เครื่องตรวจสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่น

โดย

นายปรัชญา ชำนาญค้า รหัสนักศึกษา B5212484

นายธีรพงษ์ ภูประโชติ รหัสนักศึกษา B5217779

นายสมภพ เขียววิชัย รหัสนักศึกษา B5223367

รายงานนี้เป็นส่วนหนึ่งของการศึกษาวิชา 427499 โครงงานวิศวกรรมโทรคมนาคม

หลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมโทรคมนาคม หลักสูตรปรับปรุง พ.ศ. 2546

สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

ประจำภาคการศึกษาที่ 3 ปีการศึกษา 2555

เครื่องตรวจสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่น

คณะกรรมการสอบโครงงาน

(ผู้ช่วยศาสตราจารย์ ดร.ชุติมา พรหมมาก)
กรรมการ/อาจารย์ที่ปรึกษาโครงงาน

(อ.ดร.บุญส่ง สุตะพันธ์)
กรรมการ

(ผู้ช่วยศาสตราจารย์ ดร. สมศักดิ์ วาณิชอนันต์ชัย)
กรรมการ

มหาวิทยาลัยเทคโนโลยีสุรนารี

มหาวิทยาลัยเทคโนโลยีสุรนารี อนุมัติให้นับรายงานโครงงานฉบับนี้ เป็นส่วนหนึ่งของการศึกษาระดับปริญญาตรี สาขาวิชาวิศวกรรมโทรคมนาคม รายวิชา 427499 โครงงานวิศวกรรมโทรคมนาคม ประจำปีการศึกษา 2555

หัวข้อโครงงาน เครื่องตรวจสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่น

จัดทำโดย นายปรัชญา ชำนาญค้า รหัส B5212484

นายธีรพงษ์ ภูประโชติ รหัส B5217779

นายสมภพ เขียววิชัย รหัส B5223367

อาจารย์ที่ปรึกษา ผู้ช่วยศาสตราจารย์ ดร.ชุติมา พรหมมาก

สาขาวิชา วิศวกรรมโทรคมนาคม

ภาคการศึกษาที่ 3/2555

บทคัดย่อ

โครงงานนี้เป็นการ ศึกษาและพัฒนาโครงสร้างของเครื่องตรวจสอบการ ใช้งานของสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่น (Local area networks) ว่าสามารถใช้งาน ได้หรือไม่ซึ่งปัจจุบันการเข้าหัวของ สายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่นนั้นอาจเกิดปัญหาหลายอย่าง เช่น สายขาดภายในเส้น สายลัดวงจร การเรียงสายย่อยภายในไม่ถูกต้อง เป็นต้น ซึ่งปัญหาเหล่านี้จะทำให้ สายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่นไม่สามารถนำมาใช้งานได้

โครงงานนี้ได้นำเอา Microcontroller MCS -51 เบอร์ AT89C51ED2 มาประยุกต์ใช้งานเพื่อช่วยในการตรวจสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่นโดยมีการใช้โปรแกรมภาษาซี ทำหน้าที่หลักในการควบคุม Microcontroller แล้วทำการตรวจสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่นซึ่งเข้าหัวRJ-45 และแสดงผลออกทางจอแสดงผล LCD

กิตติกรรมประกาศ

โครงงานเรื่อง เครื่องตรวจสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่น สามารถสำเร็จลุล่วงไปได้ด้วยดีเนื่องจากคณะผู้จัดทำโครงงานได้รับความช่วยเหลือด้านต่างๆจากอาจารย์ที่ปรึกษาผู้ช่วยศาสตราจารย์ดร. ชุติมาพรหมมากผู้ที่เป็นแนวความคิดริเริ่มในโครงงานเรื่องเครื่องตรวจสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่น ซึ่งได้ใช้ Microcontroller มาประยุกต์ใช้ ซึ่งได้ให้ความช่วยเหลือและให้คำปรึกษาในทุกๆด้านมาโดยตลอด และขอขอบพระคุณคณาจารย์ บุคลากร และนักศึกษาบัณฑิตศึกษาสาขาวิชาวิศวกรรมโทรคมนาคมทุกท่านที่ให้ความช่วยเหลือแก่คณะผู้จัดทำมาโดยตลอด

นอกจากนี้คณะผู้จัดทำโครงงานขอบคุณเพื่อนๆ สาขาวิศวกรรมโทรคมนาคมทุกๆ คนที่คอยให้ความช่วยเหลือคณะผู้จัดทำโครงงานในด้านต่างๆ คณะผู้จัดทำโครงงานใคร่ขอขอบพระคุณทุกๆท่านที่ได้กล่าวไปแล้วไว้ ณ ที่นี้สำหรับส่วนดีของโครงงานชิ้นนี้ขออุทิศให้แก่คณาจารย์ทุกท่านที่ได้ประสิทธิ์ประสาทวิชาความรู้ให้แก่คณะผู้จัดทำโครงงานหากโครงงานชิ้นดีมีข้อผิดพลาดประการใดทางคณะผู้จัดทำโครงงานใคร่ขอน้อมรับและขออภัยมา ณ ที่นี้

นาย ปรัชญา ชำนาญค้า

นายธีรพงษ์ ภูประโชติ

นายสมภพ เขียววิชัย

มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญ

เรื่อง หน้า

บทคัดย่อ ก

กิตติกรรมประกาศ ข

สารบัญ ค

สารบัญรูป ช

สารบัญตาราง ญ

บทที่ 1 บทนำ

1.1 มาตรฐานการสื่อสารแบบ RS-232 1

1.2มาตรฐานการสื่อสารแบบ RS-422 2

1.3 เคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่าย 3

1.4 Microcontroller 5

1.5 หลักการและเหตุผล 6

1.6วัตถุประสงค์ 7

1.7ขอบเขตงาน 8

1.8ผลที่คาดว่าจะได้รับ 8

บทที่ 2 ทฤษฎีและความรู้พื้นฐาน

2.1 บทนำ 9

2.2 มาตรฐานการสื่อสารแบบ RS-232 9

2.2.1 ความเป็นมาของระบบการสื่อสารผ่านพอร์ตอนุกรมแบบ RS-232 10

2.2.2 ลักษณะของสัญญาณ RS-232C 13

2. 2.3 ข้อต่อของมาตรฐานแบบ RS-232C 14

2.2.4 ขาสัญญาณต่างๆ ของข้อต่อแบบ DB-9 15

2.3 มาตรฐานการสื่อสารแบบ RS-422 17

2.3.1 การติดต่อสื่อสารแบบเข้าจังหวะ (Synchronous) 17

2.3.2 การติดต่อสื่อสารแบบไม่เข้าจังหวะ (Asynchronous) 18

2.3.3 ระบบการสื่อสารมาตรฐานแบบ RS-422 20

2.4 เคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่าย 23

2.4.1 สายตรง (Straight-through cable) 23

มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญ(ต่อ)

เรื่อง	หน้า
2.4.2 สายไขว้ (Crossover cable)	24
2.4.3. สายสลับ (Rollover cable)	24
2.4.4 สาย Serial	25
2.5 อุปกรณ์เครือข่าย	26
2.5.1 Switch	26
2.5.2 router	27
2.5.3 Access Point	28
2.5.3.1 Access Point Mode	28
2.5.3.2 Client Mode (AP Station / AP Client)	28
2.5.3.3. Repeater Mode	29
2.5.3.4. Bridge Mode	29
2.5.3.5 Repeater Mode	30
2.6 การใช้งานของไมโครคอนโทรลเลอร์ MCS-51	31
2.6.1ทฤษฎีพื้นฐานเกี่ยวกับMicrocontroller	33
2.6.2 โครงสร้างโดยทั่วไป ของMicrocontroller	34
2.6.3 กลุ่มของMicrocontroller	35
2.6.4 ภาษาที่ใช้เขียน โปรแกรมควบคุมMicrocontroller	37
2.7 การใช้งานจอแสดงผลผลึกเหลว (Liquid Crystal Display : LCD)	38
บทที่ 3 โครงสร้างของระบบ	44
3.1 บทนำ	44
3.2 ส่วน Hard ware	45
3.2.1 Microcontroller MCS -51	45
3.2.1.1 คุณสมบัติของบอร์ด	46
3.2.1.2 โครงสร้างบอร์ด Microcontroller MCS -51	47
3.2.1.3โครงสร้างไอซี Microcontroller MCS -51 เบอร์ AT89C51ED2 ขั้วต่อสัญญาณต่างๆ	48
3.2.1.4 ขั้วต่อสัญญาณต่างๆ	49

มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญ(ต่อ)

เรื่อง หน้า

3.2.2 จอแสดงผลผลึกเหลว (LCD) 50

3.2.2.1ตำแหน่งของขาและหน้าที่การใช้งาน ของ LCD โมดูล 51

3.2.2.2 คำสั่งควบคุมการแสดงผลของ LCD โมดูล 52

3.2.2.3 ตำแหน่ง Address ของ LCD 53

3.3 ส่วน Software 54

3.3.1 โปรแกรมที่สั่งการให้ Microcontroller ทำงาน 54

3.3.1.1 Flow chart 54

บทที่ 4 การทดสอบเครื่องตรวจสอบสายเคเบิล ที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่น 56

4.1 บทนำ 56

4.2อุปกรณ์ที่ใช้ในการทดลอง 56

4.3ออกแบบการทดลอง 56

4.4วิธีการทดลอง 57

4.4.1โปรแกรม ISIS 7 Professional 75

4.2.2เครื่องตรวจสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่น 62

4.5ผลการทดลอง 65

4.5.1โปรแกรม ISIS 7 Professional 65

4.5.1.1สายตรง 65

4.5.1.2สายไขว้ 65

4.5.1.3สายสลับ 66

4.5.1.4สายขาด 66

4.5.1.5สายที่เรียงผิด 67

4.5.2 เครื่องตรวจสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่น 68

4.5.2.1สายตรง 68

มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญ(ต่อ)

เรื่อง หน้า

4.5.2.2สายไขว้ 68

4.5.2.3สายสลับ 69

4.5.2.4สายขาด 69

4.5.2.5 สายที่เรียงผิด 70

4.5.2.6 ความยาวสาย 71

4.6 วิเคราะห์ผลการทดลอง 72

4.7 สรุปผลการทดลอง 72

บทที่ 5 สรุปผลการทดลองและข้อเสนอแนะ 73

5.1 สรุปผล 73

5.2 ประโยชน์ที่ได้รับ 74

5.3 ปัญหาและข้อเสนอแนะ 74

ประวัติผู้เขียน 75

บรรณานุกรม 76

ภาคผนวกที่1 77

ภาคผนวกที่2 86

ภาคผนวกที3 90

ภาคผนวกที่4 98

มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญรูป

เรื่อง หน้า

รูปที่ 1.1 แสดงตัวอย่างของบอร์ด Microcontroller [3] 6

รูปที่ 2.1 วงจรมาตรฐานผ่านพอร์ตอนุกรมแบบ RS-232 14

รูปที่ 2.2 ขาสัญญาณต่างๆ ของหัวต่อแบบ DB-9 15

รูปที่ 2.3 ข้อมูลที่ส่งและข้อมูลที่รับซึ่งสัมพันธ์กับสัญญาณนาฬิกา 17

รูปที่ 2.4 สัญญาณการรับส่งข้อมูลแบบไม่เข้าจังหวะ 18

รูปที่ 2.5 การส่งข้อมูล 8 บิต 11011101 แบบพาริตี้คู่ และแบบพาริตี้คี่ 19

รูปที่ 2.6 การเชื่อมต่อสัญญาณแบบสมดุลในระบบมาตรฐานการสื่อสารแบบ RS-422 21

รูปที่ 2.7 สัญญาณทางด้านเอาต์พุตของมาตรฐานการสื่อสารแบบ RS-422 22

รูปที่ 2.8 รูปแสดงสาย Serial [4] 25

รูปที่2.9 แสดงรูป Switch [5] 26

รูปที่2.10 แสดงรูป Router [6] 27

รูปที่ 2.11โครงสร้างภายในของไมโครคอนโทรลเลอร์ MCS-51 32

รูปที่ 2.12 รูปร่างและการจัดวางขาต่าง ๆ ของไมโครคอนโทรลเลอร์ MCS-51 33

รูปที่ 2.13 ส่วนประกอบของMicrocontroller [3] 35

รูปที่ 2.14 แผนภาพบล็อกของไอซีภายในจอแสดงผลผลึกเหลว 38

รูปที่ 2.16 ลักษณะและหน้าที่ของขาต่าง ๆ ของจอแสดงผลผลึกเหลว 39

รูปที่ 3.1 เครื่องตรวจสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์ท้องถิ่น 44

รูปที่ 3.2 บอร์ด Microcontroller MCS -51 เบอร์ AT89C51ED2 45

รูปที่ 3.3 โครงสร้างบอร์ด Microcontroller MCS -51 เบอร์ AT89C51ED2 47

รูปที่ 3.4 โครงสร้างไอซี Microcontroller MCS -51 เบอร์ AT89C51ED2 [7] 48

รูปที่ 3.5 การจัดเรียงสัญญาณของพอร์ต I/O ต่างๆของบอร์ด 50

รูปที่ 3.6 จอแสดงผลผลึกเหลว (LCD) โมดูล 16X2 Line [8] 50

รูปที่ 3.6 flow chart การทำงานของบอร์ด 54

รูปที่4.1. แสดงDownload File วงจรเครื่องตรวจสอบสายเคเบิลลงใน โปรแกรม ISIS 7 Professional 57

รูปที่4.2 แสดงDownload Hex File ใส่ Microcontroller 57

รูปที่4.3 แสดงเรียงสายแบบตรง 58

สารบัญรูป(ต่อ)

เรื่อง หน้า

รูปที่4.4 แสดงเรียงสายแบบไขว้ 58

รูปที่4.5 แสดงเรียงสายแบบสลับ 59

รูปที่4.6 แสดงเรียงสายแบบสายขาด 59

รูปที่4.7 แสดงเรียงสายแบบสายที่เรียงผิด 60

รูปที่4.8 แสดงกดปุ่มPlayเพื่อให้วงจรเริ่มทำงาน 60

รูปที่4.9 แสดงกดปุ่มสวิตช์ 1 ครั้งเพื่อให้วงจรทำการตรวจสอบสาย 61

รูปที่4.10 แสดงผลการตรวจสอบที่จอLCD 61

รูปที่4.11แสดงจ่ายไฟ5-12Vให้กับเครื่องตรวจสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่น 62

รูปที่4.12แสดงการทำ Reset บอร์ดMicrocontroller 62

รูปที่4.13 แสดงDownload Hex File ด้วยโปรแกรม Flip ให้กับบอร์ด Microcontroller 63

รูปที่4.14แสดงนำสายเคเบิล (UTP) ที่เตรียมไว้ มาต่อกับหัว RJ 45 ตัวเมียทั้งด้าน input และ output 63

รูปที่4.15 กด Switch เพื่อตรวจสอบสายเคเบิล (UTP) 64

รูปที่4.16 แสดงผลการทดลองที่จอLCD 64

รูปที่4.17 แสดงผลการตรวจสอบสายตรงที่จอLCD 65

รูปที่4.18 แสดงผลการตรวจสอบสายไขว้ที่จอLCD 65

รูปที่4.19 แสดงผลการตรวจสอบสายสลับที่จอLCD 66

รูปที่4.20 แสดงผลการตรวจสอบสายขาดที่จอLCD 66

รูปที่4.21 แสดงผลการตรวจสอบสายขาดที่จอLCD 67

รูปที่4.22แสดงผลการตรวจสอบสายที่เรียงผิดที่จอLCD 67

รูปที่4.23 แสดงผลการทดลองแสดงที่จอLCDของสายตรง 68

รูปที่4.24 แสดงผลการทดลองแสดงที่จอLCDของสายไขว้ 68

รูปที่4.25 แสดงผลการทดลองแสดงที่จอLCDของสายสลับ 69

รูปที่4.26 แสดงผลการทดลองแสดงที่จอLCDของสายขาด 69

รูปที่4.27 แสดงผลการทดลองแสดงที่จอLCDของสายขาด 70

รูปที่4.28 แสดงผลการทดลองแสดงที่จอLCDของสายที่เรียงผิด 70

มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญรูป(ต่อ)

เรื่อง หน้า

รูปที่4.29 แสดงผลการทดลองแสดงที่จอLCDของสายเคเบิล (UTP) ที่มีความยาว131เมตร 71

รูปที่4.30 แสดงผลการทดลองแสดงที่จอLCDของสายเคเบิล (UTP) ที่มีความยาว141เมตร 71

รูปที่4.31 แสดงผลการทดลองแสดงที่จอLCDของสายเคเบิล (UTP) ที่มีความยาว141เมตร 72

มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญตาราง

เรื่อง	หน้า
ตารางที่1.1 แสดงการจัดขาของคอนเน็กเตอร์ อนุกรมแบบ DB9 และหน้าที่การใช้งานต่างๆ[1]	2
ตารางที่1.2 การเข้าหัวแบบสายตรง (Straight-through cable EIA/TIA 568B) [2]	4
ตารางที่1.3 การเข้าหัวแบบสายไขว้ (Crossover cable EIA/TIA 568A & 568B) [2]	4
ตารางที่1.4 การเชื่อมต่อสาย Rollover [2]	5
ตารางที่ 2.1 ย่านของแรงดันไฟฟ้าในสัญญาณ	13
ตารางที่ 3.1 ตารางแสดงตำแหน่งของขา และหน้าที่การใช้งาน LCD [8]	51
ตารางที่ 3.2 ตารางแสดงคำสั่งควบคุมการแสดงผล LCD [9]	52
ตารางที่ 3.3 ตารางแสดงตำแหน่ง Address ของ LCD [9]	53

มหาวิทยาลัยเทคโนโลยีสุรนารี

บทที่1
บทนำ

1.1 มาตรฐานการสื่อสารแบบ RS-232

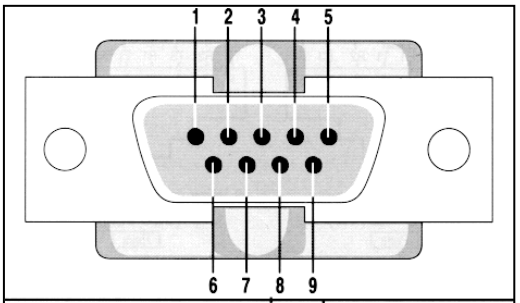
เป็นมาตรฐานการเชื่อมต่อข้อมูลแบบอนุกรมที่มีคนนิยมใช้มากที่สุด กำหนดโดย EIA (Electronics Industry Association) หรือสมาคมผู้ประกอบการอุตสาหกรรมอิเล็กทรอนิกส์ ของอเมริกา ตั้งแต่ปี 1969 โดยมีจุดเริ่มต้นจากความต้องการที่จะกำหนดมาตรฐานการเชื่อมต่อระหว่างคอมพิวเตอร์กับโมเด็มในสมัยนั้น ตัวมาตรฐานจะกำหนดสิ่งที่เกี่ยวข้องกับการเชื่อมต่อนี้ด้วยกันทั้งหมด 4 หัวข้อหลักๆ ด้วยกันคือ

1. คุณสมบัติทางไฟฟ้าของสัญญาณ
2. คุณสมบัติทางกลของการเชื่อมต่อ ซึ่งหมายถึงตัวคอนเน็กเตอร์นั่นเอง
3. หน้าที่การทำงานของวงจรสำหรับแลกเปลี่ยนข้อมูล
4. มาตรฐานการเชื่อมต่อสำหรับระบบสื่อสารเฉพาะอย่าง

มาตรฐาน RS-232-C เป็นมาตรฐาน RS-232 ที่มีการปรับปรุงแก้ไขจากมาตรฐานเดิม ซึ่งเราอาจคุ้นเคยกับชื่อนี้มากกว่า RS-232-A หรือ RS-232-B อันที่จริงแล้วยังมีมาตรฐาน RS-232-D ที่ใหม่กว่า RS-232-C โดยที่มีการเพิ่มข้อกำหนดของคอนเน็กเตอร์แบบ DB เข้าไปด้วย เช่น DB-25 ซึ่งในขณะนั้นสิทธิบัตรของตัวคอนเน็กเตอร์แบบนี้ได้หมดอายุลงพอดี จึงสามารถรวมข้อกำหนดเข้าไว้ได้

ลักษณะโดยทั่วไปของการเชื่อมต่อข้อมูลแบบอนุกรมตามมาตรฐาน RS-232 คือเป็นการสื่อสารข้อมูลแบบจุดต่อจุด ซึ่งเดิมทีเป็นการสื่อสารข้อมูลระหว่างคอมพิวเตอร์กับโมเด็ม ซึ่งจริงๆ แล้วทั้งสองฝั่งจะเป็นอะไรก็ได้ การสื่อสารเป็นแบบสองทางพร้อมกัน (Full-duplex) โดยอาจใช้สายสัญญาณอื่นร่วมเพื่อทำแฮนด์เชค (Hand-shake) หรือไม่ก็ได้ มาตรฐาน RS-232 จำกัดความยาวสายไว้ที่ 50 ฟุต (หรือประมาณ 15 เมตร) สำหรับการส่งสัญญาณที่ความเร็ว 19,200 บิทต่อวินาที โดยที่ความยาวสายจะต้องสั้นลงถ้าต้องการสื่อสารที่ความเร็วสูงขึ้น และถ้ามีสัญญาณรบกวนมากๆ เช่นในโรงงาน หรือบริเวณใกล้เครื่องจักรที่เป็นแบบมีการสวิทช์สัญญาณไฟฟ้าที่กระแสสูงๆ ก็จะทำให้ต้องมีการลดความเร็วในการส่งสัญญาณลงหรือใช้สายที่สั้นลง

มหาวิทยาลัยเทคโนโลยีสุรนารี



Pin	Signal	Pin	Signal
1	Data Carrier Detect	6	Data Set Ready
2	Received Data	7	Request to Send
3	Transmitted Data	8	Clear to Send
4	Data Terminal Ready	9	Ring Indicator
5	Signal Ground		

ตาราง1.1 แสดงการจัดขาของคอนเน็กเตอร์ อนุกรมแบบ DB9และหน้าที่การใช้งานต่างๆ [1]

1.2 มาตรฐานการสื่อสารแบบ RS-422

ถือเป็นพี่ใหญ่ของ RS-232 เป็น โปรโตคอลที่มืออาชีพต้องเลือกใช้ RS-422 ใช้ขั้วต่อแบบ DB 9 ขา หรือ DB 25 เหมือนกับ RS-232 สำหรับส่งคำสั่งและข่าวสารระหว่างเครื่องเล่นเทปและตัวควบคุม กรณีที่ใช้รหัสเวลา การใช้โปรโตคอล RS-422 จะควบคุมความแม่นยำในการเข้าหาภาพในระดับเฟรม (Frame-accurate) อย่างไรก็ตาม RS-422 มีใช้ในอุปกรณ์มืออาชีพเท่านั้น

RS422 สามารถที่จะรับส่งได้ในระยะทางที่ไกลกว่า RS232 ความเร็วในการส่งก็สูงกว่า RS232 ด้วย RS422สามารถส่งได้10Mbpsที่ความยาว 50 ฟุต และ100Kbpsที่ความยาว 4000 ฟุต เป็นแบบ full duplexหรือ haft duplexในการใช้งานจะต้องแปลงสัญญาณ RS232 เป็น RS422 เสียก่อน โดยที่ RS422 จะมีขาที่ใช้ในการส่งข้อมูล Tx สองขา ซึ่งมีเฟสตรงกันข้าม 180 องศา คือขาหนึ่งเป็นลอจิก "1" อีกขาหนึ่งก็จะเป็นลอจิก "0" ทำให้กระแสที่ไหลวนในสายมีค่าคงที่ไม่ว่าจะเป็น 1 หรือ 0 ส่วนขาที่ใช้ในการรับข้อมูลของ RS422 Rx ก็มีสองขาด้วยเช่นกัน ดังนั้นจะเห็นว่า RS422 มีสายสัญญาณเพิ่มขึ้นอีกสองเส้นเป็น 4 เส้น แต่ส่งได้ไกลกว่า RS232 แต่RS422 ก็จะต้องเสียค่าใช้จ่ายของสายสัญญาณเพิ่มขึ้น ก็เลยเกิด การส่งแบบ RS485 ซึ่งเป็นการลดสายสัญญาณออก 2 เส้น ทำให้ไม่สามารถที่จะรับส่งพร้อมๆกันได้เรียกว่า haft duplex แต่มีหลักการรับและส่งแบบเดียวกับ RS422 ทำให้ระยะทางได้ไกลเท่ากัน

1.3 เคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่าย

สาย UTP (Unshielded Twisted Pair) หรือสาย CAT (Category) เป็นสายที่ใช้เชื่อมโยงคอมพิวเตอร์ไปยังอุปกรณ์สื่อสารตามมาตรฐาน EIA/TIA เป็นสายสัญญาณที่มีพื้นที่หน้าตัดกลม ภายในมีสายนำสัญญาณที่มีฉนวนหุ้มอยู่ 4 คู่ แต่ละคู่จะประกอบไปด้วยสายทองแดง 2 เส้น บิดกันเป็นเกลียวเป็นคู่ๆไป เพราะการตีเกลียวคู่เป็นการลดสัญญาณรบกวนอยู่แล้ว สายชนิดนี้จะใช้กับระบบเครือข่ายแบบ 10BASE-T และ 100BASE-TX การใช้งานจะต้องมีการแค็มหัว RJ-45 เข้ากับสาย UTP แล้วนำไปเสียบเข้ากับ Hub มีความเร็วในการรับ-ส่งข้อมูล 10/100Mbps ปัจจุบันนิยมใช้สาย CAT 5 กันมาก เพราะสนับสนุนการรับ-ส่งข้อมูลความเร็วตั้งแต่ 10-100 Mbps แต่กำหนดความยาวได้ไม่เกิน 100 เมตร

สาย UTP ที่ใช้กันอยู่ปัจจุบันจะเป็น 100 ohm ซึ่งแต่เดิมระบบ Ethernet เน็ตเวอร์คจะทำงานที่ความเร็ว 10 Mbps ทำให้การใช้งานสาย UTP ไม่มีปัญหา แต่เมื่อระบบเน็ตเวอร์คพัฒนาจากระบบ Ethernet เป็น Fast Ethernet ทำงานที่ 100Mbps เมื่อความเร็วในการใช้งานสูงขึ้น ความถี่ที่ใช้งานภายในสายก็จะสูงขึ้นด้วย ทำให้อัตราการลดทอนสัญญาณภายในสายมีมากขึ้นด้วย ด้วยเหตุนี้การใช้งานสาย UTP และการเข้าหัว UTP จึงมีบทบาทมากขึ้น สาย UTP แบ่งตามลักษณะการนำสัญญาณความถี่ได้ดังนี้

Category 3:

การเชื่อมต่อกับอุปกรณ์ด้วยค่าความถี่สูงสุดที่ 16 MHz

Category 4:

การเชื่อมต่อกับอุปกรณ์ด้วยค่าความถี่สูงสุดที่ 20 MHz

Category 5:

การเชื่อมต่อกับอุปกรณ์ด้วยค่าความถี่สูงสุดที่ 100 MHz

Category 5e:

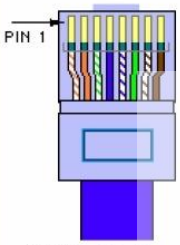
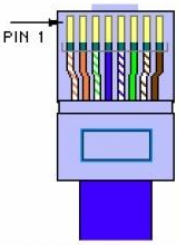
เช่นเดียวกับ Cat 5 แต่มีคุณภาพของสายที่ดีกว่า เพื่อรองรับการส่งข้อมูลแบบฟูลดูเฟล็กซ์ที่1000 Mbps ซึ่งใช้ 4 คู่สาย

มาตรฐาน EIA/TIA 568 มาตรฐานที่กำหนดขึ้นโดยความร่วมมือของ 3 องค์กร ได้แก่ สำนักงานมาตรฐานแห่งสหรัฐอเมริกา (American National Standards Institute : ANSI) , สมาคมอุตสาหกรรมอิเล็กทรอนิกส์ (Electronic Industries Association : EIA) และสมาคม อุตสาหกรรมโทรคมนาคม (Telecommunications Industry Association : TIA) โดยใช้ชื่อ มาตรฐานว่า “EIA/TIA 568” ใช้ในการเข้าหัว RJ-45 และสาย UTP

มหาวิทยาลัยเทคโนโลยีสุรนารี

1.3.1: Straight-through Cable (สายตรง)

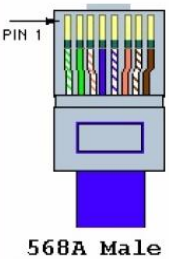
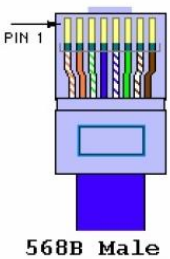
สายสัญญาณแบบ Straight-through คือการต่อสายสัญญาณที่เหมือนกันทั้งสองด้าน อาจจะเป็นแบบ 568A ทั้งสองด้าน หรือแบบ 568B ทั้งสองด้านก็ได้ จะใช้เชื่อมต่อกับอุปกรณ์ต่างชนิดกัน

ปลายสายด้านที่ 1	ลำดับสาย	Symbol	การเรียงสี	ปลายสายด้านที่ 2
	1	TD+	ขาว-ส้ม	
	2	TD-	ส้ม	
	3	RX+	ขาว-เขียว	
	4	Not Assigned	น้ำเงิน	
	5	Not Assigned	ขาว-น้ำเงิน	
	6	RX-	เขียว	
	7	Not Assigned	ขาว-น้ำตาล	
	8	Not Assigned	น้ำตาล	

ตารางที่1.2 การเข้าหัวแบบสายตรง (Straight-through cable EIA/TIA 568B)[2]

1.3.2: Crossover Cable (สายไขว้)

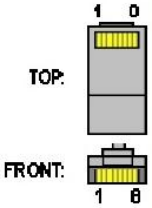
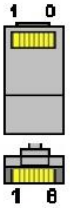
การเลือกมาตรฐานแบบใดแบบหนึ่งเป็นหัวข้างหนึ่ง ส่วนหัวอีกข้างหนึ่งก็เป็นอีกมาตรฐานหนึ่ง อาจเริ่มต้นด้วยแบบ T568A แล้วอีกข้างเป็น T568B หรือสลับกันก็ได้ โดยหัวทั้งสองข้างต้องเรียงสายไม่เหมือนกัน ใช้ในลักษณะการเชื่อมต่อกับอุปกรณ์ชนิดเดียวกัน

ปลายสายด้านที่ 1	ลำดับสาย	การเรียงสี		ลำดับสาย	การเรียงสี	ปลายสายด้านที่ 2
	1	ขาว-เขียว	↔	1	ขาว-ส้ม	
	2	เขียว	↔	2	ส้ม	
	3	ขาว-ส้ม	↔	3	ขาว-เขียว	
	4	น้ำเงิน	↔	4	น้ำเงิน	
	5	ขาว-น้ำเงิน	↔	5	ขาว-น้ำเงิน	
	6	ส้ม	↔	6	เขียว	
	7	ขาว-น้ำตาล	↔	7	ขาว-น้ำตาล	
	8	น้ำตาล	↔	8	น้ำตาล	

ตารางที่1.3 การเข้าหัวแบบสายไขว้ (Crossover cable EIA/TIA 568A & 568B)[2]

1.3.3: RollOver (config Router:สายสลับ)

เป็นสายสัญญาณที่ใช้ในการ Config Router โดยผ่าน Console Port โดยการต่อจะเรียงสีด้านทั้งสองด้านตรงข้ามกัน คือด้านที่ เป็น 1 – 8 ปลายอีกด้านเป็น 8 – 1

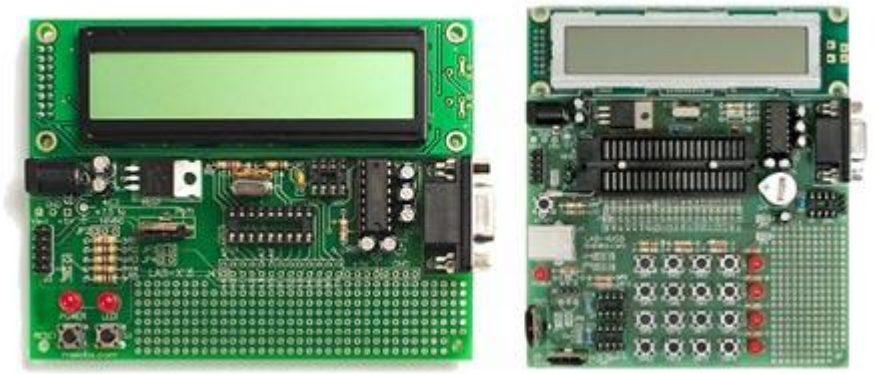
ปลายสายด้านที่ 1	ลำดับสาย	การเรียงสี		ลำดับสาย	การเรียงสี	ปลายสายด้านที่ 2
	1	ขาวส้ม	↔	8	น้ำตาล	
	2	ส้ม	↔	7	ขาว-น้ำตาล	
	3	ขาว-เขียว	↔	6	เขียว	
	4	น้ำเงิน	↔	5	ขาว-น้ำเงิน	
	5	ขาว-น้ำเงิน	↔	4	น้ำเงิน	
	6	เขียว	↔	3	ขาว-เขียว	
	7	ขาว-น้ำตาล	↔	2	ส้ม	
	8	น้ำตาล	↔	1	ขาว-ส้ม	

ตารางที่1.4 การเชื่อมต่อสาย Rollover [2]

1.4 Microcontroller

Microcontroller คือ อุปกรณ์ควบคุมขนาดเล็ก ซึ่งบรรจุความสามารถที่คล้ายคลึงกับระบบคอมพิวเตอร์ โดยในMicrocontrollerได้รวมเอาซีพียู,หน่วยความจำ และพอร์ต ซึ่งเป็นส่วนประกอบหลักสำคัญของระบบคอมพิวเตอร์เข้าไว้ด้วยกัน โดยทำการบรรจุเข้าไว้ในตัวถังเดียวกัน

Microcontroller มาจากMicro + controller ซึ่งจะหมายถึง การควบคุมขนาดเล็ก หรืออาจจะกล่าวได้ว่า Microcontroller ทำหน้าที่เสมือนคอมพิวเตอร์ขนาดเล็กที่ในควบคุมการทำงานของอุปกรณ์ไฟฟ้า หรือ ระบบควบคุมอิเล็กทรอนิกส์ โดยในการควบคุมนั้นจะควบคุมตามคำสั่งที่ได้ทำการเขียนให้Microcontroller ตัวอย่างอุปกรณ์ที่ควบคุมด้วยMicrocontroller เช่น เครื่องซักผ้าแบบอัตโนมัติ(ตั้งโปรแกรมได้) หรือ ระบบสมองกลของรถยนต์ รวมถึงเครื่องใช้ไฟฟ้าอิเล็กทรอนิกส์อีกมากมาย ซึ่งบางรุ่น ก็จะมีตัวMicrocontrollerซ่อนอยู่ภายใน โดยที่ภายในของMicrocontrollerจะมีโครงสร้างหลักอยู่ดังนี้ ส่วนประมวลผล หรือ Processing Unit คือส่วนที่ทำหน้าที่คำนวณทางคณิตศาสตร์ หรือ การตัดสินใจแบบมีเงื่อนไข



รูปที่ 1.1 แสดงตัวอย่างของบอร์ด Microcontroller [3]

1.5 หลักการและเหตุผล

ปัจจุบัน สื่อสารข้อมูลมีบทบาทต่อการดำเนินชีวิตของมนุษย์เป็นอย่างมาก โดยมีระบบอินเตอร์เน็ตซึ่งเป็นเครือข่ายคอมพิวเตอร์ขนาดใหญ่ที่มีการเชื่อมต่อระหว่างเครือข่ายหลายๆ เครือข่ายทั่วโลก โดยใช้ภาษาที่ใช้สื่อสารกันระหว่างคอมพิวเตอร์ที่เรียกว่า โพรโทคอล (Protocol) ผู้ใช้เครือข่ายนี้สามารถสื่อสารถึงกันได้ในหลายๆ ทาง อาทิเช่น อี เมล์ เว็บบอร์ด และสามารถสืบค้นข้อมูลและข่าวสารต่างๆรวมทั้งคัดลอกแฟ้มข้อมูลและโปรแกรมมาใช้ได้ ทำให้ระบบงานเป็นไปได้อย่างมีประสิทธิภาพและเป็นระบบงานแบบอัตโนมัติ

โดยสื่อที่นำมาใช้เป็นตัวกลางในการสื่อสารข้อมูล เครือข่ายท้องถิ่น มีอยู่หลากหลาย แต่ที่นิยมในปัจจุบันคือสาย UTP CAT5 เป็นสายทองแดงตีเกลียวไม่หุ้มฉนวน มีจำนวน 8 เส้น 4 คู่ ทำจากตัวนำคู่ที่มีฉนวนแยกจากกันและพันกันเป็นเกลียวอยู่ภายในปลอกหุ้ม ที่ใช้ในระบบ LAN ที่มีความเร็วปานกลาง MAXIMUM ของ SPEED อยู่ที่ 100Mbps ใช้ในระบบ RING, STAR และแบบผสม โดยในการติดตั้งสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่นให้เป็นระบบอาจจะพบปัญหา เช่น สายขาด สายลัดวงจร การเรียงสายผิด และความสามารถในการรับส่งข้อมูลของสาย ดังนั้นก่อนที่จะมีการนำเครื่องคอมพิวเตอร์มาทำการติดตั้งจริงๆเข้ากับระบบ จำเป็นที่จะต้องตรวจสอบปัญหาเหล่านี้ก่อนที่จะทำการติดตั้งเครื่องคอมพิวเตอร์เข้ากับระบบ เพื่อที่จะเป็นการประหยัดเวลา และค่าใช้จ่าย ในการต้องทำการตรวจสอบเมื่อระบบเครือข่ายได้ติดตั้งเสร็จสมบูรณ์ไปแล้ว

ในการเข้าสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่นที่ใช้สาย UTP ซึ่งเข้าหัวแบบ RJ-45 นั้นมีการเข้าได้ 3 แบบ คือ 1.แบบตรง(Straight-Through Cable) ใช้สำหรับเชื่อมต่อระหว่าง (Switch-Router),(Switch-Com),(Hub-Com) 2. แบบครอส(Crossover Cable) ใช้สำหรับ

เชื่อมต่อระหว่าง (Switch-Switch),(Switch-Hub), (Hub-Hub),(Router-Router),(Router-Com) และ (Com-Com) 3.แบบสลับ(Rollover Cable) ใช้สำหรับเชื่อมต่อระหว่าง (Com-RouterหรือSwitch)

Microcontroller มีส่วนช่วยที่สำคัญในการใช้ทดสอบสาย UTP ซึ่งเลือกใช้บอร์ด Microcontroller MCS-51 รุ่น AT89C51ED2 ของ Philips เป็น MCU ประจำบอร์ด โดยMCUรุ่นนี้จะบรรจุอยู่ภายในตัวถังแบบ 40 Pin โดย MCU ตัวนี้มีจุดเด่นในเรื่องที่สามารถรองรับการดาวน์โหลดโปรแกรมแบบ ISP (In System Programming) ผ่านพอร์ตอนุกรมได้โดยตรง จึงทำให้สามารถใช้งานได้อย่างสะดวก AT89C51ED2 ยังสามารถทำงานในโหมด X2 ซึ่งสามารถทำงานได้เร็วกว่า MCS-51 พื้นฐาน 2 เท่า (6 Clock/1 Machine Cycle) เมื่อใช้คริสตอลความถี่ที่เท่ากัน ในการทำงานในโหมด X2 นี้ AT89C51ED2 สามารถใช้คริสตอลความถี่สูงสุด 20MHz ส่วนในการทำงานในโหมด X1 สามารถใช้คริสตอลความถี่สูงสุด 40MHz ภายใน AT89C51ED2 มีหน่วยความจำโปรแกรมแบบแฟลชขนาด 64Kbyte นอกจากนั้นยังมี หน่วยความจำข้อมูลภายนอกเพิ่มเติมขนาด 1 Kbyte อยู่ภายในตัวชิพด้วย

โครงงานนี้นำเสนอการพัฒนาเครื่องตรวจสอบ สายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่น สำหรับการเชื่อมต่อเป็นประเภทที่ใช้สาย UTPซึ่งเข้าหัว RJ-45โดยใช้โปรแกรมภาษาซี เป็นตัวควบ Microcontroller ซึ่งเป็นตัวหลัก เพื่อใช้ในการตรวจสอบ สายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่นแล้วแสดงผลออกทาง จอแสดงผล LCD ตามที่เราออกแบบไว้

1.6วัตถุประสงค์

1. เพื่อศึกษาโครงสร้างสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่น
2. เพื่อศึกษาการใช้งานของสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่นแต่ละประเภท
3. เพื่อศึกษาวิธีการเข้าหัวของสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่น
4. เพื่อศึกษาการทำงานและประยุกต์การใช้งานโปรแกรม ภาษาซี และMicrocontroller
5. เพื่อสามารถใช้โปรแกรม ภาษาซี มาตรวจสอบสายพร้อมทั้งแสดงออกทาง จอแสดงผล LCDได้
6. เพื่อให้สามารถนำความรู้ที่ได้จากการศึกษาทฤษฎีต่างๆ มาพัฒนาเครื่องสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่นที่สามารถประยุกต์ใช้ได้จริง
7. เพื่อฝึกการทำงานเป็นทีม การแก้ปัญหาต่างๆที่เกิดขึ้น

1.7 ขอบเขตงาน

1 .ศึกษาและพัฒนาเครื่องทดสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่นด้วย Microcontroller

2. ศึกษาและพัฒนาโปรแกรมที่ใช้สั่งการให้ Microcontroller ตรวจสอบความผิดพลาดของสาย

3. ศึกษาและพัฒนาโปรแกรม ภาษาซี ให้สามารถรับข้อมูลจาก Microcontrollerแล้วทำการแสดงผล

1.8 ผลที่จะได้รับ

1.มีความรู้เกี่ยวกับโปรแกรม ภาษาซี และสามารถนำมาประยุกต์ใช้ให้เกี่ยวข้องกับหัวข้อโครงงานได้

2.มีความรู้เกี่ยวกับ Microcontroller และสามารถ นำมาประยุกต์ใช้ให้เกี่ยวข้องกับหัวข้อโครงงานได้

