนโทรลเลอร์

1.3 ประเภทของไมโครคอนโทรลเลอร์

1. ไมโครคอนโทรลเลอร์ที่มี CPU ในการประมวลผลขนาด 8 บิต และ 1 พอร์ตจะมีขาสัญญาณได้สูงสุดจำนวน 8 ขา เช่นไมโครคอนโทรลเลอร์ตระกูล MCS-51 ,PIC10F ,PIC12F ,PIC16F , ,PIC18F และ AVR เป็นต้น

2. ไมโครคอนโทรลเลอร์ที่มี CPU ในการประมวลผลขนาด 16 บิต และ 1 พอร์ตจะมีขาสัญญาณได้สูงสุดจำนวน 16 ขา เช่นไมโครคอนโทรลเลอร์ตระกูล MSP430 , PIC24F และ dsPIC30F เป็นต้น

3. ไมโครคอนโทรลเลอร์ที่มี CPU ในการประมวลผลขนาด 32 บิต และ 1 พอร์ตจะมีขาสัญญาณได้สูงสุดจำนวน 32 ขา เช่นไมโครคอนโทรลเลอร์ตระกูล ARM7 ,PIC32F ,STM32 ,ESP8285 ,ESP8266 และESP32 เป็นต้น

Digital And Microcontroller

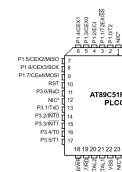
11

พื้นฐานไมโครคอนโทรลเลอร์

1.4 คุณสมบัติของไมโครคอนโทรลเลอร์ AT89C51ED2 ,PIC16F887 และ ATMEGA32

ตารางที่ 1.1 ตารางเปรียบเทียบคุณสมบัติของไมโครคอนโทรลเลอร์ตระกูลต่างๆ

ไมโครคอนโทรลเลอร์เบอร์	ตระกูล	Program Memory	Data Memory	External Memory	EEPROM	IO	PWM (ch)	SPI (ch)	Timer/Counter	I2C (ch)	UART (ch)	ADC (ch)
AT89C51ED2	MCS51	64Kbyte	256byte	1792byte	2Kbyte	32	5	1	3	-	1	-
PIC16F887	PIC16F	8Kword	368byte	-	256byte	36	2	1	3	1	1	14
ATMEGA32	AVR	32Kbyte	2Kbyte	-	1Kbyte	32	4	1	3	1	1	8



รูปที่ 1.5 รูปตำแหน่งขาสัญญาณและตัวถังของไมโครคอนโทรลเลอร์ตระกูล MCS-51 เบอร์ AT89C51ED2

Digital And Microcontroller

12

พื้นฐานไมโครคอนโทรลเลอร์

รูปที่ 1.6 รูปตำแหน่งขาสัญญาณและตัวถังของไมโครคอนโทรลเลอร์ตระกูล PIC16F เบอร์ PIC16F887

รูปที่ 1.7 รูปตำแหน่งขาสัญญาณและตัวถังของไมโครคอนโทรลเลอร์ตระกูล AVR เบอร์ ATMEGA32

Digital And Microcontroller
13

พื้นฐานไมโครคอนโทรลเลอร์

1.5 การใช้งานขาสัญญาณของไมโครคอนโทรลเลอร์

ตารางที่ 1.2 ตารางเปรียบเทียบรีจิสเตอร์ของไมโครคอนโทรลเลอร์ตระกูล MCS51 , PIC16F และ AVR ที่เกี่ยวกับการใช้งานขาสัญญาณพอร์ตเพื่อติดต่อกับอุปกรณ์อินพุตหรือเอาต์พุต

ไมโครคอนโทรลเลอร์ตระกูล	เบอร์ของอุปกรณ์	รีจิสเตอร์ที่ทำหน้าที่กำหนดทิศทางขาสัญญาณ	รีจิสเตอร์ที่ทำหน้าที่ติดต่อกับอุปกรณ์อินพุต	รีจิสเตอร์ที่ทำหน้าที่ติดต่อกับอุปกรณ์เอาต์พุต
MCS-51	AT89C51ED2	-	Px	Px
PIC16F	PIC16F887	TRISx	PORTx	PORTx
AVR	ATMEGA32	DDRx	PINx	PORTx

