



อุปกรณ์เฝ้าระวังสัญญาณรบกวนในย่านสื่อสารการบิน

โดย

นางสาวอรปภา	บุญเพชร	รหัสประจำตัว	B5407750
นางสาวอรัญญา	อารมย์เพียร	รหัสประจำตัว	B5409433
นางสาววิภากรวีร์	โสมจุมจัง	รหัสประจำตัว	B5418497

รายงานนี้เป็นส่วนหนึ่งของการศึกษาวิชา 527499 โครงการศึกษาวิศวกรรมโทรคมนาคม
หลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมโทรคมนาคม หลักสูตรปรับปรุง พ.ศ. 2554
สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี
ประจำภาคการศึกษาที่ 3 ปีการศึกษา 2557

อุปกรณ์เฝ้าระวังสัญญาณรบกวนในย่านสื่อสารการบิน

คณะกรรมการสอบโครงการงาน



(ผู้ช่วยศาสตราจารย์ ดร. มนต์ทิพย์ภา อูหารสกุล)

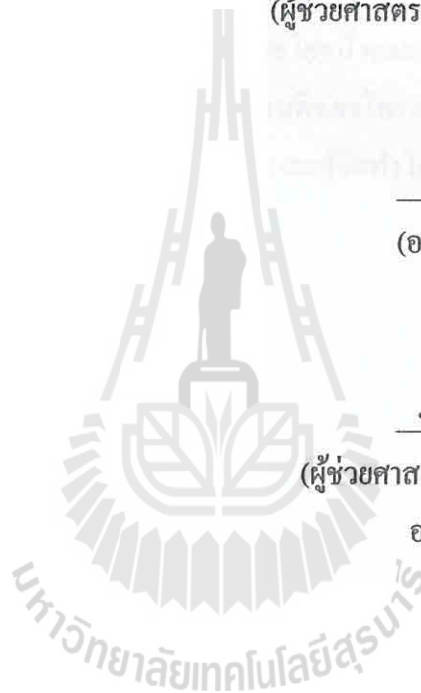


(อาจารย์ ดร. ธนเสฏฐ์ ทศศิกรพัฒน์)



(ผู้ช่วยศาสตราจารย์ ดร. รังสรรค์ ทองทา)

อาจารย์ที่ปรึกษาโครงการงาน



มหาวิทยาลัยเทคโนโลยีสุรนารีอนุมัติให้นับโครงการฉบับนี้เป็นส่วนหนึ่งของการศึกษาระดับ
ปริญญาตรี สาขาวิศวกรรมโทรคมนาคม รายวิชา 527499 โครงการวิศวกรรมโทรคมนาคม
ประจำปีการศึกษา 2557

กิตติกรรมประกาศ

โครงการฉบับนี้สำเร็จลุล่วงไปด้วยดี เนื่องจากได้รับความกรุณาจากท่านอาจารย์ที่ปรึกษาโครงการ ผู้ช่วยศาสตราจารย์ ดร.รังสรรค์ ทองทา ท่านได้ให้ความช่วยเหลือเกี่ยวกับแนวคิดการแก้ปัญหาที่เกิดขึ้น และดูแลเอาใจใส่ติดตามผลงานอย่างใกล้ชิด ซึ่งแนะข้อบกพร่องตลอดจนช่วยฝึกฝนและการสนับสนุนคณะผู้จัดทำโครงการ ให้มีความสามารถในการทำโครงการจนสำเร็จ

ขอขอบพระคุณอาจารย์และบุคลากรสาขาวิชาวิศวกรรมโทรคมนาคมทุกท่าน ที่ให้ความช่วยเหลือแก่คณะผู้จัดทำโครงการมาโดยตลอด นายปัญญา หันตุลา นักศึกษารุ่นพี่วิศวกรรมโทรคมนาคมที่คอยแนะนำและให้ความรู้ที่เป็นประโยชน์ คณะผู้จัดทำโครงการใคร่ขอขอบพระคุณทุกท่านที่ได้กล่าวไปแล้วไว้ ณ ที่นี้ สำหรับส่วนดีของโครงการชิ้นนี้ ขออุทิศให้แก่อาจารย์ทุกท่านที่ได้ประสิทธิ์ประสาทวิชาความรู้มาให้แก่คณะผู้จัดทำโครงการทำให้การทำโครงการชิ้นนี้ผ่านไปด้วยดี

ผู้จัดทำ

นางสาวอรปภา บุญเพชร
นางสาวอรทัย อารมย์เพียร
นางสาววิภัทรวีร์ โสมจุมจัง

มหาวิทยาลัยเทคโนโลยีสุรนารี

โครงการ	อุปกรณ์เฝ้าระวังสัญญาณรบกวนในย่านสื่อสารการบิน
จัดทำโดย	นางสาวอรปภา บุญเพชร นางสาวอรทัย อารมย์เพชร นางสาววิภัทรวีร์ โสมจุมจัง
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ ดร. รังสรรค์ ทองทา
สาขาวิชา	วิศวกรรมโทรคมนาคม
ภาคการศึกษาที่	3/ 2557

บทคัดย่อ (Abstract)

อุปกรณ์เฝ้าระวังสัญญาณรบกวนในย่านสื่อสารการบิน คือ อุปกรณ์ต้นแบบที่จะสามารถลดปัญหาการใช้ทรัพยากรบุคคลในการลงพื้นที่สำรวจพื้นที่ทางสำนักงาน กสทช. โดยกระบวนการทำงานที่ทำให้สิ้นเปลืองทรัพยากรบุคคลของสำนักงาน กสทช. โดยจะจัดตั้งห้องปฏิบัติการทดสอบเครื่องส่งจะทำการตรวจสอบและค้นหาคลื่นวิทยุที่มีได้รับอนุญาตในเขตพื้นที่นั้นแต่ปัญหาในการตรวจสอบคือสามารถระบุคลื่นความถี่ที่รบกวนได้อย่างชัดเจนแต่ไม่สามารถระบุพิกัดพื้นที่ของคลื่นความถี่นั้นจำเป็นต้องทำการออกสำรวจในพื้นที่ที่ระบุพิกัดและพื้นที่ที่ออกสำรวจจะใช้ทรัพยากรบุคคลเป็นจำนวนมากที่ได้แจ้งเตือนว่ามีผู้ลักลอบใช้ความถี่โดยมิได้รับอนุญาตทางคณะจัดทำโครงการนี้ขึ้นมาทางโดยทางผู้จัดทำได้สร้างอุปกรณ์ที่สามารถช่วยลดปัญหาดังกล่าวได้ลักษณะการทำงานของอุปกรณ์ชิ้นนี้สามารถติดตั้งอุปกรณ์ไว้ตามพื้นที่ต่างๆและยังสามารถตรวจสอบค้นหาสถานีที่กำลังออกอากาศอยู่ใน ณ ขณะนั้น ซึ่งผู้ตรวจสอบสามารถตรวจสอบเครื่องนี้ได้ผ่านทางระบบเครือข่ายอินเทอร์เน็ตไม่จำเป็นต้องอยู่ในพื้นที่นั้นทำให้สามารถเพิ่มประสิทธิภาพในการค้นหาได้