3.มีความรู้เกี่ยวกับ มาตรฐานการสื่อสารของสายUTP และการใช้งานของสายUTP

4.มีความรู้เกี่ยวกับ วิธีการเข้าหัวของสายUTP

5.สามารถแก้ปัญหาจากการปฏิบัติงานจริง เช่น การแก้ปัญหาต่างๆ การทำงานเป็นทีม

มหาวิทยาลัยเทคโนโลยีสุรนารี

บทที่2

หลักการและทฤษฎี

2.1 บทนำ

ในบทนี้จะกล่าวถึงทฤษฎีพื้นฐานที่เกี่ยวข้องกับเครื่องตรวจสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่นเนื้อหาจะประกอบไปด้วยมาตรฐานการสื่อสารแบบ RS-232 มาตรฐานการสื่อสารแบบ RS-422 เคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่าย อุปกรณ์เครือข่ายการใช้งานของไมโครคอนโทรลเลอร์ MCS-51 การใช้งานจอแสดงผลผลึกเหลว (Liquid Crystal Display : LCD)

2.2มาตรฐานการสื่อสารแบบ RS-232

ในยุคแรกๆคอมพิวเตอร์ทุกขนาดจะสื่อสารถึงกันผ่านทางสายสัญญาณซึ่งจะนำพาสัญญาณดิจิตอลล้วนๆในทางปฏิบัติแล้วระยะทางที่ไกลที่สุดที่สามารถส่งข้อมูลได้จะไกลประมาณ 1ไมล์โดยมีความเร็วในการส่งข้อมูลเพียง 300 บิตต่อวินาที (bps) ในรูปแบบของการสื่อสารแบบนี้ผู้ใช้คอมพิวเตอร์หรือผู้ใช้เครื่องลูกข่าย (Terminal) จำเป็นจะต้องติดตั้งสายนำสัญญาณชนิดพิเศษและต้องมีตัวทวนสัญญาณทุกๆ 2-3 พันฟุต เพื่อให้การสื่อสารข้อมูลนั้นเชื่อถือได้ระบบเครือข่ายยุคเริ่มแรกนี้ยากที่จะสื่อสารข้อมูลกับบริเวณที่อยู่นอกเหนือไปจากบริเวณที่ตั้งเครื่องไว้ด้วยมาตรฐา น ในขณะนั้นนั่นเป็นวิธีการที่ง่ายแต่ค่าใช้จ่ายสูงในการขนส่งกลุ่มข้อมูลระหว่างคอมพิวเตอร์กับผู้ใช้ โดยเฉพาะเมื่อผู้ใช้บางส่วนมองว่าการสื่อสารแบบดิจิตอลในระบบนี้เป็นเหมือนตัวขัดขวางอนาคตของการประมวลผลขนาดใหญ่ และการสื่อสารทางไกลในระบบดิจิตอลทั้งหมด ต่อมาเกิดมีแนวคิดในการใช้งานระบบนำสัญญาณสาธารณะทั่วไป(ระบบสายโทรศัพท์ธรรมดา)ในการส่งและรั บข้อมูลดิจิตอลขึ้นมา เครื่องโมเดมหรือชุดข้อมูลก็ถูกผลักดันเข้าสู่การให้บริการ

ชุดของข้อมูลจะถูกจัดเตรียมไว้ที่ศูนย์คอมพิวเตอร์แต่ละแห่งเพื่อใช้ในการผสมสัญญาณและถอดรหัสสัญญาณดิจิตอลที่จะถูกส่งไประหว่างคอมพิวเตอร์เครื่องอื่นๆที่อยู่ห่างไกลออกไปผ่านทางระบบโทรศัพท์แอนะลอก อุปกรณ์ตัวใหม่ที่เรียกว่าตัวผสมสัญญาณและถอดรหัสสัญญาณ (Modulation and Demodulation) ก็ได้กลายมาเป็นสิ่งที่เรารู้จักกันในปัจจุบันในนามของ “โมเดม” (Modem)

2.2.1 ความเป็นมาของระบบการสื่อสารผ่านพอร์ตอนุกรมแบบ RS-232

ด้วยการกำเนิดขึ้นมาของโมเดม วิถีทางในการประมวลผลจากระยะไกลก็สดใสขึ้นแม้ว่าจะมีค่าใช้จ่ายสูงและใครๆ ก็สามารถทำโมเดมขึ้นมาได้ ในเดือนพฤษภาคม ปี ค.ศ. 1960ได้มีเหตุการณ์ที่จำเป็นต้องมีการกำหนดมาตรฐานการต่อร่วม (Interface) ทางไฟฟ้าระหว่างคอมพิวเตอร์กับโมเดม

หรือชุดข้อมูล มีการวางแผนกำหนดแรงดันไฟฟ้ามาตรฐาน และชื่อเรียกมาตรฐานขึ้นมา เพื่อแยกแยะตัวนำสัญญาณในสายส่งสัญญาณที่ต่อกับคอมพิวเตอร์และโมเดมเทคโนโลยีการประมวลผลจากระยะไกล (Teleprocessing Technology) เป็นสิ่งที่ได้รับความนิยมอย่างรวดเร็ว เครื่องคอมพิวเตอร์และเครื่องเทอร์มินอลจำนวนมากที่ถูกสร้างขึ้นโดยผู้ผลิตต่างๆ กันก็จำเป็น จะต้องมีการติดต่อสื่อสารระหว่างกันมากขึ้น ถ้าปราศจากมาตรฐานแบบเดียวกันแล้วอุตสาหกรรมการประมวลผลจากระยะไกล ทั้งหมดก็จะหยุดชะงักลงเนื่องจากความไม่มีมาตรฐานนั้น

ด้วยเหตุนี้คณะกรรมการที่เรียกว่า สมาคมอุตสาหกรรมอิเล็กทรอนิกส์ (Electronic Industries Association: EIA) ซึ่งต่อไปจะขอเรียกว่า EIA ก็ได้จัดตั้งขึ้น คณะกรรมการ EIA นี้ได้วางมาตรฐานที่รู้จักในนามของ EIA RS-232(X) แม้ว่ามันจะเป็นแนวความคิดที่ยอดเยี่ยมแต่คุณลักษณะที่กำหนดขึ้นในครั้งแรกนั้นก็เป็นแบบกว้างๆ และไม่สามารถรับประกันในเรื่องของความเข้ากันได้ (ข้อกำหนด RS-232) ใหม่นี้ ก็ยังมีคู่แข่งนอกสหรัฐอเมริกา ที่รู้จักกันในนามของ CCITT หรือ Consultative Committee for International Telephone and Telegraph

ข้อเสนอของ RS-232 ได้กำหนดการต่อร่วมทั้งทางลอจิกและทางกายภาพระหว่างอุปกรณ์คอมพิวเตอร์กับโมเดม โดยพื้นฐานแล้วจำเป็นการต่อร่วม 2 ชุด เพื่อให้ได้การสื่อสารแบบดิจิตอลที่สมบูรณ์ ระหว่างคอมพิวเตอร์ 2 เครื่อง และเครื่องอุปกรณ์ปลายทางการต่อร่วมชุดแรก คือ ทางเข้าออกของเครื่องคอมพิวเตอร์ หรือเครื่องอุปกรณ์ปลายทางจะมีการต่อร่วมทั้งทางลอจิก และทางกายภาพกับโมเดม และประกอบด้วยสายนำสัญญาณหลายเส้น สำหรับควบคุมส่งและรับข้อมูลและเป็นตัวกำหนดจังหวะและเวลา กล่าวโดยสรุปก็คือการต่อร่วมระหว่างคอมพิวเตอร์กับโมเดมจะมีมาตรฐานอยู่ชุดหนึ่งและเรียกว่าการต่อร่วมแบบ EIA RS-232 การต่อร่วมชุดที่2จะเชื่อมต่อโมเดมเข้ากับช่องทางการสื่อสารต่าง ๆ ที่ต้องการ (สายโทรศัพท์ เส้นใยแสง ดาวเทียม และอื่นๆ) สำหรับผู้ใช้คอมพิวเตอร์ส่วนบุคคลส่วนใหญ่ช่องทางการสื่อสารนี้ก็คือ สาย 2 เส้นของโทรศัพท์แบบ

แอนะลอก มาตรฐาน EIA เริ่มแรกได้กำหนดสายสัญญาณในการต่อร่วมจำนวน 7 เส้น และยังไม่ได้กำหนดตัวข้อต่อ (Connector)แบบเฉพาะขึ้น แรงดันไฟฟ้าของสัญญาณถูกกำหนดว่าต้องมีค่าอย่างน้อย+3 โวลต์ หรือ -3 โวลต์แต่ต้องไม่เกิน 20 โวลต์ เมื่อเทียบกับกราวด์

ในเดือนตุลาคม ค.ศ. 1963 มาตรฐาน RS-232 ได้กลายมาเป็น RS-232A และดัดแปลงให้รวมเอาขั้วต่อแบบ DB-25 ขาเข้าไว้ด้วย โดยมีความยาวสูงสุดของสายสัญญาณอยู่ที่ 50 ฟุต มาตรฐานที่ปรับปรุงขึ้นมาใหม่นี้ก่อให้เกิดความสัมพันธ์ที่คงที่ระหว่างวงจรและจำนวนขาสัญญาณที่เฉพาะเจาะจง และยังแสดงระบบรหัสตัวอักษรสำหรับวงจรการต่อร่วมแต่ละชนิดไว้ด้วย ตัวอักษรแรกของระบบเข้ารหัสนี้กำหนด A สำหรับกราวด์, B สำหรับข้อมูล, C สำหรับการควบคุม และ D สำหรับสัญญาณนาฬิกาให้จังหวะ

วงจรพื้นฐานวงจรดั้งเดิมและการกำหนดระดับสัญญาณว่า –3 โวลต์ เป็นข้อมูล (Mark) และ +3 โวลต์ เป็นช่องว่าง (Space) ก็ยังคงเหมือนเดิม และเพิ่มเติมการกำหนดวงจรเพิ่มขึ้นอีก 10 ข้อ แรงดันไฟฟ้าสูงสุดขณะเปิดวงจรที่ยอมรับได้ถูกเปลี่ยนเป็น 25 โวลต์ กระแสไฟฟ้าสูงสุดระหว่างสายตัวนำ 2 เส้น ใดๆ รวมทั้งกราวด์จะถูกตั้งไว้ที่ 0.5 แอมป์ สายตัวนำที่มีคุณสมบัติในการตอบรับโดยอัตโนมัติ (Autoanswer Capability) ก็ได้เริ่มใช้เป็นครั้งแรก

ในเดือนตุลาคม ค.ศ.1965 ก็เกิดมาตรฐาน RS-232B ซึ่งได้กำหนดค่าอิมพีแดนซ์ที่ปลายทางซึ่งช่วยให้นักออกแบบวงจรสามารถสร้างฮาร์ดแวร์ที่มีความน่าเชื่อถือได้มากยิ่งขึ้นระดับสัญญาณขณะเปิดวงจรยังคงไม่เปลี่ยนแปลงอยู่ที่ –3 ถึง –25 โวลต์ เมื่อเป็นข้อมูลและอยู่ที่ +3 ถึง +25 โวลต์ เมื่อเป็นช่องว่าง แต่มาตรฐาน RS-232B นี้ยังเพิ่มคุณลักษณะทางแรงดันไฟฟ้าที่สำคัญโดยกำหนดว่าสายกราวด์บนขา 7 จะต่อเข้ากับสายกราวด์ของตัวถังบนขา 1 ในโมเดมซึ่งจะทำให้เกิดระดับอ้างอิงของสัญญาณระหว่างโมเดมกับเครื่องคอมพิวเตอร์หรืออุปกรณ์ปลายทาง ข้อกำหนดการการต่อร่วมระหว่างอุปกรณ์ปลายทางของข้อมูลกับอุปกรณ์สื่อสารที่ใช้การแลกเปลี่ยนข้อมูลฐานสองแบบอนุกรม (Interface Between Data TerminalEquipment and Data Communication Equipment Employing Serial Binary DataInterchange) ได้กำหนดขึ้นในเดือนสิงหาคม ค.ศ.1969 มันยังกำหนดคุณลักษณะสายตัวนำให้กระจ่างชัดเจนมากยิ่งขึ้น และระบุว่าวงจรของ RS-232 ส่วนปลายที่เหมาะสมไม่ควรจะมีแรงดันไฟฟ้าเกิน + 15 โวลต์ หรือ - 15 โวลต์

มาตรฐาน RS-232C ก็ประกาศออกมาในภายหลัง และได้กำหนดการต่อร่วมระหว่างอุปกรณ์ปลายทางของข้อมูล(Data Terminal Equipment : DTE) กับอุปกรณ์ส่วนปลายของวงจรข้อมูล (Data Circuit Equipment : DCE) อุปกรณ์เชื่อมต่อข้อมูลตามปกติแล้วจะเป็นอุปกรณ์ปลายทางที่พูดไม่ได้ (Damb terminal) เป็นอุปกรณ์ที่มีความฉลาด ตัวอย่างอุปกรณ์นี้เช่น ไมโครคอนโทรลเลอร์ หรือเครื่องคอมพิวเตอร์ส่วนบุคคลที่มีความสามารถในการที่จะสร้างบิตข้อมูลเป็นแบบอนุกรมได้ เป็นต้น อุปกรณ์วงจรข้อมูลปลายทางจะรับบิตข้อมูลจากอุปกรณ์เชื่อมต่อข้อมูลส่งออกมา ผ่านทางการต่อร่วมแบบ RS-232 และแปลงให้อยู่ในรูปที่เหมาะสมสำหรับถ่ายทอดผ่านตัวกลางการสื่อสารจากระยะไกล เช่น ผ่านทางสายโทรศัพท์ เป็นต้น

ถ้าติดตามมาตรฐาน RS-232C ต่อไป จะพบว่าในทางกายภาพแล้วขั้วต่อทางเข้าออกของอุปกรณ์เชื่อมต่อข้อมูลจะเป็นข้อต่อตัวผู้ และขั้วต่อทางเข้าออกของอุปกรณ์วงจรข้อมูลปลายทางจะเป็นข้อต่อตัวเมีย ทางเข้าออกแบบอนุกรมของเครื่องคอมพิวเตอร์ส่วนบุคคลปกติแล้วจะเป็นอุปกรณ์เชื่อมต่อข้อมูล และทางเข้าออกของโมเดมก็มักจะมีโครงสร้างเป็นอุปกรณ์วงจรข้อมูลปลายทาง

ทางเข้าออกแบบอนุกรมของเครื่องคอมพิวเตอร์จะเป็นตัวผู้ ไม่ว่าจะเป็นขั้วต่อแบบ9 ขา หรือ 25 ขา ส่วนข้อต่อของโมเดมส่วนมากแล้วจะเป็นขั้วต่อแบบตัวเมีย 25 ขา แม้ว่าขั้วต่อแบบ 9 ขา จะไม่ได้เป็นมาตรฐาน RS-232 ของ EIA แต่ในปัจจุบันนี้ก็มีการใช้อยู่ทั่วไปในการต่อร่วมแบบ RS-232 การต่อร่วมที่ใช้ขั้วต่อแบบ 9 ขา พบจำหน่ายเป็นครั้งแรกในเครื่องคอมพิวเตอร์รุ่น AT ในต้นทศวรรษที่ 80

มาตรฐานแบบ RS-232C ยังกำหนดคุณลักษณะสำหรับการสื่อสารผ่านทางสายส่งสัญญาณเสียงด้วยอัตราเร็วสูงถึง 9,600 บิตต่อวินาที ซึ่งช่วยทำให้สามารถมีการติดต่อสื่อสารเป็นแบบอะซิงโครนัสจากจุดหนึ่งไปยังอีกจุดหนึ่ง ผ่านทางโครงข่ายโทรศัพท์สาธารณะได้

มาตรฐานแบบ RS-232C ในระหว่างการส่งข้อมูล สภาวะมีข้อมูล (Mark condition) บ่งชี้ด้วยสถานะ “1” ในระบบเลขฐานสอง และสภาวะช่องว่าง (Space condition) บ่งชี้ด้วยสถานะ “0” ในระบบเลขฐานสอง สำหรับวงจรกำหนดจังหวะเวลาและควบคุมการแลกเปลี่ยนข้อมูลแล้ว การทำงานของตัวมันเองจะเป็น “ON” เมื่อแรงดันไฟฟ้าเป็นบวกมากกว่า + 3 โวลต์และจะเป็น “OFF”

เมื่อแรงดันไฟฟ้าเป็นลบมากกว่า –3 โวลต์เมื่อเทียบกับกราวด์ การทำงานนี้จะไม่สามารถกำหนดได้ ถ้าแรงดันไฟฟ้าอยู่ในช่วงของการเปลี่ยนแปลงระหว่าง –3 ถึง +3 โวลต์เมื่อเทียบกราวด์

คำว่า “Mark” และ “Space” จะใช้อยู่ทั่วไปในเอกสารกำหนดคุณลักษณะ เพื่อบรรยายสภาวะของสายนำข้อมูลหรือสายสัญญาณควบคุมในระบบ RS-232 ในตารางที่ 2.1 กำหนดค่าแรงดันไฟฟ้าต่ำสุดและสูงสุดของข้อกำหนดมาตรฐานแบบ RS-232D และมาตรฐานแบบ V.28 และข้อกำหนดที่จุดปลายของการต่อร่วมบางข้อ ปัจจุบันนี้อุปกรณ์ส่วนใหญ่ที่มีวางจำหน่ายจะเป็นไปตามมาตรฐานแบบ RS-232C หรือมาตรฐานแบบ RS-232D (มาตรฐาน CCITTV.24 และมาตรฐานแบบ V.28 ก็ยังคงมีใช้อยู่ทั่วไปอย่างกว้างขวาง) วงจรมาตรฐานแบบ RS-232 ส่วนมากไม่ได้ใช้ในการติดต่อสื่อสารระหว่างเครื่องอุปกรณ์ปลายทาง 2 เครื่อง หรือระหว่างเครื่องคอมพิวเตอร์ 2 เครื่องโดยตรง

2.2.2 ลักษณะของสัญญาณ RS-232C

เพื่อเป็นหลักประกันว่าข้อมูลถูกส่งออกไปอย่างถูกต้อง และอุปกรณ์ที่ควบคุมถูกควบคุมอย่างถูกต้องจำเป็นต้องมีข้อตกลงกันในเรื่องของสัญญาณที่ใช้มาตรฐาน RS-232Cกำหนดย่านของแรงดันไฟฟ้าในสัญญาณเพื่อสนองจุดประสงค์ดังกล่าวสามารถแสดงให้เห็นได้ดังตารางที่ 2.1

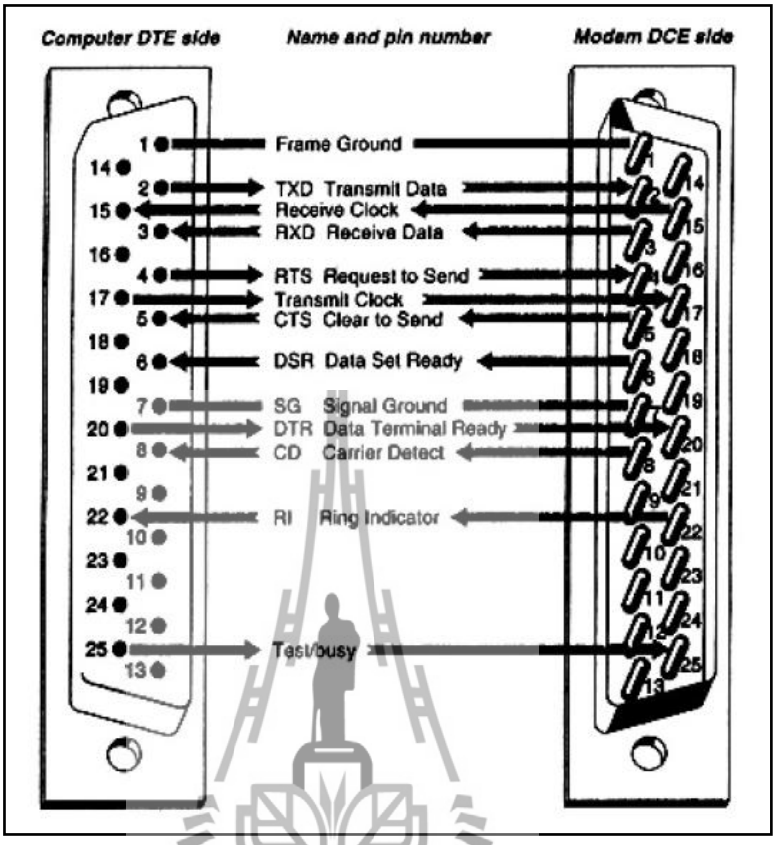
แรงดันไฟฟ้า	สถานภาพลอจิก	สถานภาพสัญญาณ	ฟังก์ชั่นการควบคุม
3 ถึง 25 โวลต์	“ 0 ”	Space	ON
-25 ถึง –3 โวลต์	“ 1 ”	Mark	OFF

ตารางที่ 2.1 ย่านของแรงดัน ไฟฟ้าในสัญญาณ

หมายเหตุ ช่วงระดับแรงดันไฟฟ้าระหว่าง –3 โวลต์ จนถึง + 3 โวลต์ จะเป็นช่วงของการเปลี่ยนแปลงลอจิกดังนั้นจึงไม่มีการระบุสถานะของสัญญาณช่วงนี้ คอมพิวเตอร์บางเครื่องใช้แต่สัญญาณลอจิกที่ออกมาเป็นสัญญาณของ RS-232C เลย ตัวอย่างเช่น อะซิงโครนัสอะแดปเตอร์ของบริษัท International Business Machines Corporation (IBM PC) ในกรณีนี้ระยะความยาวของสายที่ใช้เชื่อมต่ออาจจะสั้นกว่า 50 ฟุต เนื่องจากระดับของกราวด์เปลี่ยนไปอันเป็นผลมาจากการสูญเสียของแรงดันไฟฟ้าในความต้านทานสาย

มหาวิทยาลัยเทคโนโลยีสุรนารี

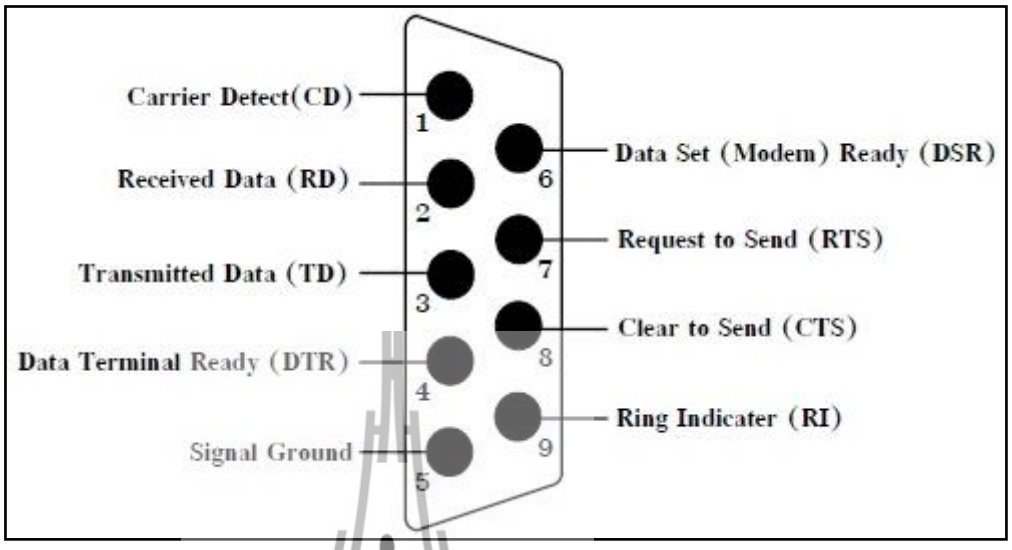
2.2.3 ข้อต่อของมาตรฐานแบบ RS-232C



รูปที่ 2.1 วงจรมาตรฐานผ่านพอร์ตอนุกรมแบบ RS-232

ในปัจจุบันเรามักใช้ข้อต่อแบบ DB-25 ในการต่อร่วมกันตามมาตรฐาน RS-232Cข้อต่อแบบนี้เทียบเท่าแบบมาตรฐาน ISO 2113 เป็นมาตรฐานที่ประกาศโดย องค์การมาตรฐานระหว่างชาติ (International Organization for Standardization : ISO) ข้อต่อตัวผู้จะใช้อุปกรณ์ที่เป็นอุปกรณ์เชื่อมต่อข้อมูลส่วนข้อต่อตัวเมียจะใช้กับอุปกรณ์ที่เป็นอุปกรณ์วงจรข้อมูลปลายทาง

2.2.4 ขาสัญญาณต่างๆ ของข้อต่อแบบ DB-9



รูปที่ 2.2 ขาสัญญาณต่างๆ ของหัวต่อแบบ DB-9

ขา 1 Carrier detect โมเดมจะส่งสัญญาณที่อยู่ในสถานะ ON (ลอจิก “0”) ไปบอกคอมพิวเตอร์ เมื่อได้รับสัญญาณจากโมเดมของอีกฝ่ายหนึ่ง สัญญาณนี้จะไปจุดไดโอดเปล่งแสงบอกว่าได้รับสัญญาณจากโมเดมอีกฝ่ายหนึ่งแล้ว หลอดไดโอดเปล่งแสงจะอยู่บนหน้าเครื่องของโมเดมเอง

ขา 2 Receive data อุปกรณ์เชื่อมต่อข้อมูล จะรับสัญญาณจาก อุปกรณ์วงจรข้อมูลปลายทางเข้ามาทางขานี้ เมื่อไม่มีสัญญาณส่งเข้ามา สถานะของลอจิกที่ขานี้จะเป็น “1”

ขา 3 Transmit data สัญญาณของขานี้จะถูกส่งจาก อุปกรณ์เชื่อมต่อข้อมูล ไปยังอุปกรณ์เชื่อมต่อข้อมูล หรือ อุปกรณ์วงจรข้อมูลปลายทางตัวอื่น เช่น คอมพิวเตอร์ตัวอื่นเครื่องพิมพ์ หรือโมเดม เป็นต้น เมื่อไม่มีสัญญาณส่งออก สถานะของลอจิกที่ขานี้จะเป็น “1”

ขา 4 Data terminal ready อุปกรณ์เชื่อมต่อข้อมูล จะเปิดสัญญาณนี้ให้เปิด (ลอจิก“0”) เมื่อพร้อมที่จะติดต่อกับอุปกรณ์วงจรข้อมูลปลายทาง โดยส่วนมากอุปกรณ์วงจรข้อมูลปลายทาง จะไม่รายงานสถานภาพของตัวเอง เช่น CD , DSR และ CTS เป็นต้น ให้อุปกรณ์เชื่อมต่อข้อมูลได้ทราบ

มหาวิทยาลัยเทคโนโลยีสุรนารี

ดังนั้นหากต้องการให้อุปกรณ์วงจรข้อมูลปลายทางรายงานสถานะภาพต้องเปิดสัญญาณ DTR ให้อยู่ในสถานะเปิด

ขา 5 Signal ground ทำหน้าที่เป็นระดับแรงดันไฟฟ้าอ้างอิง ให้สายสัญญาณทุกๆ สายจะมีแรงดันไฟฟ้าเป็น 0 โวลต์

ขา 6 Data set ready เมื่อสัญญาณขานี้อยู่ในสถานะเปิด (ลอจิก “0”) เป็นการบอกอุปกรณ์เชื่อมต่อข้อมูล ว่า อุปกรณ์วงจรข้อมูลปลายทาง ต่อเรียบร้อยพร้อมที่จะส่งข้อมูลได้แล้ว

ขา 7 Request to send ใช้สำหรับส่งสัญญาณจากอุปกรณ์เชื่อมต่อข้อมูล ไปยัง อุปกรณ์วงจรข้อมูลปลายทาง เพื่อเป็นการร้องขอ ที่จะส่งสัญญาณออกทางขา 2 สัญญาณนี้จะใช้ควบคู่กับขา CTS หรือ Clear to send หากอุปกรณ์ วงจรข้อมูลปลายทาง ได้รับสัญญาณ RTS แล้วจะตรวจสอบตัวเองว่าพร้อมที่จะรับข้อมูลได้หรือยัง หากพร้อมที่จะรับข้อมูลแล้ว จะทำการส่งสัญญาณไปบอกขา CTS

ขา 8 Clear to send เมื่อสัญญาณนี้อยู่ในสภาวะปิด (ลอจิก “1”) หมายความว่าอุปกรณ์รับกำลังบอกว่าพร้อมที่จะรับข้อมูลแล้ว

ขา 9 Ring indicator สัญญาณนี้ใช้ในโมเดมที่เป็นระบบโต้ตอบอัตโนมัติ (AutoAnswer) สัญญาณนี้จะเปิด เมื่อมีสัญญาณกระดิ่งเข้ามา และ ปิด ระหว่างเสียงดังของกระดิ่ง

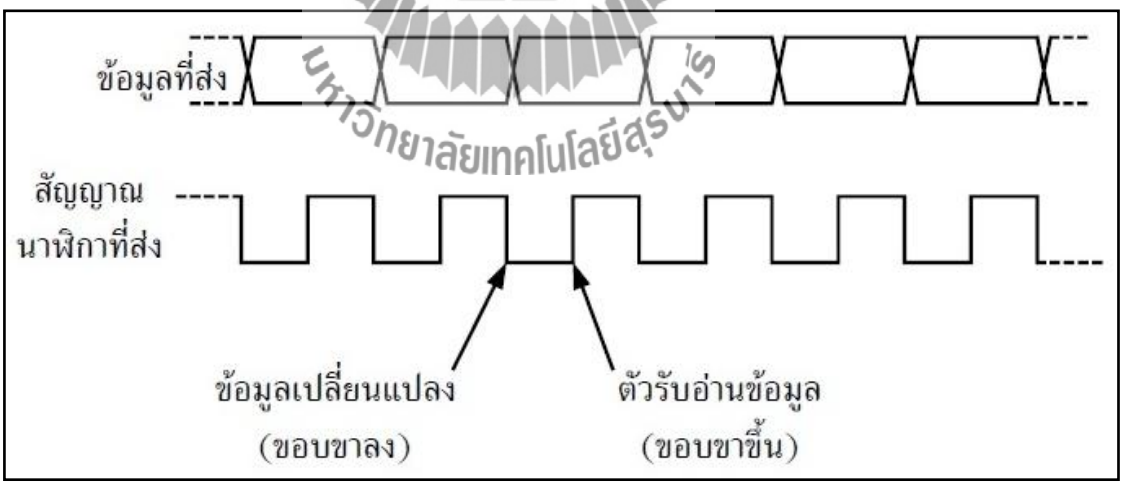
มหาวิทยาลัยเทคโนโลยีสุรนารี

2.3 มาตรฐานการสื่อสารแบบ RS-422

การส่งข้อมูลแบบอนุกรมตามมาตรฐาน RS-232 เป็นที่นิยมกันมากและเป็นแบบแรกที่ใช้สื่อสารกันในระบบคอมพิวเตอร์ส่วนบุคคล จากคุณสมบัติของมาตรฐานการส่งข้อมูลแบบ RS-232 ที่ง่ายต่อการเข้าใจและกฎเกณฑ์ไม่มากจึงทำให้มีการใช้กันอย่างแพร่หลายแต่กระนั้นมาตรฐานการส่งข้อมูลแบบ RS-232 ก็ยังมีขีดจำกัดอยู่หลายอย่างในการสื่อสารซึ่งทำให้เป็นระบบที่ไม่สมบูรณ์ จากที่กล่าวมายังมีมาตรฐานหนึ่งที่มีคุณสมบัติที่เหนือกว่ามาตรฐานแบบ RS-232 นั่นก็คือมาตรฐานการส่งข้อมูลแบบ RS-422 ซึ่งสามารถทำงานได้ในสภาวะแวดล้อมที่เลวร้ายต่างๆ เช่น โรงงานอุตสาหกรรม เป็นต้น และมาตรฐานการส่งข้อมูลแบบ RS-422 ยังสามารถส่งข้อมูลได้ไกลกว่ารวมถึงความเร็วในการรับส่งข้อมูลก็สูงกว่าด้วย

2.3.1 การติดต่อสื่อสารแบบเข้าจังหวะ (Synchronous)

การสื่อสารข้อมูลแบบเข้าจังหวะนี้ จะใช้สัญญาณนาฬิกาเป็นตัวกำหนดตำแหน่งข้อมูลที่ส่งมา โดยตัวส่งจะเปลี่ยนแปลงข้อมูลที่ตำแหน่งขอบขาลงของสัญญาณนาฬิกา ส่วนตัวรับจะอ่านข้อมูลที่ตำแหน่งขอบขาขึ้นของสัญญาณนาฬิกา ซึ่งข้อมูลจะยังคงสภาวะอยู่เช่นเดิมไม่มีการเปลี่ยนแปลงลักษณะการส่งจะเป็นดังรูปที่ 2.3



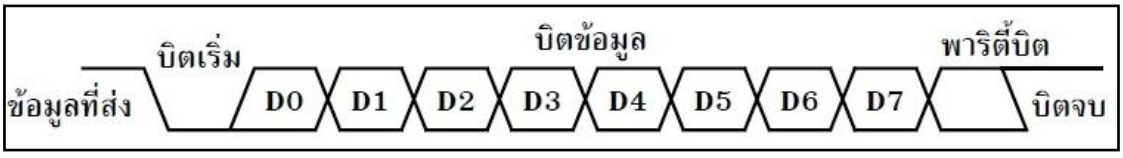
รูปที่ 2.3 ข้อมูลที่ส่งและข้อมูลที่รับซึ่งสัมพันธ์กับสัญญาณนาฬิกา

การติดต่อสื่อสารแบบเข้าจังหวะจะได้เปรียบหรือมีข้อดีก็ตรงที่สามารถส่งข้อมูลได้เร็วและความเที่ยงตรงสูงกว่าการสื่อสารแบบไม่เข้าจังหวะ (Asynchronous) อย่างไรก็ตามการติดต่อสื่อสารแบบนี้มีความยุ่งยากและละเอียดอ่อนกว่าการสื่อสารแบบไม่เข้าจังหวะ การติดต่อสื่อสารแบบเข้าจังหวะส่วนใหญ่จะใช้สำหรับระบบคอมพิวเตอร์ขนาดเล็ก และในคอมพิวเตอร์ระบบเมนเฟรม เพราะระบบต่าง ๆ ต้องใช้ความรวดเร็วในการติดต่อสื่อสาร และความถูกต้องแม่นยำในการทำงาน

2.3.2 การติดต่อสื่อสารแบบไม่เข้าจังหวะ (Asynchronous)

การติดต่อสื่อสารแบบนี้เป็นแบบที่ง่าย และมีความซับซ้อนน้อย นิยมใช้กับเครื่องคอมพิวเตอร์ส่วนบุคคลทั่วไป การติดต่อสื่อสารแบบไม่เข้าจังหวะไม่จำเป็นต้องมีสัญญาณนาฬิกาเข้ามาเกี่ยวข้อง การติดต่อสื่อสารแบบไม่เข้าจังหวะนี้จะต้องกำหนดความเร็วในการรับส่งข้อมูลให้เท่ากัน ข้อมูลจะถูกส่งทีละชุดคือ จะมีบิตเริ่ม (Start bit) และบิตจบ (Stop bit) เป็นตัวบ่งบอกส่วนเริ่มต้นและส่วนสิ้นสุดของข้อมูล ซึ่งบิตเริ่มจะมีสภาวะลอจิกเป็นระดับต่ำ (Low)ตามด้วยข้อมูลอาจจะเป็น 7 บิต หรือ 8 บิต ก็แล้วแต่ ต่อจากนั้นก็ตามด้วยพาริตี้บิต และปิดท้ายด้วยบิตจบ ซึ่งมีสภาวะลอจิกเป็นระดับสูง (High) มีความกว้างเท่ากับ 1 1.5 หรือ2 เท่าของบิตข้อมูล ดังแสดงในรูปที่ 2.4 ซึ่งจุดมุ่งหมายของการมีบิตจบก็คือ

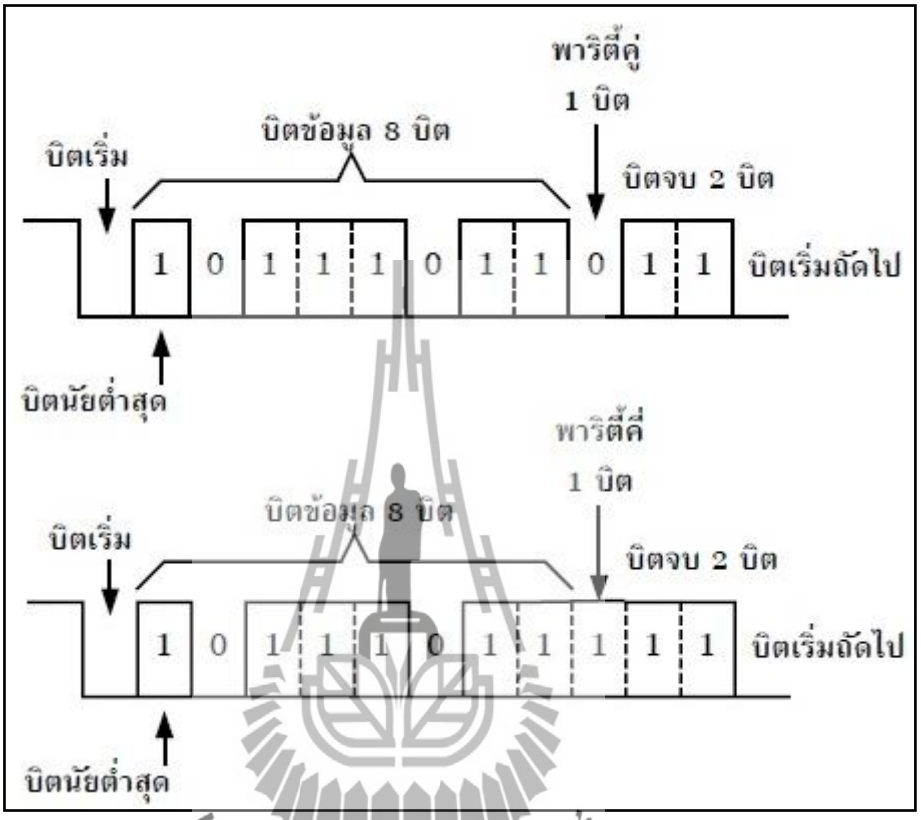
- ใช้ตรวจสอบข้อมูลที่รับได้ว่า ถูกต้องหรือไม่ ถ้าไม่มีการพบบิตจบหลังจากสิ้นสุดการรับข้อมูลแล้ว แฟลกจะแสดงข้อผิดพลาดให้ทราบ
- เพื่อให้ทราบได้ว่าข้อมูลชุดก่อน ได้สิ้นสุด และพร้อมที่จะส่งข้อมูลชุดใหม่ได้



รูปที่ 2.4 สัญญาณการรับส่งข้อมูลแบบไม่เข้าจังหวะ

ยกตัวอย่างการส่งข้อมูล 8 บิตมี ข้อมูลเป็น 11011101 การส่งข้อมูล 8 บิตนี้ จะเป็นดังรูปที่ 2.8 จากรูปจะเป็นการส่งข้อมูลแบบมีพาริตี้คู่ และแบบพาริตี้คี่ ซึ่งความสำคัญของการส่งข้อมูลแบบมีพาริตี้คือเพื่อตรวจสอบว่าข้อมูลที่รับได้นั้นถูกต้องตามที่ส่งมาหรือไม่การติดต่อสื่อสารแบบนี้มักใช้ไอซีที่ทำหน้าที่ในการเป็นตัวรับข้อมูลแบบไม่เข้าจังหวะที่มีชื่อว่า UART (Universal Asynchronous Reciver/Transmitter) ซึ่งใช้สัญญาณความถี่นาฬิกาเป็นตัวกำหนดอัตราการรับส่ง

ข้อมูล โดยมีค่าเท่ากับ 16 เท่าของอัตราการรับส่งข้อมูล เช่น มีค่าเป็น 153600 รอบต่อวินาที สำหรับอัตราการรับส่งข้อมูล 9600 บิตต่อวินาที โดยปกติถ้ายังไม่มีการส่งข้อมูลใด ๆ ในสายส่งสัญญาณจะเป็นลอจิก สูง อยู่



รูปที่ 2.5 การส่งข้อมูล 8 บิต 11011101 แบบพาริตี้คู่ และแบบพาริตี้คี่

เมื่อมีการเริ่มส่งข้อมูลสภาวะของสายสัญญาณจะเปลี่ยนเป็นค่าเป็นระดับ ต่ำ แสดงถึงบิตเริ่ม ตัวรับส่งสัญญาณข้อมูล UART จะหน่วงเวลาโดยการนับสัญญาณนาฬิกาไป 8 ลูกคลื่นเพื่อตรวจสอบอีกครั้งเพื่อความแน่ใจ (ที่จุดนี้เป็นจุดกึ่งกลางของบิต เพราะ 1 บิตจะกว้าง16 ลูกคลื่นของสัญญาณนาฬิกา) ถ้าพบว่ามีสถาวะลอจิกเป็นค่า สูง แสดงว่าเมื่อสักครู่เป็นสัญญาณรบกวนก็จะไม่มีการสนใจ แต่ถ้าหากพบว่าลอจิกยังคงเป็นค่า ต่ำ มันก็จะเริ่มรับข้อมูลโดยการนับสัญญาณนาฬิกาไปทีละ 16 ลูกคลื่น แล้วอ่านข้อมูลไปเก็บในชิฟต์รีจิสเตอร์ทีละบิต

หลังจากข้อมูลทั้งหมดและพาริตี้บิตถูกรับมาหมดแล้วบิตที่จะตามมาคือบิตจบ ถ้าเป็นลอจิก ต่ำ ตัวรับส่งสัญญาณ UART จะระบุว่าไม่สามารถรับข้อมูลได้อย่างถูกต้อง และแสดง

ข้อผิดพลาดขึ้นมา เพราะฉะนั้นอัตราการรับส่งข้อมูลที่ถูกต้องเที่ยงตรงทั้งตัวรับและตัวส่ง จึงมีความสำคัญมาก เพราะถ้าพลาดไปแล้ว จะทำให้การรับส่งข้อมูลผิดพลาดทันที