Digital And Microcontroller
14

พื้นฐานไมโครคอนโทรลเลอร์

2. ชุดคำสั่งภาษาแอสเซมบลีของไมโครคอนโทรลเลอร์

2.1 ชุดคำสั่งภาษาแอสเซมบลีของไมโครคอนโทรลเลอร์ตระกูล MCS-51

ไมโครคอนโทรลเลอร์ MCS-51 มีรูปแบบชุดคำสั่งภาษาแอสเซมบลีทั้งหมดประมาณ 111 รูปแบบชุดคำสั่ง และแต่ละรูปแบบชุดคำสั่งจะมีขนาดของข้อมูลที่ไม่เท่ากัน โดยสามารถแบ่งออกเป็นกลุ่มได้ 5 กลุ่ม ดังนี้

1. กลุ่มคำสั่งการโอนย้ายข้อมูล (Data Transfer Instructions) มี 28 รูปแบบชุดคำสั่ง
2. กลุ่มคำสั่งทางคณิตศาสตร์ (Arithmetic Operation instructions) มี 24 รูปแบบชุดคำสั่ง
3. กลุ่มคำสั่งทำงานลอจิก (Logical Operation Instructions) มี 25 รูปแบบชุดคำสั่ง
4. กลุ่มคำสั่งจัดการข้อมูลระดับบิต (Boolean Variable Manipulated Instructions) มี 17 รูปแบบชุดคำสั่ง
5. กลุ่มคำสั่งการกระโดด (Program Branching Instructions) มี 17 รูปแบบชุดคำสั่ง

Digital And Microcontroller
15

พื้นฐานไมโครคอนโทรลเลอร์

2.2 ชุดคำสั่งภาษาแอสเซมบลีของไมโครคอนโทรลเลอร์ตระกูล PIC16F

ไมโครคอนโทรลเลอร์ PIC16F มีรูปแบบชุดคำสั่งภาษาแอสเซมบลีทั้งหมดประมาณ 35 รูปแบบชุดคำสั่งและแต่ละรูปแบบชุดคำสั่งจะมีขนาดของข้อมูลเท่ากันที่ 14 บิต โดยสามารถแบ่งออกเป็นกลุ่มได้ 3 กลุ่ม ดังนี้

1. กลุ่มคำสั่งกระทำกับข้อมูลแบบไบนารี (Byte-Oriented File Register Operation) มี 18 รูปแบบชุดคำสั่ง
2. กลุ่มคำสั่งกระทำกับข้อมูลแบบบิต (Bit-Oriented File Register Operation) มี 4 รูปแบบชุดคำสั่ง
3. กลุ่มคำสั่งกระทำกับค่าข้อมูลคงที่ (Literal and Control Operation) มี 13 รูปแบบชุดคำสั่ง

Digital And Microcontroller
16

พื้นฐานไมโครคอนโทรลเลอร์

2.3 ชุดคำสั่งภาษาแอสเซมบลีของไมโครคอนโทรลเลอร์ตระกูล AVR

ไมโครคอนโทรลเลอร์ AVR มีรูปแบบชุดคำสั่งภาษาแอสเซมบลีทั้งหมดประมาณ 131 รูปแบบชุดคำสั่ง โดยสามารถแบ่งออกเป็นกลุ่มได้ 5 กลุ่ม ดังนี้

1. กลุ่มคำสั่งกระทำทางคณิตศาสตร์และลอจิก (Arithmetic and Logic Instructions) มี 28 รูปแบบชุดคำสั่ง
2. กลุ่มคำสั่งการกระโดด (Branch Instructions) มี 36 รูปแบบชุดคำสั่ง
3. กลุ่มคำสั่งการโอนย้ายข้อมูล (Data transfer Instructions) มี 35 รูปแบบชุดคำสั่ง
4. กลุ่มคำสั่งการกระทำบิต (Bit and Bit-Test Instructions) มี 28 รูปแบบชุดคำสั่ง
5. กลุ่มคำสั่งการโอนย้ายข้อมูล (MCU control Instructions) มี 4 รูปแบบชุดคำสั่ง