สารบัญ

เรื่อง	หน้า
กิตติกรรมประกาศ.....	ก
บทคัดย่อ.....	ข
สารบัญ.....	ค
สารบัญรูปภาพ.....	ง
สารบัญตาราง.....	ฉ
บทที่	
1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์.....	2
1.3 ขอบเขตการดำเนินการ.....	2
1.4 ขั้นตอนการดำเนินงาน.....	2
1.5 ประโยชน์คาดว่าจะได้รับ.....	2
2 ทฤษฎีพื้นฐานที่เกี่ยวข้อง	3
2.1 กล่าวนำ.....	3
2.2 มาตรฐานทางเทคนิคของเครื่องส่งวิทยุกระจายเสียงสำหรับชุมชน.....	4
2.3 ระบบสื่อสารวิทยุ.....	10
2.3.1 การผสมคลื่นทางความสูงหรือแบบ AM (Amplitude Modulation).....	10
2.3.2 การผสมคลื่นทางความถี่หรือแบบ FM (Frequency Modulation).....	11
2.3.3 การผสมคลื่นทางเฟสหรือแบบ PM (Phase Modulation).....	12
2.3.4 การผสมสัญญาณรหัสพัลส์ (PMC).....	12
2.3.5 การแยกคลื่นสัญญาณแบบผสม AM.....	13
2.3.6 การแยกคลื่นสัญญาณแบบผสม FM และ PM.....	14
2.3.7 คลื่นเสียงวิทยุ.....	14
2.3.8 คลื่นวิทยุ.....	1

สารบัญ (ต่อ)

เรื่อง บทที่	หน้า
2.4 โมดูลเครื่องรับวิทยุ FM สำหรับไมโครคอนโทรลเลอร์.....	16
2.4.1 คุณสมบัติที่สำคัญ.....	17
2.4.2 การทำงาน.....	18
2.4.3 รีจิสเตอร์ควบคุมการทำงานของโมดูลวิทยุเอฟเอ็ม.....	19
2.4.4 การติดต่อกับบอร์ดไมโครคอนโทรลเลอร์.....	25
2.4.5 การใช้งานของเอฟเอ็มโมดูล.....	26
2.5 การเชื่อมต่ออุปกรณ์แบบ I ² C 2.5.1 การเชื่อมต่ออุปกรณ์แบบ I ² C.....	27
2.5.1 การเชื่อมต่ออุปกรณ์แบบ I ² C.....	27
2.5.2 การเขียน-อ่านข้อมูลแบบ I ² C.....	28
2.5.3 การรับ-ส่งข้อมูลแบบ I ² C BUS MCU.....	28
2.5.4 การกำหนดสถานะเริ่มต้นและสถานะสิ้นสุดแบบ I ² C BUS.....	28
2.5.5 รหัสควบคุมของ I ² C BUS.....	29
2.5.6 ช่วงเวลารับส่งบิตข้อมูลของ I ² C BUS.....	29
2.6 บอร์ดไมโครคอนโทรลเลอร์ Arduino UNO R3.....	30
2.6.1 คุณสมบัติของบอร์ดไมโครคอนโทรลเลอร์ Arduino UNO R3.....	32
2.6.2 การใช้งานของบอร์ดไมโครคอนโทรลเลอร์ Arduino UNO R3.....	33
2.7 โปรแกรม Visual Basic 2010.....	34
2.8 โปรแกรม Team Viewer.....	35
3. การออกแบบอุปกรณ์ไฟระว่างสัญญาณในย่านสื่อสารการบิน.....	36
3.1 กล่าวนำ.....	36
3.2 องค์ประกอบและหลักการทำงานของอุปกรณ์ไฟระว่างสัญญาณในย่านสื่อสารการบิน.....	37
3.2.1 การควบคุมการทำงานของโมดูลวิทยุเอฟเอ็ม.....	38
3.2.2 การออกแบบคำสั่งสำหรับควบคุม.....	40
3.2.3 การเขียนโปรแกรมคำสั่งควบคุม.....	45
3.2.4 การออกแบบโปรแกรมคำสั่งควบคุม.....	50

สารบัญ (ต่อ)

เรื่อง บทที่	หน้า
4. การใช้งานและการทดสอบอุปกรณ์ต้นแบบ	58
4.1 กล่าวนำ	58
4.2 การติดตั้งและการใช้งานอุปกรณ์รับสัญญาณวิทยุ	58
4.3 การใช้งาน	69
4.4 การใช้งานเครื่องรับวิทยุ	73
4.5 การใช้งานโปรแกรม Team Viewer	75
4.6 การทดสอบการรับสัญญาณวิทยุในช่องความถี่ต่างๆ	78
4.6.1 ทดสอบรับคลื่นสัญญาณวิทยุของอุปกรณ์ต้นแบบเปรียบเทียบกับ เครื่องรับวิทยุทั่วไป	79
4.6.2 การทดสอบการค้นหาสถานีแบบอัตโนมัติของอุปกรณ์ต้นแบบ เปรียบเทียบกับวิทยุทั่วไป	82
4.6.3 การทดสอบการค้นหาสถานีแบบอัตโนมัติของอุปกรณ์ต้นแบบ	84
4.7 สรุป	86
5. ข้อสรุปของโครงการ	87
5.1 กล่าวนำ	87
5.2 ปัญหาที่พบและแนวทางในการแก้ไขปัญหา	87
5.3 ข้อเสนอแนะ	88
5.4 แนวทางการพัฒนา	89
5.5 บทสรุปของโครงการ	89
บรรณานุกรม	90
ภาคผนวก	91
ประวัติผู้เขียน	99