เมื่อตัวรับอ่านข้อมูลได้ 8 หรือ 9 บิต (รวมกับพาริตี้บิต) ได้อย่างถูกต้อง บิตต่อมาจะเป็นบิตจบ ซึ่งมีลอจิก สูง มีช่วงเวลาการคงสภาวะเป็น 1 1.5 หรือ 2 ของช่วงเวลาแต่ละบิตหากถูกต้องหมดทุกอย่าง ตัวรับส่งสัญญาณข้อมูล UART จะทำการตรวจสอบข้อมูลที่ได้รับมาว่าถูกต้องหรือไม่ โดยตรวจสอบว่าเป็นพาริตี้คู่ หรือ พาริตี้คี่ ถ้าพาริตี้ไม่ถูกต้องตามกำหนดแฟลกจะบอกข้อผิดพลาดทันที

ตัวรับส่งสัญญาณข้อมูล UART ส่วนมากสามารถรับข้อมูลเข้ามาที่ละ 2 ชุด โดยชุดแรกเก็บไว้ในชิฟต์รีจิสเตอร์ และที่เหลือเก็บไว้ในโฮลดิ้งรีจิสเตอร์ (Holding Register) ข้อมูลที่รับได้จะเลื่อนชิฟต์รีจิสเตอร์ไปยังโฮลดิ้งรีจิสเตอร์ เพื่อรอให้ตัวประมวลผลกลางอ่านข้อมูลออกไปเมื่อเลื่อนข้อมูลไปแล้วชิฟต์รีจิสเตอร์ ก็จะรับข้อมูลเข้ามาใหม่เพื่อเก็บและเตรียมส่งให้กับโฮลดิ้งรีจิสเตอร์ต่อไป

อย่างไรก็ตามถ้าตัวประมวลผลกลางไม่อ่านข้อมูลที่อยู่ในโฮลดิ้งรีจิสเตอร์ก่อนที่ข้อมูลตัวใหม่จะถูกส่งมา ข้อมูลเก่าที่เก็บไว้จะถูกทับโดยข้อมูลตัวใหม่ ทำให้ข้อมูลไม่ครบถ้วน ทำให้เกิดข้อผิดพลาดในการรับส่งข้อมูลได้ ในปัจจุบันได้มีการพัฒนา ตัวรับส่งสัญญาณข้อมูล UARTที่ดีกว่าเดิม ซึ่งจะมีการเก็บข้อมูลแบบ เข้าก่อนออกก่อน (First In Firtst Out : FIFO)หมายความว่า จะเก็บข้อมูลเป็นชุด ๆ ชุดละ 16 ตัวอักษร พอเต็มก็จะมีการทับของข้อมูลช่วยทำให้ตัวประมวลผลกลางมีเวลาในการอ่านข้อมูลมากขึ้นสำหรับวิธีอื่นที่จะป้องกันการสูญหายของข้อมูลก็มีดังนี้

- ควบคุมอัตราเร็วการรับส่งข้อมูล คือ ตัวรับข้อมูลสามารถที่จะสั่งให้ตัวส่งข้อมูลหยุดการส่งข้อมูลชั่วคราว เมื่อมีการรับข้อมูลไม่ทัน

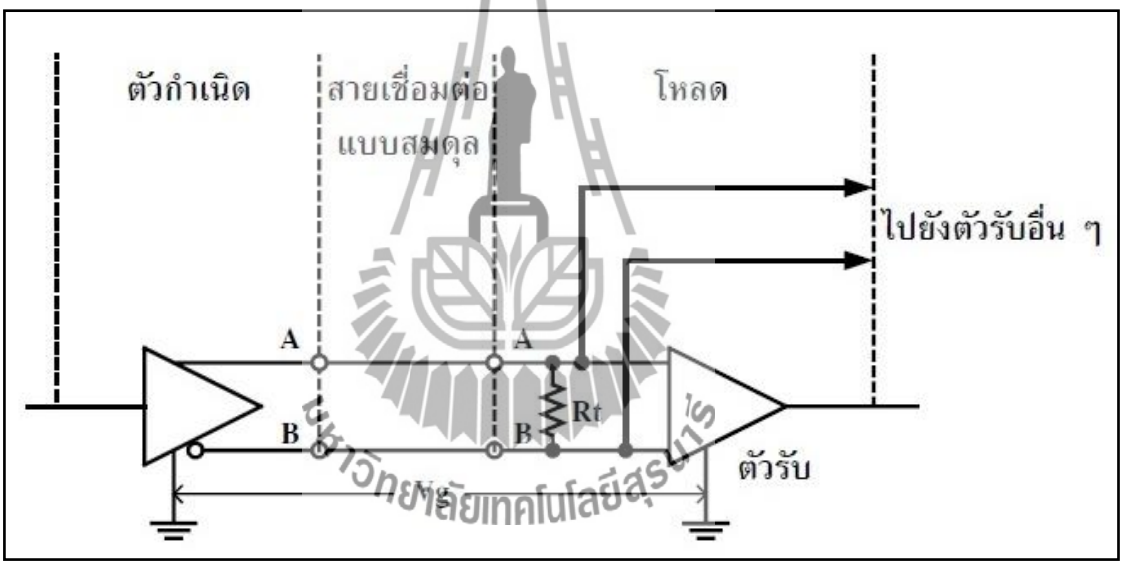
- ควบคุมทางอุปกรณ์อิเล็กทรอนิกส์ โดยใช้สัญญาณต่าง ๆ ที่พอร์ต RS-232 ซึ่งเป็นพอร์ตการสื่อสาร เพื่อบอกความพร้อมของการรับส่งข้อมูล ซึ่งมีชื่อของสายสัญญาณว่า RTS(Request To Send) และสายสัญญาณ CTS (Clear To send)

2.3.3 ระบบการสื่อสารมาตรฐานแบบ RS-422

เมื่อไม่นานมานี้ EIA ได้ออกแบบมาตรฐานชนิดใหม่โดยพัฒนาจากของเดิมเพื่อมาใช้แทนมาตรฐานแบบ RS-232 ในส่วนขีดจำกัดของมาตรฐานแบบ RS-232 โดยได้ออกแบบให้มาตรฐานแบบ RS-422 มีความสามารถส่งข้อมูลได้ไกล ๆ คือสามารถส่งได้ไกลถึง 4,000 ฟุตและความเร็วใน

การส่งข้อมูลถึง 10 ล้านบิตต่อวินาที ที่มีความสามารถสูงเช่นนี้เพราะใช้การแยกสายสัญญาณออกเป็น 2 เส้น ต่อ 1 สัญญาณ โดยไม่ใช้กราวด์ร่วม เรียกว่า เป็นระบบสมดุลอีกทั้งใช้ระดับสัญญาณที่ + 5 โวลต์ จึงไม่จำเป็นต้องใช้แหล่งจ่ายไฟเพิ่มเติมเหมือนกับมาตรฐานแบบ RS-232 อย่างไรก็ตามที่ความเร็ว 90 กิโลบิตต่อวินาที ยอมให้ส่งได้ไกลสูงสุด 4,000 ฟุตความเร็ว 2 เมกะบิตต่อวินาที ที่ระยะทาง 200 ฟุต และด้วยความเร็ว 10 เมกะบิตต่อวินาทีที่ระยะทาง 35 ฟุต

ใน 1 เส้นสายสัญญาณ สายทั้ง 2 เส้นจะตีเกลียวเข้าด้วยกันทำให้มีอิมพิแดนซ์เท่ากันและกระแสไฟฟ้าในสายจะเท่ากันแต่ทิศทางตรงข้ามกัน สัญญาณรบกวนจะหักล้างกันหมดไปส่วนสายกราวด์นั้นจะใช้เพียงอ้างอิงระดับแรงดันไฟฟ้าเท่านั้น โดยไม่ได้เป็นทางผ่านของสัญญาณแต่อย่างใด ลักษณะการเชื่อมต่อแบบนี้แสดงดังรูปที่ 2.6

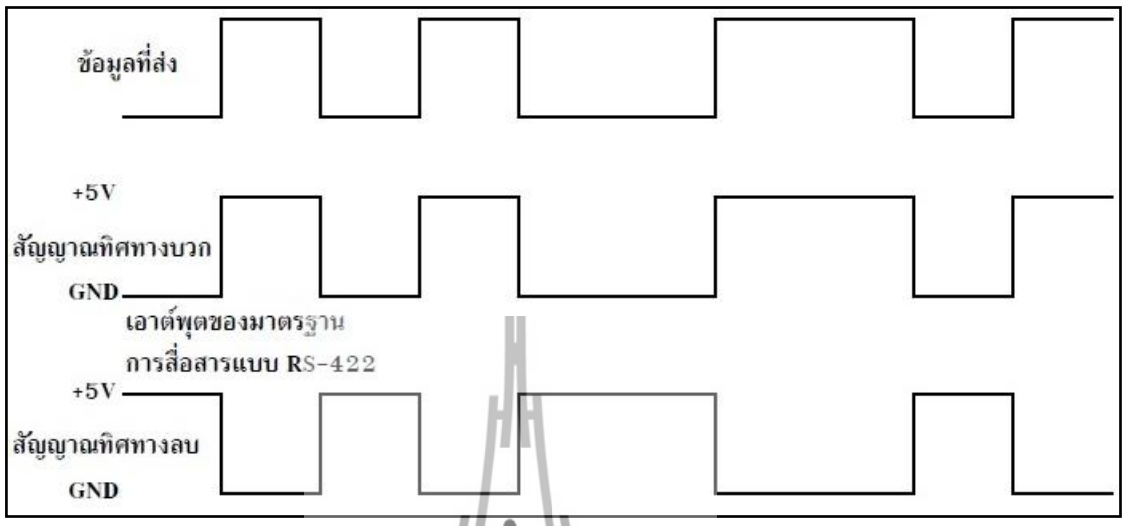


รูปที่ 2.6 การเชื่อมต่อสัญญาณแบบสมดุลในระบบมาตรฐานการสื่อสารแบบ RS-422

ในการส่งสัญญาณแบบนี้นั้น ขณะที่เส้นหนึ่งมีแรงดันไฟฟ้า + 5 โวลต์ เส้นที่คู่กันจะเป็น0 โวลต์ และในทางตรงกันข้าม เมื่อสายเส้นหนึ่งเป็น 0 โวลต์ อีกเส้นหนึ่งจะเป็น + 5 โวลต์ดังนั้นการเปลี่ยนแปลงแรงดันไฟฟ้าระหว่างสาย 2 เส้น จะมีค่าเป็น 10 โวลต์ ดังรูปที่ 2.10

สัญญาณมาตรฐานการสื่อสารแบบ RS-422 จะแบ่งเป็น 1 คู่สายต่อ 1 สัญญาณเพื่อลดค่าอิมพิแดนซ์ในสาย ทำให้ผลของความไวต่อสัญญาณรบกวนมีน้อยมาก ถึงแม้มาตรฐานการสื่อสารแบบ RS-422 จะมีคุณสมบัติดีกว่ามาตรฐานการสื่อสารแบบ RS-232 แต่ก็ยังมีความนิยม

น้อย ปัจจุบันกำลังเป็นที่นิยมใช้กันในโรงงานอุตสาหกรรมที่เป็นแบบอัตโนมัติและต้องการความเชื่อถือสูง



รูปที่ 2.7 สัญญาณทางด้านเอาต์พุตของมาตรฐานการสื่อสารแบบ RS-422

มหาวิทยาลัยเทคโนโลยีสุรนารี

2.4 เคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่าย

สายสัญญาณที่ได้รับความนิยมใช้ในการเชื่อมต่ออุปกรณ์สื่อสารในเครือข่ายคือสาย UTP (Unshielded Twisted Pair) เป็นสายมีขนาดเล็กไม่มีชีลด์ห่อหุ้ม มีเส้นตีเกลียวเป็นคู่ๆ เพื่อลดสัญญาณรบกวน ในการเชื่อมต่อจะใช้หัวต่อแบบ RJ-45 จำนวน 2 หัวต่อสาย 1 เส้น สามารถต่อสายได้ยาวสูงสุดประมาณ 100 เมตร

ชนิดของสาย UTP ที่มีใช้งานในปัจจุบัน มีดังนี้

- Category 3 (CAT 3) วิ่งที่ความเร็ว 10 Mbps ใช้ในเครือข่ายประเภท 10BASET Ethernet
- Category 5 (CAT 5) วิ่งที่ความเร็ว 100 Mbps ใช้ในเครือข่ายประเภท Fast Ethernet
- Category 5e (CAT 5e) วิ่งที่ความเร็ว 1000 Mbps (1 Gbps) ใช้ในเครือข่ายประเภท Gigabit Ethernet
- Category 6 (CAT 6) วิ่งที่ความเร็ว 1000 Mbps (1 Gbps) ใช้ในเครือข่ายประเภท Gigabit Ethernet

สาย CAT 5 และ CAT 5e เป็นสายที่ใช้กันมากในเครือข่าย เนื่องจากเครือข่ายในปัจจุบันเป็นเครือข่ายประเภท Fast Ethernet และ Gigabit Ethernet สายชนิดดังกล่าวประกอบด้วยสายไฟ twisted pair จำนวน 4 คู่ แต่ในการเชื่อมต่ออุปกรณ์เครือข่ายจะใช้เพียง 2 คู่เท่านั้น การเรียงตัวของ twisted pair ในการเข้าหัว RJ-45 ของสาย UTP มีอยู่ 3 แบบคือ

2.4.1 สายตรง (Straight-through cable)

ใช้ในการเชื่อมต่อเครื่องคอมพิวเตอร์เข้ากับ Ethernet Switch/Hub หรือ Cisco router กับ Ethernet Switch/Hub การเรียงตัวของสายไฟในหัว RJ-45 ทั้งสองด้านมีลักษณะดังนี้

ขาที่ 1 สีขาว-ส้ม	----------	ขาที่ 1 สีขาว-ส้ม
ขาที่ 2 สีส้ม	----------	ขาที่ 2 สีส้ม
ขาที่ 3 สีขาว-เขียว	----------	ขาที่ 3 สีขาว-เขียว
ขาที่ 4 สีน้ำเงิน	----------	ขาที่ 4 สีน้ำเงิน
ขาที่ 5 สีขาว-น้ำเงิน	----------	ขาที่ 5 สีขาว-น้ำเงิน
ขาที่ 6 สีเขียว	----------	ขาที่ 6 สีเขียว
ขาที่ 7 สีขาว-น้ำตาล	----------	ขาที่ 7 สีขาว-น้ำตาล
ขาที่ 8 สีน้ำตาล	----------	ขาที่ 8 สีน้ำตาล

2.4.2 สายไขว้ (Crossover cable)

ใช้ในการเชื่อมต่อเครื่องคอมพิวเตอร์กับเครื่อ งคอมพิวเตอร์, เชื่อมต่อ router กับ router, เชื่อมต่อ Switch/Hub กับ Switch/Hub การเรียงตัวของสายไฟในหัว RJ-45 ทั้งสองด้านมีลักษณะดังนี้

ขาที่ 1 สีขาว-ส้ม	----------	ขาที่ 1 สีขาว-เขียว
ขาที่ 2 สีส้ม	----------	ขาที่ 2 สีเขียว
ขาที่ 3 สีขาว-เขียว	----------	ขาที่ 3 สีขาว-ส้ม
ขาที่ 4 สีน้ำเงิน	----------	ขาที่ 4 สีน้ำเงิน
ขาที่ 5 สีขาว-น้ำเงิน	----------	ขาที่ 5 สีขาว-น้ำเงิน
ขาที่ 6 สีเขียว	----------	ขาที่ 6 สีส้ม
ขาที่ 7 สีขาว-น้ำตาล	----------	ขาที่ 7 สีขาว-น้ำตาล
ขาที่ 8 สีน้ำตาล	----------	ขาที่ 8 สีน้ำตาล

2.4.3. สายสลับ (Rollover cable)

ใช้ในการเชื่อมต่อเครื่องคอมพิวเตอร์เข้ากับ Console port ของ router หรือ switch โดย Console port เป็นพอร์ตประเภท asynchronous serial port ที่ใช้ในการส่งและรับตัวอักษร ASCII เพื่อทำการ config พารามิเตอร์ให้กับ router หรือ switch การเรียงตัวของสายไฟในหัว RJ-45 ทั้งสองด้านมีลักษณะดังนี้

ขาที่ 1 ต่อกับขาที่ 8

ขาที่ 2 ต่อกับขาที่ 7

ขาที่ 3 ต่อกับขาที่ 6

ขาที่ 4 ต่อกับขาที่ 5

ขาที่ 5 ต่อกับขาที่ 4

ขาที่ 6 ต่อกับขาที่ 3

ขาที่ 7 ต่อกับขาที่ 2

ขาที่ 8 ต่อกับขาที่ 1

มหาวิทยาลัยเทคโนโลยีสุรนารี

2.4.4 สาย Serial

Serial Port คือ พอร์ตอนุกรม ในการสื่อสารข้อมูลนั้นพอร์ตอนุกรมจะมีความเร็วในการสื่อสารที่ช้ากว่าแบบ ขนาน เพราะการเคลื่อนย้ายข้อมูลแบบอนุกรมนั้นเป็นการส่งข้อมูลครั้งละ 1 บิต แต่พอร์ตขนานนั้นสามารถส่งข้อมูลทีละหลายๆ บิต พร้อมๆ กันได้ แต่ข้อดีของการสื่อสารข้อมูลแบบอนุกรมคือ สามารถส่งข้อมูลได้ในระยะทางที่ไกลกว่าแบบขนาน และใช้สายสัญญาณที่น้อยกว่าการสื่อสารข้อมูลแบบขนาน

รูปที่ 2.8 รูปแสดงสาย Serial [4]

2.5 อุปกรณ์เครือข่าย

2.5.1 Switch

อุปกรณ์ Switch จะทำงานในการ รับ-ส่งข้อมูลที่สามารถส่งข้อมูลจากพอร์ตหนึ่งของอุปกรณ์ ไปยังเฉพาะพอร์ตปลายทางที่เชื่อมต่ออยู่กับอุปกรณ์หรือเครื่องคอมพิวเตอร์ที่ต้องการส่งข้อมูลไปหาเท่านั้น ซึ่งจากหลักการทำงานในลักษณะนี้ ทำให้พอร์ตที่เหลือของอุปกรณ์ Switch ที่ไม่เกี่ยวข้องกับการรับ-ส่งข้อมูลนั้น สามารถทำการ รับ-ส่งข้อมูลกันได้พร้อมกันในเวลาเดียวกัน Switch (สวิตซ์) บางทีก็เรียกว่า Switching Hub (สวิตชิ่งฮับ) ซึ่งในช่วงแรกนั้นจะเรียกว่า Bridge (บริดจ์) เหตุผลที่เรียกว่าบริดจ์ในช่วงแรกนั้น เพราะส่วนใหญ่บริดจ์จะมีแค่สองพอร์ต และใช้สำหรับแยกคอลลิชันโดเมน ปัจจุบันที่เรียกว่า Switch เพราะหมายถึง บริดจ์ที่มีมากกว่าสองพอร์ตนั่นเอง

Switch จะฉลาดกว่า Hub คือ Switch สามารถส่งข้อมูลที่ได้รับมาจากพอร์ตหนึ่งไปยังเฉพาะพอร์ตที่เป็นปลายทางเท่านั้น ทำให้คอมพิวเตอร์ที่เชื่อมต่อกับพอร์ตที่เหลือสามารถส่งข้อมูลถึงกัน และกันได้ในเวลาเดียวกัน การทำเช่นนี้ทำให้อัตราการส่งข้อมูล หรือแบนด์วิธไม่ขึ้นอยู่กับจำนวนคอมพิวเตอร์ที่เชื่อมต่อเข้ากับ Switch คอมพิวเตอร์ทุกเครื่องจะมีแบนด์วิธเท่ากับแบนด์วิธของ Switch

ด้วยข้อดีนี้เครือข่ายที่ติดตั้งใหม่ในปัจจุบันส่วนใหญ่จะนิยมใช้ Switch มากกว่า Hub เพราะจะไม่มีปัญหาเกี่ยวกับการชนกันของข้อมูลในเครือข่าย

รูปที่2.9 แสดงรูป Switch [5]

2.5.2 router

Router คือ อุปกรณ์ที่ทำหน้าที่เชื่อมต่อระบบเครือข่ายอย่างหนึ่ง ซึ่งถ้าแปลความหมายคำว่า Route ก็คือ ถนน นั่นเอง ดังนั้น การเชื่อมต่อคอมพิวเตอร์ด้วย Router ทำให้เราสามารถเชื่อมต่อคอมพิวเตอร์ได้มากกว่าหนึ่งเครื่องในเวลาเดียวกัน ซึ่ง Router นั้นจะมีซอฟต์แวร์ที่ใช้ในการควบคุมการทำงานเรียกว่า Internetwork Operating System (IOS) และตัว Router จะมีช่องที่ใช้เสียบต่อสายสัญญาณเรียกว่า Port LAN ซึ่งโดยทั่วไปมักมี 4 Ports หรือมากกว่า ใน Router 1 ตัว

หน้าที่หลักของ Router คือการหาเส้นทางในการส่งผ่านข้อมูลที่ดีที่สุด และเป็นตัวกลางในการส่งต่อข้อมูลไปยังเครือข่ายอื่น ทั้งนี้ Router สามารถเชื่อมโยงเครือข่ายที่ใช้สื่อสัญญาณหลายแบบแตกต่างกันได้ไม่ว่าจะเป็น Ethernet, Token Rink หรือ FDDI ทั้งๆที่ในแต่ละระบบจะมี packet เป็นรูปแบบของตนเองซึ่งแตกต่างกัน โดยโปรโตคอลที่ทำงานในระดับบนหรือ Layer 3 ขึ้นไปเช่น IP, IPX หรือ AppleTalk เมื่อมีการส่งข้อมูลก็จะบรรจุข้อมูลนั้นเป็น packet ในรูปแบบของ Layer 2 คือ Data Link Layer เมื่อ Router ได้รับข้อมูลมาก็จะตรวจดูใน packet เพื่อจะทราบว่าใช้โปรโตคอลแบบใด จากนั้นก็จะตรวจดูเส้นทางส่งข้อมูลจากตาราง Routing Table ว่าจะต้องส่งข้อมูลนี้ไปยังเครือข่ายใดจึงจะต่อไปถึงปลายทางได้ แล้วจึงบรรจุข้อมูลลงเป็น Packet ของ Data Link Layer ที่ถูกต้องอีกครั้ง เพื่อส่งต่อไปยังเครือข่ายปลายทาง

คุณสมบัติของ Router

1.ทำหน้าที่คล้าย Swich ทำให้เชื่อมต่อได้หลายเครื่องพร้อมกัน

2.บางรุ่นรองรับการทำงาน Wire หรือ Wireless

3.เป็น ADSL Modem ในตัว (เฉพาะบางรุ่นเท่านั้น)

4.Firewall /IPsec VPN (รองรับการเชื่อมต่อทางไกลแบบมี security)

5.Antivirus (รุ่นใหม่ๆ ของ Router บางรุ่น จะมี antivirus program ฝังอยู่ด้วย)

รูปที่2.10 แสดงรูป Router [6]

2.5.3 Access Point

Access Point หมายถึง อุปกรณ์จุดเข้าใช้งานเครือข่ายไร้สาย ทำหน้าที่รองรับการเชื่อมโยงจากเครื่องลูกข่าย เป็นอุปกรณ์ที่ทำหน้าที่คล้ายคลึงกับ switching hub ของระบบเครือข่ายปกติครับ โดย accessPoint ทำหน้าที่รับส่งข้อมูลทางคลื่นความถี่กับ Wireless Card ซึ่งติดตั้งบนเครื่องของผู้ใช้แต่ละคน

Access Point หรือเรียกกันสั้นๆ ว่าAP (เอ-พี) ซึ่งจะทำหน้าที่เป็นจุดกระจายและเชื่อมต่อสัญญาณไร้สาย เพื่อเชื่อมต่ออุปกรณ์ไร้สายทุกชนิด (ที่ทำงานภายใต้มาตรฐานของ IEEE802.11) เข้าด้วยกัน นอกจากจะทำหน้าที่เป็น Access Point แล้ว AP ที่ดียังสามารถทำหน้าที่อื่นๆ เพื่อช่วยให้ระบบเครือข่ายไร้สายตอบสนองความต้องการของคุณได้อย่างถึงขีดสุด หน้าที่ต่างๆ ของ AP ที่ดี ที่จะช่วยสร้างระบบเครือข่ายไร้สายของคุณให้ทรงประสิทธิภาพสูงสุดอย่างแท้จริง ซึ่ง AP นั้นก็มีอยู่หลายโหมดด้วยกันคือ

2.5.3.1 Access Point Mode

ซึ่งเป็นหน้าที่หลักโดยกำเนิดของ AP ทุกตัวและเป็นที่มาของชื่อเรียกของเจ้าอุปกรณ์ตัวนี้ AP ที่ทำหน้าที่เป็น Access Point จะว่าไปแล้วก็เปรียบเสมือนสวิตซ์ในการสร้างระบบเครือข่ายผ่านสาย (ไม่ว่าจะเป็นสาย UTP หรือสาย Fiber Optic) โดย AP จะทำหน้าที่ เป็นจุดเชื่อมต่อระหว่างอุปกรณ์ต่างๆ ที่รองรับรับระบบเครือข่ายไร้สายเข้าด้วยกัน ไม่ว่าจะเป็น Computer, Print Server, Cameraหรือ อุปกรณ์พกพาต่างๆ (Smart Phone /PDA) เพื่อให้ใช้ทรัพยากรในวงแลนรวมกัน ทั้งซอฟท์แวร์ อาทิแชร์ไฟล์ แชร์โปรแกรม แชร์อินเตอร์เน็ต หรือ ฮาร์ดแวร์ อาทิ การแชร์ Printer เป็นต้น Access Point Modeนี้จึงเป็นหัวใจหลักของการสร้างระบบเครือข่ายไร้สาย ที่ต้องการจะเชื่อมต่ออุปกรณ์ไร้สายเข้าด้วยกัน และเป็นเพียงโหมดเดียวที่ให้เครื่องลูกข่าย เชื่อมโยงเข้ากับ Access Point ได้ นอกจากนั้นจะเป็นการเชื่อมกันระหว่าง Access Point ด้วยกันเอง

2.5.3.2 Client Mode (AP Station / AP Client)

ใน Mode นี้ AP จะทำหน้าที่ในลักษณะเดียวกันกับ WirelessCard (หรือ Wireless Adapter อื่นๆ) คือทำหน้าที่เป็นตัวลูกข่าย และเชื่อมต่อผ่านทางสัญญาณไร้สายกับAP เท่านั้น โดยจะไม่สามารถกระจายสัญญาณไร้สายไปยังอุปกรณ์ชิ้นอื่นๆ ได้อีก การใช้งานใน Mode นี้เหมาะสำหรับการอำนวยความสะดวกให้กับStation ที่ไม่พร้อมสำหรับการใช้งานไร้สาย แต่พร้อมสำหรับการเป็น

ส่วนหนึ่งในวง LAN เช่น เครื่องชั่งน้ำหนักและพิมพ์ Label ในศูนย์การค้า (โดยเฉพาะในแผนกผักผลไม้)ที่ใช้ดึงข้อมูลจาก Database แล้วคำนวณออกมาเป็นราคาสินค้า โดยไม่ต้องเชื่อมต่ออุปกรณ์ดังกล่าวกับระบบฐานข้อมูลด้วยสายซึ่งเกะกะ หรือตั้งอยู่ในจุดที่ไม่สะดวกในการติดตั้งสายหรือจะใช้ AP ใน Mode นี้สำหรับการเชื่อมต่อวงแลน 2 วงที่อยู่ห่างกัน เข้าด้วยกัน หรือจะใช้กับเครื่อง Macintosh ที่ไม่ต้องการซื้อWireless Card ซึ่งมีราคาค่อนข้างสูงมาใช้งาน โดยสามารถนำ AP มาใช้งานแทนได้

2.5.3.3. Repeater Mode

ทำหน้าที่เป็น Repeater ก็คือการทำหน้าที่ รับสัญญาณไร้สายมาเพื่อ กระจายต่อ โดยระบบเครือข่ายสำหรับการเชื่อมต่อในลักษณะนี้ต้องอยู่ในวงแลนเดียวกันเท่านั้นไม่ใช่การสร้างหรือการเพิ่มวงแลนแต่อย่างใด และ AP ที่ ทำหน้าที่เป็น Repeater จะต้องอยู่ในรัศมีของสัญญาณจาก Access Point การเชื่อมต่อใน Repeater Mode นั้น จะสามารถสร้าง Hop ได้ทั้งหมด 8 Hop(1 AP + 8 Repeater) โดยแต่ละ Hop ที่เกิดขึ้น จะทำให้สัญญาณเครือข่ายไร้สายช้าลงตามความหน่วงและตามสภาวะแวดล้อม และหากเกิด Network Down จำเป็นอย่างยิ่งที่จะต้องไล่ตรวจสอบ AP ทีละตัว ทำให้เกิดปัญหายุ่งยากในการดูแลระบบเครือข่ายไร้สาย นอกจากนี้ การใช้ Repeater จะทำให้การเชื่อมต่อกับเครื่องลูกข่ายทั้งหมดไปรวมอยู่ที่ Access Point ซึ่งต้อง Load งานหนักและอาจจะพาลแฮงค์ ไปทั้งระบบเลยก็ได้ การตั้งค่า AP ด้วย Repeater Mode จึงเหมาะสำหรับการแก้ไขระบบเครือข่ายที่ได้รับการออกแบบมาผิด เนื่องจากไม่ถูกสำรวจความต้องการใช้งานให้ดีก่อนที่จะสร้างระบบเครือข่าย หรือใช้สำหรับกระจายสัญญาณไปยังจุดอับสัญญาณจริงๆ เนื่องด้วยรูปแบบของสถานที่ใช้งาน เช่น ตามซอกลึบของอาคาร การใช้งาน Repeater Mode นั้นค่อนข้างจะเป็นการฝืนธรรมชาติของการใช้งานระบบเครือข่ายไร้สายแบบปกติชาวโลกใช้กัน จึงควรใช้สำหรับการแก้ปัญหาเฉพาะหน้ามากกว่าจะใช้เป็นทางเลือกหลักในการวางระบบเครือข่ายไร้สายที่ต้องการประสิทธิภาพสูงสุด

2.5.3.4. Bridge Mode (WDA: Wireless Distribution Architecture / WDS: Wireless Distribution System)

สำหรับใน Mode นี้ AP จะทำหน้าที่เหมือนเป็นสะพาน เชื่อมระหว่างวงแลนเข้าหากัน จะเรียกง่ายๆ ก็คือBridge Mode ทำให้วงแลน 2 วง ที่ต่างคนต่างทำงานกันเป็นปกติอยู่แล้ว สามารถ

เชื่อมต่อเข้าหากันได้และต่างก็สามารถเข้าถึงอุปกรณ์ของอีกวง แลนหนึ่งได้ (แตกต่างจาก Client Mode ตรงนี้ Client Modeจะไม่สามารถเชื่อมต่อไปยัง อุปกรณ์ไร้สายเครื่องอื่นๆ ได้ แต่ใน Bridge Mode นี้ทำได้) การเชื่อมต่อในลักษณะ Bridge Mode ทำได้ ทั้งแบบ Point to Point (PtP) คือเชื่อมระหว่างวงแลน 2 วงเข้าด้วยกันและการเชื่อมต่อแบบ Point to Multi-Point (PtMP) นั่นก็คือสามารถเชื่อมต่อวงแลนมากกว่า 2วงแต่สูงสุดไม่ควรจะเกิน 7 Bridge เนื่องจาก จะทำให้การเชื่อมต่อช้าลง เนื่องจากความหน่วง (เช่นเดียวกันกับRepeater Mode) ไม่ใช่ AP ทุกตัวที่จะสามารถทำงานได้ครบทั้ง 4 Mode ดังนั้นก่อนจะตัดสินใจเลือกAP ตัวใด ควรสอบถามจากเจ้าของผลิตภัณฑ์ให้ แน่ใจก่อนว่า AP ที่คุณซื้อนั้นสามารถใช้งานใน Modeที่คุณต้องการได้ เพื่อให้การจ่ายเงิน ของคุณเกิดประโยชน0สูงสุด ที่สำคัญ AP ในแต่ละ Mode ล้วนแล้วแต่มีวัตถุประสงค์ในการ ใช้งานที่แตกต่างกันไป จะเลือกใช้ Mode ใด ก็ขึ้นอยู่กับความต้องการ และลักษณะของระบบเครือข่ายที่คุณต้องการ ดังนั้นก่อนที่จะสร้างระบบเครือข่าย สิ่งสำคัญคือต้องศึกษาความต้องการให้ดี เพื่อที่จะได้ออกแบบระบบเครือข่ายที่รองรับการทำงานของคุณอย่างแท้จริง

2.5.3.5 Repeater Mode

โหมดนี้เป็นเหมือนการขยายระยะส่งของระบบ Wireless LAN ครับ โดยติดตั้งAccess Point เพิ่มขึ้น บริเวณที่สัญญาณของ Access Point ตัวหลักเริ่มจาง ทำให้สามารถเพิ่มระยะส่งของทั้งระบบออกไปอีก

มหาวิทยาลัยเทคโนโลยีสุรนารี

2.6 การใช้งานของไมโครคอนโทรลเลอร์ MCS-51

ไมโครคอนโทรลเลอร์ เป็นไมโครโปรเซสเซอร์ประเภทหนึ่งที่ได้รับการออกแบบมาเพื่อใช้งานกับระบบควบคุมที่มีขนาดเล็ก โดยภายในไอซีคอนโทรลเลอร์หนึ่งตัวจะประกอบด้วยการทำงานหลักของคอมพิวเตอร์อย่างครบถ้วนเช่น หน่วยประเมินผลกลาง หน่วยความจำพอร์ตในการติดต่อหรือควบคุมอุปกรณ์ต่างๆเป็นต้น หากว่าเป็นการใช้งานไมโครโปรเซสเซอร์ทั่วไปก็จะต้องใช้ไอซีภายนอกมาประกอบเพื่อที่จะมาใช้ทำหน้าที่เหล่านี้ ดังนั้นจึงอาจกล่าวได้ว่าไมโครคอนโทรลเลอร์เป็นระบบคอมพิวเตอร์เพื่อควบคุมที่สมบูรณ์ โดยบรรจุอยู่ในตัวไอซีเพียงหนึ่งตัวเท่านั้น ในบางครั้งจึงอาจจะพบได้ว่ามีการเรียกชื่อของไมโครคอนโทรลเลอร์ว่าเป็นระบบไมโครคอมพิวเตอร์ชิพเดี่ยว ในไมโครคอนโทรลเลอร์ที่นิยมใช้กันอย่างแพร่หลายคือตระกูล 8051

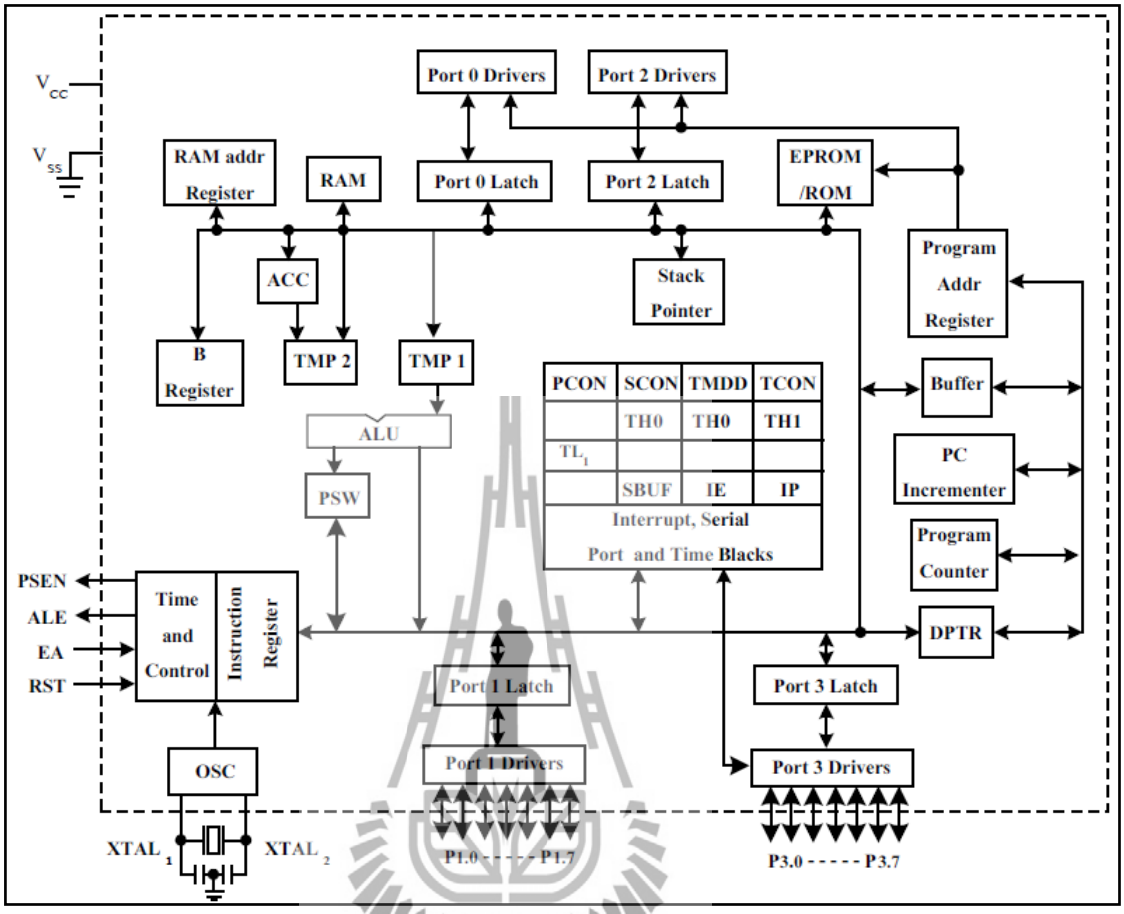
จากรูปที่ 2.11 แสดงให้เห็นถึงโครงสร้างและหน่วยการทำงานที่เป็นระบบพื้นฐานของไมโครคอนโทรลเลอร์ต่าง ๆ ที่จัดอยู่ภายในตระกูล MCS-51 นี้ประกอบด้วย

- หน่วยประมวลผลกลางขนาด 8 บิต
- หน่วยประมวลผลสำหรับข้อมูลแบบบิต
- ความสามารถในการอ้างตำแหน่งของหน่วยความจำโปรแกรม 64 กิโลไบต์
- ความสามารถในการอ้างตำแหน่งของหน่วยความจำข้อมูล 64 กิโลไบต์
- หน่วยความจำโปรแกรมภายในขนาด 4 กิโลไบต์ แบบ อีพรอม (เบอร์ 8751)
- หน่วยความจำแบบ แรม ภายในจำนวน 1 กิโลไบต์
- พอร์ตอินพุต/เอาต์พุตแบบขนานจำนวน 32 เส้น ซึ่งแยกทำงานได้อย่างอิสระ
- วงจรนับ/จับเวลาขนาด 16 บิต จำนวนสองวงจร
- วงจรสื่อสารแบบอนุกรมแบบดูเพลกซ์เต็ม
- วงจรควบคุมการขัดจังหวะจากแหล่งกำเนิดสัญญาณ 6 ประเภท พร้อมกำหนดลำดับ
- วงจรออสซิลเลเตอร์ภายใน

ซึ่งโครงสร้างการทำงานทั้งหมดของไมโครคอนโทรลเลอร์จะอาศัยหลักการทำงานที่เกี่ยวข้องกันโดยอาศัยหลักการทำงานที่เป็นตามโครงสร้างเสมอ ไมโครคอนโทรลเลอร์ตระกูลนี้

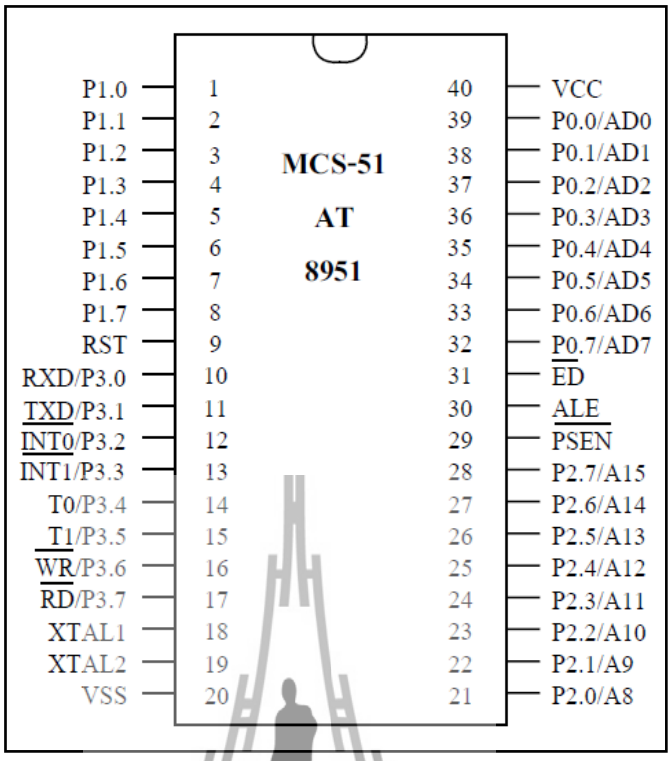
มหาวิทยาลัยเทคโนโลยีสุรนารี

มักจะมีรูปร่างของไอซีเป็นแบบแบนขนาด 40 ขา ดังแสดงในรูปที่ 2.15



รูปที่ 2.11 โครงสร้างภายในของไมโครคอนโทรลเลอร์ MCS-51

มหาวิทยาลัยเทคโนโลยีสุรนารี



รูปที่ 2.12 รูปร่างและการจัดวางขาต่าง ๆ ของไมโครคอนโทรลเลอร์ MCS-51

แต่ละขาสัญญาณจะมีหน้าที่ที่ระบุชัดเจนตามสัญลักษณ์ชื่อย่อที่กำกับไว้ในแต่ละขา อย่างไรก็ตามจะมีบางขาสัญญาณที่อาจจะมีหน้าที่ได้มากกว่าหนึ่งอย่าง (ซึ่งเขียนกำกับไว้ว่า Alternate Functions ในรูปที่ 2.2) ซึ่งจะไม่สามารถใช้งานในเวลาเดียวกันได้ ดังตัวอย่างเช่น ขาสัญญาณบิต 0 ของพอร์ต 3 (ใช้ตัวย่อเป็น P3.0) อาจจะใช้เป็นขาสัญญาณทางเอาต์พุต หรืออินพุต ตามปกติ