Digital And Microcontroller
17

พื้นฐานไมโครคอนโทรลเลอร์

2.4 ชุดคำสั่งภาษาแอสเซมบลีของไมโครคอนโทรลเลอร์ที่ใช้สำหรับการเขียนโปรแกรมเพื่อจำลองการทำงานเป็นลอจิกเกต

2.4.1 กลุ่มคำสั่งจัดการข้อมูลระดับบิตของไมโครคอนโทรลเลอร์ MCS-51 ได้แก่

CLR C	ORL C,bit
CLR bit	ORL C,/bit
SETB C	MOV C,bit
SETB bit	MOV bit,C
CPL C	JC rel
CPL bit	JNC rel
ANL C,bit	JB bit,rel
ANL C,/bit	JNB bit,rel
	JBC bit,rel

Digital And Microcontroller
18

พื้นฐานไมโครคอนโทรลเลอร์

2.4.2 กลุ่มคำสั่งจัดการข้อมูลระดับบิตของไมโครคอนโทรลเลอร์ PIC16F ได้แก่

BCF f,b ; ทำให้บิตของข้อมูล ณ ตำแหน่งที่ b ของหน่วยความจำข้อมูลที่ f มีค่าเป็นลอจิก '0'

BSF f,b ; ทำให้บิตของข้อมูล ณ ตำแหน่งที่ b ของหน่วยความจำข้อมูลที่ f มีค่าเป็นลอจิก '1'

BTFSF f,b ; ทำการตรวจสอบบิตของข้อมูล ณ ตำแหน่งที่ b ของหน่วยความจำข้อมูลที่ f ถ้ามีค่าเป็นลอจิก '0' ให้ข้ามการประมวลผลไป 1 คำสั่ง

BTFSB f,b ; ทำการตรวจสอบบิตของข้อมูล ณ ตำแหน่งที่ b ของหน่วยความจำข้อมูลที่ f ถ้ามีค่าเป็นลอจิก '1' ให้ข้ามการประมวลผลไป 1 คำสั่ง

พื้นฐานไมโครคอนโทรลเลอร์

2.4.3 กลุ่มคำสั่งจัดการข้อมูลระดับบิตของไมโครคอนโทรลเลอร์ AVR ได้แก่

CBI P,b ; ทำให้บิตของข้อมูล ณ ตำแหน่งที่ b ของหน่วยความจำที่ I/O มีค่าเป็นลอจิก '0'

SBI P,b ; ทำให้บิตของข้อมูล ณ ตำแหน่งที่ b ของหน่วยความจำที่ I/O มีค่าเป็นลอจิก '1'

SBIC P,b ; ทำการตรวจสอบบิตของข้อมูล ณ ตำแหน่งที่ b ของหน่วยความจำ I/O ถ้ามีค่าเป็นลอจิก '0' ให้ข้ามการประมวลผลไป 1 คำสั่ง

SBIS P,b ; ทำการตรวจสอบบิตของข้อมูล ณ ตำแหน่งที่ b ของหน่วยความจำ I/O ถ้ามีค่าเป็นลอจิก '1' ให้ข้ามการประมวลผลไป 1 คำสั่ง

พื้นฐานไมโครคอนโทรลเลอร์

2.5 การเขียนโปรแกรมภาษาแอสเซมบลี

การเขียนโปรแกรมภาษาแอสเซมบลีประกอบด้วย 4 ส่วนหลัก ซึ่งแยกได้โดยใช้ปุ่ม Tab คือ

1. ลาเบล (Label) ใช้ในการอ้างอิงบรรทัดโคโรรที่ดหนึ่งของโปรแกรมที่ทำการเขียนขึ้น โดยลาเบลจะต้องเขียนตามหลังด้วยเครื่องหมายโคลอน ":"
2. รหัสนิมิก (Mnemonic) เป็นส่วนแสดงคำสั่งของไมโครคอนโทรลเลอร์ที่ต้องการให้กระทำ
3. โอเปอเรนด์ (Operand) เป็นส่วนที่แสดงถึงตัวกระทำหรือถูกกระทำและข้อมูลที่ใช้ในการกระทำตามคำสั่งที่กำหนดโดยรหัสนิมิกก่อนหน้านี้
4. คอมเมนต์ (Comment) เป็นส่วนที่ผู้เขียนโปรแกรมเขียนขึ้นเพื่อใช้ในการอธิบายคำสั่งที่กระทำ หรือผลของการกระทำคำสั่งในบรรทัดหรือโปรแกรมย่ออื่น ๆ โดยคอมเมนต์จะต้องเขียนตามหลังด้วยเครื่องหมายโคลอน ";" และการเขียนภาษาแอสเซมบลีในแต่ละบรรทัดจะมีหรือไม่มีในส่วนคอมเมนต์ก็ได้