สารบัญตาราง

ตารางที่	หน้า
2.1 แสดงข้อมูลของรีจิสเตอร์0x02.....	20
2.2 แสดงข้อมูลของรีจิสเตอร์0x03.....	21
2.3 แสดงข้อมูลของรีจิสเตอร์0x04.....	22
2.4 แสดงข้อมูลของรีจิสเตอร์0x05.....	23
2.5 แสดงข้อมูลของรีจิสเตอร์0x06.....	23
2.6 แสดงข้อมูลของรีจิสเตอร์0x0A และ แสดงข้อมูลของรีจิสเตอร์0x0B.....	24
4.1 แสดงการทดสอบรับคลื่นสัญญาณวิทยุของอุปกรณ์ต้นแบบเปรียบเทียบกับ เครื่องรับวิทยุทั่วไป.....	79
4.2 แสดงสรุปผลการทดสอบอุปกรณ์ต้นแบบเปรียบเทียบกับเครื่องรับวิทยุทั่วไป.....	82
4.3 แสดงการทดสอบการค้นหาสถานีแบบอัตโนมัติของอุปกรณ์ต้นแบบ เปรียบเทียบกับวิทยุทั่วไป.....	83
4.4 แสดงสรุปผลการทดสอบการค้นหาสถานีแบบอัตโนมัติของวิทยุทั่วไป.....	84
4.5 แสดงการทดสอบการค้นหาสถานีแบบอัตโนมัติของอุปกรณ์ต้นแบบ.....	84
4.6 แสดงสรุปการทดลองใช้อุปกรณ์ต้นแบบในการค้นหาสถานีวิทยุแบบอัตโนมัติ.....	85

สารบัญรูปภาพ

รูปที่	หน้า
2.1 แสดงการแบ่งช่องความถี่สัญญาณวิทยุระบบ FM.....	6
2.2 แสดงขีดจำกัดการแพร่กระจาย.....	7
2.3 แสดงการ Intermediation Spectrum.....	9
2.4 แสดงหลักการส่งข้อมูลข่าวสารทางคลื่นวิทยุ.....	10
2.5 แสดงการผสมคลื่นทางความถี่หรือแบบ AM (Amplitude Modulation).....	11
2.6 แสดงการผสมคลื่นทางความถี่หรือแบบ FM (Frequency Modulation).....	11
2.7 แสดงลักษณะการผสมคลื่นทางเฟส PM (Phase Modulation).....	12
2.8 แสดงลักษณะการผสมสัญญาณแบบรหัสพัลส์ PCM.....	13
2.9 แสดงลักษณะการแยกคลื่นสัญญาณผสมแบบ AM (Amplitude Modulation).....	13
2.10 แสดงลักษณะคลื่นเสียงที่เราสามารถได้ยินได้.....	14
2.11 แสดงลักษณะการส่งคลื่นวิทยุเอเอ็ม.....	15
2.12 แสดงลักษณะการส่งคลื่นแบบเอฟเอ็ม.....	16
2.13 FM Module เบอร์ RDA5707SS.....	16
2.14 แสดงหน้าตาและการจัดขาและหน้าที่ขาใช้งานทั้งหมดของ FM.....	17
2.15 แสดงไดอะแกรมการทำงานภายในของไอซี RDA5707SS ที่ใช้ในโมดูล.....	19
2.16 แสดงตัวอย่างวงจรเชื่อมต่อไมโครคอนโทรลเลอร์อย่างง่ายที่สุดเพื่อใช้งานโมดูลวิทยุ FM.....	25
2.17 แสดงภาพส่วนประกอบของ FM Module เบอร์ RDA5707SS.....	26
2.18 แสดงลักษณะการเชื่อมต่ออุปกรณ์แบบ I ² C BUS.....	27
2.19 แสดงรูปแบบการเขียน/อ่านข้อมูลแบบ I ² C BUS.....	29
2.20 แสดง I ² C BUS START และ STOP Condition.....	28
2.21 แสดง I ² C BUS (Control Byte).....	29
2.22 แสดงการรับส่งบิตข้อมูลของ I ² C BUS.....	29
2.23 แสดงบอร์ด Arduino UNO R3.....	30

สารบัญรูปภาพ(ต่อ)

รูปที่	หน้า
2.24 แสดงส่วนประกอบของบอร์ด Arduino UNO R3.....	31
2.25 แสดงการใช้งานของบอร์ด Arduino UNO R3.....	33
2.26 แสดงโปรแกรม Visual Basic 2010.....	34
3.1 แสดงภาพรวมของระบบเฟิร์มแวร์สัญญาณรบกวนในย่านสื่อสารการบิน.....	36
3.2 แสดงองค์ประกอบของระบบเฟิร์มแวร์สัญญาณรบกวนในย่านสื่อสารการบิน.....	37
3.3 แสดงการควบคุมการทำงานของ FM Module RDA5807SS.....	39
3.4 แสดงแผนผังการออกแบบคำสั่งควบคุม.....	41
3.5 แสดงหน้าต่างการใช้งานของโปรแกรมควบคุม.....	45
3.6 แสดงการตั้งค่า Board.....	46
3.7 แสดงการตั้งค่า Port.....	46
3.8 แสดงการเขียนโปรแกรมลงบนหน้าต่าง IDE.....	47
3.9 แสดงการคอมไพล์โปรแกรม.....	47
3.10 แสดงข้อความการคอมไพล์ไม่สำเร็จ.....	48
3.11 แสดงข้อความการ UPLOND โปรแกรม.....	48
3.12 แสดงการ UPLOND โปรแกรม.....	49
3.13 แสดงข้อความการ UPLOND โปรแกรมไม่สำเร็จ.....	49
3.14 แสดงหน้าต่างการใช้งานโปรแกรม.....	50
3.15 แสดงฟังก์ชันการใช้งานการสร้างโปรแกรม.....	51
3.16 แสดงหน้าต่างการใช้งานโปรแกรม.....	51
3.17 แสดงการสร้าง Object.....	52
3.18 แสดงรายละเอียดของฟอร์ม.....	52
3.19 แสดงหน้าต่างการเขียนโค้ดควบคุม.....	53
3.20 แสดงหน้าต่างการกำหนดหรือแก้ไขโค้ดคำสั่งควบคุม.....	53
3.21 แสดงหน้าต่างการออกแบบโปรแกรมคำสั่งควบคุม.....	54