2.6.1ทฤษฎีพื้นฐานเกี่ยวกับMicrocontroller

Microcontroller คือ อุปกรณ์ควบคุมขนาดเล็ก ซึ่งบรรจุความสามารถที่คล้ายคลึงกับระบบคอมพิวเตอร์ โดยในMicrocontrollerได้รวมเอาซีพียู,หน่วยความจำ และพอร์ต ซึ่งเป็นส่วนประกอบหลักสำคัญของระบบคอมพิวเตอร์เข้าไว้ด้วยกัน โดยทำการบรรจุเข้าไว้ในตัวถังเดียวกัน

Microcontroller มาจากMicro + controller ซึ่งจะหมายถึง การควบคุมขนาดเล็ก หรืออาจจะกล่าวได้ว่า Microcontroller ทำหน้าที่เสมือนคอมพิวเตอร์ขนาดเล็กที่ในควบคุมการทำงานของอุปกรณ์ไฟฟ้า หรือ ระบบควบคุมอิเล็กทรอนิกส์ โดยในการควบคุมนั้นจะควบคุมตามคำสั่งที่ได้ทำ

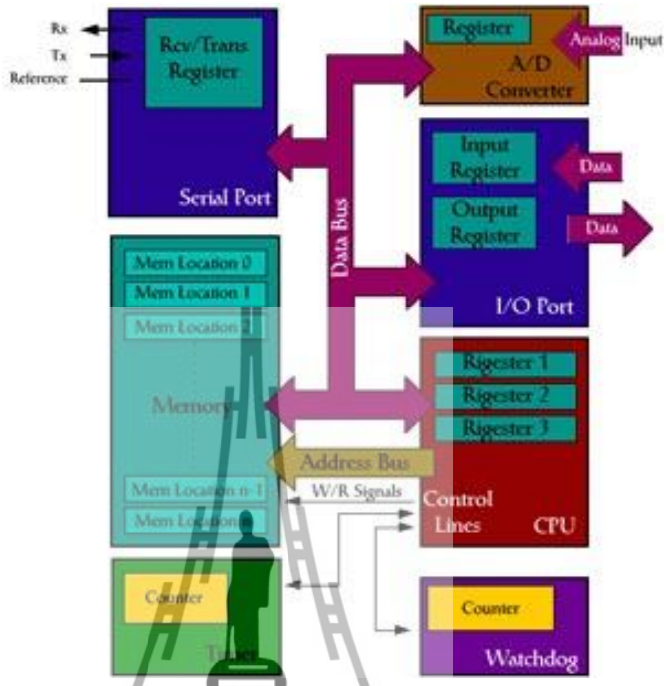
การเขียนให้ Microcontroller ตัวอย่างอุปกรณ์ที่ควบคุมด้วย Microcontroller เช่น เครื่องซักผ้าแบบอัตโนมัติ(ตั้งโปรแกรมได้) หรือ ระบบสมองกลของรถยนต์ รวมถึงเครื่องใช้ไฟฟ้าอิเล็กทรอนิกส์อีกมากมาย ซึ่งบางรุ่น ก็จะมีตัว Microcontrollerซ่อนอยู่ภายใน โดยที่ภายในของ Microcontrollerจะมีโครงสร้างหลักอยู่ดังนี้ ส่วนประมวลผล หรือ Processing Unit คือส่วนที่ทำหน้าที่คำนวณทางคณิตศาสตร์ หรือ การตัดสินใจแบบมีเงื่อนไข

2.6.2 โครงสร้างโดยทั่วไป ของMicrocontroller

สามารถแบ่งออกมาได้เป็น 5 ส่วนใหญ่ๆ ดังต่อไปนี้

1. หน่วยประมวลผลกลางหรือซีพียู (CPU : Central Processing Unit)
2. หน่วยความจำ (Memory) สามารถแบ่งออกเป็น 2 ส่วน คือ หน่วยความจำที่มีไว้สำหรับเก็บโปรแกรมหลัก (Program Memory) เปรียบเสมือนฮาร์ดดิสก์ของเครื่องคอมพิวเตอร์ตั้งโต๊ะ คือข้อมูลใดๆ ที่ถูกเก็บไว้ในนี้จะไม่สูญหายไปแม้ไม่มีไฟเลี้ยง อีกส่วนหนึ่งคือหน่วยความจำข้อมูล (Data Memory) ใช้เป็นเหมือนกกระดาษทดในการคำนวณของซีพียู และเป็นที่พักข้อมูลชั่วคราวขณะทำงาน แต่หากไม่มีไฟเลี้ยง ข้อมูลก็จะหายไปคล้ายกับหน่วยความแรม (RAM) ในเครื่องคอมพิวเตอร์ทั่วๆ ไป แต่สำหรับMicrocontrollerสมัยใหม่ หน่วยความจำข้อมูลจะมีทั้งที่เป็นหน่วยความจำแรม ซึ่งข้อมูลจะหายไปเมื่อไม่มีไฟเลี้ยง และเป็นอีอีพรอม (EEPROM : Erasable Electrically Read-Only Memory) ซึ่งสามารถเก็บข้อมูลได้แม้ไม่มีไฟเลี้ยง
3. ส่วนติดต่อกับอุปกรณ์ภายนอก หรือพอร์ต (Port) มี 2 ลักษณะคือ พอร์ตอินพุต (Input Port) และพอร์ตส่งสัญญาณหรือพอร์ตเอาต์พุต (Output Port) ส่วนนี้จะใช้ในการเชื่อมต่อกับอุปกรณ์ภายนอก ถือว่าเป็นส่วนที่สำคัญมาก ใช้ร่วมกันระหว่างพอร์ตอินพุต เพื่อรับสัญญาณ อาจจะด้วยการกดสวิตช์ เพื่อนำไปประมวลผลและส่งไปพอร์ตเอาต์พุต เพื่อแสดงผลเช่น การติดสว่างของหลอดไฟ เป็นต้น
4. ช่องทางเดินของสัญญาณ หรือบัส (BUS) คือเส้นทางการแลกเปลี่ยนสัญญาณข้อมูลระหว่าง ซีพียู หน่วยความจำและพอร์ต เป็นลักษณะของสายสัญญาณ จำนวนมากอยู่ภายในตัวMicrocontroller โดยแบ่งเป็นบัสข้อมูล (Data Bus) , บัสแอดเดรส (Address Bus) และบัสควบคุม (Control Bus)
5. วงจรกำเนิดสัญญาณนาฬิกา นับเป็นส่วนประกอบที่สำคัญมากอีกส่วนหนึ่ง เนื่องจากการทำงานที่เกิดขึ้นในตัวMicrocontroller จะขึ้นอยู่กับการกำหนดจังหวะ หาก

สัญญาณนาฬิกามีความถี่สูง จังหวะการทำงานก็จะสามารถทำได้ถี่ขึ้นส่งผลให้ Microcontrollerตัวนั้น มีความเร็วในการประมวลผลสูงตามไปด้วย



รูปที่ 2.13 ส่วนประกอบของMicrocontroller [3]

ในปัจจุบัน Microcontroller ได้ถูกนำมาใช้ในงานออกแบบวงจรอิเล็กทรอนิกส์มากมาย ตั้งแต่วงจรที่มีขนาดเล็กไปจนถึงวงจรที่มีความซับซ้อนมากขึ้น และ Microcontrollerในปัจจุบันก็มีประสิทธ์ภาพสูงมากขึ้น ทำงานเร็วมากขึ้นและมีเครื่องมือในการพัฒนามากขึ้น ดังนั้นก่อนที่จะใช้งานMicrocontrollerในการออกแบบวงจรอิเล็กทรอนิกส์

2.6.3 กลุ่มของMicrocontroller

Microcontrollerมีด้วยกันหลายประเภทแบ่งตามสถาปัตยกรรม (การผลิตและกระบวนการทำงานระบบการประมวลผล) ที่มีใช้ในปัจจุบันยกตัวอย่างดังนี้

1. ไมโครคอนโทรลเลอร์ตระกูล PIC (บริษัทผู้ผลิต Microchip ไมโครชิป)
2. ไมโครคอนโทรลเลอร์ตระกูล MCS51 (บริษัทผู้ผลิต Atmel,Phillips)
3. ไมโครคอนโทรลเลอร์ตระกูล AVR (บริษัทผู้ผลิต Atmel)

4. ไมโครคอนโทรลเลอร์ตระกูล ARM7,ARM9 (บริษัทผู้ผลิต Atmel,Phillips,AnalogDevice,Sumsung,STMicroelectronics)
5. ไมโครคอนโทรลเลอร์ตระกูล Basic Stamp (บริษัทผู้ผลิต Parallax)
6. ไมโครคอนโทรลเลอร์ตระกูล PSOC (บริษัทผู้ผลิต CYPRESS)
7. ไมโครคอนโทรลเลอร์ตระกูล MSP (บริษัทผู้ผลิต Texas Intruments)
8. ไมโครคอนโทรลเลอร์ตระกูล 68HC (บริษัทผู้ผลิต MOTOROLA)
9. ไมโครคอนโทรลเลอร์ตระกูล H8 (บริษัทผู้ผลิต Renesas)
10. ไมโครคอนโทรลเลอร์ตระกูล RABBIT (บริษัทผู้ผลิต RABBIT SEMICONDUCTOR)
11. ไมโครคอนโทรลเลอร์ตระกูล Z80 (บริษัทผู้ผลิต Zilog)

กลุ่ม PIC เป็นMicrocontrollerที่ออกแบบจาก Microchip และเป็นMicrocontrollerที่นิยมนำมาใช้งาน อย่างกว้างขวาง กลุ่ม dsPIC เป็นMicrocontrollerที่ออกแบบจาก Microchip และเป็น MicrocontrollerdsPICMicrocontroller 16 บิต

สัญลักษณ์ Microchip

สัญลักษณ์ Atmel

กลุ่ม AVR Microcontroller ตระกูลนี้ออกแบบและผลิตโดย Atmel และจะมีสถาปัตยกรรมแบบ RISC มีเบอร์ให้เลือกใช้หลายเบอร์ ราคาไม่สูงมากนัก ตระกูลในกลุ่ม AVR ATmega AVR Device เช่น เช่น ATmega8 ATmega32 กลุ่ม ARM Microcontroller ชนิดนี้ จะเป็นMicrocontrollerที่มีความเร็ว และการประมวลแบบ 16 บิตขึ้นไป เหมาะสำหรับงานที่ต้องการการประมวลผลที่เร็ว เช่น ARM ตระกูล 68HC11 เป็นMicrocontrollerที่ผลิตโดย Motorola

สัญลักษณ์ motorola

BASIC Stamp เป็นMicrocontrollerที่ออกแบบโดย Parallax ซึ่งมีความสามาในการเขียนโปรแกรมด้วยภาษาเบสิก

สัญลักษณ์ Parallax

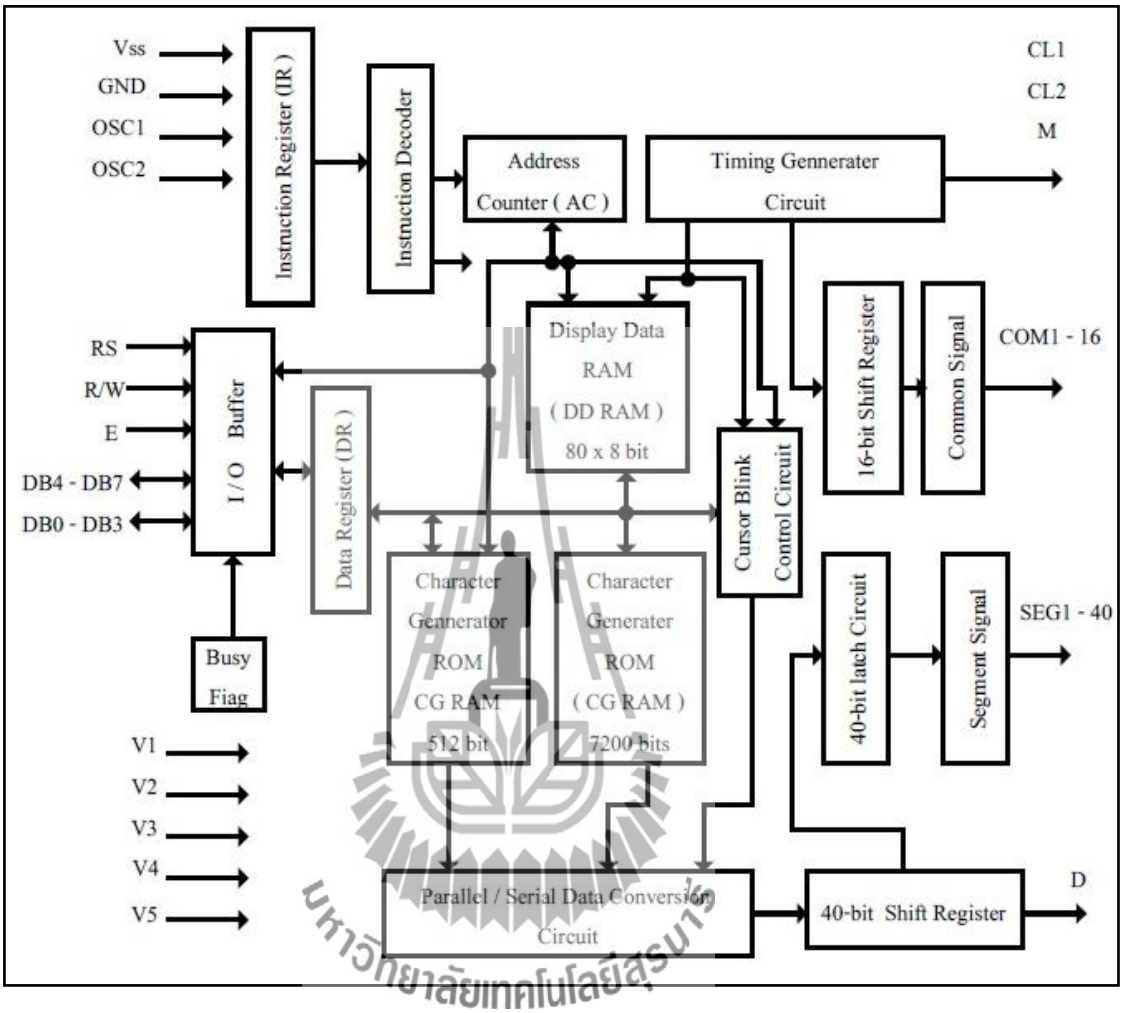
2.6.4 ภาษาที่ใช้เขียน โปรแกรมควบคุมMicrocontroller

1. ภาษา Assembly
2. ภาษา Basic
3. ภาษา C
4. ภาษา Pascal

ภาษาดัง กล่าวที่กล่าวในเบื้องต้น Microcontrollerบางตระกูล จะใช้ได้ครบทุกภาษา แต่บางตระกูลจะใช้ได้บางภาษา ขึ้นอยู่กับบริษัทผู้ผลิต Software (โดยทั่วไปจะเรียกว่า Editor And Complier) ที่ใช้เขียนภาษาMicrocontrollerนั้นจะผลิตออกมาให้ Support หรือไม่ Webmaster Talk: ผมขอสรุปในเบื้องต้นแบบง่ายๆ Microcontroller เปรียบเทียบรถยนต์ทั่วไปรถยนต์ มีหลายบริษัทผู้ผลิต,ในแต่ละบริษัทก็มีอยู่หลายรุ่นหลายแบบ รถยนต์มีระบบทุกอย่างพร้อมขึ้นอยู่กับเราจะขับหรือควบคุมการใช้งานนั่นเอง

Microcontrollerก็เช่นกัน มีหลายบริษัทผู้ผลิตในแต่ละบริษัทผู้ผลิต ก็จะมีหลายเบอร์ให้เลือกใช้งาน ,Microcontrollerก็มีระบบต่างๆอยู่เพียบ พร้อม ส่วนการใช้งานขึ้นอยู่กับเราจะเขียนโปรแกรมควบคุมให้มันทำงานตามที่เราต้อง การเท่านั้นเอง

2.7 การใช้งานจอแสดงผลผลึกเหลว (Liquid Crystal Display : LCD)



รูปที่ 2.14 แผนภาพบล็อกของไอซีภายในจอแสดงผลผลึกเหลว

อุปกรณ์ในปัจจุบันนี้ที่ใช้ในการแสดงผลการทำงานของระบบต่าง ๆ นั้น โดยทั่วไปจะใช้จอแสดงผลผลึกเหลวเสียเป็นส่วนใหญ่ ไม่ว่าจะเป็นเครื่องเล่นวิดีโอเทป เครื่องมือวัดคุมเครื่องคอมพิวเตอร์ รวมไปถึงการใช้งานทางด้าน งานอิเล็กทรอนิกส์ หรือแม้แต่เพจเจอร์ที่ใช้จอแสดงผลผลึกเหลว ที่สามารถที่จะแสดงผลออกเป็นข้อความที่เป็นตัวเลข ตัวอักษร และลักษณะอื่น ๆ ซึ่งสามารถพอที่จะแบ่งคุณลักษณะของจอแสดงผลผลึกเหลวออกเป็นพวก ๆ ดังนี้

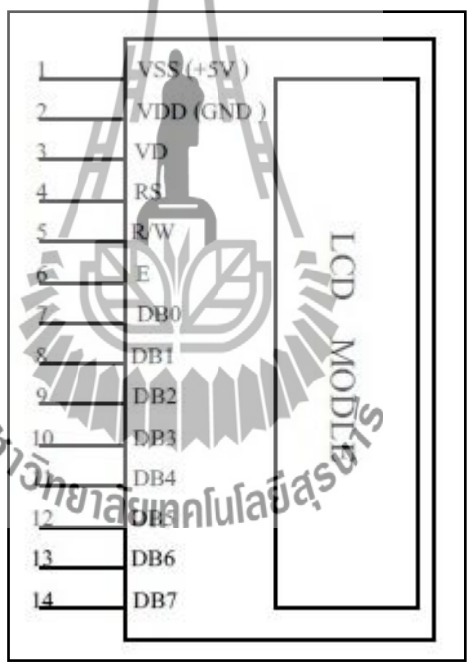
- คุณลักษณะทั่ว ๆ ไปของจอแสดงผลผลึกเหลว
- กราฟฟิกของจอแสดงผลผลึกเหลว
- จอแสดงผลผลึกเหลว ชนิด ไดโอดเปล่งแสง 7 หลัก

โดยในแต่ละแบบนี้จะมีส่วนประกอบใหญ่ ๆ แบ่งได้เป็น

1. จอแสดงผลผลึกเหลวเป็นตัวแสดงผลให้มองเห็นในลักษณะการเปิดและปิดกับแสงก็คือส่วนที่เป็นกระจกบรรจุผลึก

2. ตัวขับ (Driver) เป็นตัวรับสัญญาณจากตัวควบคุมมาขับผลักจอแสดงผล โดยมีหมายเลขที่นิยมใช้ในคือ หมายเลข HD44100H และหมายเลข MSM5259

3. ตัวควบคุม (Controller) เป็นตัวรับข้อมูลจากอุปกรณ์ภายนอกเข้ามา และจะทำการจัดการควบคุมจอแสดงผลผลึกเหลวให้ทำงานแสดงผลต่าง ๆ เช่นการลบจอภาพการเกิดตัวอักษร เป็นต้น โดยมีเบอร์ไอซีที่นิยมใช้กัน คือ เบอร์ HD44780 ซึ่งจะใช้ในแบบคุณลักษณะ ทั่ว ๆ ไป และเบอร์ของไอซีเบอร์ HD61830 จะใช้ในแบบ กราฟฟิก



รูปที่ 2.15 ลักษณะและหน้าที่ของขาต่าง ๆ ของจอแสดงผลผลึกเหลว

หน้าที่การทำงานของตำแหน่งขาต่าง ๆ ของจอแสดงผลผลึกเหลว

ขาที่ 1 : VSS ทำหน้าที่ เป็นกราวด์

ขาที่ 2 : VDD ทำหน้าที่ เป็น รับแรงดันไฟฟ้า +5 โวลต์

ขาที่ 3 : Vo ทำหน้าที่ ปรับความสว่างของจอด้วยแรงดันไฟฟ้า +5 โวลต์

ขาที่ 4 : RS ทำหน้าที่ เลือกรีจิสเตอร์ ให้แสดงสภาวะการทำงาน

ขาที่ 5 : R/W ทำหน้าที่ เลือกการอ่านหรือเขียน (0 = เขียน ; 1 = อ่าน)

ขาที่ 6 : E ทำหน้าที่ อีนาเบิลการอ่านหรือเขียนข้อมูลในจอแสดงผลผลึกเหลว

ขาที่ 7 : D0 ทำหน้าที่ รับ - ส่ง ข้อมูลอินพุต / เอาต์พุตบิตต่ำสุด

ขาที่ 8 : D1 ทำหน้าที่ รับ - ส่ง ข้อมูลอินพุต / เอาต์พุตบิตที่ 2

ขาที่ 9 : D2 ทำหน้าที่ รับ - ส่ง ข้อมูลอินพุต / เอาต์พุตบิตที่ 3

ขาที่ 10 : D3 ทำหน้าที่ รับ - ส่ง ข้อมูลอินพุต / เอาต์พุตบิตที่ 4

ขาที่ 11 : D4 ทำหน้าที่ รับ - ส่ง ข้อมูลอินพุต / เอาต์พุตบิตที่ 5

ขาที่ 12 : D5 ทำหน้าที่ รับ - ส่ง ข้อมูลอินพุต / เอาต์พุตบิตที่ 6

ขาที่ 13 : D6 ทำหน้าที่ รับ - ส่ง ข้อมูลอินพุต / เอาต์พุตบิตที่ 7

ขาที่ 14 : D7 ทำหน้าที่ รับ - ส่ง ข้อมูลอินพุต / เอาต์พุตบิตสูงสุด

การทำงานของแต่ละขาของจอแสดงผลผลึกเหลวนี้ จะถูกกำหนดโดยโปรแกรมที่ผู้เขียนเป็นผู้กำหนดการทำงานจอแสดงผลผลึกเหลวนั้น ๆ ว่าจะให้ขาใดทำหน้าที่ใด

- รายละเอียดของคำสั่งไอซีเบอร์ HD44780

1. การเคลียร์ จอแสดงผล

RS	R/W	DB7							DB0
0	0	0	0	0	0	0	0	0	1

มหาวิทยาลัยเทคโนโลยีสุรนารี

คำสั่งจะเป็นการเขียนข้อมูลย่อยหรือ SPACE (ASCII 20H) เข้าไปในหน่วยความจำแบบ ดีดี แรม ทั้งหมดและทำการกำหนดค่าตำแหน่งของหน่วยความจำแบบ ดีดี แรม เป็นศูนย์ตัว เคอร์เซอร์ จะกลับไปอยู่ตำแหน่งบนสุดซ้ายมือของจอภาพ SET1/D = 1,S ไม่มีการเปลี่ยน

2. การกลับไปจุดเริ่มต้น

RS	R/W	DB7							.DB0
0	0	0	0	0	0	0	0	0	.

คำสั่งนี้จะทำการกำหนดตำแหน่งหน่วยความจำแบบ ดีดี แรม เป็นศูนย์ตัว เคอร์เซอร์ จะกลับไปอยู่ตำแหน่งบนสุดซ้ายมือของจอภาพข้อมูลในจอภาพไม่เปลี่ยน

3. การเข้าสู่โหมดการกำหนด (SET)

RS	R/W	DB7					DB0
0	0	0	0	0	0	0	0	I/D	S

บิต I/D : โดยจะเป็นตัวกำหนดให้ว่าเมื่อมีการเขียนหรืออ่านข้อมูลแล้วจะเป็นผลทำให้ตำแหน่งของหน่วยความจำแบบ ดีดี แรม เพิ่มขึ้นหนึ่งหรือลดลงหนึ่งโดย

1 = เพิ่ม

0 = ลดลงหนึ่ง

บิต S : โดยจะเป็นตัวกำหนดแสดงผลโดยถ้า S = 1 จะเป็นการใส่ข้อมูลเข้าไปแล้วตัวเคอร์เซอร์ อยู่กับที่ข้อมูลจะถูกดันไปทางซ้าย ถ้า S = 0 ข้อมูลจะอยู่กับที่ตัวเคอร์เซอร์จะถูกดันไปทางขวามือ

4. เคอร์เซอร์ หรือ การเลื่อนข้อมูลการแสดงผล

RS	R/W	DB7					DB0
0	0	0	0	0	1	S/C	R/C	.	.

เป็นคำสั่งกำหนดให้ตำแหน่งเคอร์เซอร์หรือข้อมูลไปทิศทางซ้าย หรือขวาโดยไม่ต้องใช้คำสั่งเขียนหรืออ่าน

มหาวิทยาลัยเทคโนโลยีสุรนารี

5. การกำหนดตำแหน่งของหน่วยความจำแบบ ซีจี แรม

RS	R/W	DB7							DB0
0	0	0	1	A	A	A	A	A	A

ในไอซี HD44780 มีหน่วยความจำอยู่ 2 ชุด คือ ชุดแสดงผลข้อมูลของหน่วยความจำแบบแรม (ดีดี แรม) จำนวน 80x8 บิต และ ชุดแสดงลักษณะทั่วไปของหน่วยความจำแบบรอม หน่วยความจำแบบ ซีจี แรม จำนวน 512 บิต และ 7200 บิต คำสั่งนี้จะเป็นการกำหนดตำแหน่งในหน่วยความจำแบบ ซีจี แรม โดยต้องทำการกำหนดตำแหน่งก่อนเขียนหรืออ่านข้อมูลจากหน่วยความจำแบบ ซีจี แรม ด้วย

6. การกำหนดตำแหน่งหน่วยความจำแบบ ดีดี แรม

RS	R/W	DB7							DB0
0	0	1	A	A	A	A	A	A	A

เป็นคำสั่งเพื่อกำหนดตำแหน่งในหน่วยความจำแบบ ดีดี แรม ในการเขียนหรืออ่านค่าจากหน่วยความจำแบบ ดีดี แรม (DD RAM) คือส่วนที่จะแสดงผลหน้าจอแสดงผลผลึกเหลวโดยจำนวนตำแหน่งที่จะเกิดขึ้นบนจอจะอยู่กับ เซตค่า N ด้วย

N = 0 (1 บรรทัด) ตำแหน่ง จะอยู่ที่ตำแหน่งที่ OOH-4FH

N = 1 (2 บรรทัด) ตำแหน่ง จะอยู่ที่ตำแหน่งที่ OOH-27H สำหรับบรรทัดที่ 1 และตำแหน่งที่ 40H-67H สำหรับบรรทัดที่ 2 แบบการจัดตำแหน่งของหน่วยความจำแบบ ดีดี แรมหน้าจอแสดงผลผลึกเหลว แบบ 16 ตัวอักษร 1 บรรทัด 16 ตัวอักษร 2 บรรทัด 16 ตัวอักษร4 บรรทัด 20 ตัวอักษร 1 บรรทัด 20 ตัวอักษร 2 บรรทัด และ 40 ตัวอักษร 2 บรรทัด

7. การเขียนข้อมูลที่หน่วยความจำแบบ ซีจี หรือแบบ ดีดี แรม

RS	R/W	DB7							DB0
1	0	D	D	D	D	D	D	D	D

เป็นคำสั่งเขียนข้อมูลเข้าไปในหน่วยความจำแบบ ซีจี หรือแบบ ดีดี แรม โดยเมื่อเขียนข้อมูลและตำแหน่ง จะเพิ่มหรือลดโดยอัตโนมัติตามคำสั่งที่กำหนด ข้อกำหนดที่จะรู้ว่าเป็นการเขียนข้อมูลของหน่วยความจำแบบ ซีจี แรม หรือแบบ ดีดี แรม ทำได้โดยการกำหนดตำแหน่งของหน่วยความจำแบบ ซีจี แรม หรือแบบ ดีดี แรม ขึ้นมาก่อนจะเขียนข้อมูล

8. การอ่านข้อมูลจากหน่วยความจำแบบ ซีจี และแบบ ดีดี แรม

RS	R/W	DB7							DB0
1	1	D	D	D	D	D	D	D	D

เป็นคำสั่งการอ่านค่าข้อมูลจากหน่วยความจำแบบ ซีจี แรม หรือแบบ ดีดี แรม และหน่วยความจำแบบ ซีจี แรม นี้ควรจะใช้คำสั่งกำหนดตำแหน่งก่อนเพื่อให้รู้ว่าข้อมูลที่อ่านได้เป็น

มหาวิทยาลัยเทคโนโลยีสุรนารี

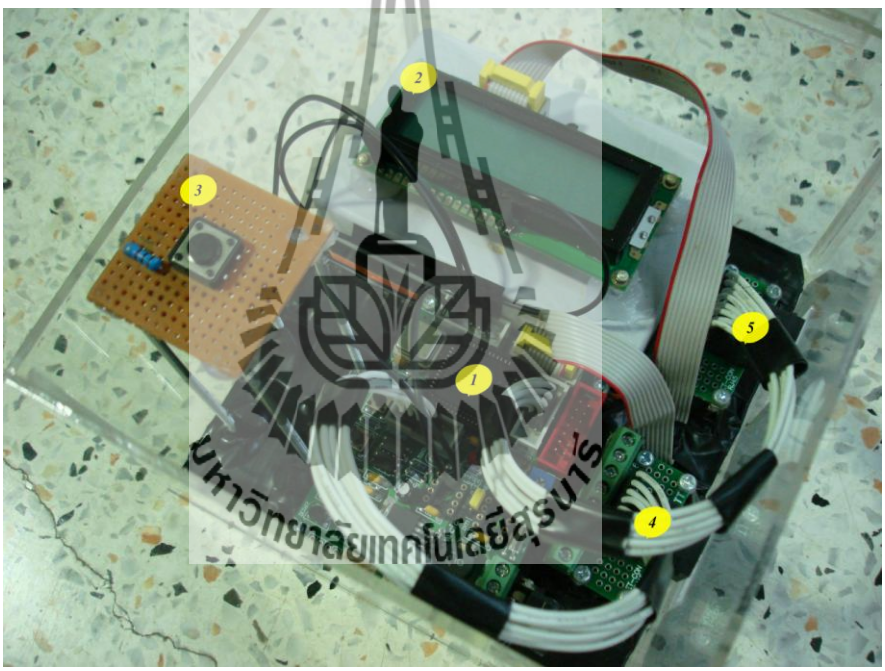
หน่วยความจำแบบ ดีดี หรือแบบ ซีจี แรม จากตารางการทำงาน จะเห็นได้ว่าการใช้งานของจอแสดงผลผลึกเหลวนั้นง่ายเพียงแต่ส่งคำสั่งเริ่มแรกและกำหนดความต้องการในขนาดตัวอักษร เคอร์เซอร์ หลังจากนั้นก็สามารถเขียนตัวอักษรเข้าไปในหน่วยความจำแบบ ดีดี แรม ตามตารางที่ให้มานั้นก็จะเกิดอักษรในจอแสดงผลผลึกเหลว

มหาวิทยาลัยเทคโนโลยีสุรนารี

บทที่ 3
โครงสร้างของระบบ

3.1 บทนำ

เนื้อหาในบทนี้จะกล่าวถึงโครงสร้างของเครื่องตรวจสอบสาย เคเบิล (UTP) ตลอดจนกระบวนการทำงานที่มีองค์ประกอบหลักคือ มีทั้งในส่วนของ Hardware และ Software ในส่วนของ Hardware ก็จะเกี่ยวกับ ข้อมูล คุณสมบัติ การทำงานของ Microcontroller และจอแสดงผลผลึกเหลว(LCD)ส่วน Software นั้นจะเกี่ยวกับ โปรแกรมที่ควบคุมให้ Microcontroller ทำการตรวจสอบสายเคเบิล (UTP) แล้วแสดงผลออกทางจอแสดงผลผลึกเหลว(LCD)



รูปที่ 3.1 เครื่องตรวจสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์ท้องถิ่น

หมายเลข 1. คือ บอร์ดMicrocontroller MCS -51 เบอร์ AT89C51ED2

หมายเลข 2. คือ จอแสดงผล LCD

หมายเลข 3. คือ สวิตซ์ เพื่อให้เครื่องทำการตรวจสอบสาย

หมายเลข 4. คือ Output

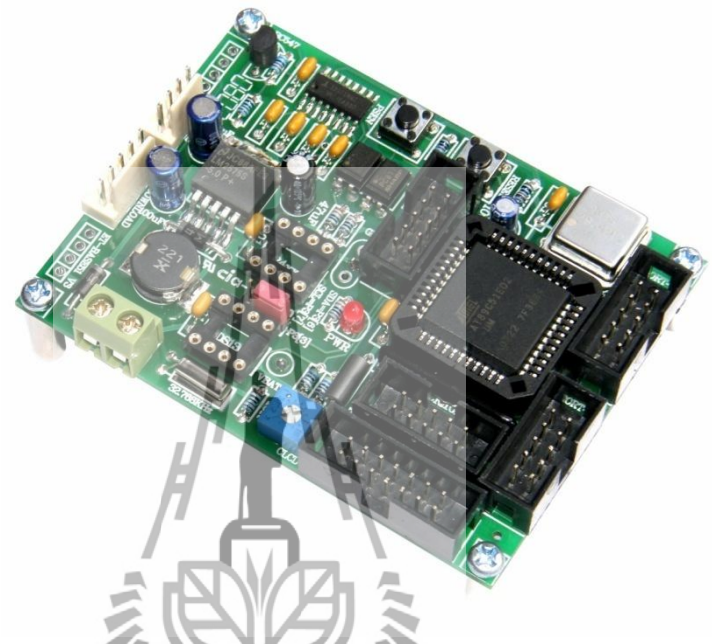
หมายเลข 5. คือ Input

มหาวิทยาลัยเทคโนโลยีสุรนารี

3.2 ส่วน Hard ware

ในส่วนนี้จะอธิบายเกี่ยว ข้อมูล คุณสมบัติ ของอุปกรณ์ต่าง ๆ ที่เกี่ยวข้องกับ Microcontroller และจอแสดงผลผลึกเหลว(LCD)

3.2.1 Microcontroller MCS -51



รูปที่ 3.2 บอร์ด Microcontroller MCS -51 เบอร์ AT89C51ED2

เป็นบอร์ดไมโครคอนโทรเลอร์ในตระกูล MCS51 ขนาด PLCC-44 Pin ซึ่งเลือกใช้ไมโครคอนโทรเลอร์ เบอร์ AT89C51ED2 ของ ATMEL เป็น MCU ประจำบอร์ด โดย MCU รุ่นนี้จะบรรจุอยู่ภายในตัวถังแบบ PLCC-44 โดย MCU ตัวนี้จะมีจุดเด่น คือ เรื่องของความเร็วในการประมวลผลซึ่งสามารถทำงานได้ด้วยความถี่สูงสุด 60MHz ที่ 12 Clock / 1 Machine Cycle นอกจากนี้แล้วยังมีความเพียบพร้อมด้วยอุปกรณ์พื้นฐานต่างๆที่จำเป็นต่อการใช้งาน ไม่ว่าจะเป็นหน่วยความจำสำหรับเก็บข้อมูลแบบ EEPROM ขนาด 2 Kbyte หรือหน่วยความจำใช้งานแบบ RAM ซึ่งมีมากถึง 1792 Byte ส่วนในด้านของอุปกรณ์ Peripheral นั้นก็นับว่าครบถ้วนเหมาะแก่การนำไปประยุกต์ใช้งานเกี่ยวกับการควบคุมและประมวลผลต่างๆได้เป็นอย่างดี โดยจะมีทั้ง SPI, UART, Watchdog, Timer/Counter, PWM ฯลฯ โดยการออกแบบโครงสร้างของบอร์ดนั้นจะเน้นเรื่องขนาดของบอร์ดให้มีขนาดเล็กเพื่อให้ง่ายต่อการนำไปประยุกต์ใช้งาน และสะดวกต่อการพัฒนาโปรแกรม

3.2.1.1 คุณสมบัติของบอร์ด

1.เลือกใช้ MCU ตระกูล MCS51 เบอร์ AT89C51ED2 ของ ATMEL เป็น MCU ประจำบอร์ด โดยคุณสมบัติเด่นๆของ MCU ได้แก่

- o มีหน่วยความจำ Flash 64KByte
- o มีหน่วยความจำ RAM ขนาด 1792 Byte
- o มีหน่วยความจำ EEPROM ขนาด 2KByte
- o มีพอร์ต I/O ขนาด 8 บิต จำนวน 4 พอร์ต (P0,P1,P2 และ P3)
- o มีวงจรสื่อสารอนุกรม UART จำนวน 1 พอร์ต และมีวงจรสื่อสาร SPI จำนวน 1 พอร์ต
- o มีวงจร Timer/Counter ขนาด 16 บิต จำนวน 3 ชุด
- o มีวงจร Watchdog, Power-ON Reset, Capture/Compare ,PWM

2.ใช้ Oscillator แบบโมดูลค่าความถี่ 29.4912MHz ซึ่งสามารถกำหนดการทำงานของ MCU ให้ทำงานในโหมดความเร็ว 2 เท่า (X2 Mode) ได้ ทำให้ MCU สามารถประมวลผลด้วยความเร็ว
สูงสุดที่ 58.9824 MHz

3. มีพอร์ตสื่อสารอนุกรมแบบ RS232 จำนวน 1 ช่อง สำหรับใช้ในการ Download โปแกรมให้กับบอร์ด และประยุกต์ใช้งานทั่วไป

4. มีขั้ว ET-DOWNLOAD สำหรับรองรับการ Download HEX File แบบอัตโนมัติร่วมกับโปรแกรม Flip เพิ่มความสะดวกในการใช้งาน

5.มีขั้วต่อสัญญาณ I/O แบบ TTL แบบ Header ขนาด 2x5 จำนวน 4 ชุด(P0,P1,P2 และP3)

6. มี I2C RTC เบอร์ DS1307 (Option) พร้อม Battery Backup

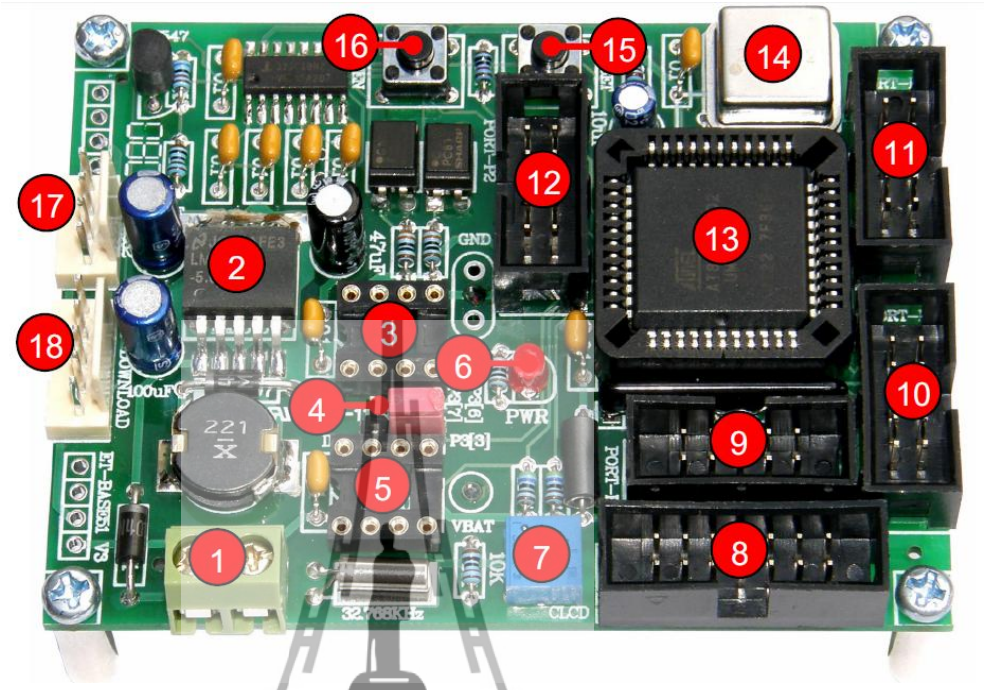
7. มี I2C EEPROM ตระกูล 24XXX (Option)

8. Header 14Pin สำหรับ Character LCD ควบคุมด้วย Port-P0 พร้อม VR ปรับความสว่าง

9. Power AC/DC Input พร้อม Regulate แบบ Switching เบอร์ LM2575 ขนาด 5V/1A ลดปัญหาความร้อนจากวงจร Regulate และ LED แสดงสถานะแหล่งจ่าย Power

10. ขนาด PCB Size เล็กเพียง 8 x 6 cm.