พื้นฐานไมโครคอนโทรลเลอร์

ตารางที่ 1.3 แสดงตัวอย่างการแบ่งพื้นที่หน้ากระดาษในการเขียนโปรแกรมภาษาแอสเซมบลี

Label	Mnemonic	Operand	Comment
Start:	ORG MOV MOV END	0000H SP,#128-32 R2,#20	;Set Stack Address ;R2 = 14H

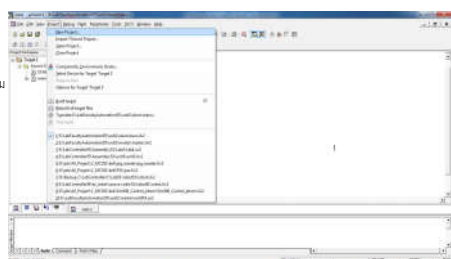
พื้นฐานไมโครคอนโทรลเลอร์

3. การใช้งานซอฟต์แวร์เพื่อเขียนโปรแกรมภาษาแอสเซมบลีของไมโครคอนโทรลเลอร์

3.1 การใช้งานโปรแกรม Keil uVision3 สำหรับเขียนโปรแกรมภาษาแอสเซมบลีของไมโครคอนโทรลเลอร์ตระกูล MCS51

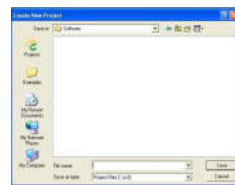


รูปที่ 1.28 รูปแสดงไอคอนของโปรแกรม Keil uVision3

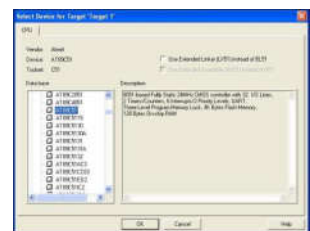


รูปที่ 1.29 รูปแสดงโปรแกรม Keil uVision3

พื้นฐานไมโครคอนโทรลเลอร์

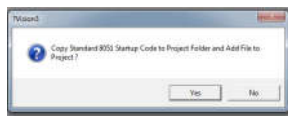


รูปที่ 1.30 รูปแสดงหน้าต่าง Create New Project

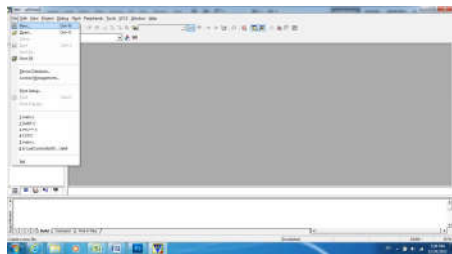


รูปที่ 1.31 รูปแสดงหน้าต่าง Select Device for Target 'Target1'

พื้นฐานไมโครคอนโทรลเลอร์



รูปที่ 1.32 รูปแสดงหน้าต่างให้เลือกรสร้างไฟล์ 8051 Startup Code

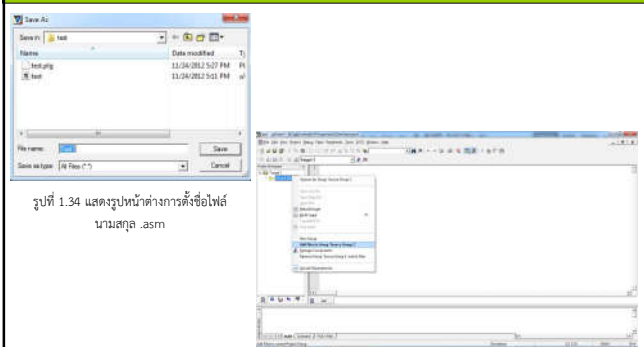


รูปที่ 1.33 รูปแสดงขั้นตอนสร้าง new file

Digital And Microcontroller

25

พื้นฐานไมโครคอนโทรลเลอร์



รูปที่ 1.34 แสดงรูปหน้าต่างการตั้งชื่อไฟล์นามสกุล .asm

รูปที่ 1.35 แสดงหน้าต่างการเลือกการเลือก Add File to Group "Source Group 1"