สารบัญรูปภาพ(ต่อ)

รูปที่	หน้า
3.21 แสดงหน้าต่างแสดงผลส่วนที่หนึ่ง.....	54
3.22 แสดงหน้าต่างโปรแกรมคำสั่งควบคุมส่วนที่สอง.....	55
3.23 แสดงหน้าต่างโปรแกรมคำสั่งควบคุมส่วนที่สาม.....	55
3.24 แสดงหน้าต่างโปรแกรมคำสั่งควบคุมส่วนที่สี่.....	56
3.25 แสดงการรันโปรแกรมเพื่อทดสอบ.....	57
3.26 แสดงการรันโปรแกรมเพื่อทดสอบเมื่อเกิดการ Error.....	57
3.27 แสดงหน้าต่างเมื่อรันโปรแกรมสำเร็จเรียบร้อย.....	57
4.1 แสดงการขีดสายอากาศ.....	59
4.2 (ก) การเชื่อมต่อสาย Audio Output ที่อุปกรณ์.....	59
(ข) การต่อสาย Audio Output เข้ากับช่อง Line in ของคอมพิวเตอร์.....	59
4.3 (ค) การเชื่อมต่อสาย USB ที่อุปกรณ์ต้นแบบ.....	60
(ง) การเชื่อมต่อสาย USB เข้ากับเครื่องคอมพิวเตอร์.....	60
4.4 แสดงการเลือกโหมด.....	60
4.5 แสดง Stereo Mix.....	60
4.6 แสดงการตั้งค่าเสียงคอมพิวเตอร์.....	61
4.7 แสดงการตั้งค่าเสียงให้ดัง.....	61
4.8 แสดงการตั้งค่าเสียงในกรณีที่ใช้อุปกรณ์ไม่ปรากฏ.....	62
4.9 แสดงการทดสอบ Line In.....	62
4.10 แสดงการเลือก Line In.....	63
4.11 แสดงการเลือก Line In ไม่ได้.....	63
4.12 แสดงวิธีการติดตั้ง Device (1).....	64
4.13 แสดงวิธีการติดตั้ง Device (2).....	65
4.14 แสดงทำการลง Device.....	65

สารบัญรูปภาพ(ต่อ)

รูปที่	หน้า
4.15 แสดงการเลือกตำแหน่งไฟล์	66
4.16 แสดงรูปการปรากฏตำแหน่งไฟล์.....	67
4.17 แสดงกดเลือกคำสั่งเพื่อดำเนินการต่อ.....	67
4.18 แสดงการติดตั้งสำเร็จ.....	68
4.19 แสดง Com Port	68
4.20 แสดงหน้าต่างการออกแบบ โปรแกรมคำสั่งควบคุม	69
4.21 แสดงการเลือก COM Port.....	70
4.22 แสดงการเลือก Baud Rate	70
4.23 แสดงการเลือก Connect	71
4.24 แสดง Status การเชื่อมต่อ.....	71
4.25 แสดงสถานะ Receiver Status	72
4.26 แสดงการเริ่มระบบใหม่.....	72
4.27 แสดงการเลื่อนแท็บความถี่.....	73
4.28 แสดงการป้อนตัวเลขความถี่.....	73
4.29 แสดง SCAN UP.....	74
4.30 แสดง SCAN DOWN.....	74
4.31 แสดงหน้าต่างโปรแกรม Team Viewer (1).....	75
4.32 แสดงหน้าต่างโปรแกรม Team Viewer (2).....	76
4.33 แสดงหน้าต่างการใช้งาน Control Remote Computer.....	76
4.34 แสดงหน้าต่าง Team Viewer Authentication.....	77
4.35 แสดงการเชื่อมต่อเสร็จหน้าจอของเครื่องคอมพิวเตอร์ที่ติดตั้งอยู่กับเครื่องรับวิทยุสื่อสาร.....	77
4.36 แสดงเมนู View.....	78
4.37 แสดงตำแหน่งของการทดสอบในพื้นที่บริเวณอาคารสุรนิทัศน์.....	78

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันพบว่าได้มีการอนุญาตให้มีผู้ทดลองประกอบกิจการวิทยุกระจายเสียงหรือเดิมเรียกว่า “วิทยุชุมชน” มีการจัดตั้งสถานีวิทยุกระจายเสียงในย่านความถี่ 87.50 - 107.50 MHz ซึ่งสถานีวิทยุมีทั้งที่ได้รับอนุญาตที่มีได้รับอนุญาตจากสำนักงานคณะกรรมการกิจการกระจายเสียง กิจการโทรทัศน์และกิจการโทรคมนาคมแห่งชาติ(กสทช.) ซึ่งเป็นสาเหตุสำคัญที่ทำให้เกิดการรบกวนคลื่นวิทยุที่ใช้ในกิจการควบคุมการจราจรทางอากาศ เนื่องจากการรบกวนที่เกิดขึ้นทำให้การสื่อสารระหว่างนักบินกับศูนย์ควบคุมการจราจรทางอากาศภาคพื้นดินเกิดความไม่ชัดเจน ส่งผลกระทบต่อความปลอดภัยทางการบิน อาจส่งผลให้เกิดอุบัติเหตุร้ายแรงต่อชีวิตและทรัพย์สินได้