3.2.1.2 โครงสร้างบอร์ด Microcontroller MCS -51



รูปที่ 3.3 โครงสร้างบอร์ด Microcontroller MCS -51 เบอร์ AT89C51ED2

- หมายเลข 1 คือ ขั้วต่อแหล่งจ่ายไฟเลี้ยงวงจรของบอร์ด ใช้กับแหล่งจ่ายไฟตรง 7-12VDC
- หมายเลข 2 คือ IC Regulate แบบ Switching ขนาด 5V/1A
- หมายเลข 3 คือ Socket สำหรับติดตั้งหน่วยความจำ I2C ตระกูล 24XXX
- หมายเลข 4 คือ Jumper สำหรับ ตัด ต่อสัญญาณ Interrupt จาก DS1307 กับ INT1(P3.3)
- หมายเลข 5 คือ Socket สำหรับติดตั้ง RTC แบบ I2C เบอร์ DS1307
- หมายเลข 6 คือ LED สำหรับแสดงสถานะ ของแหล่งจ่ายไฟ +5V ของบอร์ด
- หมายเลข 7 คือ VR ปรับค่า สำหรับใช้ปรับความสว่างของหน้าจอแสดงผล LCD
- หมายเลข 8 คือ ขั้วต่อ 14PIN IDE สำหรับเชื่อมต่อกับ LCD แบบ Character
- หมายเลข 9 คือ ขั้วต่อ PORT-P0 สำหรับเชื่อมต่อสัญญาณ P0[0..7] ออกไปใช้งาน
- หมายเลข 10 คือ ขั้วต่อ PORT-P1 สำหรับเชื่อมต่อสัญญาณ P1[0..7] ออกไปใช้งาน
- หมายเลข 11 คือ ขั้วต่อ PORT-P3 สำหรับเชื่อมต่อสัญญาณ P3[2..7] ออกไปใช้งาน
- หมายเลข 12 คือ ขั้วต่อ PORT-P2 สำหรับเชื่อมต่อสัญญาณ P2[0..7] ออกไปใช้งาน
- หมายเลข 13 คือ MCU เบอร์ AT89C51ED2 ซึ่งเป็น MCU ตระกูล MCS51 จาก ATMEL

- หมายเลข 14 คือ Oscillator Module ค่า 29.4912 MHz
- หมายเลข 15 คือ SW RESET ใช้สำหรับรีเซ็ตการทำงานของ MCU ภายในบอร์ด
- หมายเลข 16 คือ Switch PSEN ใช้สำหรับกำหนดสถานะโลจิก “0” ให้ขา PSEN สำหรับใช้ใน

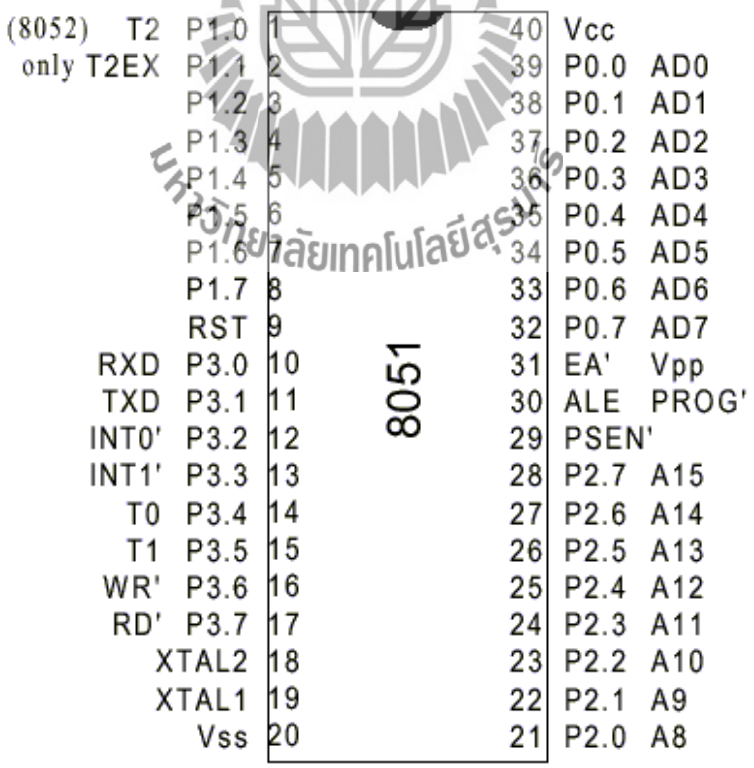
ขั้นตอนของการ Download HEX File ให้กับหน่วยความจำ Flash ของ MCU ในบอร์ด เมื่อต้องการใช้การ Download แบบ Manual

- หมายเลข 17 คือ ขั้วต่อ RS232 หรือ Serial Port สำหรับติดต่อกับอุปกรณ์มาตรฐาน RS232 และ

ใช้เป็น ISP Download Connector สำหรับ Download HEX File ให้กับ MCU ของบอร์ด เมื่อต้องการใช้การ Download แบบ Manual

- หมายเลข 18 คือ ขั้วต่อ ET-DOWNLOAD ใช้เป็น ISP Download Connector สำหรับ Download HEX File ให้กับ MCU ของบอร์ด เมื่อต้องการใช้การ Download แบบ Auto

3.2.1. 3 โครงสร้างไอซี Microcontroller MCS -51 เบอร์ AT89C51ED2



รูปที่ 3.4 โครงสร้างไอซี Microcontroller MCS -51 เบอร์ AT89C51ED2 [7]

ขาที่สำคัญของไมโครคอนโทรลเลอร์

1. ขา Vcc เป็นขารับแรงดันไฟกระแสตรง +5 Vdc

2.ขา GND เป็นขากราวด์

3. พอร์ต 0 (P0.0-P0.7) เป็นพอร์ตอินพุตเอาต์พุทแบบ 2 ทิศทาง สำหรับใช้งานทั่วไป

3. พอร์ต 0 (P0.0-P0.7) เป็นพอร์ตอินพุตเอาต์พุทแบบ 2 ทิศทาง สำหรับใช้งานทั่วไป

4. พอร์ต 1 (P1.0-P1.7) เป็นพอร์ตอินพุตเอาต์พุทแบบ 2 ทิศทาง สำหรับใช้งานทั่วไป

5. พอร์ต 2 (P2.0-P2.7) เป็นพอร์ตอินพุตเอาต์พุทแบบ 2 ทิศทาง สำหรับใช้งานทั่วไป

6. พอร์ต 3 (P3.0-P3.7) เป็นพอร์ตอินพุตเอาต์พุทแบบ 2 ทิศทาง สำหรับใช้งานทั่วไป

7. ขารีเซต (RST) ใช้สำหรับการรีเซตการทำงานของไมโครคอนโทรลเลอร์ โดยการรีเซตจะต้องคงสภาวะ high อย่างน้อยนาน 2 Machine cycle ในขณะที่ออสซิลเลเตอร์กำลังทำงานอยู่

8. ขา ALE/PROG เป็นขาสัญญาณเพื่อทำหน้าที่ควบคุมการแล็ช (Latch) ค่าตำแหน่งแอดเดรสไบต์ตำ (Address Latch Enable)

9. ขา PSEN (Program Store Enable) ทำหน้าที่เป็นสัญญาณสโตปเพื่ออ่านคำสั่งจากหน่วยความจำภายนอก เมื่อไมโครคอนโทรลเลอร์ประมวลผลคำสั่ง จากหน่วยความจำภายนอก ขานี้จะส่งสัญญาณสโตปจำนวน 2 ครั้งในแต่ละ Machine cycle แต่ในขณะที่ติดต่อกับหน่วยความจำข้อมูลภายนอกจะไม่มีการส่ง สัญญาณสโตปแต่อย่างใด

10. ขา EA/Vcc (External Access Enable/Vcc) เป็นขาสำหรับการเลือกใช้หน่วยความจำโปรแกรมจากภายในหรือภายนอก โดยมีสถานะเป็น 0 และ 1 และขานี้ยังทำหน้าที่รับแรงดันไฟสำหรับ โปรแกรม (Vcc) ขนาด 12 โวลต์ เพื่อใช้ในระหว่างการโปรแกรมหน่วยความจำโปรแกรม (EPROM)

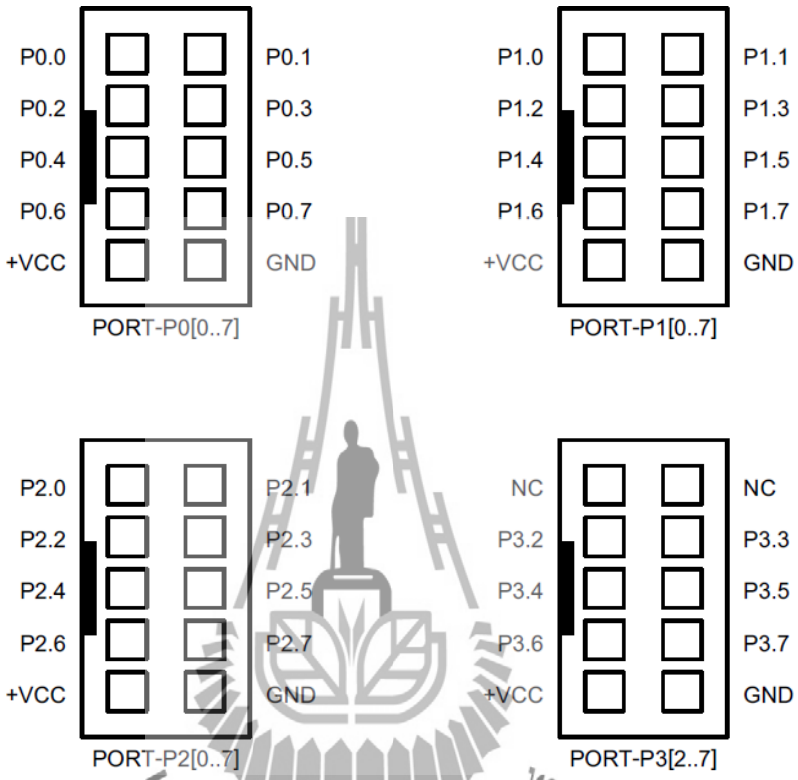
11. ขา XTAL1 และขา XTAL2 เป็นขาใช้งานของวงจรอินเวอร์ติ้งออสซิลเลเตอร์แอมปลิไฟเออร์ (Inverting Oscillator Amplifier) สำหรับใช้ต่อร่วมกับคริสตอลภายนอก

มหาวิทยาลัยเทคโนโลยีสุรนารี

3.2.1.4 ขั้วต่อสัญญาณต่างๆ

สำหรับขั้วต่อสัญญาณของพอร์ต I/O จาก MCU นั้น จะถูกออกแบบและจัดเตรียมไว้ ผ่านทางขั้วต่อแบบ IDE Header ขนาด 2x5 จำนวน 4 ชุด คือ PORT-P0, PORT-P1, PORT-P2 และ PORT-P3ตามลำดับ โดยที่ขั้วต่อสัญญาณแต่ละชุด จะประกอบไปด้วยสัญญาณของ I/O ที่เชื่อมต่อมาจากขาสัญญาณของ MCU โดยตรงทั้งหมด โดยแต่ละพอร์ตจะมีสัญญาณพอร์ตละ 8 บิต ยกเว้น PORT-P3 ซึ่งจะมีเพียง 6 บิต คือ P3.2-P3.7 เท่านั้น ส่วน P3.0 และ P3.1 จะถูกสงวนไว้ใช้งานเป็นขาสัญญาณ RXDและ TXD สำหรับรับส่งข้อมูลของ RS232 ซึ่งสัญญาณทั้ง 2 เส้น (P3.0

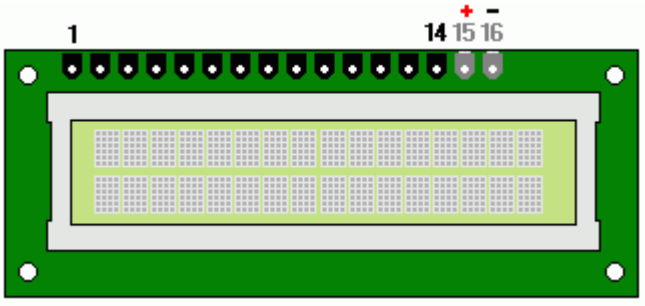
และ P3.1) จะถูกเชื่อมต่อผ่านวงจรLine Driver (MAX3232) สำหรับแปลงระดับสัญญาณจากระดับโลจิก TTL ของ MCU ให้เป็นสัญญาณแรงดันตามมาตรฐานของ RS232 โดยสัญญาณที่ได้รับการแปลงเป็นแบบ RS232 จะถูกเชื่อมต่อไปรอไว้ที่ขั้วต่อแบบ CPA ขนาด 4 PIN (RS232) และ CPA 5-PIN (ET-DOWNLOAD) โดยการจัดเรียงสัญญาณของแต่ละชุด จะเป็นดังรูป



รูปที่ 3.5 การจัดเรียงสัญญาณของพอร์ต I/O ต่างๆของบอร์ด

3.2.2 จอแสดงผลผลึกเหลว (LCD)

ในที่นี้เราจะใช้ จอแสดงผลผลึกเหลว (LCD) โมดูล 16X2 Line คือจะสามารถแสดงผลได้ทั้งหมด 16 ตัวอักษร 2 บรรทัด นั่นเอง



รูปที่ 3.6 จอแสดงผลผลึกเหลว (LCD) โมดูล 16X2 Line [8]

3.2.2.1ตำแหน่งของขาและหน้าที่การใช้งานของ LCD โมดูล

Pin No.	Symbol	Description	Level	Function	
1	VSS	Ground	-	0V	Ground
2	VDD	Power Supply	-	+5V	ต่อกับแรงดันไฟเลี้ยง +5V
3	VO	LCD Contr	-	-	ต่อกับแรงดันเพื่อปรับความเข้มของการแสดงผล
4	RS	Register Select	H/L	RS = 0 หมายถึงต้องการติดต่อกับรีจิสเตอร์คำสั่ง (Instruction Register) RS = 1 หมายถึงต้องการติดต่อกับรีจิสเตอร์ข้อมูล (Data Register)	
5	R/W	Read/Write	H/L	R/W = 0 หมายถึงต้องการเขียนข้อมูลไปยัง LCD โมดูล R/W = 1 หมายถึงต้องการอ่านข้อมูลจาก LCD โมดูล	
6	E	Enable	H, H->L	Enable Signal	
7 - 14	DB0-DB7	Data Bus	H/L	Data Bus Line	
15	A	Back Light A	-	Back Light +5V (สำหรับรุ่นที่มี Back Light)	
16	K	Back Light K	-	Back Light 0V (สำหรับรุ่นที่มี Back Light)	

ตารางที่ 3.1 ตารางแสดงตำแหน่งของขา และหน้าที่การใช้งาน LCD [8]

มหาวิทยาลัยเทคโนโลยีสุรนารี

3.2.2.2 คำสั่งควบคุมการแสดงผลของ LCD โมดูล

Instruction		RS	R/W	Command Code (binary)								Description
				7	6	5	4	3	2	1	0	
1	Clear Display	0	0	0	0	0	0	0	0	0	1	Clear entire display and move cursor home (address 0)
2	Home Display	0	0	0	0	0	0	0	0	1	0	Move cursor home and return display to home position.
3	Entry Mode Set	0	0	0	0	0	0	0	1	M	S	Sets cursor direction (M: 0=left, 1=right) and display scrolling (S: 0=no scroll, 1=scroll)
4	Display/Cursor	0	0	0	0	0	0	1	D	C	B	Sets display on/off (D), cursor on/off (C) and blinking cursor (B). (0=off, 1=on)
5	Cursor or Display Shift	0	0	0	0	0	1	C	M	0	0	Cursor or Display Shift (C: 0=cursor, 1=display) left or right (M: 0=left, 1=right).
6	Function Set	0	0	0	0	1	D	N	F	0	0	Data bus size (D: 0=4-bits, 1=8-bits), lines No.(N: 0=1-line, 1=2-lines) and font size (F: 0=5x7, 1=5x10)
7	Set CG-RAM Address	0	0	0	1	CGRAM ADDRESS						Move pointer to Character Generator RAM location specified by address (ADDRESS)
8	Set DD-RAM Address	0	0	1	DDRAM ADDRESS							Move cursor to Display Data RAM location specified by address (ADDRESS)
9	Busy, ADD.Read	0	1	BF	ADDRESS							Read Busy flag, And Address Read
10	CGRAM,DDRAM WR	1	0	WRITE DATA								Write Data to DDRAM or CGRAM
11	CGRAM,DDRAM RD	1	1	READ DATA								Read Data to DDRAM or CGRAM

ตารางที่ 3.2 ตารางแสดงคำสั่งควบคุมการแสดงผล LCD [9]

มหาวิทยาลัยเทคโนโลยีสุรนารี

รายละเอียดคำสั่ง

❖ เคลียร์การแสดงผล (Clear Display)

7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	1

Bit 0=1 เคลียร์การแสดงผล
เคอร์เซอร์กลับไปอยู่ที่มุมซ้ายมือสุด

❖ Home Display

7	6	5	4	3	2	1	0
0	0	0	0	0	0	1	-

Bit1=1เคอร์เซอร์กลับไปอยู่ที่มุมซ้ายมือสุด
ข้อมูลไม่มีการเปลี่ยนแปลง

3.2.2.3 ตำแหน่ง Address ของ LCD

Line1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Line2	64	65	66	67	68	69	79	71	72	73	74	75	76	77	78	79

ตารางที่ 3.3 ตารางแสดงตำแหน่ง Address ของ LCD [9]

กรณีใช้งาน LCD โมดูลแบบ 2x16 ต้องการให้แสดงผลที่บรรทัดที่ 2 ใน column แรกจะมีค่า address = 64(dec) โดยการนำคำสั่งนี้คือ 10000000 หรือ 0x80 มา OR กับ address 0x80 OR 64(dec) = 0xC0 จะเป็นชุดคำสั่งที่ใช้เขียนไปยังโมดู LCD

3.3 ส่วน Software

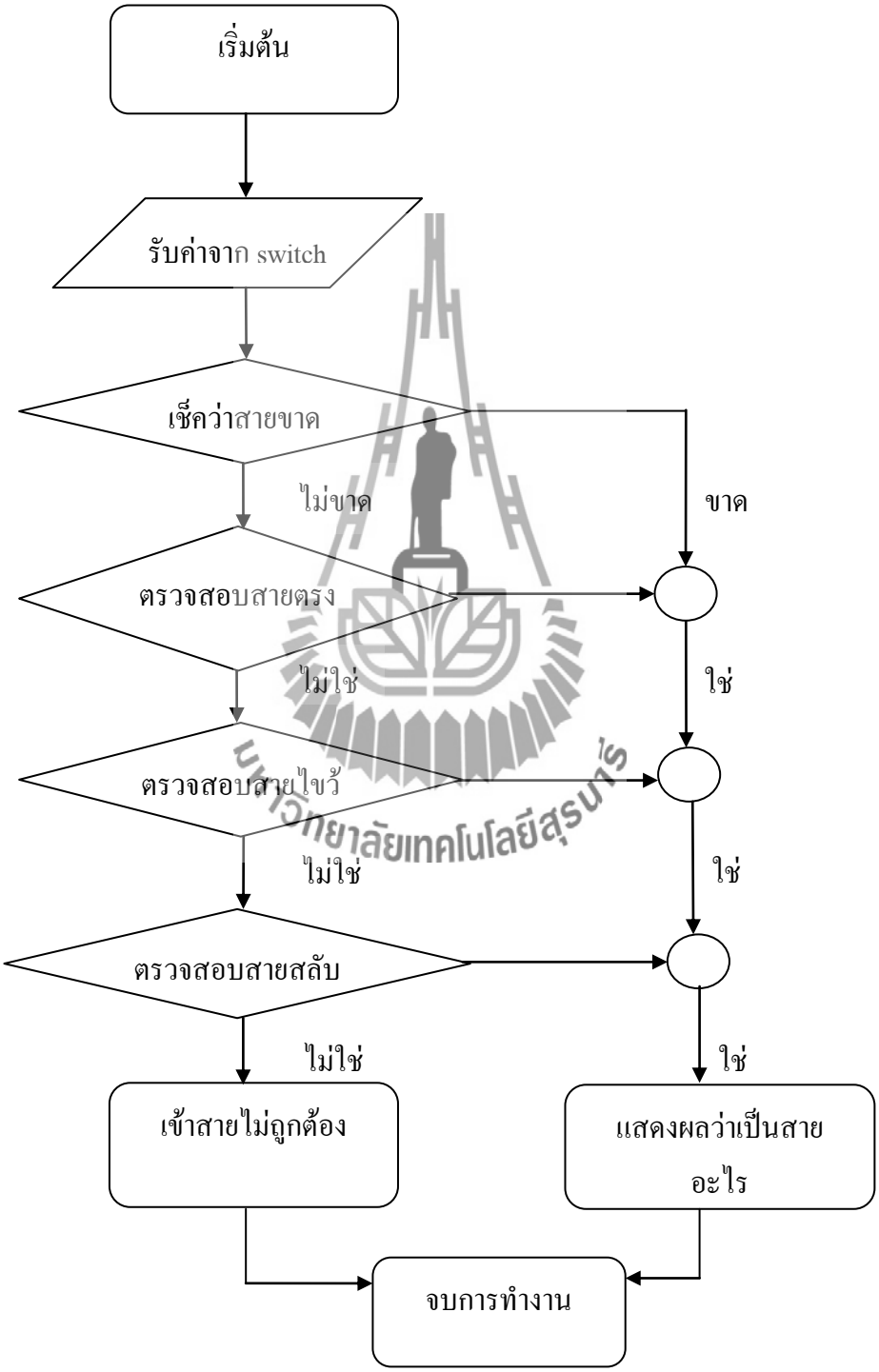
ในส่วนนี้จะประกอบไปด้วย โปรแกรมที่เกี่ยวข้องระบบนี้คือ โปรแกรมที่สั่งการให้ Microcontroller ทำการตรวจสอบสายเคเบิล (UTP) โดยใช้โปรแกรม Keil uVision4 ในการเขียนคำสั่งนี้ขึ้นมา และใช้โปรแกรม ISIS 7 Professional ในการทดสอบการตรวจสอบสายเคเบิล (UTP) ก่อนที่จะทำการเบิร์นโปรแกรมลงบอร์ด ส่วนการใช้งานโปรแกรมจะอธิบายในภาคผนวก

3.3.1 โปรแกรมที่สั่งการให้ Microcontroller ทำงาน

โดยใช้โปรแกรม Keil uVision4 ในการเขียนคำสั่งโปรแกรมที่สั่งการให้MCSซึ่งแสดงอยู่ภาคผนวก

3.3.1.1 Flow chart

การทำงานของโปรแกรม ที่ใช้สั่งการให้ Microcontroller ตรวจสอบสาย UTP



รูปที่ 3.6 flow chart การทำงานของบอร์ด

อธิบายการทำงานของโปรแกรม

เริ่มต้นโปรแกรมจะ ทำการรับค่าจากสวิตซ์ก่อน จากนั้นโปรแกรมจะทำการตรวจสอบสายตีเกลียวทุกๆเส้น ทั้งหมด 8 เส้น โดยการกำหนดให้ Output ส่งเป็น Logic 0 ออกไป แล้วดูว่า Input รับได้ Logic 0 หรือ Logic 1 ถ้าฝั่ง Input รับได้ Logic 1 แสดงว่าสายเส้นนั้นขาดนั้นเอง

ถ้าเกิด Input รับได้ Logic 1 เพียงเส้นเดียวโปรแกรมก็จะจบการทำงานทันที

แต่ถ้า Input รับได้ Logic 0 หมด แสดงว่าสายตีเกลียว 8 เส้น ผ่านหมด จากนั้นโปรแกรมจะทำการตรวจสอบว่าเป็นสาย UTP ประเภทไหน

โดยการ ส่ง Logic 1 ไปที่ละเส้นของสายตีเกลียว แล้วดูว่า Input รับได้ ตรงกับที่ Output ส่งมาให้หรือไม่ โดยการเริ่มตรวจสอบที่สายตรงก่อน ถ้าใช้โปรแกรมก็จะจบการทำงานทันที แล้วจะแสดงผลออกทางจอ LCD ว่าเป็นสายตรง

แต่ถ้าไม่ใช่โปรแกรมก็จะทำการตรวจสอบสายแบบไขว้ต่อไป ถ้าใช้โปรแกรมก็จะจบการทำงานทันที แล้วจะแสดงผลออกทางจอ LCD ว่าเป็นสายไขว้

แต่ถ้าไม่ใช่โปรแกรมก็จะทำการตรวจสอบสายแบบสลับต่อไป ถ้าใช้โปรแกรมก็จะจบการทำงานทันที แล้วจะแสดงผลออกทางจอ LCD ว่าเป็นสายสลับ แต่ถ้าไม่ใช่โปรแกรมก็จะจบการทำงานทันที แสดงว่ามีการเข้าสายผิดนั่นเอง

มหาวิทยาลัยเทคโนโลยีสุรนารี

บทที่4
การทดสอบเครื่องตรวจสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่น

4.1 บทนำ

บทนี้จะเป็นการทดลองเพื่อทดสอบความ สามารถ ของเครื่องตรวจสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่น ว่าสามารถตรวจสายเคเบิลได้คลาดเคลื่อนมากน้อยเพียงใด โดยทำการทดสอบหลายกรณี ทั้งตรวจสอบว่าเป็นสายประเภทใด สายตรง สายไขว้ สายสลับ สายขาด สายที่เรียงผิด รวมถึงตรวจสอบความยาวของสายมีผลต่อการใช้งาน เพื่อทดสอบว่า เครื่องตรวจสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่นนี้ใช้งานได้จริงหรือไม่

4.2อุปกรณ์ที่ใช้ในการทดลอง

1.เครื่องตรวจสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่น ประกอบไปด้วย

- Microcontroller รุ่น AT89C51ED2
-จอ LCD จำนวน 1 ตัว
-หัว RJ 45 ตัวเมีย จำนวน 2 ตัว
-สวิตช์ จำนวน 1 ตัว
- Adapter จำนวน 1 ตัว

2.สายเคเบิล (สายUTP)

3.คอมพิวเตอร์

4.สาย RS232

5.โปรแกรม FLIP 2.4.6, โปรแกรม ISIS 7 Professional

มหาวิทยาลัยเทคโนโลยีสุรนารี

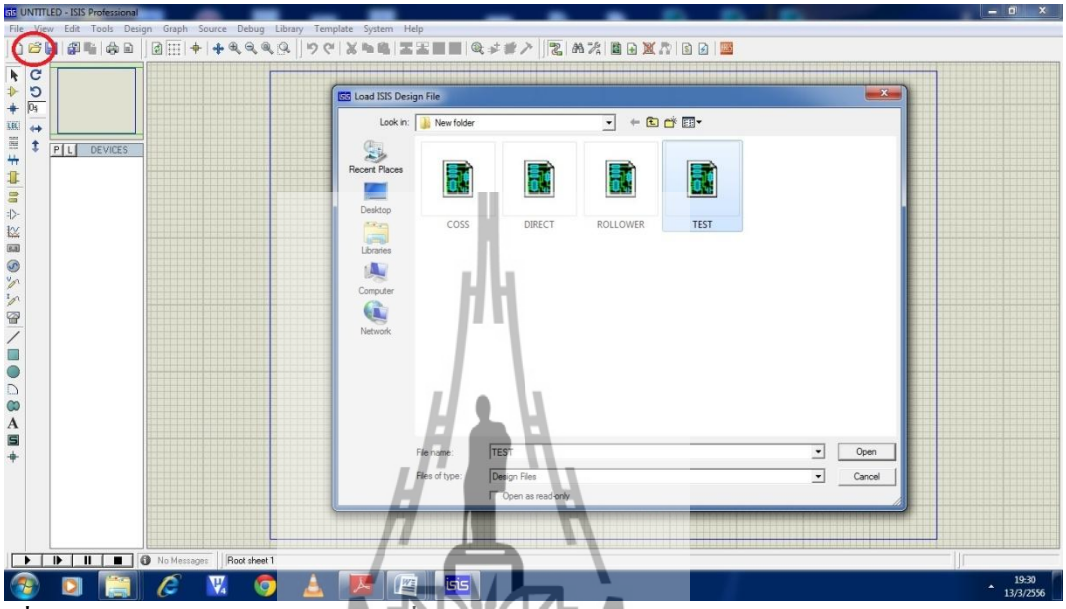
4.3 ออกแบบการทดลอง

การทดลองจะแบ่งออกเป็น 2 ส่วน ส่วนแรกจะเป็นส่วนของการใช้โปรแกรม ISIS 7 Professional เพื่อทำการตรวจสอบcodeที่ได้ทำการเขียนไว้ในโปรแกรม Keil uVision 4 ก่อนที่จะทำการโหลดลงบอร์ดว่าถูกต้องหรือมีข้อผิดพลาดตรงไหน ส่วนที่สองจะเป็นการตรวจสอบสาย UTPโดยเครื่องตรวจสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่น

4.4วิธีการทดลอง

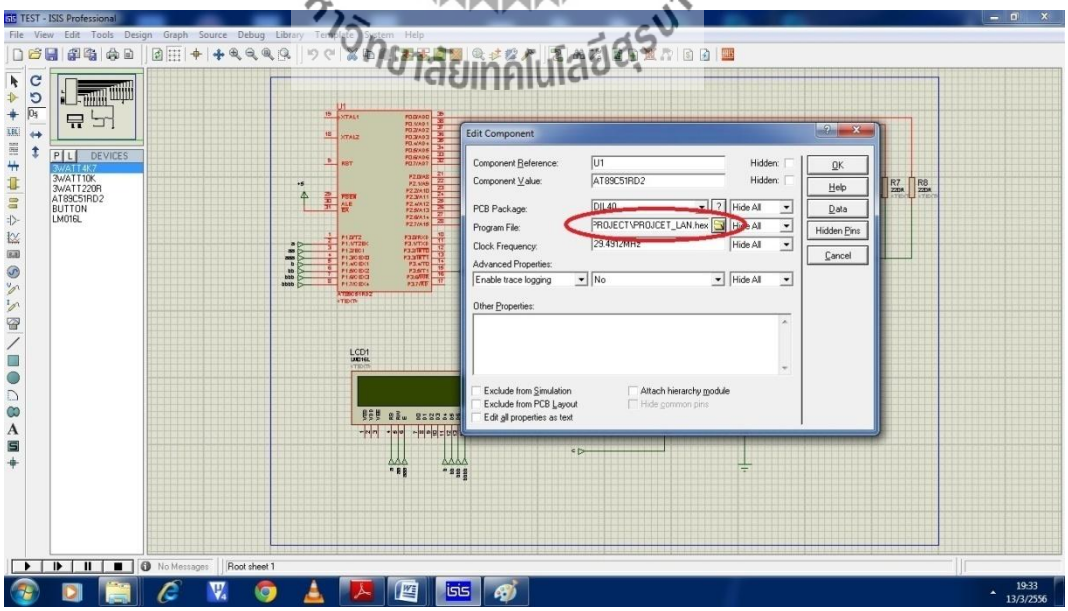
4.4.1โปรแกรม ISIS 7 Professional

1. Download File วงจรเครื่องตรวจสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่น ลงในโปรแกรม ISIS 7 Professional



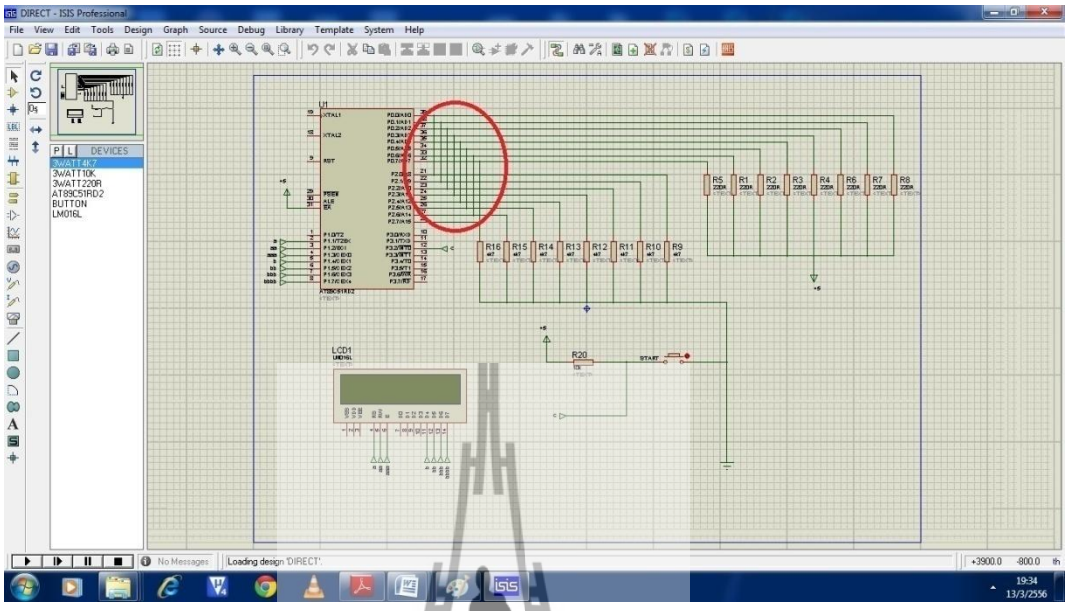
รูปที่4.1. แสดงDownload File วงจรเครื่องตรวจสอบสายเคเบิลลงในโปรแกรม ISIS 7 Professional

2. Download Hex File ใส่ Microcontroller

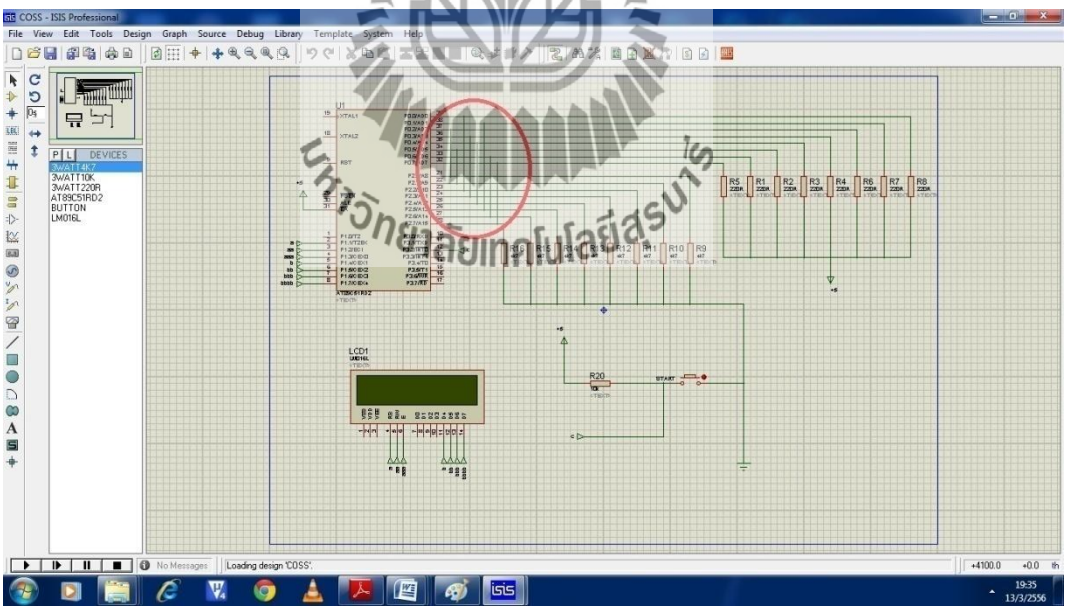


รูปที่4.2 แสดงDownload Hex File ใส่ Microcontroller

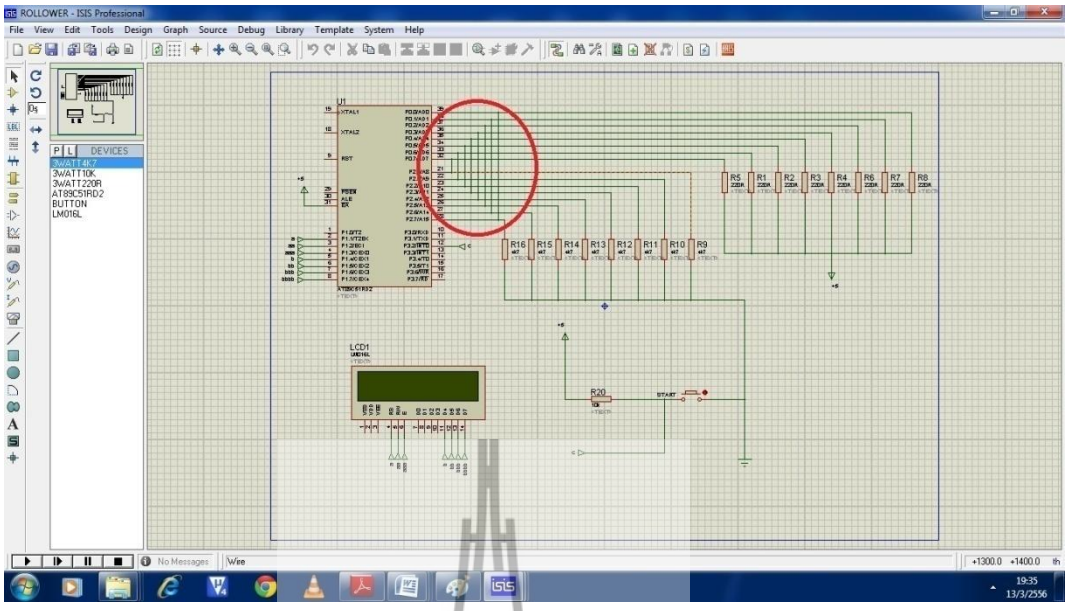
3.ทำการเรียงสายตามประเภทที่เราจะตรวจสอบ



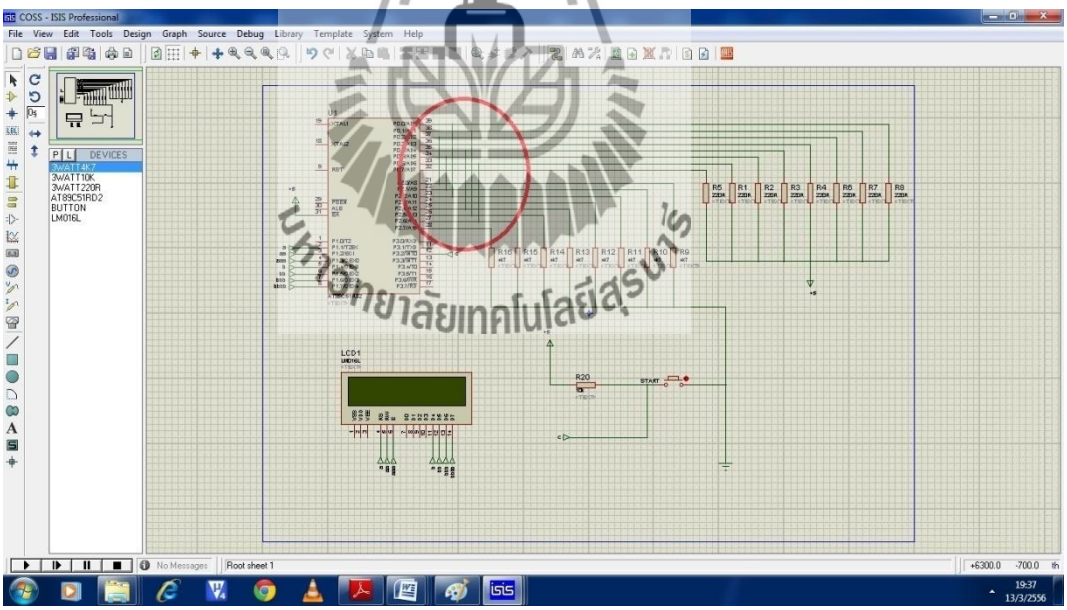
รูปที่4.3 แสดงเรียงสายแบบตรง



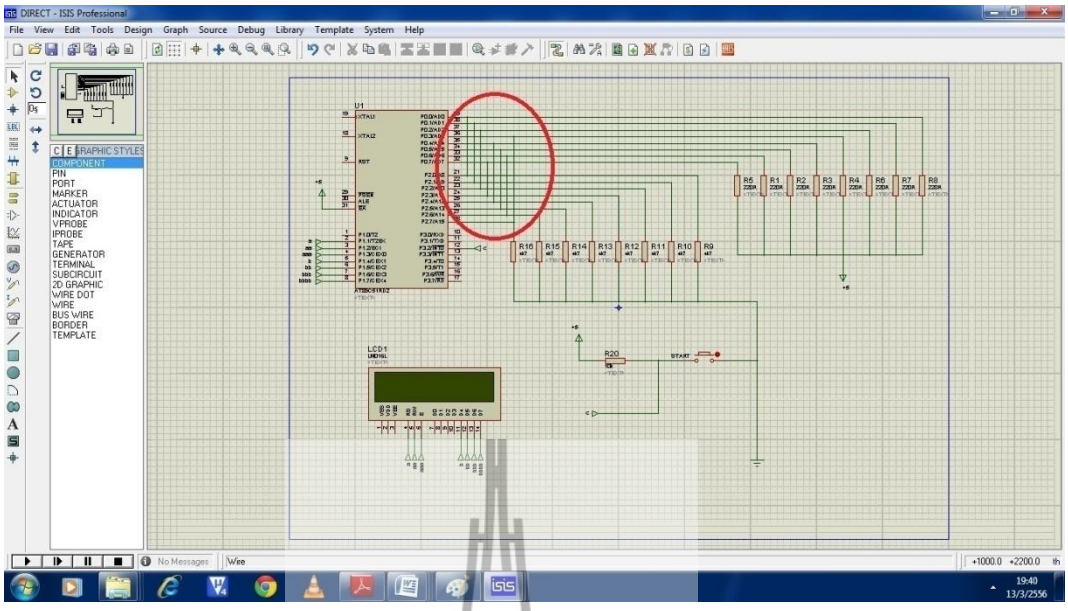
รูปที่4.4 แสดงเรียงสายแบบไขว้



รูปที่4.5 แสดงเรียงสายแบบสลับ

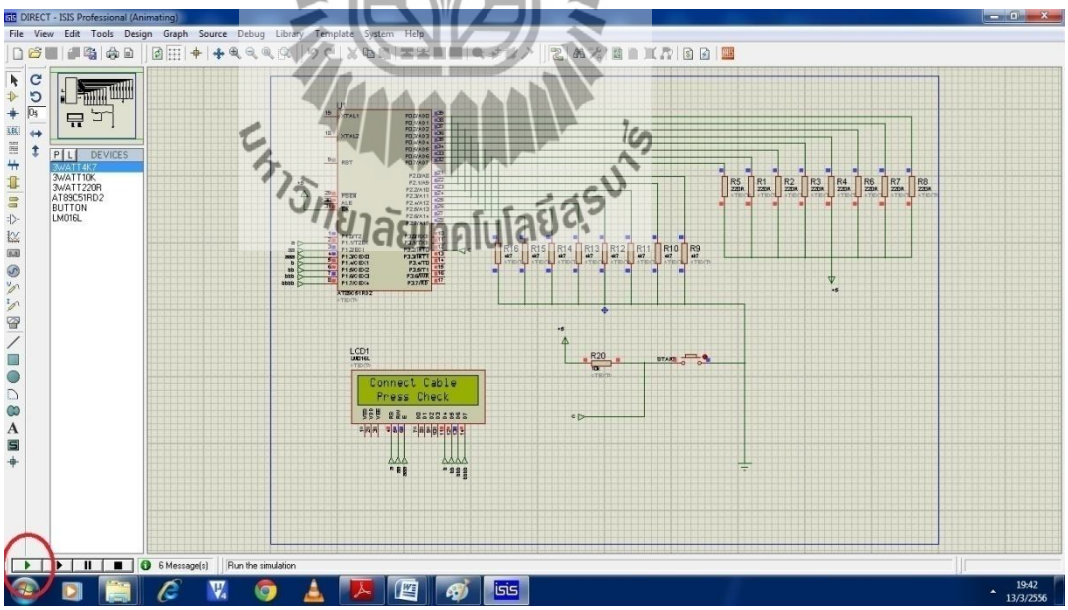


รูปที่4.6 แสดงเรียงสายแบบสายขาด



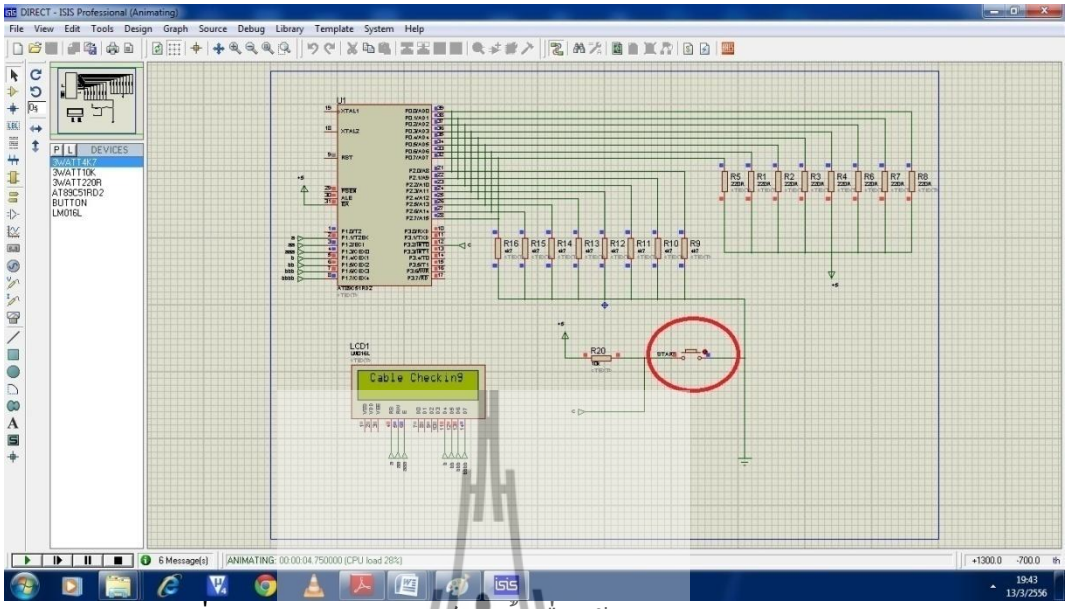
รูปที่4.7 แสดงเรียงสายแบบสายที่เรียงผิด

4.กดปุ่มPlayเพื่อให้วงจรเริ่มทำงาน



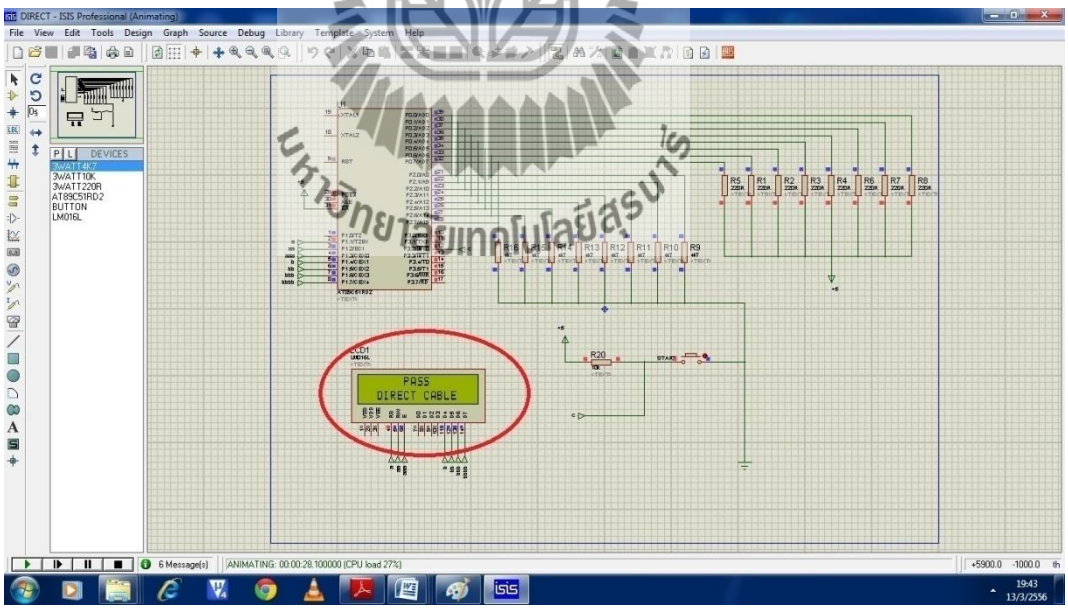
รูปที่4.8 แสดงกดปุ่มPlayเพื่อให้วงจรเริ่มทำงาน