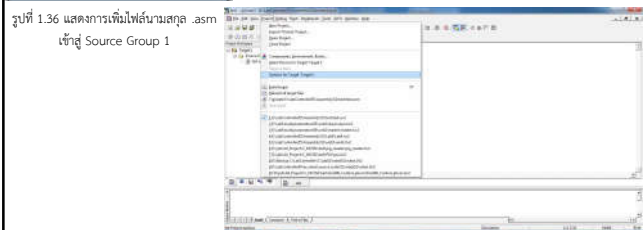
Digital And Microcontroller

26

พื้นฐานไมโครคอนโทรลเลอร์



รูปที่ 1.37 แสดงขั้นตอนการเลือกเมนู Options for Target "Target 1"

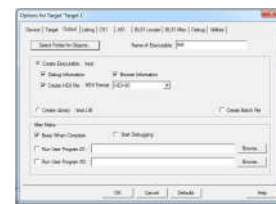


รูปที่ 1.36 แสดงการเพิ่มไฟล์นามสกุล .asm เข้าสู่ Source Group 1

Digital And Microcontroller

27

พื้นฐานไมโครคอนโทรลเลอร์



รูปที่ 1.38 รูปแสดงการเลือกเมนู Create HEX File

เมื่อทำตามขั้นตอนเรียบร้อยแล้วให้ทำการเขียนโปรแกรมภาษาแอสเซมบลีบน Work Sheet เมื่อทำการเขียนโปรแกรมเรียบร้อยแล้วให้ทำการ Assembler โปรแกรม โดยเข้าเมนู Project > Build target แล้วให้สังเกตบริเวณหน้าต่างโปรแกรมด้านล่างว่ามีข้อความ 0 Error หรือไม่ ถ้าไม่ให้ทำการแก้ไขโปรแกรมแล้วทำการ Build target ใหม่จนกว่าหน้าต่าง Output จะปรากฏข้อความ 0 Error

Digital And Microcontroller

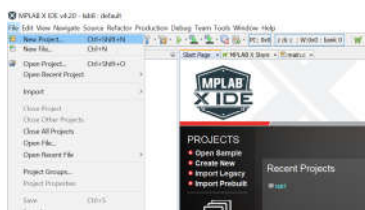
28

พื้นฐานไมโครคอนโทรลเลอร์

3.2 การใช้งานโปรแกรม MPLAB X สำหรับเขียนโปรแกรมภาษาแอสเซมบลีของไมโครคอนโทรลเลอร์ตระกูล PIC16F