ถึงแม้ในปัจจุบันสำนักงาน กสทช. ได้มีการจัดตั้งห้องปฏิบัติการทดสอบเครื่องส่งวิทยุขึ้นเพื่อลดปัญหาเครื่องส่งวิทยุกระจายเสียงที่ไม่ได้มาตรฐานและนอกจากนี้สำนักงาน กสทช. กำหนดให้ผู้ที่มีความประสงค์ที่จะทดลองประกอบกิจการวิทยุกระจายเสียงจะต้องขอขึ้นทะเบียนกับสำนักงาน กสทช. เพื่อจัดตั้งสถานีวิทยุ แต่ยังมีผู้ประกอบการลักลอบใช้ความถี่โดยมิได้รับอนุญาตเป็นจำนวนมาก ทำให้ปัญหาการรบกวนและปัญหาการลักลอบใช้ความถี่โดยมิได้รับอนุญาตยังคงเกิดขึ้นอย่างต่อเนื่อง

ในปัจจุบันของสำนักงาน กสทช. จะทำการตรวจสอบและค้นหาคลื่นวิทยุที่มีได้รับอนุญาตในเขตพื้นที่นั้น แต่ปัญหาในการตรวจสอบคือสามารถระบุคลื่นความถี่ที่รบกวนได้อย่างชัดเจนแต่ไม่สามารถระบุพิกัดพื้นที่ของคลื่นความถี่นั้นจำเป็นต้องทำการออกสำรวจในพื้นที่เขตรับผิดชอบและพื้นที่ที่ออกสำรวจจะใช้ทรัพยากรบุคคลเป็นจำนวนมากที่ได้รับการแจ้งเตือนว่ามีเกิดการลักลอบใช้งานคลื่นความถี่โดยมิได้รับอนุญาต ซึ่งกระบวนการข้างต้นนี้ทำให้เสียทรัพยากรบุคคลมากเกินไปและอาจมีการดำเนินการที่ล่าช้าและไม่มีประสิทธิภาพ

ดังนั้นทางคณะผู้จัดทำโครงการตระหนักถึงปัญหา จึงจัดทำโครงการนี้ขึ้นมาทางโดยทางผู้จัดทำได้สร้างอุปกรณ์ที่สามารถช่วยลดปัญหาดังกล่าวได้ ลักษณะการทำงานของอุปกรณ์ชิ้นนี้สามารถติดตั้งอุปกรณ์ไว้ตามพื้นที่ต่างๆและยังสามารถตรวจสอบค้นหาสถานีที่กำลังออกอากาศอยู่ใน ณ ขณะนั้น ซึ่งผู้ตรวจสอบสามารถตรวจสอบเครื่องนี้ได้ผ่านทางระบบเครือข่ายอินเทอร์เน็ตไม่จำเป็นต้องอยู่ในพื้นที่นั้นทำให้สามารถเพิ่มประสิทธิภาพในการค้นหาได้

1.2 วัตถุประสงค์

เพื่อศึกษาการออกแบบและสร้างอุปกรณ์เฝ้าระวังสัญญาณรบกวนในย่านสื่อสารการบิน

1.3 ขอบเขตการดำเนินงาน

- 1.3.1 ศึกษาที่มาของปัญหา อุปกรณ์เฝ้าระวังสัญญาณรบกวนในย่านสื่อสารการบิน
- 1.3.2 ออกแบบและสร้างอุปกรณ์เฝ้าระวังสัญญาณรบกวนในย่านสื่อสารการบิน
- 1.3.3 สามารถควบคุม และรับข้อมูลข่าวสารผ่านระบบเครือข่ายอินเทอร์เน็ตได้
- 1.3.4 ทดสอบชุดอุปกรณ์นี้ได้ในสภาพแวดล้อมจริง

1.4 ขั้นตอนการดำเนินงาน

- 1.4.1 ศึกษาค้นคว้าข้อมูลเบื้องต้นที่เกี่ยวกับกิจการวิทยุกระจายเสียงในประเทศไทย
- 1.4.2 ศึกษาค้นคว้าข้อมูลเกี่ยวกับอุปกรณ์ที่ใช้ในโครงการ
- 1.4.3 ออกแบบและสร้างอุปกรณ์แล้วนำไปใช้ทดสอบจริง
- 1.4.4 สรุปผลการทดลองเขียนรายงาน และนำเสนอโครงการ

1.5 ประโยชน์ที่คาดว่าจะได้รับ

- 1.5.1 สร้างอุปกรณ์เฝ้าระวังสัญญาณรบกวนในย่านสื่อสารการบิน
- 1.5.2 อุปกรณ์เฝ้าระวังสัญญาณรบกวนในย่านสื่อสารการบิน ยังทำให้ประหยัดทางด้านงบประมาณ ประหยัดทางด้านบุคลากร ประหยัดเวลา
- 1.5.3 ได้รับความรู้ทางการเขียนโปรแกรมลงบอร์ด Arduino UNO R3
- 1.5.4 สามารถบูรณาการเอาความรู้จากวิชาต่าง ๆ ที่ได้รับมาจัดทำโครงการ
- 1.5.5 สามารถนำอุปกรณ์ที่ประกอบแล้วมาทดสอบใช้งานจริงเพื่อให้ได้ตามวัตถุประสงค์