5.กดปุ่มสวิตช์ 1 ครั้งเพื่อให้วงจรทำการตรวจสอบสาย



รูปที่4.9 แสดงกดปุ่มสวิตช์ 1 ครั้งเพื่อให้วงจรทำการตรวจสอบสาย

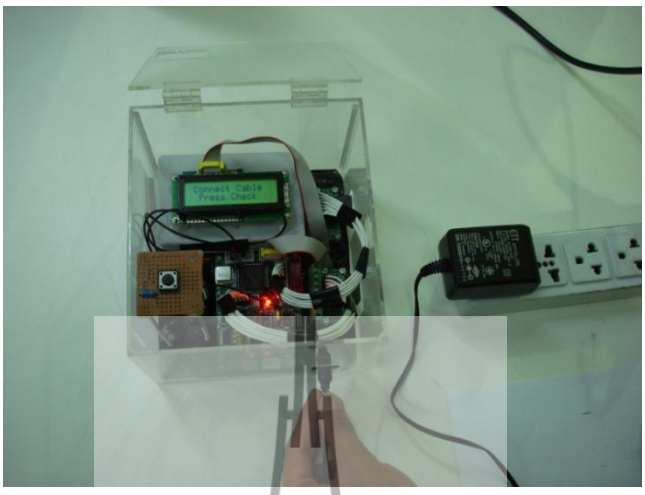
6.ดูผลการตรวจสอบที่จอLCD



รูปที่4.10 แสดงผลการตรวจสอบที่จอLCD

4.2.2เครื่องตรวจสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่น

1.จ่ายไฟ 5-12V ให้กับเครื่องตรวจสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่น



รูปที่4.11แสดงจ่ายไฟ5-12Vให้กับเครื่องตรวจสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่น

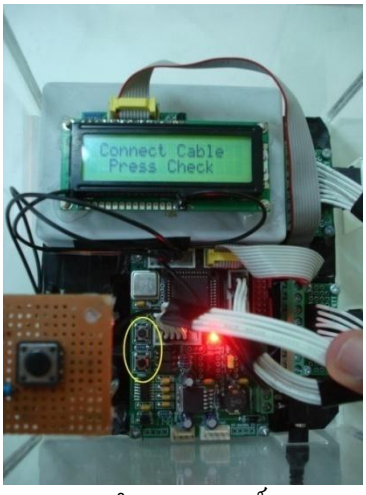
2. ทำการ Reset บอร์ดMicrocontroller มีขั้นตอนดังนี้

1. กดสวิตช์ PSEN ค้างไว้เพื่อกำหนดสถานะขาสัญญาณ PSEN ให้เป็น “0”

2. กดสวิตช์ RESET เพื่อส่งสัญญาณ RESET ให้ MCU โดยสวิตช์ PSEN ต้องกดค้างอยู่เช่นเดิม

3. ปล่อยสวิตช์ RESET เพื่อให้ MCU พ้นจากสภาวะการ Reset (สวิตช์ PSEN ยังกดค้างอยู่)

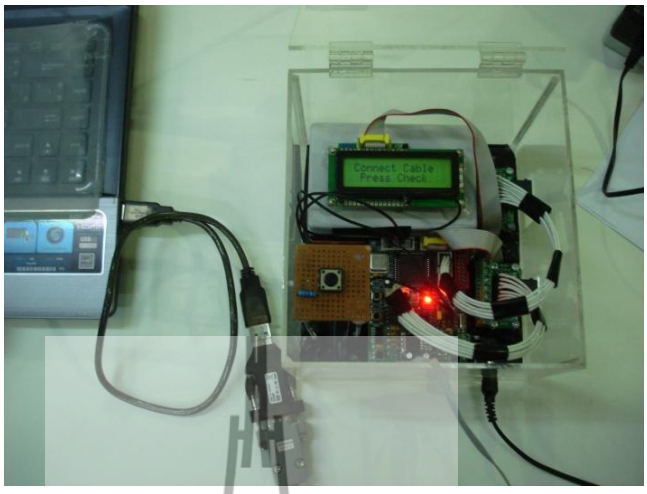
4. ปล่อยสวิตช์ PSEN เป็นลำดับสุดท้าย



รูปที่4.12แสดงการทำ Reset บอร์ดMicrocontroller

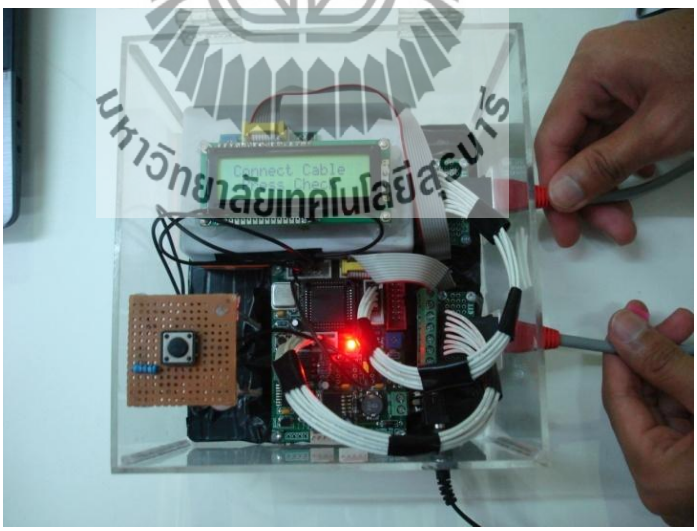
มหาวิทยาลัยเทคโนโลยีสุรนารี

3. Download Hex File ด้วยโปรแกรม Flip ให้กับบอร์ด Microcontroller ผ่านทางพอร์ตสื่อสารอนุกรม RS232



รูปที่4.13 แสดงDownload Hex File ด้วยโปรแกรม Flip ให้กับบอร์ด Microcontroller

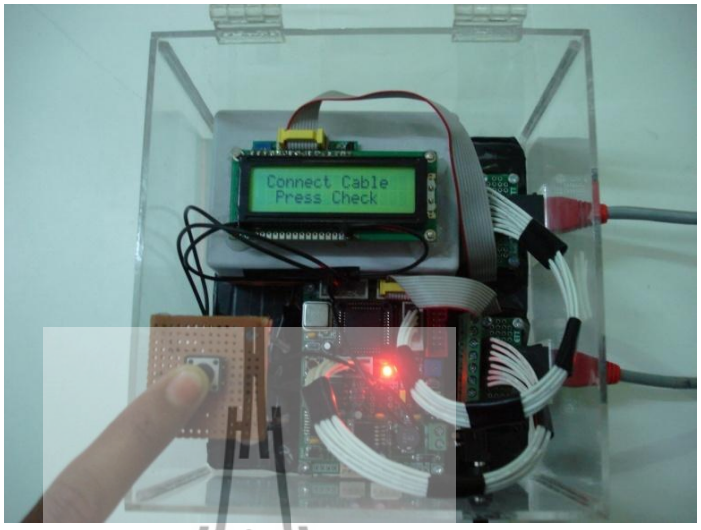
4. นำสายเคเบิล (UTP) ที่เตรียมไว้ มาต่อกับหัว RJ 45 ตัวเมียทั้งด้าน input และ output



รูปที่4.14แสดงนำสายเคเบิล (UTP) ที่เตรียมไว้ มาต่อกับหัว RJ 45 ตัวเมียทั้งด้าน input และ output

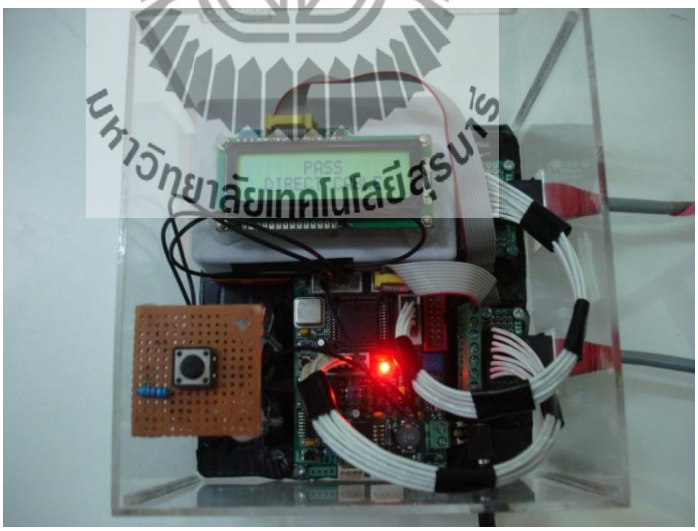
มหาวิทยาลัยเทคโนโลยีสุรนารี

5. กด Switch เพื่อตรวจสอบสายเคเบิล (UTP)



รูปที่4.15 กด Switch เพื่อตรวจสอบสายเคเบิล (UTP)

6.ดูผลการทดลองแสดงที่จอLCD



รูปที่4.16 แสดงผลการทดลองที่จอLCD

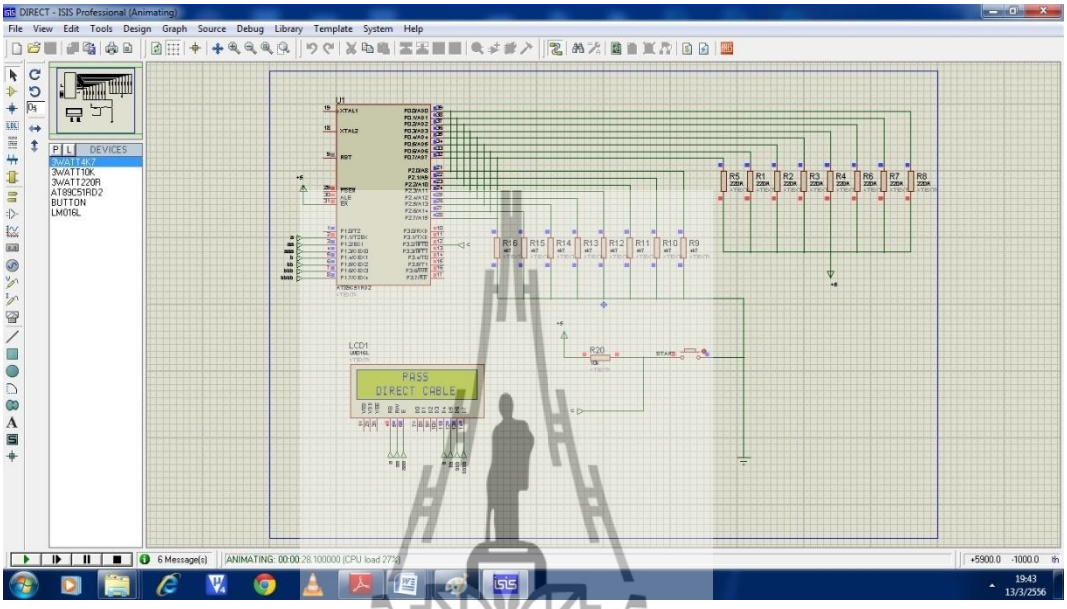
มหาวิทยาลัยเทคโนโลยีสุรนารี

4.5ผลการทดลอง

4.5.1โปรแกรม ISIS 7 Professional

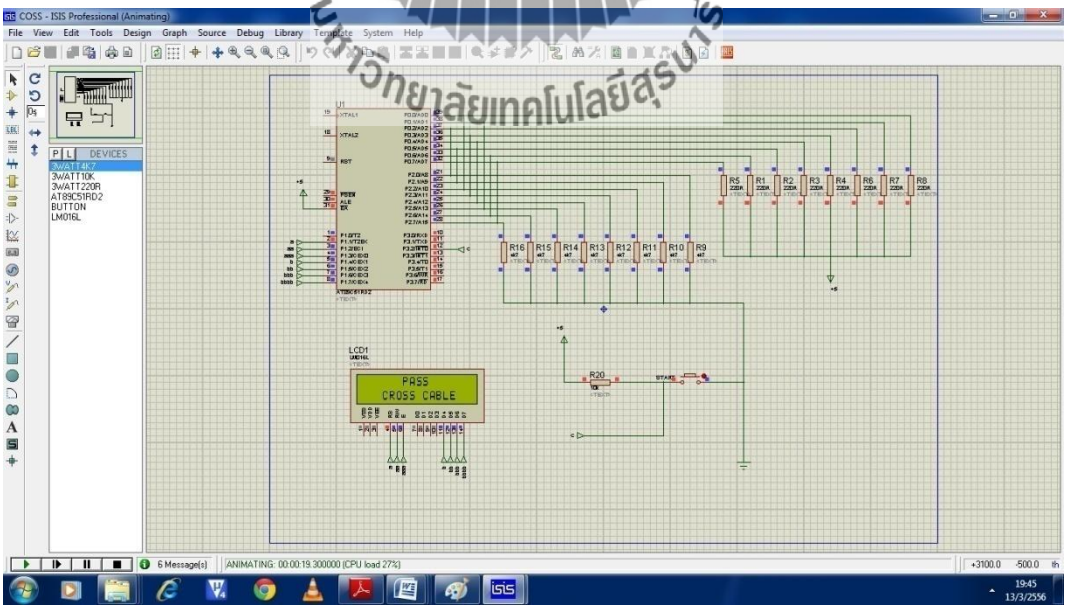
ผลการทดลองส่วนที่ 1

4.5.1.1สายตรง



รูปที่4.17 แสดงผลการตรวจสอบสายตรงที่จอLCD

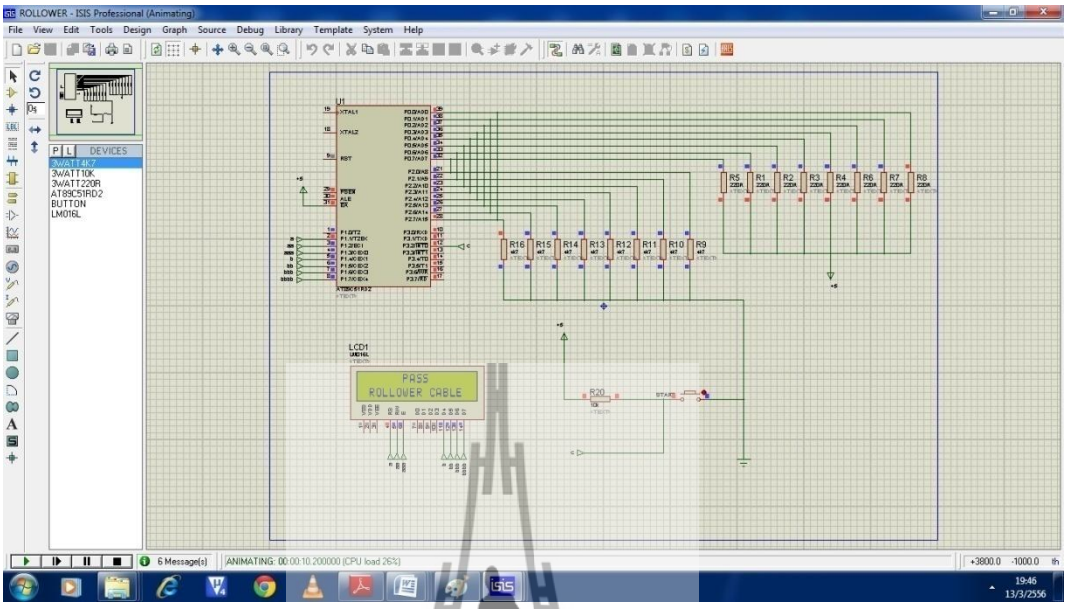
4.5.1.2สายไขว้



รูปที่4.18แสดงผลการตรวจสอบสายไขว้ที่จอLCD

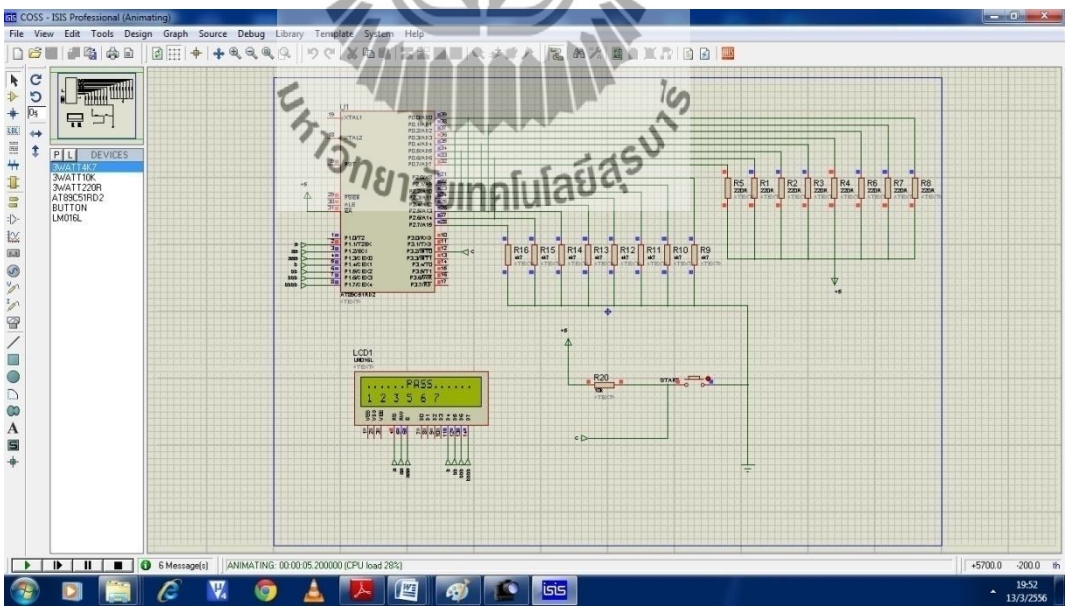
มหาวิทยาลัยเทคโนโลยีสุรนารี

4.5.1.3สายสลับ

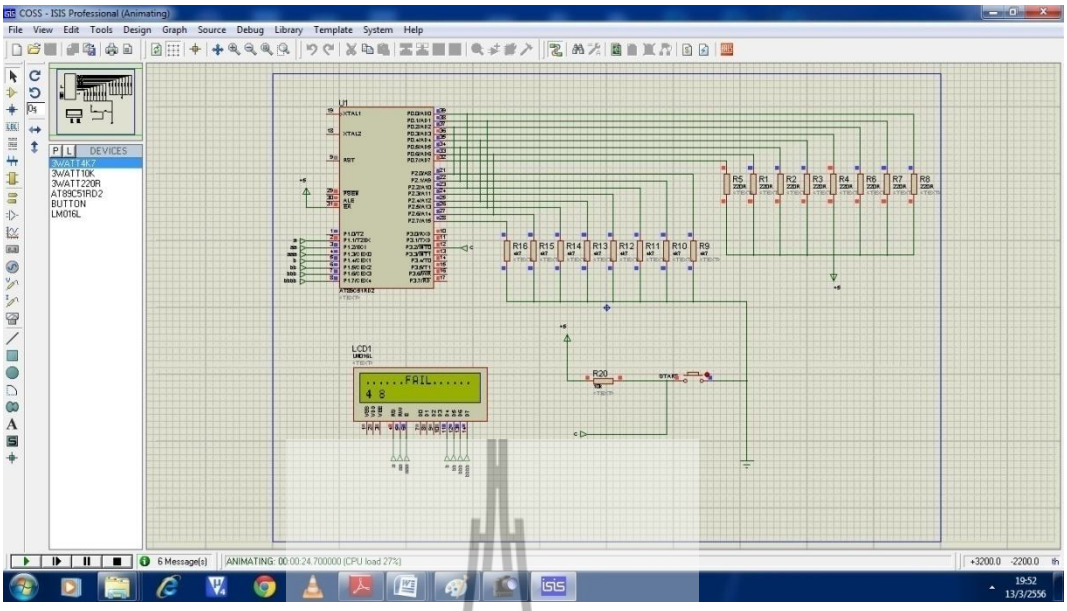


รูปที่ 4.19 แสดงผลการตรวจสอบสายสลับที่จอLCD

4.5.1.4สายขาด

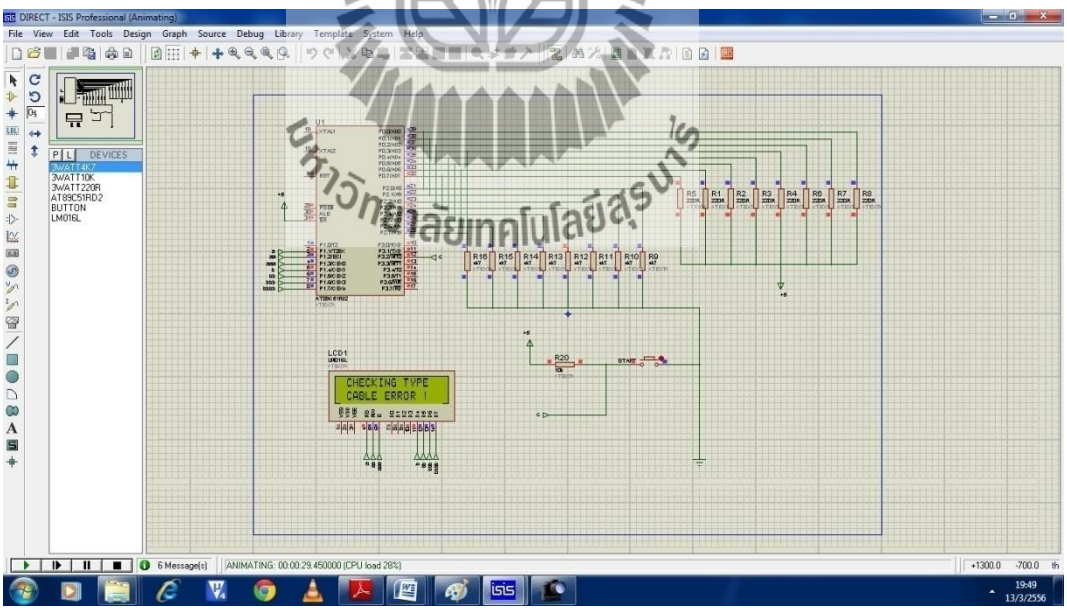


รูปที่4.20 แสดงผลการตรวจสอบสายขาดที่จอLCD



รูปที่4.21 แสดงผลการตรวจสอบสายขาดที่จอLCD

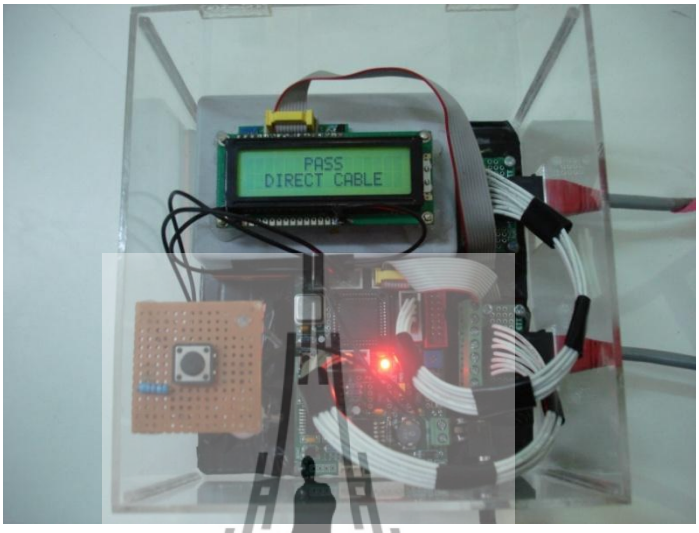
4.5.1.5สายที่เรียงผิด



รูปที่4.22แสดงผลการตรวจสอบสายที่เรียงผิดที่จอLCD

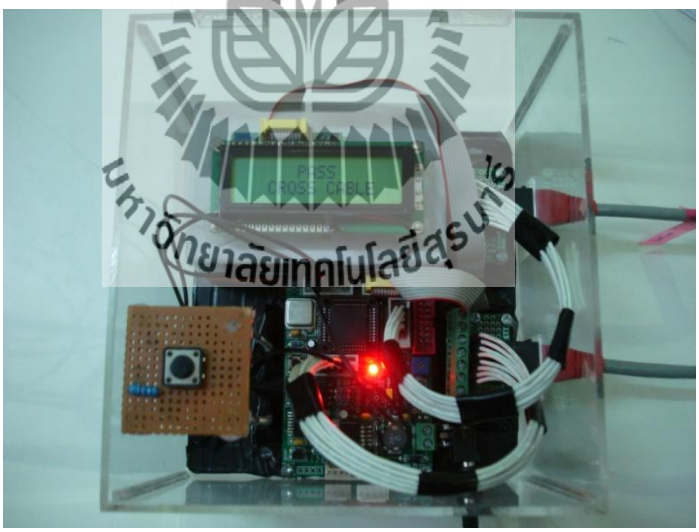
4.5.2 เครื่องตรวจสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่น ผลการทดลองส่วนที่ 2

4.5.2.1สายตรง



รูปที่4.23 แสดงผลการทดลองแสดงที่จอLCDของสายตรง

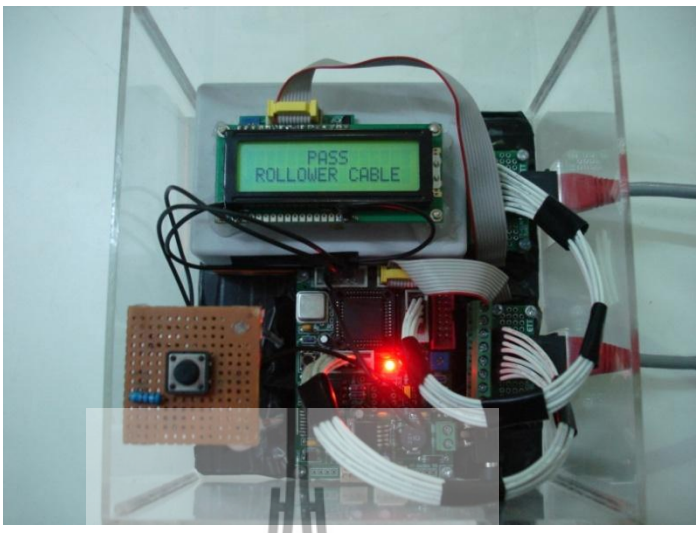
4.5.2.2สายไขว้



รูปที่4.24 แสดงผลการทดลองแสดงที่จอLCDของสายไขว้

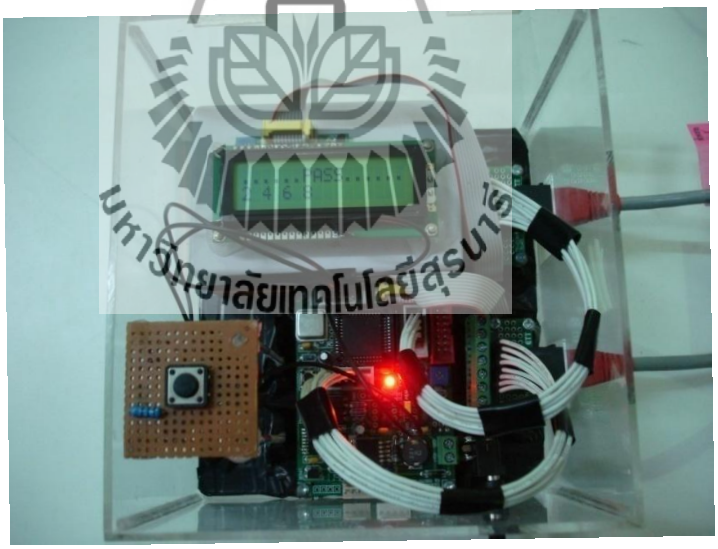
มหาวิทยาลัยเทคโนโลยีสุรนารี

4.5.2.3สายสลับ



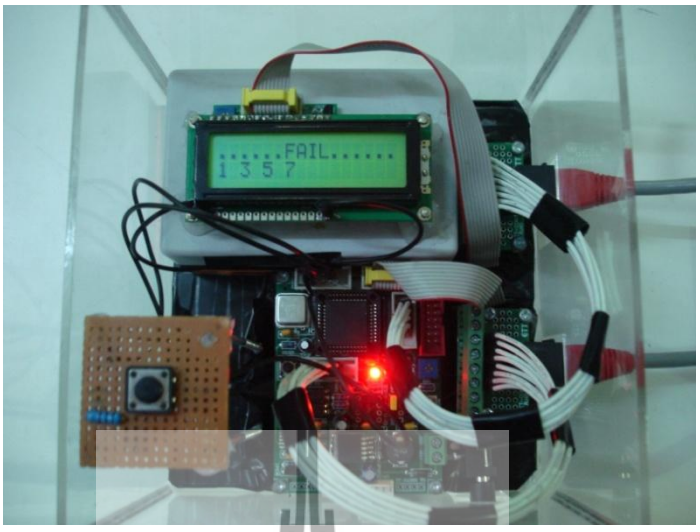
รูปที่4.25 แสดงผลการทดลองแสดงที่จอLCDของสายสลับ

4.5.2.4สายขาด



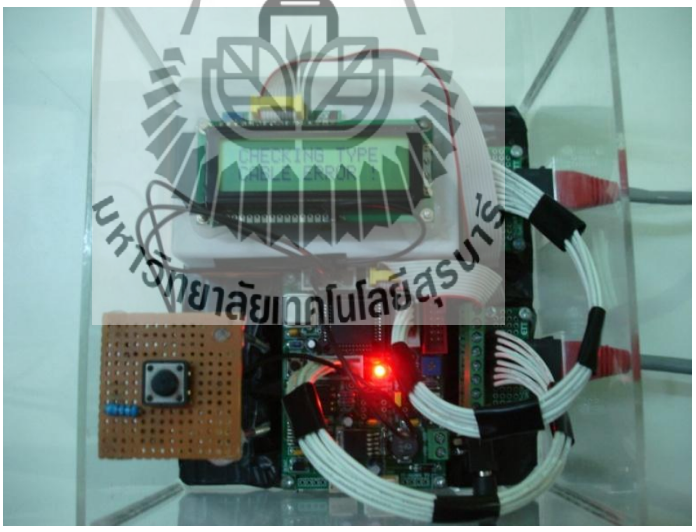
รูปที่4.26 แสดงผลการทดลองแสดงที่จอLCDของสายขาด

มหาวิทยาลัยเทคโนโลยีสุรนารี



รูปที่4.27 แสดงผลการทดลองแสดงที่จอLCDของสายขาด

4.5.2.5 สายที่เรียงผิด

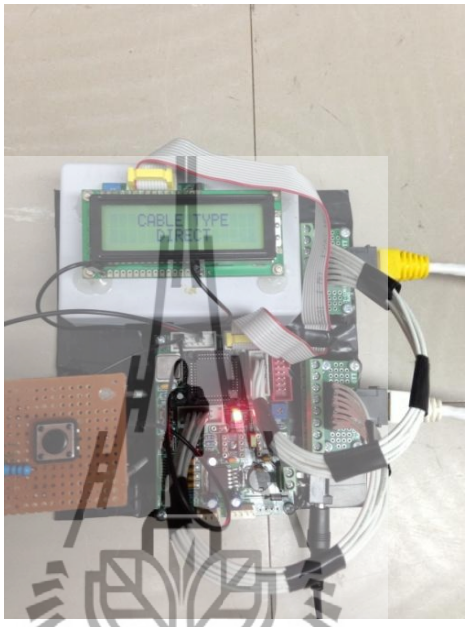


รูปที่4.28 แสดงผลการทดลองแสดงที่จอLCDของสายที่เรียงผิด

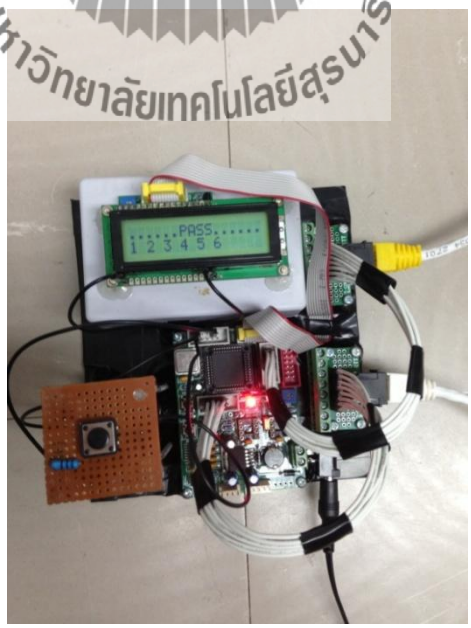
มหาวิทยาลัยเทคโนโลยีสุรนารี

4.5.2.6 ความยาวสาย

เป็นการทดลองเพื่อตรวจสอบประสิทธิภาพของเครื่องตรวจสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่นว่ามีความสามารถในการตรวจสอบสายเคเบิลที่มีความยาวมากๆ ซึ่งในการทดลองนี้ใช้สายเคเบิล(UTP) ที่มีความยาว141เมตร

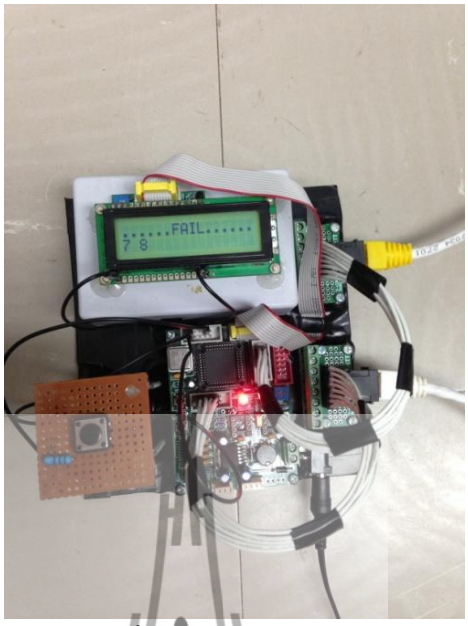


รูปที่4.29 แสดงผลการทดลองแสดงที่จอ LCDของสายเคเบิล (UTP) ที่มีความยาว131 เมตร



รูปที่4.30 แสดงผลการทดลองแสดงที่จอ LCD ของสายเคเบิล (UTP) ที่มีความยาว141 เมตร

มหาวิทยาลัยเทคโนโลยีสุรนารี



รูปที่4.31 แสดงผลการทดลองแสดงที่จอLCDของสายเคเบิล (UTP) ที่มีความยาว 141 เมตร

4.6 วิเคราะห์ผลการทดลอง

เครื่อง ตรวจสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่นนี้มีการส่งแบบลอจิก"1"และลอจิก"0" ซึ่งจากการทดลองการทำงานจะไม่มีผลต่อความยาวเมื่อนำตรวจสอบกับสายเคเบิล (UTP) ดังนั้นจากการทดลองโดยใช้สายเคเบิล (UTP) ที่มีความยาว141เมตรกับเครื่องตรวจสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่นปรากฏว่าเครื่องสามารถทำการตรวจสอบได้ปกติ

4.7 สรุปผลการทดลอง

จากการผลการทดลอง เครื่องตรวจสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่นนี้สามารถตรวจสอบสายตรง สายไขว้ สายสลับ สายขาด สายที่เรียงผิด สามารถใช้งานได้จริง รวมถึงความยาวของสายไม่มีผลต่อการตรวจสอบของเครื่อง

บทที่ 5

สรุปผลการทดลองและข้อเสนอแนะ

5.1 สรุปผล

งานวิจัยนี้ ผู้วิจัยได้ทำเครื่องตรวจสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่นซึ่งได้มีส่วนประกอบหลักๆ 2 ส่วนคือ ส่วนโปรแกรม Keil uVision 4 สำหรับเขียนคำสั่งป้อนให้กับบอร์ดMicrocontroller และส่วนโปรแกรม ISIS 7 Professional ไว้สำหรับออกแบบวงจรในการตรวจสอบสายเคเบิล (UTP)

โดยหลักการทำงานในส่วนของ Microcontroller โปรแกรมจะเริ่มต้นโดยจากการรับค่าจากสวิตซ์ จากนั้นโปรแกรมจะทำการตรวจสอบสายตีเกลียวทั้ง 8เส้น โดยการกำหนดให้ Output ส่งเป็น Logic 0 ออกไปทุกๆเส้น จากนั้น input จะดูว่าได้รับ Logic 0 หรือLogic 1

ถ้าฝั่ง Input ได้รับ Logic 0 ทั้งหมดทุกๆเส้นโปรแกรมจะทำการแสดงผลว่า สายทั้ง8เส้นมีการเชื่อมต่อกันทั้งหมด (แต่ยังไม่รู้ว่าสายนี้เข้าถูกหรือผิด) ผ่านทางจอLCD

จากนั้นโปรแกรมจะตรวจสอบด้าน Input ได้รับ Logic ตรงตามที่กำหนดในโปรแกรมไว้โดยเริ่มจากตรวจสอบสายตรงก่อน ถ้าตรงตามที่กำหนดไว้โปรแกรมจะแสดงผลว่าเป็นสายตรงทางจอLCDและจบการทำงาน แต่ถ้าไม่ตรงตามโปรแกรมจะทำการตรวจสอบสายไขว้ต่อไปและแสดงผลทางจอ LCD แต่ถ้าไม่ตรงตามโปรแกรมจะทำการตรวจสอบสายสลับต่อไปและแสดงผลทางจอLCD แต่ถ้าเกิด input รับได้ logic ที่ไม่ตรงกับทั้ง3ประเภทโปรแกรมจะแสดงผลว่า error คือเข้าสายผิดนั่นเอง และโปรแกรมจะจบการทำงาน

ถ้าฝั่ง Input ได้รับ Logic 1 เส้นใดเส้นหนึ่งหรือทั้งหมดก็ตามโปรแกรมจะแสดงผลว่ามีสายที่ผ่านมีกี่เส้น เส้นใดบ้าง และสายที่ขาดมีกี่เส้น เส้นไดบ้าง แล้วจบการทำงานทันที โดยโปรแกรมไม่สามารถบอกได้ว่าสายที่ขาดนั่นเป็นสายประเภทไหน