รูปที่ 1.39 แสดงรูปไอคอนของโปรแกรม MPLAB X IDE V4.20



รูปที่ 1.40 แสดงวิธีการสร้าง New project

Digital And Microcontroller

29

พื้นฐานไมโครคอนโทรลเลอร์



รูปที่ 1.41 แสดงหน้าต่าง New project Choose Project

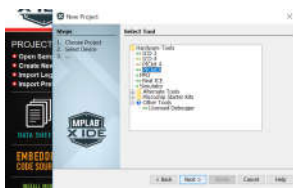


รูปที่ 1.42 แสดงหน้าต่าง Select Device

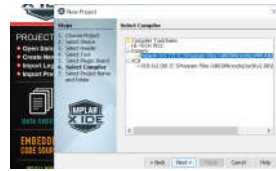
Digital And Microcontroller

30

พื้นฐานไมโครคอนโทรลเลอร์



รูปที่ 1.43 แสดงหน้าต่าง Select Tool

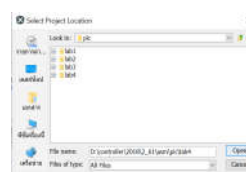


รูปที่ 1.44 หน้าต่างแสดง Select Compiler

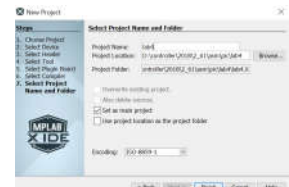
Digital And Microcontroller

31

พื้นฐานไมโครคอนโทรลเลอร์



รูปที่ 1.45 แสดงหน้าต่าง Select Project Location

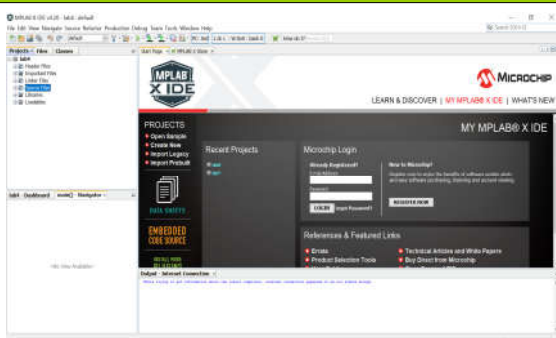


รูปที่ 1.46 แสดงหน้าต่าง Select Project Name and Folder

Digital And Microcontroller

32

พื้นฐานไมโครคอนโทรลเลอร์

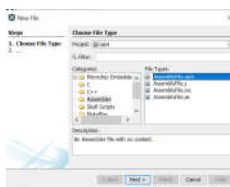


รูปที่ 1.47 แสดงการเตรียมการสร้างไฟล์ asm

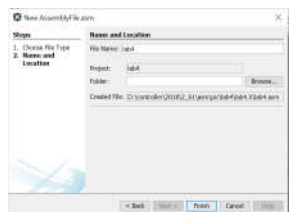
Digital And Microcontroller

33

พื้นฐานไมโครคอนโทรลเลอร์



รูปที่ 1.48 หน้าต่างแสดงให้เลือกชนิดของไฟล์ที่ต้องการสร้าง