บทที่ 2

ทฤษฎีพื้นฐานที่เกี่ยวข้อง

2.1 กล่าวนำ

ในบทนี้จะกล่าวถึงทฤษฎีที่เกี่ยวข้องกับการออกแบบและสร้างอุปกรณ์ต้นแบบของอุปกรณ์เฝ้าระวังสัญญาณรบกวนในย่านสื่อสารการบิน โดยแนวคิดนี้เริ่มจากสำนักงานคณะกรรมการกิจการกระจายเสียงกิจการโทรทัศน์และกิจการโทรคมนาคมแห่งชาติ(กสทช.) ได้มีคลื่นวิทยุเข้ามารบกวนที่ใช้ในกิจการควบคุมการจราจรทางอากาศทำให้การสื่อสารระหว่างนักบินกับศูนย์ควบคุมการจราจรทางอากาศภาคพื้นดินเกิดความไม่ชัดเจน ส่งผลกระทบต่อความปลอดภัยทางการบิน อาจส่งผลให้เกิดอุบัติเหตุร้ายแรงต่อชีวิตและทรัพย์สินถึงแม้สำนักงาน กสทช. ได้มีการจัดตั้งห้องปฏิบัติการทดสอบเครื่องส่งวิทยุขึ้น เพื่อลดปัญหาเครื่องส่งวิทยุกระจายเสียงที่ไม่ได้มาตรฐาน และนอกจากนี้ยังกำหนดให้ผู้ที่มีความประสงค์ที่จะทดลองประกอบกิจการวิทยุกระจายเสียงจะต้องขอขึ้นทะเบียนกับสำนักงาน กสทช. เพื่อจัดตั้งสถานีวิทยุแต่ยังมีผู้ประกอบการลักลอบใช้ความถี่โดยมิได้รับอนุญาตเป็นจำนวนมาก ทำให้ปัญหาการรบกวนและปัญหาการลักลอบใช้ความถี่โดยมิได้รับอนุญาตยังคงเกิดขึ้นอย่างต่อเนื่อง แต่ในการแก้ปัญหาเบื้องต้นของสำนักงาน กสทช. จะทำการตรวจสอบและค้นหาคลื่นวิทยุที่มิได้รับอนุญาตในเขตพื้นที่นั้น แต่ปัญหาในการตรวจสอบคือ สามารถระบุคลื่นความถี่ที่รบกวนได้อย่างชัดเจนแต่ไม่สามารถระบุพิกัดพื้นที่ของคลื่นความถี่นั้น จำเป็นต้องทำการออกสำรวจในพื้นที่เขตรับผิดชอบและพื้นที่ที่ออกสำรวจจะใช้ทรัพยากรบุคคลเป็นจำนวนมากที่ได้รับการแจ้งเตือนว่ามีเกิดการลักลอบใช้งานคลื่นความถี่ โดยมิได้รับอนุญาตซึ่งกระบวนการข้างต้นนี้ ทำให้เสียทรัพยากรบุคคลมากเกินไปและอาจมีการดำเนินการที่ล่าช้าและไม่มีประสิทธิภาพแนวคิดเริ่มแรกของการทำโครงการครั้งนี้ คือการสร้างอุปกรณ์ต้นแบบของอุปกรณ์เฝ้าระวังสัญญาณรบกวนในย่านสื่อสารการบิน โดยอาศัยผ่านทางระบบเครือข่ายอินเทอร์เน็ตและอุปกรณ์ชิ้นนี้สามารถติดตั้งอุปกรณ์ไว้ตามพื้นที่ต่างๆ และยังสามารถตรวจสอบค้นหาสถานีที่กำลังออกอากาศอยู่ในพื้นที่นั้นซึ่งสามารถเพิ่มประสิทธิภาพในการค้นหาได้



2.2 มาตรฐานทางเทคนิคของเครื่องส่งวิทยุกระจายเสียงสำหรับชุมชน

ตามพระราชบัญญัติองค์กรจัดสรรคลื่นความถี่และกำกับการวิทยุกระจายเสียง วิทยุโทรทัศน์และกิจการโทรคมนาคม พ.ศ. 2543 คณะกรรมการกิจการวิทยุกระจายเสียงแห่งชาติ (กสทช.) มีหน้าที่พิจารณาอนุญาตและกำกับดูแลการใช้คลื่นความถี่เพื่อกิจการวิทยุกระจายเสียง และกิจการวิทยุโทรทัศน์และการกำหนดมาตรฐานและลักษณะพึงประสงค์ทางเทคนิคของอุปกรณ์ที่ใช้ในการประกอบกิจการวิทยุกระจายเสียงและวิทยุโทรทัศน์เนื่องจากขณะนี้ยังไม่มี การแต่งตั้ง กสทช.เพื่อทำหน้าที่ตามที่กฎหมายบัญญัติ ทำให้ไม่สามารถพิจารณาจัดสรรคลื่นความถี่ออก ใบอนุญาตประกอบกิจการ หรืออนุญาตให้ประกอบกิจการเพิ่มเติมไม่ได้ ตามบทบัญญัติมาตรา 80 ของ พ.ร.บ. ดังกล่าวเพื่อบรรเทาปัญหาการตั้งสถานีวิทยุกระจายเสียงสำหรับชุมชนเป็นการชั่วคราว คณะรัฐมนตรีได้มีมติในการประชุมวันที่ 16 สิงหาคม 2548 กำหนดกรอบกติกาทางเทคนิคและ หลักเกณฑ์การตั้งสถานีวิทยุชุมชน โดยกำหนดกำลังส่งไม่เกิน 30 วัตต์ความสูงเสาอากาศไม่เกิน 30 เมตรจากระดับพื้นดิน และรัศมีการออกอากาศไม่เกิน 15 กิโลเมตรแม้ว่ายังไม่มีการอนุญาตจัดตั้ง สถานีวิทยุชุมชนถูกต้องตามกฎหมายอย่างเป็นทางการ แต่ปัจจุบันมีการส่งวิทยุกระจายเสียงของสถานี วิทยุชุมชน ทั้งในพื้นที่เขตกรุงเทพมหานคร ปริมณฑล และในเขตภูมิภาคทั่วประเทศ ประมาณ 3,000-4,000 สถานี ทำให้เกิดการแพร่กระจายคลื่นแปลกปลอมที่ก่อให้เกิดการรบกวนต่อข่ายการ สื่อสารอื่น โดยเฉพาะข่ายสื่อสารทางการบินที่ใช้สำหรับการควบคุมการจราจรทางอากาศ ซึ่งมี ผลกระทบต่อความปลอดภัยของชีวิตและทรัพย์สินโดยตรง ดังนั้นคณะกรรมการกิจการ โทรคมนาคมแห่งชาติ ได้แต่งตั้ง คณะอนุกรรมการเฉพาะกิจจัดทำมาตรฐานเทคนิคสำหรับเครื่อง วิทยุคมนาคม เพื่อรับผิดชอบการศึกษามาตรฐานสากล และยกร่างมาตรฐานทางเทคนิคสำหรับ เครื่องส่งวิทยุคมนาคมที่เหมาะสมของเครื่องส่งวิทยุกระจายเสียงสำหรับชุมชนย่านความถี่วิทยุ 87.5 MHz ถึง 108 MHz เพื่อป้องกันและลดการรบกวนที่เกิดขึ้นกับข่ายสื่อสารในกิจการวิทยุ คมนาคมอื่น โดยเฉพาะกิจการทางการบิน (aeronautical) ในย่านความถี่วิทยุใกล้เคียง 108-137 MHz