ผลของการวิจัยพบว่า หลังจาก ได้ทดสอบ เครื่องตรวจสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายท้องถิ่น ด้วยสายเคเบิล (UTP) ประเภทต่างๆสามารถตรวจสอบได้จริง รวมไปถึงตรวจสอบสายเคเบิล (UTP) ที่มีความยาวมากกว่า 100เมตร ที่เป็นเหตุนี้เป็นอันเนื่องมาจากการทำงานของเครื่องใช้หลักการของlogic ซึ่งสามารถตรวจสอบได้ระยะทางไกลๆ

5.2 ประโยชน์ที่ได้รับ

1. ได้ศึกษาการใช้งานของ microcontroller
2. ได้ศึกษาการใช้งานโปรแกรม Keil uVision 4
3. ได้ศึกษาการใช้งานโปรแกรม ISIS 7 Professional
4. สามารถเขียนโปรแกรมสั่งการให้ microcontroller ทำการตรวจสอบสายได้
5. โครงงานสามารถนำไปพัฒนาให้มีประสิทธิภาพมากขึ้นได้
6. สามารถแก้ปัญหาจากการปฏิบัติงานจริง เช่น การแก้ปัญหาต่างๆ การทำงานเป็นทีม

5.3 ปัญหาและข้อเสนอแนะ

สำหรับผู้ที่สนใจในการตรวจสอบสายเคเบิล(UTP) รวมไปถึงสายเคเบิลประเภทต่างๆ สามารถนำไปพัฒนาเพิ่มเติมต่อให้มีประสิทธิภาพมากขึ้น หรือนำไปประยุกต์ใช้ร่วมกับวิธีการอื่นต่อไป

สำหรับปัญหาที่พบ คือ

•ปัญหาในการ download hex file ลง Microcontroller ซึ่งเกิดจากการ reset บอร์ด

•ปัญหาจากการใช้โปรแกรม ISIS 7 Professional ซึ่งต้องทำความเข้าใจในการสร้างวงจรขึ้นมาเพื่อใช้ในการตรวจสอบโปรแกรมที่เราได้เขียนไว้

ข้อเสนอแนะ คือ

•ปัญหาในการเขียนโปรแกรม Keil uVision 4 เช่น การกำหนดข้อแม้ในการตรวจสอบว่าสายขาดหรือไม่ขาด รวมถึงตรวจสอบสายแต่ละประเภท

•ทำการพัฒนาให้เครื่องตรวจสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์ท้องถิ่น สามารถทำการวัดระยะความยาวของสายเคเบิล (UTP) ได้ ซึ่งในโครงงานนี้ไม่ครอบคลุมถึงการวัดระยะทางของสาย

•ทำการพัฒนาให้เครื่องตรวจสอบสายเคเบิลที่ใช้ในการเชื่อมต่ออุปกรณ์ท้องถิ่น สามารถตรวจสอบสายเคเบิล (UTP) ว่าเป็นสายประเภทไหนได้ ถึงแม้ว่าสายจะขาด ซึ่งในโครงงานนี้ไม่ครอบคลุมถึงการบอกประเภทของสายได้ ถ้าสายมีการขาดหรือไม่ครบทั้ง8เส้น

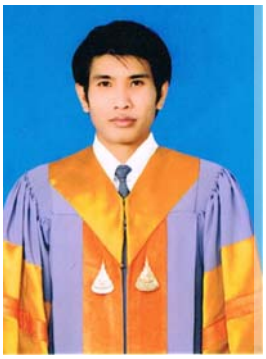
มหาวิทยาลัยเทคโนโลยีสุรนารี

ประวัติผู้เขียน

ชื่อ: นายปรัชญา ชำนาญค้า

เกิด: 18 พฤศจิกายน 2533

การศึกษา: ระดับมัธยมศึกษาตอนปลายจาก
โรงเรียนคำเขื่อนแก้วชนูปถัมภ์
อำเภอคำเขื่อนแก้ว จังหวัดยโสธร เมื่อ พ.ศ. 2551
ปัจจุบันเป็นนักศึกษาชั้นปีที่ 4
สาขาวิชาวิศวกรรมโทรคมนาคม
สำนักวิชาวิศวกรรมศาสตร์
มหาวิทยาลัยเทคโนโลยีสุรนารี



ชื่อ: นาย ธีรพงษ์ ภูประโชติ

เกิด: 23 กุมภาพันธ์ 2533

การศึกษา: ระดับมัธยมศึกษาตอนปลายจาก
โรงเรียนยางตลาดวิทยาคาร
อำเภอยางตลาด จังหวัดกาฬสินธุ์ เมื่อ พ.ศ. 2551
ปัจจุบันเป็นนักศึกษาชั้นปีที่ 4
สาขาวิชาวิศวกรรมโทรคมนาคม
สำนักวิชาวิศวกรรมศาสตร์
มหาวิทยาลัยเทคโนโลยีสุรนารี

ชื่อ: นาย สมภพ เขียววิชัย

เกิด: 13 สิงหาคม 2533

การศึกษา: ระดับมัธยมศึกษาตอนปลายจาก
โรงเรียนราชวินิตบางแคปานขำ
เขตบางแค จังหวัดกรุงเทพมหานคร เมื่อ พ.ศ. 2551
ปัจจุบันเป็นนักศึกษาชั้นปีที่ 4
สาขาวิชาวิศวกรรมโทรคมนาคม
สำนักวิชาวิศวกรรมศาสตร์
มหาวิทยาลัยเทคโนโลยีสุรนารี

มหาวิทยาลัยเทคโนโลยีสุรนารี

บรรณานุกรม

[1] http://www.riverplusblog.com/2011/06/18/serial-communication/

[2] https://www.evo.in.th/knowledgebase/8/-RJ-45-.html

[3] http://www.thaieasyelec.com/electronics-in-chapter/what-is-zigbee.html

[4] http://riverplusblog.com/2011/06/18/serial-communication/

[5] http://www.moosolve.com/?mod=viewaticle&aid=64

[6] http://www.com5dow.com/basic-computer/267-Router-%E0%B8%84%E0%B8%B7%E0%B8%AD%E0%B8%AD%E0%B8%B0%E0%B9%84%E0%B8%A3.html

[7] http://www.elecnet.chandra.ac.th/learn/elearning/microprocessor/8051.html

[8] http://www.thaimicrotron.com/Referrence/LCD/LCD-Module1.htm

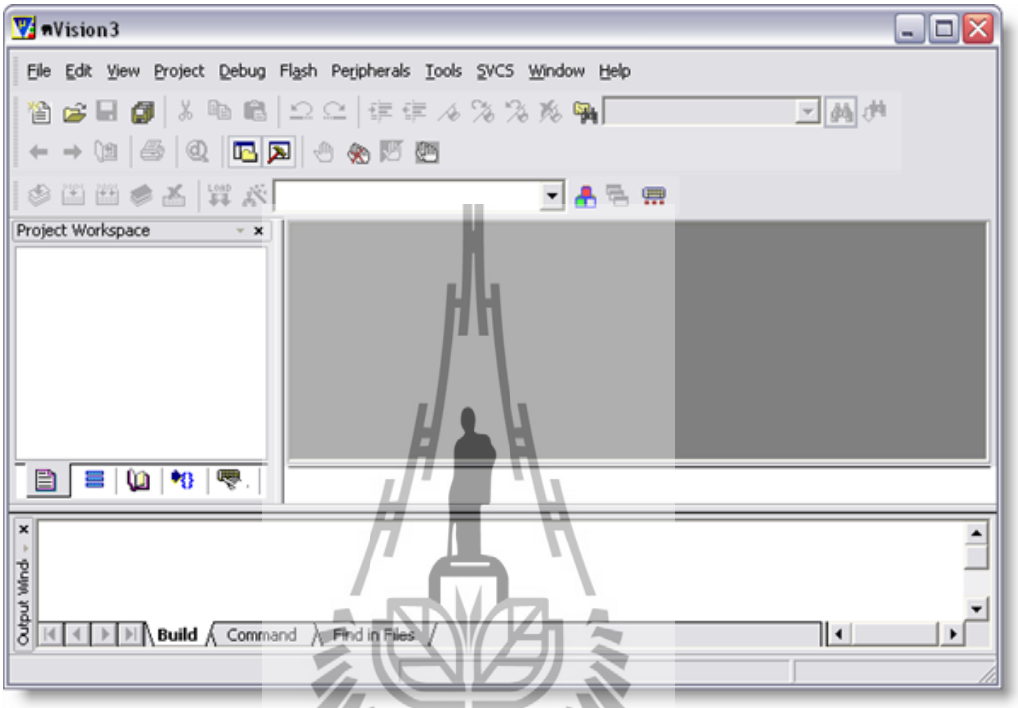
[9] http://www.thaimicrotron.com/Referrence/LCD/LCD-Module2.htm

มหาวิทยาลัยเทคโนโลยีสุรนารี

ภาคผนวกที่1

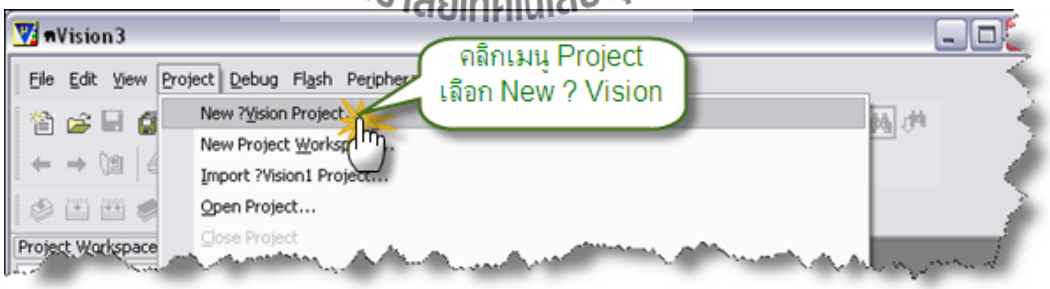
การใช้งานโปรแกรม Keil uVision 4 ในการเขียนภาษาซี

1. เมื่อเปิดโปรแกรมจะได้ดังรูป (ถ้ามีงานเก่าให้ปิดโดยการคลิกที่เมนู Project เลือก Close Project)



รูปที่ 1 แสดงหน้าแรกของโปรแกรม

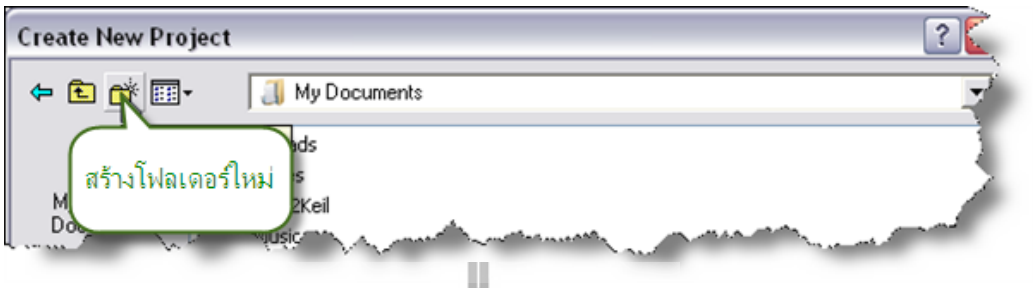
2. สร้าง Project ไฟล์ใหม่ (ทำทุกครั้งถ้าเขียนโปรแกรมตัวใหม่)



รูปที่ 2 แสดงการสร้าง Project

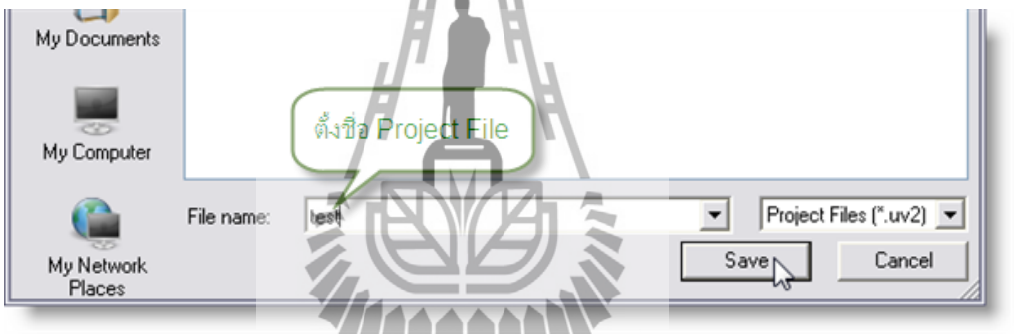
มหาวิทยาลัยเทคโนโลยีสุรนารี

3.สร้างโฟลเดอร์ใหม่เพื่อแยกงานออกจากส่วนอื่นเนื่องจากการเขียนโปรแกรมแต่ละครั้งจะได้ไฟล์หลายไฟล์



รูปที่ 3 แสดงการสร้างโฟลเดอร์

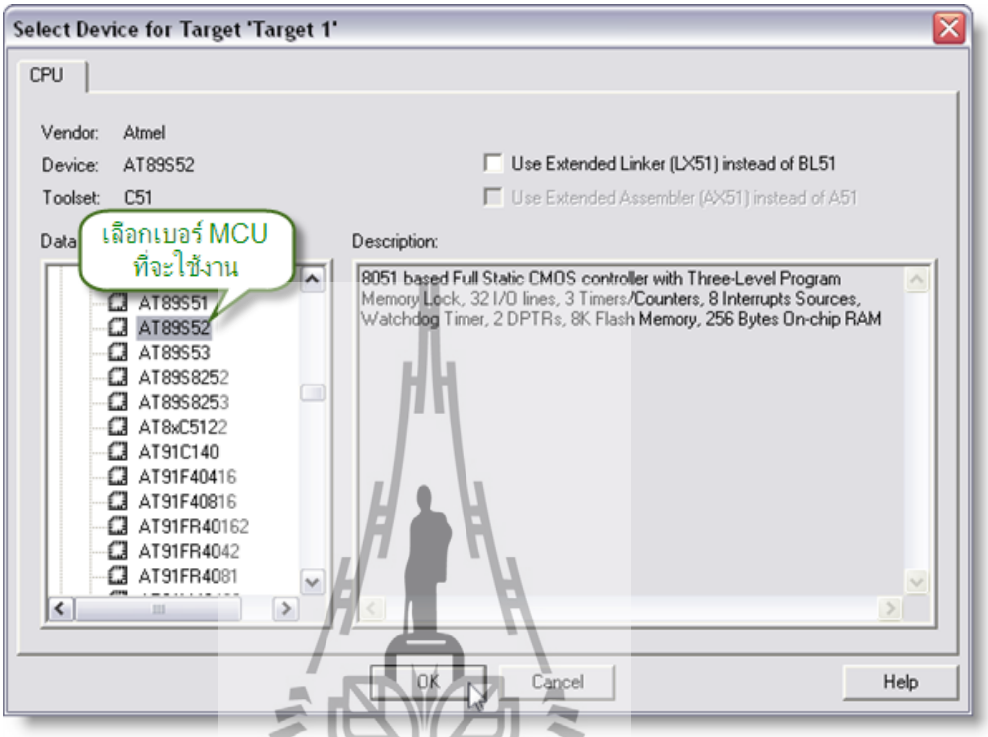
4. ตั้งชื่อ Project File



รูปที่ 4 แสดงการตั้งชื่อ Project File

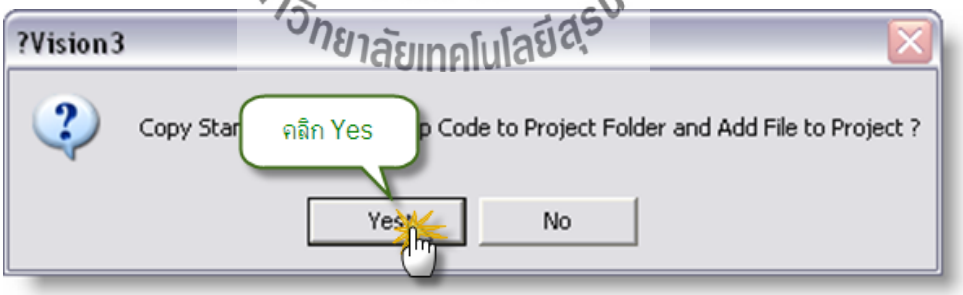
มหาวิทยาลัยเทคโนโลยีสุรนารี

5. เมื่อตั้งชื่อ Project File เสร็จจะปรากฏหน้าต่างเลือกเบอร์ไมโครคอนโทรลเลอร์ที่ใช้งาน (สำหรับการทดลองในที่นี้เลือกใช้เบอร์ AT89S52 ของ ATMEL)



รูปที่ 5 แสดงหน้าต่างเลือกเบอร์ไมโครคอนโทรลเลอร์ที่ใช้งาน

6. คลิก Yes



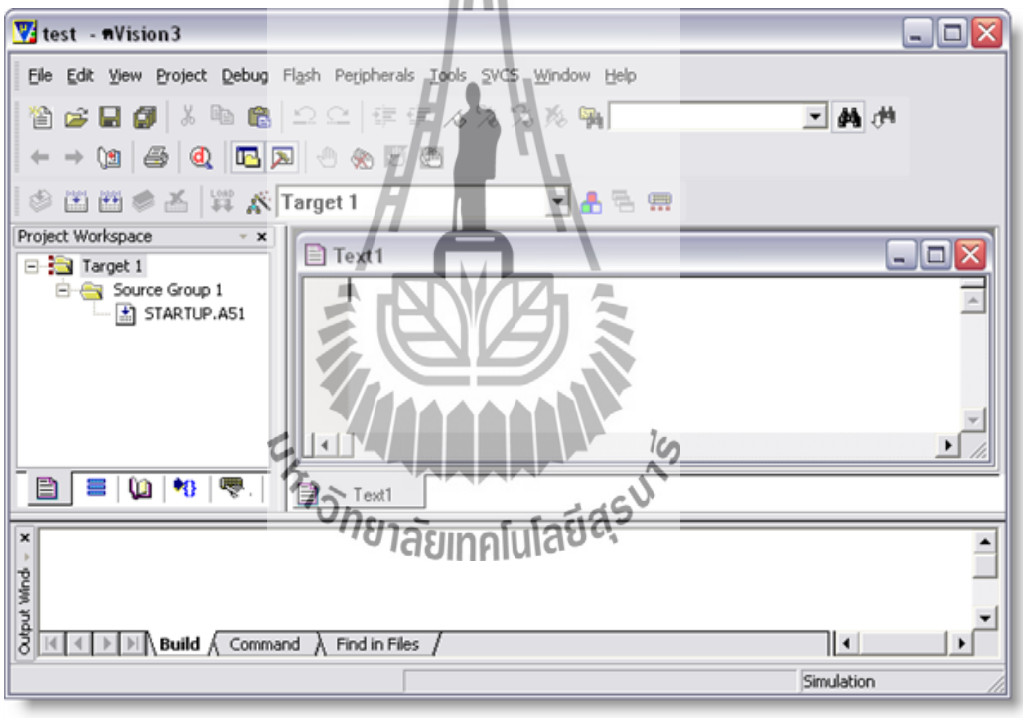
รูปที่ 6 หลังจากเลือกเบอร์ไมโครคอนโทรลเลอร์ที่ใช้งาน

มหาวิทยาลัยเทคโนโลยีสุรนารี

7. สร้างไฟล์เอกสารเพื่อใช้พิมพ์โปรแกรมภาษาซี

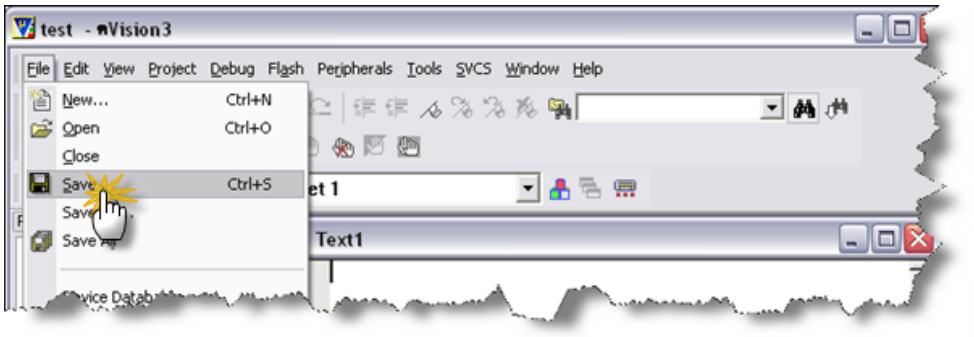
รูปที่ 7 แสดงการสร้างไฟล์เอกสารเพื่อใช้พิมพ์โปรแกรมภาษาซี

8. จะปรากฏไฟล์ชื่อ Text ดังรูป



รูปที่ 8 แสดงหน้าต่างไฟล์ชื่อ Text ดังรูป

9. Save ไฟล์

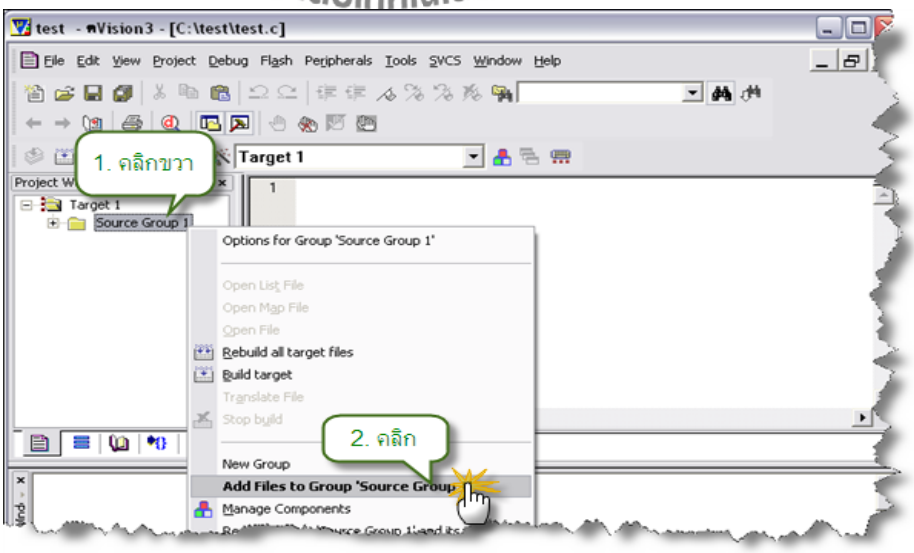


รูปที่ 9 แสดงหน้าต่าง Save ไฟล์

10. ตั้งไฟล์ โดยใช้สกุล .c

รูปที่ 10 แสดงหน้าต่างตั้งไฟล์ โดยใช้สกุล .c

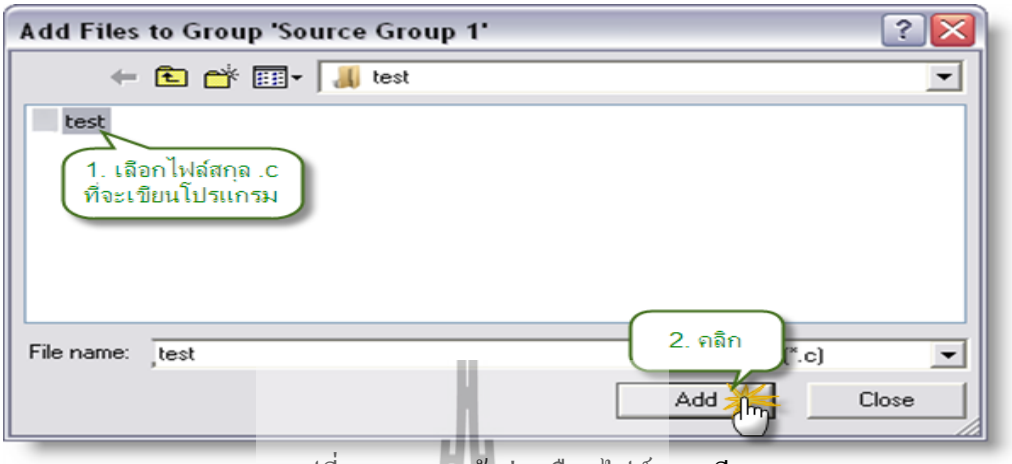
11.เพิ่มไฟล์ลงใน Source Group



รูปที่ 11 แสดงหน้าต่างเพิ่มไฟล์ลงใน Source Group

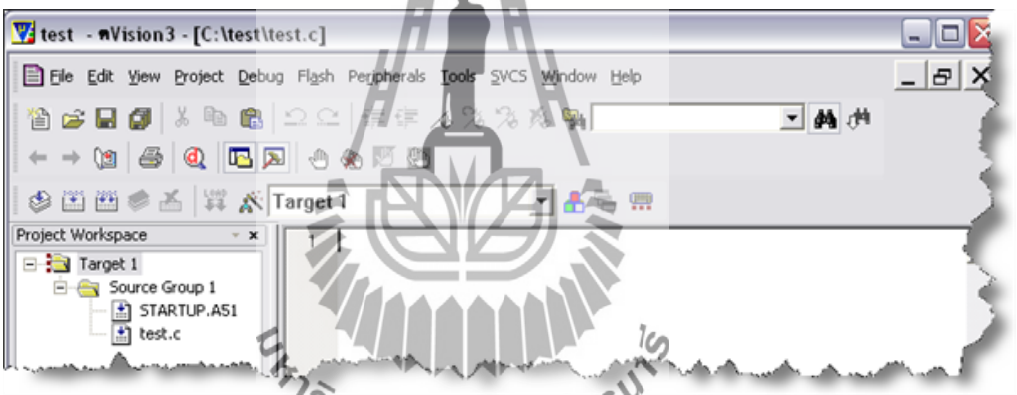
มหาวิทยาลัยเทคโนโลยีสุรนารี

12.เลือกไฟล์ภาษาซี (ที่ทำไว้)



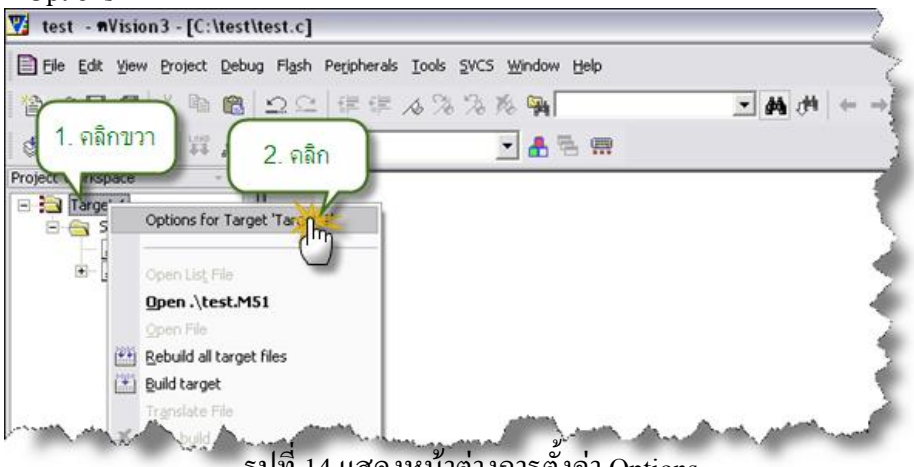
รูปที่ 12 แสดงหน้าต่างเลือกไฟล์ภาษาซี

13.จะได้



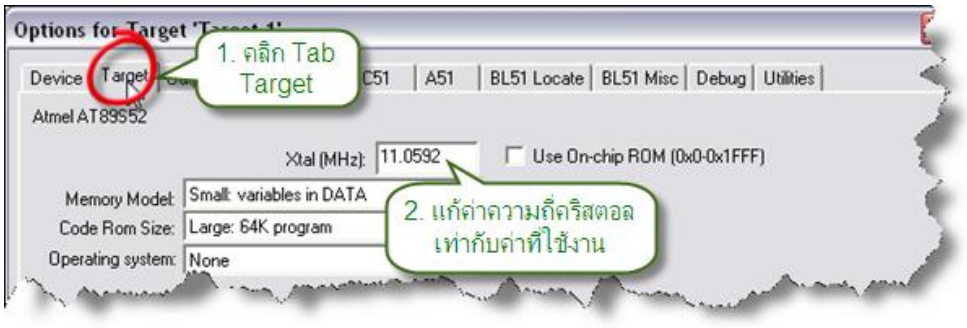
รูปที่ 13 แสดงหน้าต่างหลังจากเลือกไฟล์ภาษาซี

14. ตั้งค่า Options



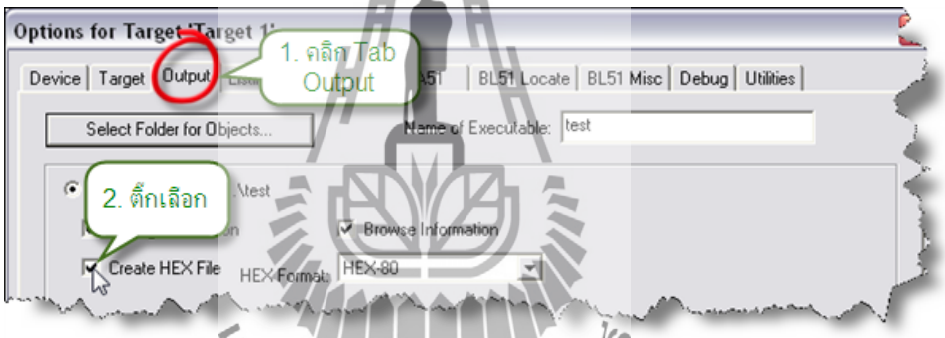
รูปที่ 14 แสดงหน้าต่างการตั้งค่า Options

15. แก้ค่าคริสตอลให้ตรงกับวงจรใช้งานจริง



รูปที่ 15 แสดงหน้าต่างการแก้ค่าคริสตอลให้ตรงกับวงจรใช้งานจริง

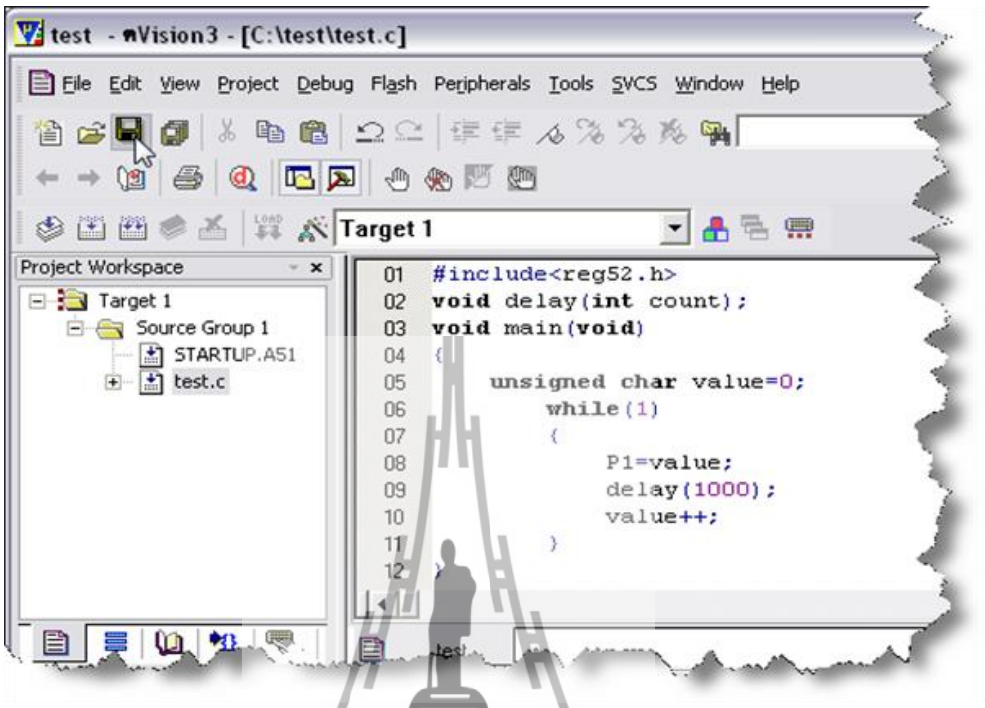
16.เซตเพื่อให้สร้างไฟล์ภาษาเครื่อง (HEX File)



รูปที่ 16 แสดงหน้าต่างการสร้างไฟล์ภาษาเครื่อง (HEX File)

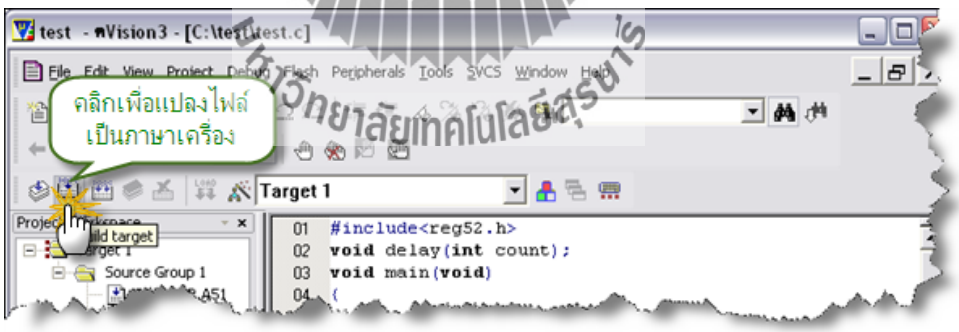
มหาวิทยาลัยเทคโนโลยีสุรนารี

17. พิมพ์ชุดคำสั่งที่เป็นภาษาซีตามที่ต้องการ



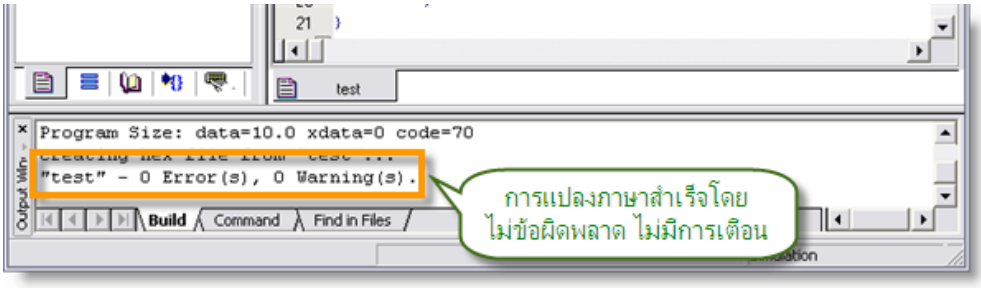
รูปที่ 17 แสดงหน้าต่างการพิมพ์ภาษาซี

18.คลิกปุ่มแปลงให้ได้ไฟล์เป็นภาษาเครื่อง (HEX File)



รูปที่ 18 แสดงหน้าต่างการแปลงเป็นภาษาเครื่อง

19.ผลเมื่อได้เมื่อไม่มีข้อผิดพลาด



รูปที่ 19 แสดงหน้าต่างกรณีไม่เกิดข้อผิดพลาด

20. ในกรณีที่เกิดข้อผิดพลาด

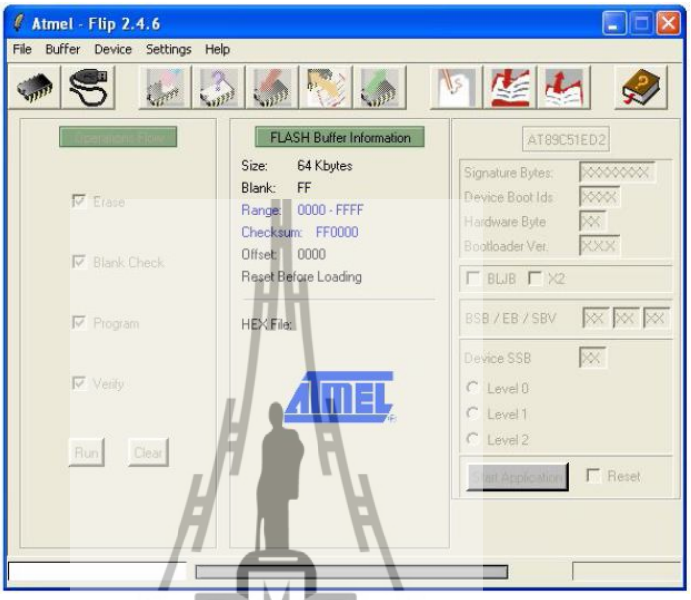
รูปที่ 20 แสดงหน้าต่างกรณีเกิดข้อผิดพลาด

มหาวิทยาลัยเทคโนโลยีสุรนารี

ภาคผนวกที่2

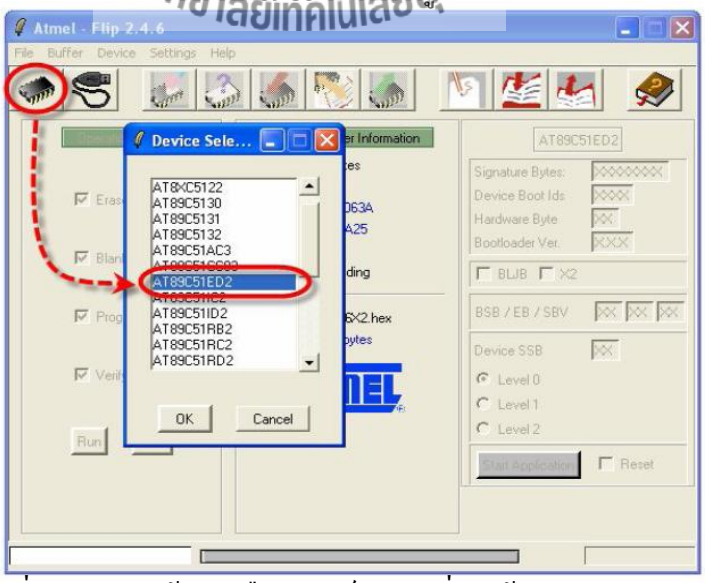
การใช้งานโปรแกรม FLIP 2.4.6 ในการ Download HEX File ให้กับบอร์ด

1. สั่ง Run Program Flip เพื่อเริ่มต้นขั้นตอนของการ Download



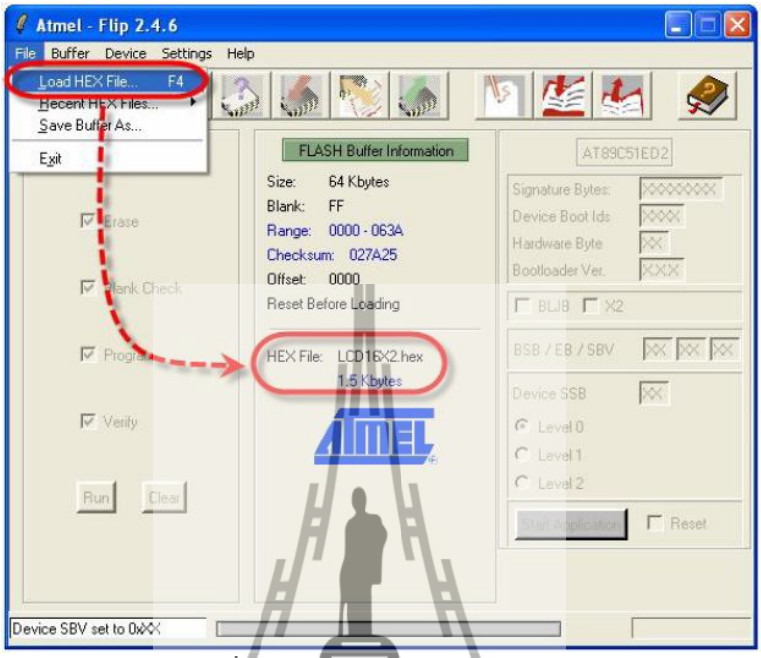
รูปที่ 21 แสดงหน้าต่างโปรแกรม

2. เลือกรายการอุปกรณ์ ซึ่งก็คือ เบอร์ MCU ที่จะใช้ทำการ Download โดยให้เลือกที่เมนูคำสั่ง Device -> Select หรือ คลิกเมาส์ที่ Icon รูป IC แล้วเลือกกำหนดเบอร์ของ MCU ให้ตรงกับที่ติดตั้งไว้จริงในบอร์ด ซึ่งในที่นี้ให้เลือกเป็น AT89C51ED2 ดังรูป



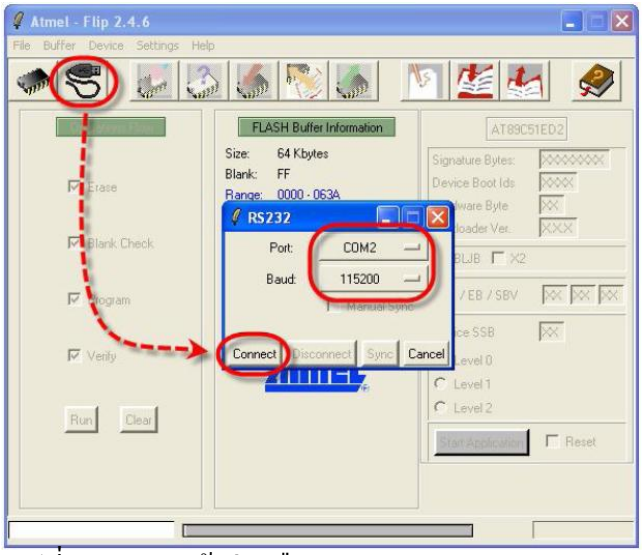
รูปที่ 22 แสดงหน้าต่างเลือก เบอร์ MCU ที่จะใช้ทำการ Download

3. ทำการสั่ง Load HEX File ที่ต้องการจะ Download ให้กับ MCU โดยให้ทำการคลิกเมาส์เลือกที่เมนูคำสั่ง File -> Load HEX File… แล้วเลือก File ที่ ดังตัวอย่าง