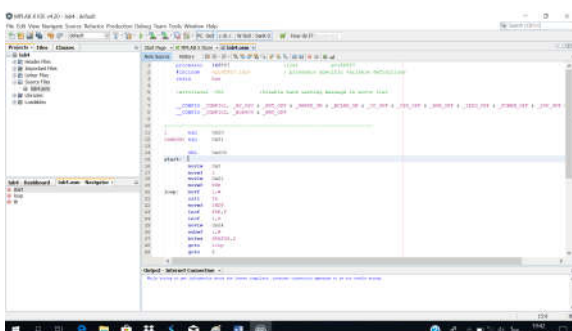


รูปที่ 1.49 รูปหน้าต่างแสดงให้ยอมรับการสร้างไฟล์นามสกุล .asm

Digital And Microcontroller

34

พื้นฐานไมโครคอนโทรลเลอร์

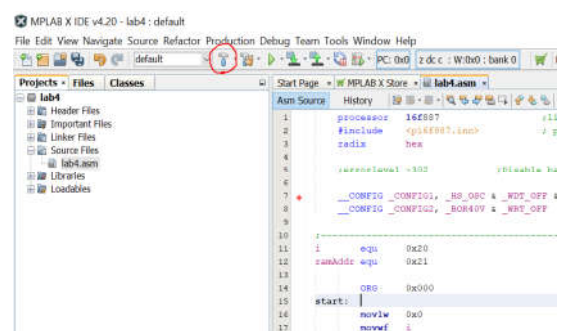


รูปที่ 1.50 รูปตัวอย่างการเขียนโปรแกรมแอสเซมบลีของ PIC16F

Digital And Microcontroller

35

พื้นฐานไมโครคอนโทรลเลอร์

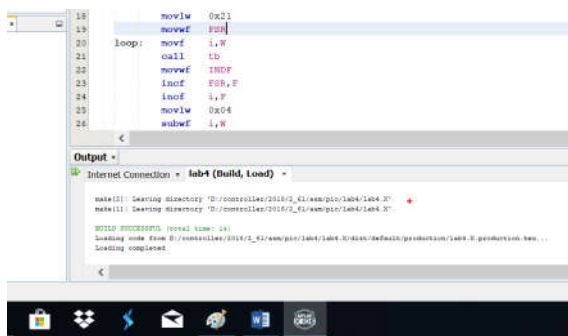


รูปที่ 1.51 รูปแสดงตำแหน่งเครื่องหมายการแอสเซมบลีของไมโครคอนโทรลเลอร์ PIC16F

Digital And Microcontroller

36

พื้นฐานไมโครคอนโทรลเลอร์



รูปที่ 1.52 แสดงผลการแอสเซมบลีของไมโครคอนโทรลเลอร์ PIC16F

Digital And Microcontroller

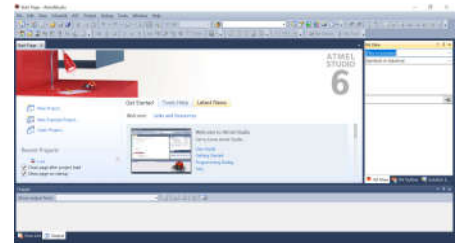
37

พื้นฐานไมโครคอนโทรลเลอร์

3.3 การใช้งานโปรแกรม AVR Studio 6.2 สำหรับเขียนโปรแกรมภาษาแอสเซมบลีของไมโครคอนโทรลเลอร์ตระกูล AVR



รูปที่ 1.53 รูปแสดงไอคอนของโปรแกรม AVR Studio 6.2

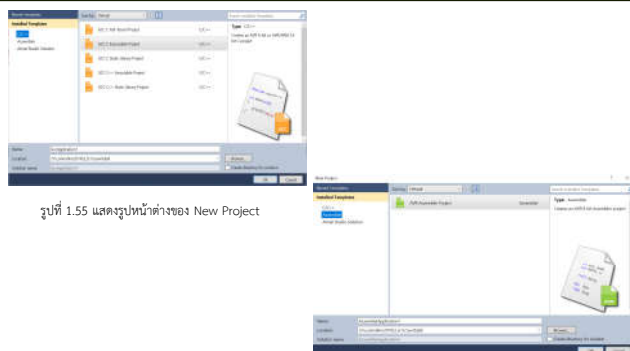


รูปที่ 1.54 รูปภาพแสดงหน้าต่างของโปรแกรม AVR Studio 6.2

Digital And Microcontroller

38

พื้นฐานไมโครคอนโทรลเลอร์



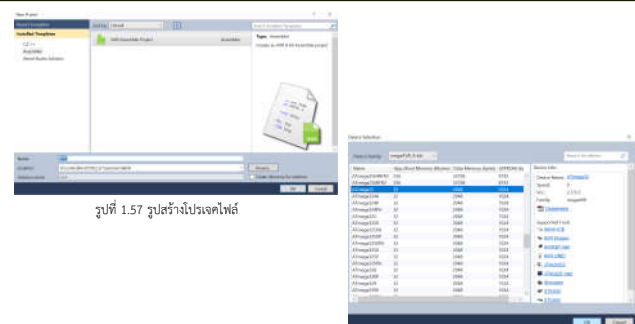
รูปที่ 1.55 แสดงรูปหน้าต่างของ New Project