คณะอนุกรรมการเฉพาะกิจจัดทำมาตรฐานทางเทคนิคสำหรับเครื่องวิทยุคมนาคม ได้ดำเนินการศึกษาเรื่องดังกล่าวแล้ว และได้ทำ รายงานผลการศึกษการจัดทำมาตรฐานทางเทคนิค ของเครื่องส่งวิทยุกระจายเสียงสำหรับชุมชน ย่านความถี่วิทยุ 87.5 MHz ถึง 108 MHz (พฤศจิกายน 2550) และมาตรฐานและลักษณะพึงประสงค์ทางด้านเทคนิคเครื่องส่งวิทยุกระจายเสียงสำหรับ

ชุมชน ย่านความถี่วิทยุ 87.5 MHz ถึง 108.0 MHz เพื่อให้ กทข. นำเสนอ กทข. (อธิบดีกรมประชาสัมพันธ์ เป็นเลขานุการคณะกรรมการกิจการวิทยุกระจายเสียงและวิทยุแห่งชาติ) หนังสือของ กทข. ลงวันที่ 27 ธันวาคม 2550 ในการศึกษาครั้งนี้นำกรอบและเกณฑ์การตั้งสถานีวิทยุชุมชนที่กำหนดกำลังส่งไม่เกิน 30 วัตต์ ความสูงเสาอากาศไม่เกิน 30 เมตร และรัศมีการออกอากาศไม่เกิน 15 กิโลเมตรตามที่กำหนดโดยมติคณะรัฐมนตรีเป็นเกณฑ์ แนวทางปฏิบัติในการจัดตั้งสถานีวิทยุกระจายเสียงสำหรับชุมชน มีหลักในการพิจารณา คือ

- การกำหนดพื้นที่การให้บริการ เพื่อไม่ให้มีการรบกวนกัน
- จะต้องมีการกำกับการใช้ความถี่วิทยุให้สอดคล้องกับแผนการใช้ความถี่วิทยุ

จะเห็นได้ว่าการจัดตั้งสถานีวิทยุชุมชนจะต้องพิจารณาเงื่อนไขทางเทคนิคคือการรบกวนเป็นหลักสำคัญ และการกำหนดแผนจัดสรรคลื่นความถี่วิทยุชุมชนจะต้องออกแบบไม่ให้มีการรบกวนกัน จึงจะขอสรุปข้อกำหนดการรบกวนของสถานีวิทยุตามที่มีการศึกษาจากเอกสารดังกล่าว คือ

1. การแบ่งช่องความถี่สัญญาณวิทยุระบบ FM เพื่อใช้ในการพิจารณาจัดตั้งวิทยุชุมชน ความถี่ 87.5-108 MHz แบ่งแต่ละช่อง 250 kHz แบ่งได้ 81 ช่องความถี่ ดังรูปที่ 2.1

ช่องที่	ความถี่ (MHz)	ช่องที่	ความถี่ (MHz)	ช่องที่	ความถี่ (MHz)
1	87.75	28	94.50	55	101.25
2	88.00	29	94.75	56	101.5
3	88.25	30	95.00	57	101.75
4	88.50	31	95.25	58	102
5	88.75	32	95.50	59	102.25
6	89.00	33	95.75	60	102.5
7	89.25	34	96.00	61	102.75
8	89.50	35	96.25	62	103
9	89.75	36	96.50	63	103.25
10	90.00	37	96.75	64	103.5
11	90.25	38	97.00	65	103.75
12	90.50	39	97.25	66	104
13	90.75	40	97.50	67	104.25

14	91.00	41	97.75	68	104.5
15	91.25	42	98.00	69	104.75
16	91.5	43	98.25	70	105
17	91.75	44	98.50	71	105.25
18	92.00	45	98.75	72	105.5
19	92.25	46	99.00	73	105.75
20	92.5	47	99.25	74	106
21	92.75	48	99.50	75	106.25
22	93.00	49	99.75	76	106.5
23	93.25	50	100.00	77	106.75
24	93.50	51	100.25	78	107
25	93.75	52	100.50	79	107.25
26	94.00	53	100.75	80	107.5
27	94.25	54	101.00	81	107.75

รูปที่ 2.1 ตารางแสดงการแบ่งช่องความถี่สัญญาณวิทยุระบบ FM

2. ข้อกำหนดเครื่องส่ง (rated carrier power) กำลังส่งที่ออกอากาศที่สายอากาศ จะต้องมีความไม่เกิน 30 วัตต์ (ไม่ได้กำหนดว่าสายอากาศเป็นชนิดใด มีกำลังขยายหรือไม่)

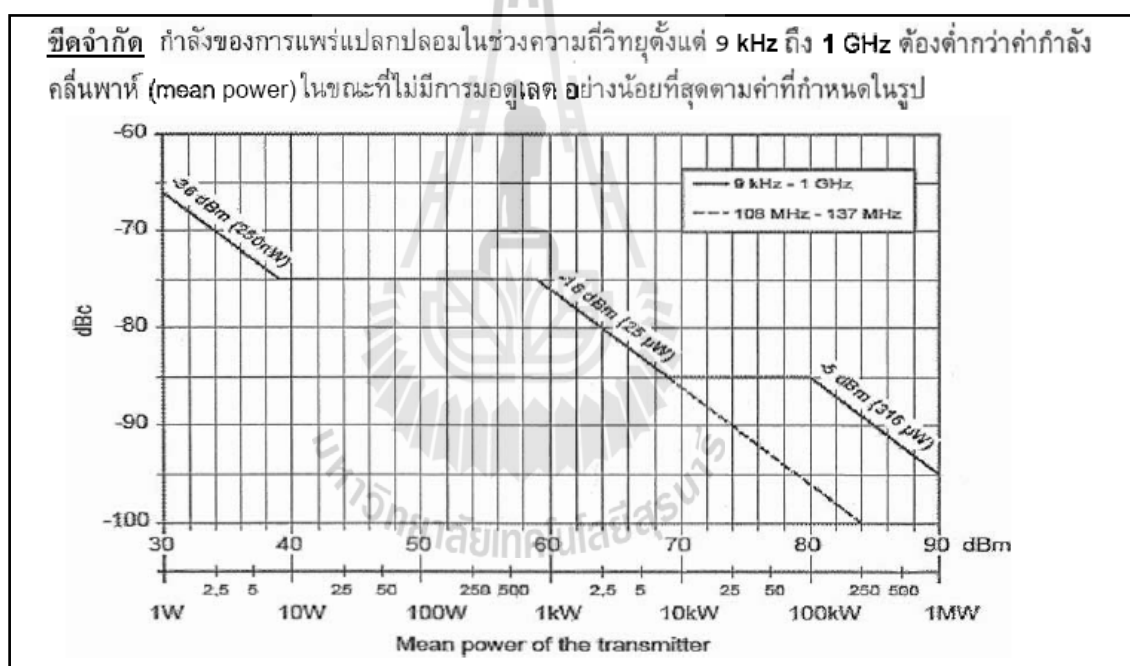
3. ลักษณะการรบกวน เครื่องส่งวิทยุกระจายเสียงที่มีลักษณะทางเทคนิคที่ไม่เหมาะสม จะก่อให้เกิดการรบกวนต่อการใช้ความถี่วิทยุในข่ายสื่อสารอื่น เรียกว่า การแพร่แปลกปลอม (conducted spurious emissions) คือ เครื่องส่งวิทยุสร้างความถี่ที่เกินกว่าที่กำหนดให้ออกอากาศ มาด้วย มี 2 ลักษณะ คือ