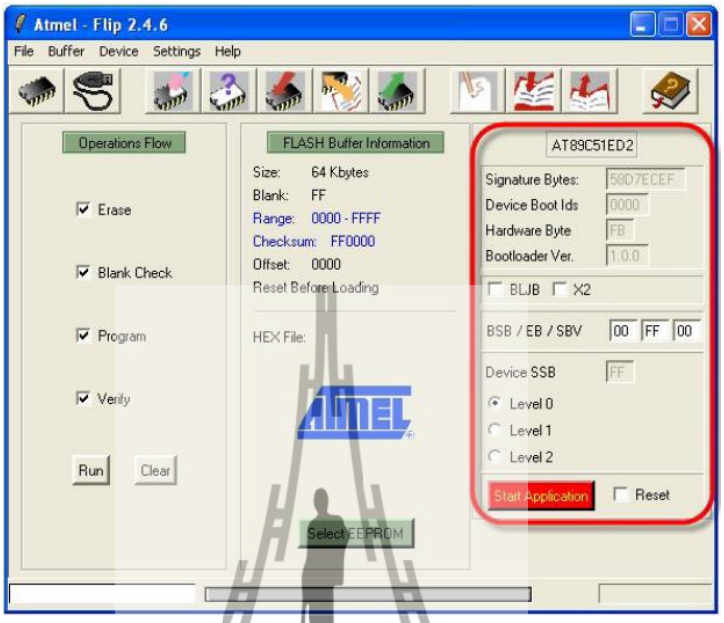
รูปที่ 23 แสดงหน้าต่าง Load HEX File

4.ทำการเลือกการเชื่อมต่อกับบอร์ด โดยให้เลือกที่เมนูคำสั่ง Settings -> Communications… ->RS232 แล้วเลือกกำหนดหมายเลข Comport ให้ตรงกับที่เชื่อมต่อไว้จริง ถ้าใช้การ Downloadแบบ Manual ให้ทำการ Reset MCU ให้ทำงานใน Monitor Mode ให้เรียบร้อยก่อน แต่ถ้าใช้แบบ Auto ให้เลือกใช้สาย ET-DOWNLOAD แทน แล้วเลือก Connect



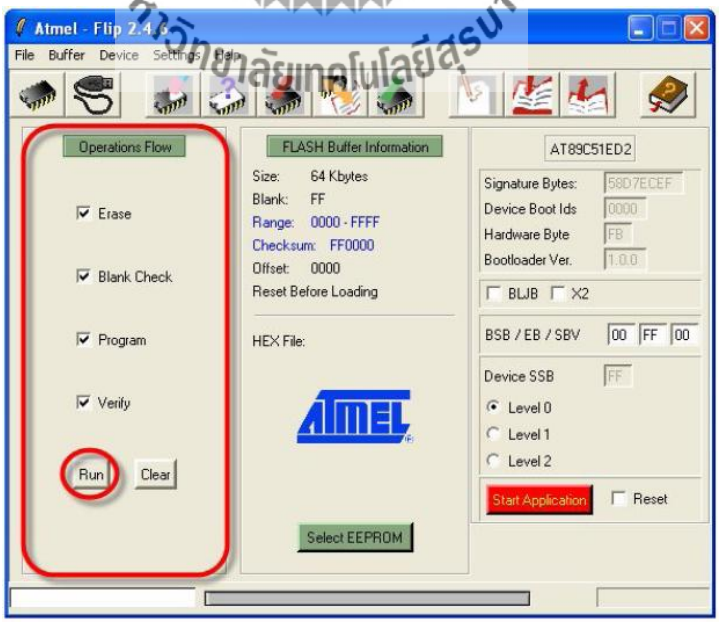
รูปที่ 24 แสดงหน้าต่างเลือกกำหนดหมายเลข Comport

5. หลังจากที่โปรแกรมสามารถเชื่อมต่อกับ MCU ของบอร์ดได้ จะแสดงค่าพารามิเตอร์ต่างๆขอ ง MCU ให้ทราบทางหน้าจอของโปรแกรมด้วยดังตัวอย่าง

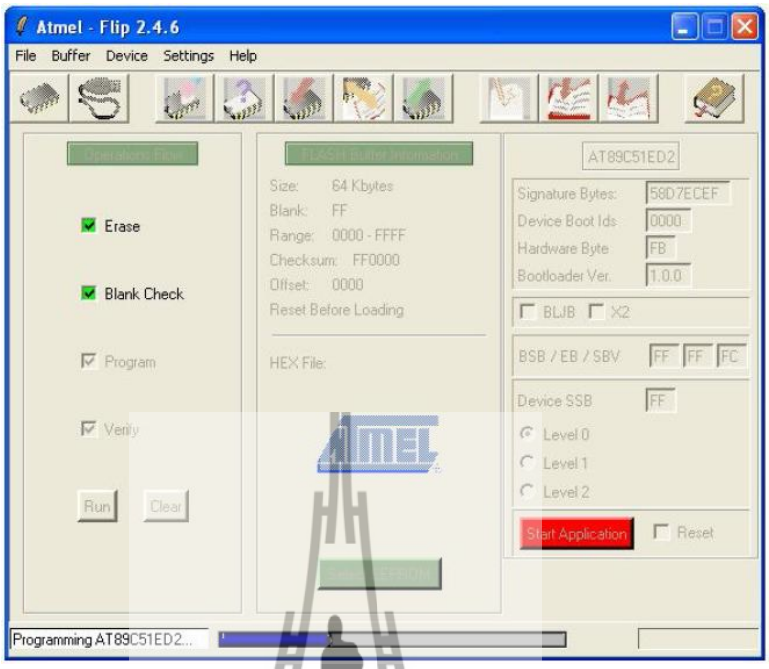


รูปที่ 25 แสดงหน้าต่างหลังจากที่โปรแกรมสามารถเชื่อมต่อกับ MCU ของบอร์ดได้

6. สั่ง Run เพื่อเริ่มต้นทำการ Download โดยให้ส่วนของ Operations Flow ให้เลือกการทำงานของคำสั่งต่างให้ครบทั้งหมด คือ Erase, Blank Check, Program และ Verify แล้วเลือก Run จากนั้นให้รอจนโปรแกรมทำงานทุกขั้นตอนจนเสร็จ

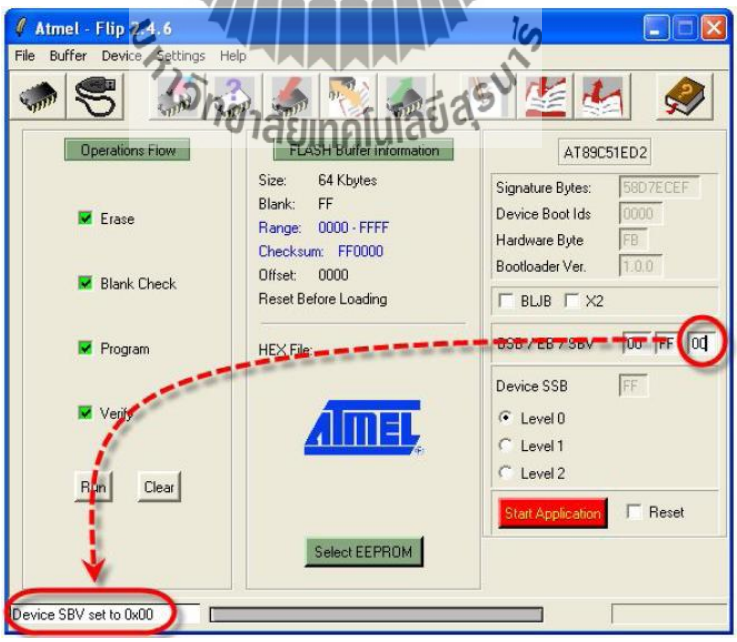


รูปที่ 26 แสดงหน้าต่างการสั่ง Run เพื่อเริ่มต้นทำการ Download



รูปที่ 27 แสดงหน้าต่างการสั่ง Run เพื่อเริ่มต้นทำการ Download

7. เมื่อการทำงานของโปรแกรมเสร็จเรียบร้อยแล้ว ให้ทำการเปลี่ยนค่า Device จาก FC เป็น 00 โดยให้คลิกเมาส์ในช่องแสดงค่า แล้วป้อนค่าใหม่ คือ 00 เข้าไปแทนที่ดังตัวอย่าง จากนั้นก็ให้เลือกที่ Start Application หรือ กดสวิตช์ Reset เพื่อให้บอร์ดเริ่มทำงานได้ทันที

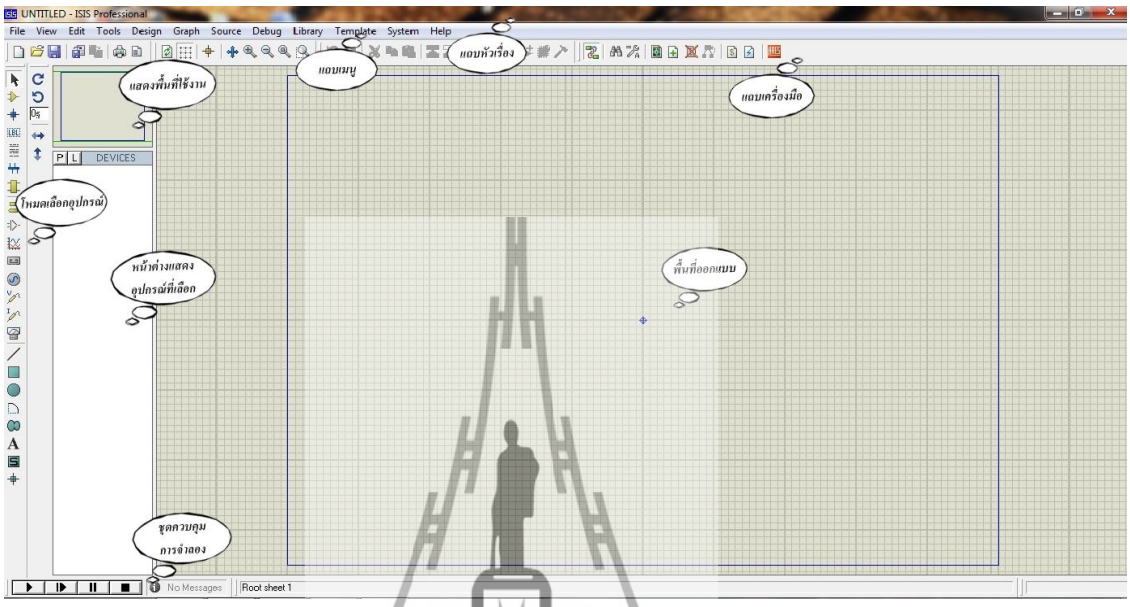


รูปที่ 28 แสดงหน้าต่างหลังจาก Download เสร็จ

ภาคผนวกที่ 3

การใช้งานโปรแกรม ISIS 7 Professional ในการตรวจสอบสาย UTP

3.1 ส่วนประกอบของโปรแกรม ISIS



รูปที่ 29 แสดงหน้าต่างหลักของโปรแกรม

- แถบหัวเรื่อง (Title) คือ ส่วนที่บ่งบอกถึงชื่อ โปรแกรมที่ใช้อยู่ ณ ปัจจุบัน
- แถบเครื่องมือ (Toolbar) แสดงเป็นปุ่มเครื่องมือที่ใช้งานบ่อย ๆ โดยด้านบนจะเป็นเครื่องมือเกี่ยวกับการจัดการไฟล์และมุมมองของโปรแกรม ส่วนด้านซ้ายเป็นเครื่องมือการสร้างวงจรบนพื้นที่ทำงาน
- แถบเมนู (Menus) เป็นส่วนที่รวมคำสั่งที่ใช้ทั้งหมดของโปรแกรมไว้ในเมนูต่าง ๆ
- หน้าต่างแสดงอุปกรณ์ที่เลือก/แสดงพื้นที่ใช้งาน (Overview window) ใช้แสดงรูปร่างอุปกรณ์ที่เลือกใช้งาน และใช้เลื่อนพื้นที่ทำงานได้อีกด้วย
- โหมดเลือกอุปกรณ์ (Component Mode) ใช้เลือกอุปกรณ์ชนิดต่าง ๆ ได้ภายในโหมดนี้
- หน้าต่างแสดงรายการอุปกรณ์ในโหมดที่เลือก (Object Selector) แสดงชื่อรายการอุปกรณ์ที่เลือกไว้ในโหมดเลือกอุปกรณ์ (Component Mode) เพื่อจะนำมาสร้างวงจรในพื้นที่ทำงาน
- พื้นที่ทำงาน (Editing Window) เป็นพื้นที่ออกแบบวงจรไฟฟ้า
- ชุดควบคุมการจำลอง (Simulation Controls) เป็นแถบของปุ่มควบคุม การจำลองการทำงานของวงจร

มหาวิทยาลัยเทคโนโลยีสุรนารี

3.2 เครื่องมือในการใช้งาน

เมื่อเปิดไฟล์ Schematic ขึ้นมาแล้ว จะเห็นแถบเครื่องมืออยู่มากมาย ซึ่งเรามาดูกันดีกว่าว่าแต่ละคำสั่งมีหน้าที่ทำอะไรได้บ้าง

File And Printing Commands

รูปที่ 30 เครื่องมือ File And Printing Commands

New file คือ ปุ่มที่ใช้สำหรับสร้างไฟล์ Schematic ใหม่

Open Design ใช้สำหรับเปิดไฟล์ Schematic ที่สร้างไว้แล้ว

Save Design ใช้สำหรับบันทึกไฟล์ Schematic ที่กำลังออกแบบอยู่ ณ ปัจจุบัน

Import Section ใช้เปิดไฟล์เอกสารที่มีนามสกุล .SEC

Export Section ใช้บันทึกไฟล์เอกสารให้เป็นนามสกุล .SEC

Print Design เป็นคำสั่งพิมพ์งานออกทางเครื่องปริ๊นเตอร์

Display commands

รูปที่ 31 เครื่องมือDisplay commands

Toggle Grid ใช้สำหรับเปิด-ปิด กริดบนพื้นที่ทำงาน

Center At Cursor เป็นคำสั่งให้เมาส์ชี้อยู่ในตำแหน่งตรงกลางหน้าจอ

Zoom In ใช้ขยายพื้นที่ทำงานให้มีขนาดใหญ่ขึ้นทุกครั้งที่คลิกเลือกคำสั่งนี้

Zoom Out ใช้ย่อพื้นที่ทำงานให้มีขนาดเล็กขึ้นทุกครั้งที่คลิกเลือกคำสั่งนี้

Zoom To View Entire Sheet เมื่อใช้คำสั่งนี้พื้นที่ทำงานจะอยู่เต็มกรอบสีน้ำเงินทุกครั้ง

Zoom To Area คำสั่งนี้เมื่อลากเมาส์คลุมส่วนที่ต้องการ จะทำให้ส่วนนั้นขยายเต็มหน้าจอ

มหาวิทยาลัยเทคโนโลยีสุรนารี

Design Tools

รูปที่ 32 เครื่องมือ Design Tools

Undo Changes เป็นคำสั่งย้อนกลับไปยังการกระทำที่ผ่านมา

Redo Changes เป็นคำสั่งให้กลับไปยังการกระทำเดิม เมื่อใช้คำสั่ง Undo ไปแล้ว

Cut To Clipboard ใช้ลบอุปกรณ์ เมื่อตัวอุปกรณ์กลายเป็นสีแดงจะมีคำสั่งนี้ขึ้นมา

Coppy To Clipboard ใช้คัดลอกอุปกรณ์ โดยคลิกหรือลากเมาส์คลุมอุปกรณ์ที่ต้องการให้กลายเป็นสีแดง โดยคำสั่งนี้จะต้องใช้ควบคู่กับคำสั่ง Paste From Clipboard

Paste From Clipboard เป็นคำสั่งวางอุปกรณ์ที่ถูกคัดลอกด้วยคำสั่ง Copy To Clipboard

Block Copy เป็นคำสั่งวางอุปกรณ์ เมื่อคลิกหรือลากเมาส์คลุมอุปกรณ์ ที่ต้องการให้กลายเป็นสีแดง ก็จะมีคำสั่งนี้ขึ้นมา

Block Move เป็นคำสั่งเคลื่อนย้ายอุปกรณ์ ที่คลิกหรือลากเมาส์คลุมอุปกรณ์ ที่ต้องการให้กลายเป็นสีแดง ก็จะมีคำสั่งนี้ขึ้นมา

Block Rotate เป็นคำสั่งหมุนอุปกรณ์ ที่คลิกหรือลากเมาส์คลุมอุปกรณ์ ที่ต้องการให้กลายเป็นสีแดง ก็จะมีคำสั่งนี้ขึ้นมา โดยใช้ค่ามุมที่ต้องการหมุนได้ตามต้องการ

Block Delete เป็นคำสั่งหมุนอุปกรณ์ ที่คลิกหรือลากเมาส์คลุมอุปกรณ์ ที่ต้องการให้กลายเป็นสีแดง ก็จะมีคำสั่งนี้ขึ้นมา โดยใช้ค่ามุมที่ต้องการหมุนได้ตามต้องการ

Pick parts from libraries ให้เรียกหน้า Pick Device ขึ้นมา ซึ่งเป็นหน้าที่ใช้เลือกไลบรารีและอุปกรณ์ต่าง ๆ

Make Device เป็นคำสั่งเรียกหน้าต่าง Make Device ของอุปกรณ์ ที่คลิกให้กลายเป็นสีแดงขึ้นมา

Packaging Tool เป็นคำสั่งเรียกหน้าต่าง Package Device ของอุปกรณ์ที่คลิกให้กลายเป็นสีแดงขึ้นมา

Decompose เป็นคำสั่งเรียกคุณสมบัติของอุปกรณ์ ที่คลิกให้กลายเป็นสีแดงขึ้นมา

Design Tools

รูปที่ 33 เครื่องมือ Design Tools

Wire Auto router ใช้เดินสายสัญญาณแบบกึ่งอัตโนมัติ

Design Explorer คำสั่งเรียกดูค่าต่าง ๆ ของอุปกรณ์ภายในพื้นที่ทำงาน

New Sheet ใช้สร้างไฟล์ Schematic ใหม่

Bill of Material ใช้แสดงเอกสารเกี่ยวกับอุปกรณ์ที่ใช้ภายในพื้นที่ทำงาน

Electrical Rules Check เป็นคำสั่งตรวจเช็คทางไฟฟ้า

Net list to Ares เป็นคำสั่งให้วงจรไฟฟ้า นำเข้าไปยังโปรแกรม Ares เพื่อออกแบบลายวงจรพิมพ์

Main Mode Icons

รูปที่ 34 เครื่องมือMain Mode Icons

Selection Mode เป็นคำสั่งเลือกไปใช้เมาส์

Component Mode ใช้เลือกอุปกรณ์อิเล็กทรอนิกส์ต่าง ๆ

Junction Dot Mode เป็นจุดเชื่อมต่อสายสัญญาณสองเส้นให้ต่อถึงกัน

Wire Label Mode ใช้กำหนดชื่อให้กับสายสัญญาณภายในวงจร

Script Mode ใช้เขียนข้อความลงตามต้องการ โดยต้องคลิกที่พื้นที่ว่าง แล้วหน้าต่าง EditScript จะปรากฏขึ้นมา

Buses Mode เป็นคำสั่งเดินสายสัญญาณบัส

Sub-Cercuit Mode ใช้เลือกอุปกรณ์ที่เป็นจุดต่อร่วมภายในวงจร

มหาวิทยาลัยเทคโนโลยีสุรนารี

Gadget Icons

รูปที่ 35 เครื่องมือ Gadget Icons

Terminal Mode เป็นจุดต่อขาอุปกรณ์จากจุดหนึ่งไปยังอีกจุดหนึ่ง โดยไม่ต้องเดินสายสัญญาณเชื่อมให้ถึงกัน

Device Pin Mode เป็นคำสั่งเลือกขาอุปกรณ์

Graph Mode เป็นคำสั่งเลือกกราฟชนิดต่าง ๆ

Generator Mode เป็นคำสั่งเลือกแหล่งจ่ายไฟชนิดต่าง ๆ

Voltage Probe Mode เป็นจุดวัดแรงดันไฟฟ้าภายในวงจร

Current Probe Mode เป็นจุดวัดกระแสไฟฟ้าภายในวงจร

Multi Meter Mode เป็นคำสั่งเลือกเครื่องมือวัดชนิดต่าง ๆ

2D Graphics

รูปที่ 36 เครื่องมือ 2D Graphics

Line Mode ใช้สร้างเส้นตรง หรือใช้เป็นสัญญาณก็ได้

Box Mode ใช้สร้างกรอบสี่เหลี่ยม

Circle Mode ใช้สร้างวงกลม

Arc Mode ใช้สร้างเส้นโค้ง

2D Path Mode ใช้สร้างเส้นตรงแบบต่อเนื่อง แต่ถ้ากดปุ่มคีย์ ค้างไว้ ก็จะเปลี่ยนเป็นเส้นโค้ง

Text Mode ใช้ผิดข้อความต่าง ๆ ลงบนพื้นที่ทำงาน

Symbol Mode เป็นคำสั่งเลือกสัญลักษณ์ Rotor ชนิดต่าง ๆ

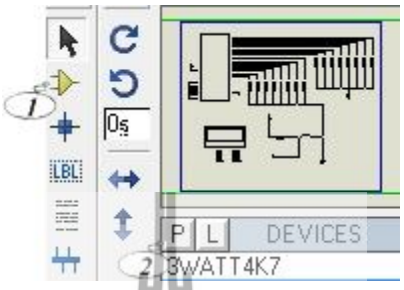
Marker Mode เป็นคำสั่งเลือกจุดชนิดต่าง ๆ

มหาวิทยาลัยเทคโนโลยีสุรนารี

3.3 การเลือก – หาอุปกรณ์อิเล็กทรอนิกส์

1. คลิกที่ปุ่ม เพื่อค้นหาอุปกรณ์

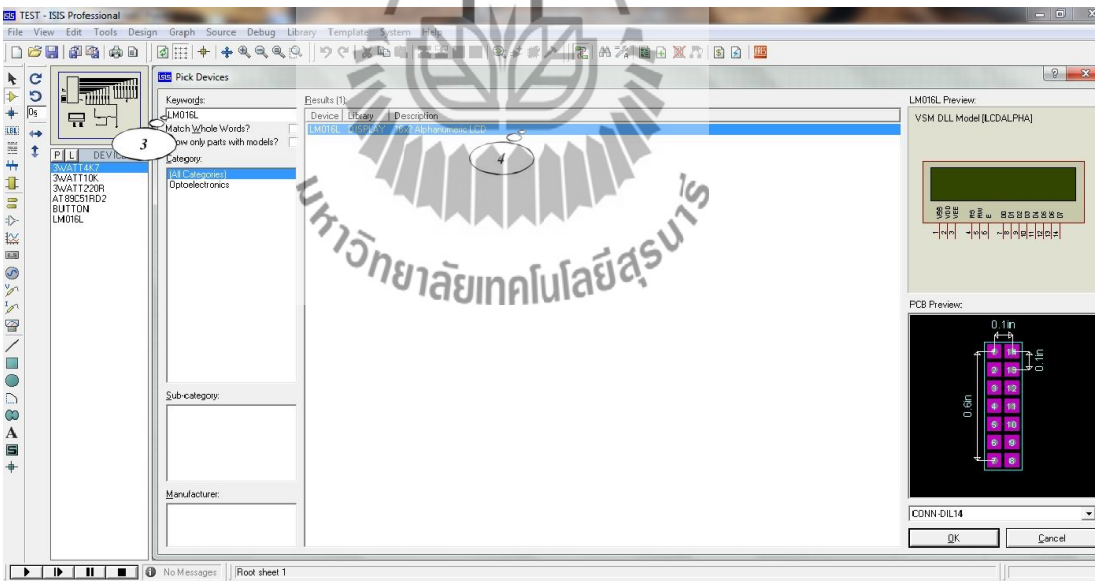
2. คลิกที่ปุ่ม เพื่อเลือกอุปกรณ์จากไลบรารี ก็จะปรากฏหน้าต่าง Pick Device ขึ้นมา



รูปที่ 37 แสดงการหาอุปกรณ์

3. คลิ๊กเลือกไลบรารีตามต้องการ ในที่นี้เลือกคลิกที่ไลบรารี Transistor

4. ดับเบิ้ลคลิกที่ชื่ออุปกรณ์ที่ต้องการเลือก ในที่นี้เลือก 2N3904 แล้วก็ปิดหน้าต่าง Pick Device

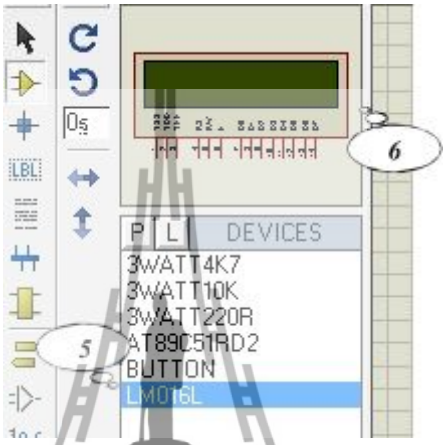


รูปที่ 38 หน้าต่างในการหาอุปกรณ์

มหาวิทยาลัยเทคโนโลยีสุรนารี

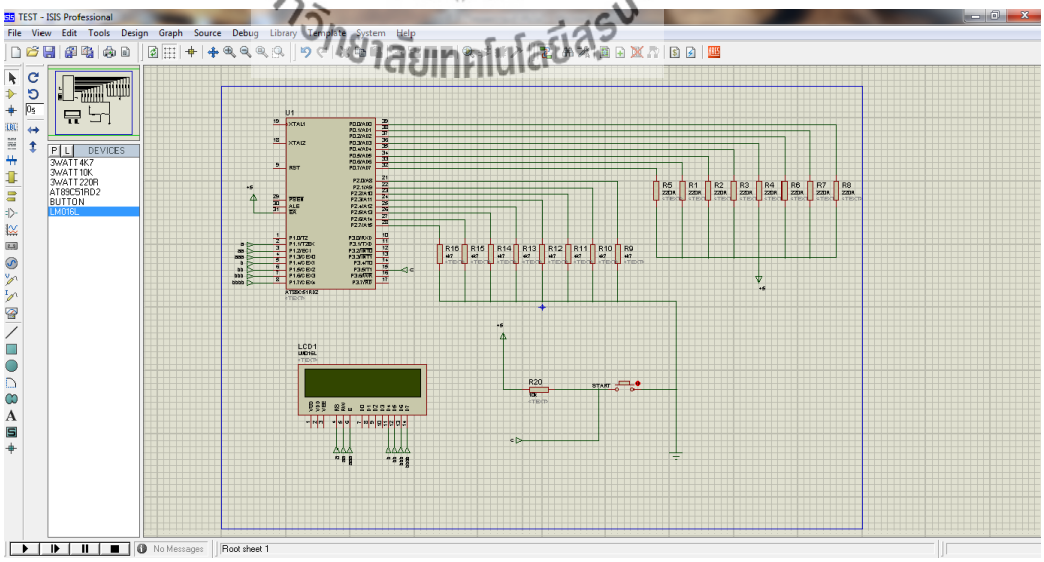
5. จะสังเกตเห็นว่า รายการอุปกรณ์ได้เข้ามายังช่องอุปกรณ์เรียบร้อยแล้ว ซึ่งในขั้นตอนที่ 4 เราสามารถเลือกอุปกรณ์ ที่ต้องการใช้ได้กี่ชนิดก็ได้ จะทำให้รายการอุปกรณ์ที่เราเลือกเข้ามาอยู่ในช่องนี้เสมอ

6. รูปภาพอุปกรณ์ที่เราเลือกใช้



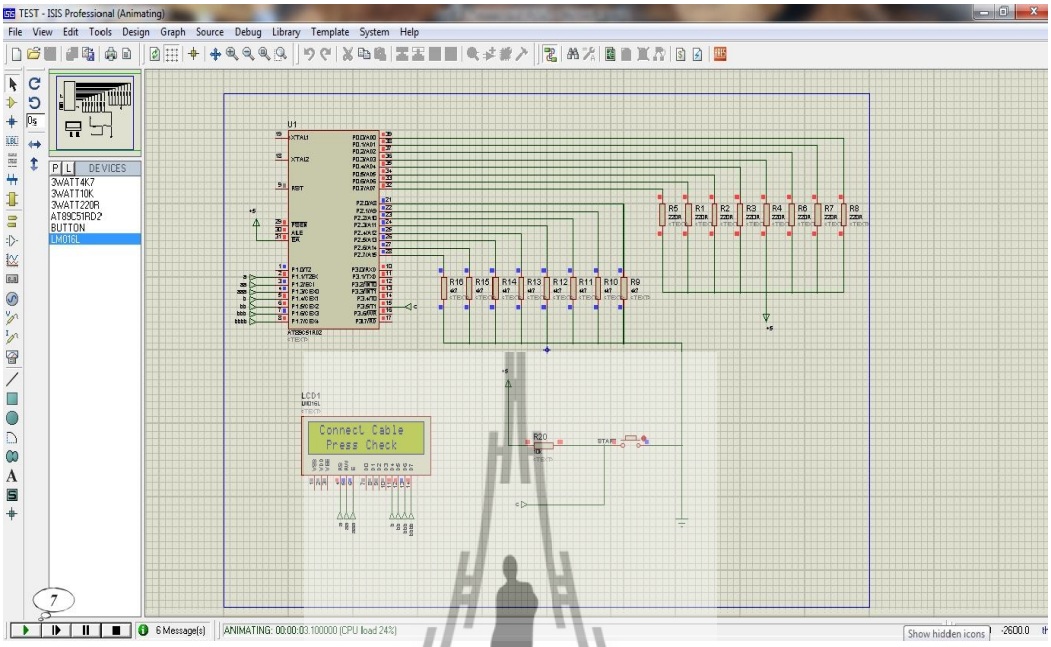
รูปที่ 39 แสดงหน้าตาอุปกรณ์

รูปภาพแสดงการต่อวงจรที่เสร็จสิ้นแล้ว



รูปที่ 40 แสดงวงจรที่ต่อเสร็จแล้ว

7. คลิก Play เพื่อให้วงจรที่สร้างขึ้นมาทำงาน



รูปที่ 41 แสดงการทำงานของวงจร

มหาวิทยาลัยเทคโนโลยีสุรนารี

ภาคผนวกที่4

คำสั่งโปรแกรมที่สั่งการให้ Microcontroller ทำการตรวจสอบสาย UTP

```
#include <at89c51xd2.h>
#include<stdio.h>
#include<string.h>
////////////// LCD PIN /////////////////
#define LCDPORT P1
#define RS P1_1;
#define RW P1_2;
#define E P1_3;
#define IN P0
#define OUT P2
////////////// SET BIT TO PRESS BUTTOM ... ////////////
sbit S1 = P3^2;
////////////// SET BIT TO CONNECT CABLE...(INPUT) P2 ////////////////
sbit I0 = P0^0;
sbit I1 = P0^1;
sbit I2 = P0^2;
sbit I3 = P0^3;
sbit I4 = P0^4;
sbit I5 = P0^5;
sbit I6 = P0^6;
sbit I7 = P0^7;
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
////////////// SET BIT TO CONNECT CABLE...(OUTPUT) P3
sbit O0 = P2^0;
sbit O1 = P2^1;
sbit O2 = P2^2;
sbit O3 = P2^3;
sbit O4 = P2^4;
sbit O5 = P2^5;
sbit O6 = P2^6;
sbit O7 = P2^7;
////////////// Define Varible //////////////
char lcd_buf[30];
int h1;
int g1;
unsigned int d[10],x =1,z=1;
unsigned int e[10],j=1;
int h,h2 = 0;
bit status=0;
int t = 1;

#define lcd_delay 10

void show(void);
int chk_cable();
void error();
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
////////////////// Delay ////////////
void delay(unsigned int j)
{
unsigned int i,k;
for(i=0;i<j;i++)
{
for(k=0;k<50;k++);
}
}
///////////////// SEND COMMAND TO LCD ////////
void _lcd_init_write(unsigned char a)
{
P1_1 = 0;
P1_2 = 0;
LCDPORT = a;
P1_3=1;
delay(lcd_delay);
P1_3=0;
}
////////////////// Recieve COMMAND ////////////
void lcd_com(unsigned char a){
unsigned char temp;
if(status){
status=0;
goto ___nextstate;
}
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
P1_1=0;
___nextstate:
P1_2=0;
temp=a;
temp&=0xF0;
LCDPORT&=0x0F;
LCDPORT|=temp;
P1_3=1;
delay(lcd_delay);
P1_3=0;
temp=a<<4;
temp&=0xF0;
LCDPORT&=0x0F;
LCDPORT|=temp;
P1_3=1;
delay(lcd_delay);
P1_3=0;
}
///////////////// Recieve String ////////////////////
void lcd_data(unsigned char a){
status=1;
P1_1=1;
lcd_com(a);
}
```

```
////////////////// INITIAL LCD ////////////
void lcd_init(void){
delay(lcd_delay);
_lcd_init_write(0x30);
delay(lcd_delay);
_lcd_init_write(0x30);
delay(lcd_delay);
_lcd_init_write(0x30);
delay(lcd_delay);
_lcd_init_write(0x20);
delay(lcd_delay);
lcd_com(0x28);
delay(lcd_delay);
lcd_com(4);
delay(lcd_delay);
lcd_com(0x85);
delay(lcd_delay);
lcd_com(6);
delay(lcd_delay);
lcd_com(1);
delay(lcd_delay);
}
////////////// INTPUT STRING AND SEND /////////////////
void lcd_puts(char *aaa){
unsigned int i=0;
for(;aaa[i]!=0;i++)lcd_data(aaa[i]);
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
}
void main(){
P0 = 0xff;  // input pin
P2 = 0x00;
delay(1000);
lcd_init();
lcd_com(0X0C);
delay(1000);
lcd_com(0X80); // Line 1
lcd_puts(" Check type LAN ");
lcd_com(0XC0);// line 2
lcd_puts("   And Fixed ");
delay(9000);
show();
while(1){
 if(S1==0){
   h2 = chk_cable();
            if(h2==9){
                     h2 = 0;
                     ////////////////// Direct /////////////////
                     lcd_com(0x01);
                     lcd_com(0x80);
                     lcd_puts(" CHECKING TYPE  ");
                     lcd_com(0xC0);
                     lcd_puts("     CABLE ?    ");
                     delay(2000);
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        ////////////////////////////////////
while(t){
        OUT = 0x01;
        delay(2000);
                if(I0==1){
                OUT = 0x02;
                delay(2000);
                        if(I1==1){
                        OUT = 0x04;
                        delay(2000);
                                if(I2==1){
                                OUT = 0x08;
                                delay(2000);
                                        if(I3==1){
                                        OUT = 0x10;
                                        delay(2000);
                                                if(I4==1){
                                                OUT = 0x20;
                                                delay(2000);
                                                        if(I5==1){
                                                        OUT = 0x40;
                                                        delay(2000);
                                                                if(I6==1){
                                                                OUT = 0x80;
                                                                delay(2000);
                                                                        if(I7==1){
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
                                                    lcd_com(0x01);
                                                    lcd_com(0x80);
        lcd_puts("     PASS     ");
        lcd_com(0xC0);
        lcd_puts("  DIRECT CABLE  ");
        delay(90000);
        t = 0;
        break; }
}
        else{
        lcd_com(0xC0);
        lcd_puts(" CABLE ERROR !");
        t=0;
        delay(7000);
        break; }
}
                else{
                lcd_com(0xC0);
                lcd_puts(" CABLE ERROR !");
                t=0;
                delay(7000);
                break; }
}
                        else{
                        lcd_com(0xC0);
                        lcd_puts(" CABLE ERROR !");
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
                t=0;
                delay(7000);
                break; }
}
                else{
                lcd_com(0xC0);
                lcd_puts(" CABLE ERROR !");
                t=0;
                delay(7000);
                break; }
}
                else{
                lcd_com(0xC0);
                lcd_puts(" CABLE ERROR !");
                t=0;
                delay(7000);
                break; }
}
                else{
                lcd_com(0xC0);
                lcd_puts(" CABLE ERROR !");
                t=0;
                delay(7000);
                break ; }
}
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
else if(I2==1){
        OUT = 0x02;
        delay(2000);
                if(I5==1){
                OUT = 0x04;
                delay(2000);
                        if(I0==1){
                        OUT = 0x08;
                        delay(2000);
                                if(I3==1){
                                OUT = 0x10;
                                delay(2000);
                                        if(I4==1){
                                        OUT = 0x20;
                                        delay(2000);
                                                if(I1==1){
                                                OUT = 0x40;
                                                delay(2000);
                                                        if(I6==1){
                                                        OUT = 0x80;
                                                        delay(2000);
                                                                if(I7==1){
                                                                lcd_com(0x01);
                                                                lcd_com(0x80);
        lcd_puts("      PASS      ");
        lcd_com(0xC0);
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        lcd_puts("   CROSS CABLE  ");
        delay(9000);;
        t = 0;
        break; }
}
        else{
        lcd_com(0xC0);
        lcd_puts(" CABLE ERROR !");
        t=0;
        delay(7000);
        break; }
}
                else{
                lcd_com(0xC0);
                lcd_puts(" CABLE ERROR !");
                t=0;
                delay(7000);
                break; }
}
                        else{
                        lcd_com(0xC0);
                        lcd_puts(" CABLE ERROR !");
                        t=0;
                        delay(7000);
                        break; }
}
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
                                else{
                                lcd_com(0xC0);
                                lcd_puts(" CABLE ERROR !");
                                t=0;
                                delay(7000);
                                break; }
}
                                else{
                                lcd_com(0xC0);
                                lcd_puts(" CABLE ERROR !");
                                t=0;
                                delay(7000);
                                break; }
}
                                else{
                                lcd_com(0xC0);
                                lcd_puts(" CABLE ERROR !");
                                t=0;
                                delay(7000);
                                break ;}
}
else if(I7==1){
        OUT = 0x02;
        delay(2000);
                if(I6==1){
                OUT = 0x04;
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
                delay(2000);
                        if(I5==1){
                        OUT = 0x08;
                        delay(2000);
                                if(I4==1){
                                OUT = 0x10;
                                delay(2000);
                                        if(I3==1){
                                        OUT = 0x20;
                                        delay(2000);
                                                if(I2==1){
                                                OUT = 0x40;
                                                delay(2000);
                                                        if(I1==1){
                                                        OUT = 0x80;
                                                        delay(2000);
                                                                if(I0==1){
                                                                lcd_com(0x01);
                                                                lcd_com(0x80);
        lcd_puts("      PASS      ");
        lcd_com(0xC0);
        lcd_puts(" ROLLOWER CABLE ");
        delay(9000);
        t = 0;
        break; }
}
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        else{
        lcd_com(0xC0);
        lcd_puts(" CABLE ERROR !");
        t=0;
        delay(7000);
        break;}
}
                else{
                lcd_com(0xC0);
                lcd_puts(" CABLE ERROR !");
                t=0;
                delay(7000);
                break; }
}
                        else{
                        lcd_com(0xC0);
                        lcd_puts(" CABLE ERROR !");
                        t=0;
                        delay(7000);
                        break; }
}
                                else{
                                lcd_com(0xC0);
                                lcd_puts(" CABLE ERROR !");
                                t=0;
                                delay(7000);
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
                                    break; }
}
                                        else{
                                        lcd_com(0xC0);
                                        lcd_puts(" CABLE ERROR !");
                                        t=0;
                                        delay(7000);
                                        break; }
}
                                          else{
                                          lcd_com(0xC0);
                                          lcd_puts(" CABLE ERROR !");
                                          t=0;
                                          delay(7000);
                                          break ;}
}
else{
        lcd_com(0xC0);
        lcd_puts(" CABLE ERROR !");
        t=0;
        delay(7000);
        break ;}
}
  ////////////////////////////////////////////////////
        t = 1;
        h2 = 0;
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
            P2 = 0x00;
              }
            else{
            P2 = 0x00;
            }
              }
}
}
void show(){
lcd_com(0X01);
lcd_com(0X80);
lcd_puts(" Connect Cable");
lcd_com(0XC0);
lcd_puts("  Press Check");
}
void error(){
lcd_com(0X01);
lcd_com(0X80);
lcd_puts("  CABLE ERROR !");
lcd_com(0xC0);
lcd_puts(" OR UNCONNECT..");
}
//////////////////////////////////////////////
int chk_cable(){
lcd_init();
lcd_com(0X01);
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
lcd_com(0X80);
lcd_puts(" Cable Checking");
delay(9000);
      if((I0==0)&&(I1==0)&&(I2==0)&&(I3==0)&&(I4==0)&&(I5==0)&&(I6==0)&&(I7==0))
                        {
                          x = 9;
                          lcd_com(0XC0);
                          lcd_puts(" CABLE PASS ALL");
                          delay(8000);
                            goto send;
                        }
                        else if((I0==0)||(I1==0)||(I2==0)||(I3==0)||(I4==0)||(I5==0)||(I6==0)||(I7==0)){
h = 0;
x = 1;
j = 1;
z = 1;
                        if(I0==1){
                                    e[j] = 1;
                                    h++;
                                    j=j+1;
                          }
                        if(I1==1){
                                    e[j] = 2;
                                    h++;
                                    j=j+1;
                        }
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
if(I2==1){
        e[j] = 3;
        h++;
        j=j+1;
}
if(I3==1){
        e[j] = 4;
        h++;
        j=j+1;
}
if(I4==1){
        e[j] = 5;
        h++;
        j=j+1;
}
if(I5==1){
        e[j] = 6;
        h++;
        j=j+1;
}
if(I6==1){
        e[j] = 7;
        h++;
        j=j+1;
}
if(I7==1){
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
            e[j] = 8;
            h++;
            j=j+1;
}
  delay(1000);
 ////////////////////////////////////////////////////
if(I0==0){
            d[z] = 1;
            z++;
            x++;
 }
if(I1==0){
            d[z] = 2;
            z++;
            x++;
}
if(I2==0){
            d[z] = 3;
            z++;
            x++;
}
if(I3==0){
            d[z] = 4;
            z++;
            x++;
}
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
if(I4==0){
        d[z] = 5;
        z++;
        x++;
}
if(I5==0){
        d[z] = 6;
        z++;
        x++;
}
if(I6==0){
        d[z] = 7;
        z++;
        x++;
}
if(I7==0){
        d[z] = 8;
        x++;
}
delay(1000);
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
////////////////////////////////////
lcd_com(0X01);
lcd_com(0X80);
lcd_puts(".....PASS.....");
lcd_com(0XC0);
for(g1=1;g1<x;g1++){
```

```
sprintf(lcd_buf,"%d ",d[g1]);
lcd_puts((unsigned char *)lcd_buf);
}
delay(10000);
if(h>=1){
lcd_com(0X01);
lcd_com(0X80);
 lcd_puts("......FAIL.....");
 lcd_com(0XC0);
 for(h1=1;h1<=h;h1++){
sprintf(lcd_buf,"%d ",e[h1]);
lcd_puts((unsigned char *)lcd_buf);
            }
}
delay(10000);
                        }
                        else{
                          error();
                                delay(9000);
                                x = 0;
                        }
return(x);
send: return(x);
}
```

มหาวิทยาลัยเทคโนโลยีสุรนารี