รูปที่ 1.56 รูปเลือกเครื่องมือในการ compile เป็นแบบ Assembler

Digital And Microcontroller

39

พื้นฐานไมโครคอนโทรลเลอร์



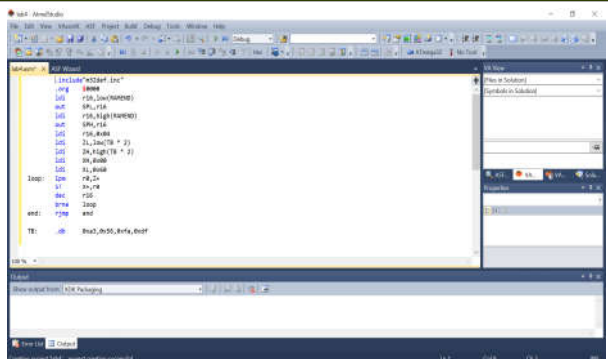
รูปที่ 1.57 รูปสร้างโปรเจกต์ไฟล์

รูปที่ 1.58 รูปหน้าต่างแสดงการเลือกเบอร์ไมโครคอนโทรลเลอร์

Digital And Microcontroller

40

พื้นฐานไมโครคอนโทรลเลอร์

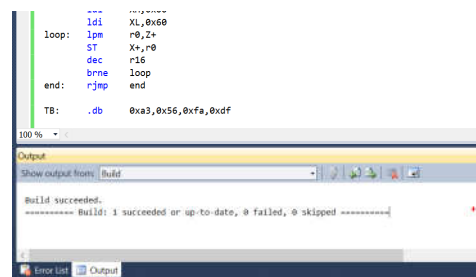


รูปที่ 1.59 รูปแสดงตัวอย่างการเขียนโปรแกรมภาษาแอสเซมบลีของไมโครคอนโทรลเลอร์ AVR

Digital And Microcontroller

41

พื้นฐานไมโครคอนโทรลเลอร์



รูปที่ 1.60 รูปแสดงการ Assembler แล้วไม่เกิดข้อผิดพลาด

Digital And Microcontroller

42

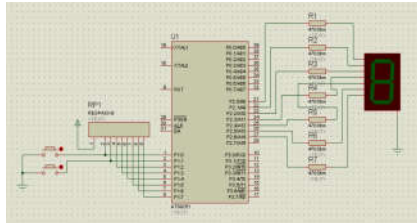
พื้นฐานไมโครคอนโทรลเลอร์

4. การใช้โปรแกรม Proteus เพื่อจำลองการทำงานของวงจรไมโครคอนโทรลเลอร์

4.1 การใช้โปรแกรม Proteus เพื่อจำลองการทำงานของวงจรไมโครคอนโทรลเลอร์ตระกูล MCS51



รูปที่ 1.61 รูปไอคอน Proteus ISIS

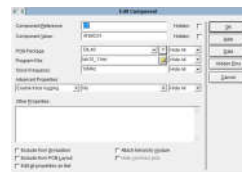


รูปที่ 1.62 ตัวอย่างการต่อวงจรเพื่อทดสอบการทำงานของไมโครคอนโทรลเลอร์ MCS-51

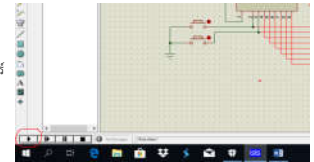
Digital And Microcontroller

43

พื้นฐานไมโครคอนโทรลเลอร์



รูปที่ 1.63 หน้าต่างการกำหนดคุณสมบัติของไมโครคอนโทรลเลอร์ตระกูล MCS-51 เบอร์ AT89C51



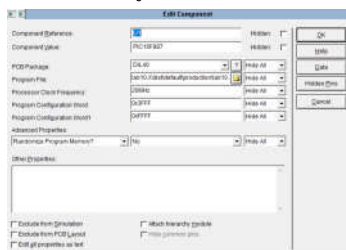
รูปที่ 1.64 แสดงตำแหน่งการคลิกใน play เพื่อเริ่มต้นการจำลองการทำงานของวงจรไมโครคอนโทรลเลอร์ตระกูล MCS-51 เบอร์ AT89C51

Digital And Microcontroller

44

พื้นฐานไมโครคอนโทรลเลอร์

4.2 การใช้โปรแกรม Proteus เพื่อจำลองการทำงานของวงจรไมโครคอนโทรลเลอร์ตระกูล PIC16F



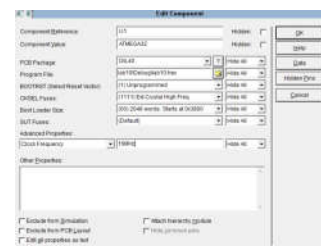
รูปที่ 1.67 หน้าต่างการกำหนดคุณสมบัติของไมโครคอนโทรลเลอร์ตระกูล PIC16F เบอร์ PIC16F887

Digital And Microcontroller

45

พื้นฐานไมโครคอนโทรลเลอร์

4.3 การใช้โปรแกรม Proteus เพื่อจำลองการทำงานของวงจรไมโครคอนโทรลเลอร์ตระกูล AVR



รูปที่ 1.71 หน้าต่างการกำหนดคุณสมบัติของไมโครคอนโทรลเลอร์ตระกูล AVR เบอร์ ATMEGA32

Digital And Microcontroller

46