3.1 การแพร่ฮาร์โมนิก (harmonic emission) เป็นการสร้างความถี่ที่มีค่าเป็นทวีคูณของความถี่ที่กำหนด คือความถี่ที่สูง 2 เท่า 3 เท่า 4 เท่า 5 เท่า ของความถี่ที่กำหนดในการออกอากาศ ซึ่งเครื่องส่งวิทยุที่ไม่ได้มาตรฐานจะไปรบกวนความถี่ข่ายสื่อสารอื่นๆหรือย่านความถี่ในการส่งวิทยุ โทรทัศน์ระยะเบียบคณะกรรมการบริหารวิทยุกระจายเสียงและวิทยุโทรทัศน์ ว่าด้วยการกำหนดลักษณะพึงประสงค์ทางเทคนิคสำหรับสถานีวิทยุกระจายเสียง พ.ศ. 2520 ซึ่งกำหนด การควบคุมความแรงของสัญญาณ

3.2 คลื่นความถี่แปลกปลอม (Spurious Emissions) เครื่องส่งวิทยุจะต้องมีวงจรลดทอนกำลังของคลื่นฮาร์โมนิกที่สองและวิธีการจำกัดกำลังของคลื่นความถี่แปลกปลอมอื่น ๆ ไม่ให้มีค่าเกินจากที่กำหนด คือจะต้องมีความแรงของคลื่นความถี่แปลกปลอมส่วนที่แรงที่สุด วัดที่จุดต่อเข้าสายส่งของระบบสายอากาศต่ำกว่าความแรงของคลื่นความถี่มูล (Fundamental Frequency) ซึ่งวัดได้ที่จุดเดียวกันไม่น้อยกว่า 60 dB และค่าความแรงดังกล่าวจะต้องมีค่าไม่เกิน 1 มิลลิวัตต์

สำหรับข้อกำหนดของการส่งวิทยุชุมชน คือ คลื่นความถี่แปลกปลอมต่ำกว่าความแรงของคลื่นความถี่มูล (Fundamental Frequency) ซึ่งวัดได้ที่จุดเดียวกันไม่น้อยกว่า 45 dB (กำลังส่ง 30 วัตต์)

ดังรูปที่ 2.2



รูปที่ 2.2 แสดงขีดจำกัดการแพร่รบกวน

4. ค่าผิดพลาดทางความถี่ (frequency error) หมายถึงค่าแตกต่างระหว่างความถี่คลื่นพาห้ในขณะที่ไม่มีการมอดูเลตกับความถี่ที่ระบุ (nominal frequency) ของภาคเครื่องส่ง

ขีดจำกัด ค่าผิดพลาดทางความถี่จะต้องไม่เกิน ± 2 kHz ของคลื่นความถี่คลื่นพาห้ในขณะที่ไม่มีการมอดูเลต (สำหรับคลื่นความถี่หลัก การควบคุมความถี่ของเครื่องส่งวิทยุ “ความถี่ที่ส่งออก

อากาศ ในย่าน 87-108 MHz จะคลาดเคลื่อนจากค่าความถี่จัดสรรได้ไม่เกิน 20 ส่วนในล้านส่วนของค่าความถี่จัดสรร สมมติว่า ความถี่จัดสรร 100 MHz ดังนั้นความคลาดเคลื่อนของความถี่เครื่องส่งวิทยุ = $20 \div 1,000,000 \times 100 \times 1,000,000 = 20,000 \text{ Hz} = 20 \text{ kHz}$

5. ค่าเบี่ยงเบนทางความถี่ (frequency deviation) หมายถึงค่าแตกต่างที่มากที่สุดระหว่างความถี่ขณะใดขณะหนึ่ง (instantaneous frequency) เมื่อไม่มีการมอดูเลตกับความถี่คลื่นพาห้ขณะที่ไม่มีการมอดูเลต

ขีดจำกัด ค่าเบี่ยงเบนทางความถี่จะต้องไม่เกิน $\pm 75 \text{ kHz}$ ซึ่งการผสมสัญญาณวิทยุในระบบ FM เรียกว่าการผสมสัญญาณ 100% (Mod 100%) และอยู่ใน Necessary Bandwidth ที่กำหนด $\pm 100 \text{ KHz}$

6. ซิงโครนัสแอมพลิจูดมอดูเลต (Synchronous AM : AM due to FM) หมายถึงการเปลี่ยนแปลงแรงดันสูงสุด (peak voltage) ของส่วนประกอบกระแสสลับ (a.c. component) ที่ปรากฏทางด้านออกของเครื่องส่ง เมื่อมีการมอดูเลตที่ส่งให้เกิดการกระเพื่อมของคลื่นพาห้ คล้ายกับการมอดูเลตแอมพลิจูด

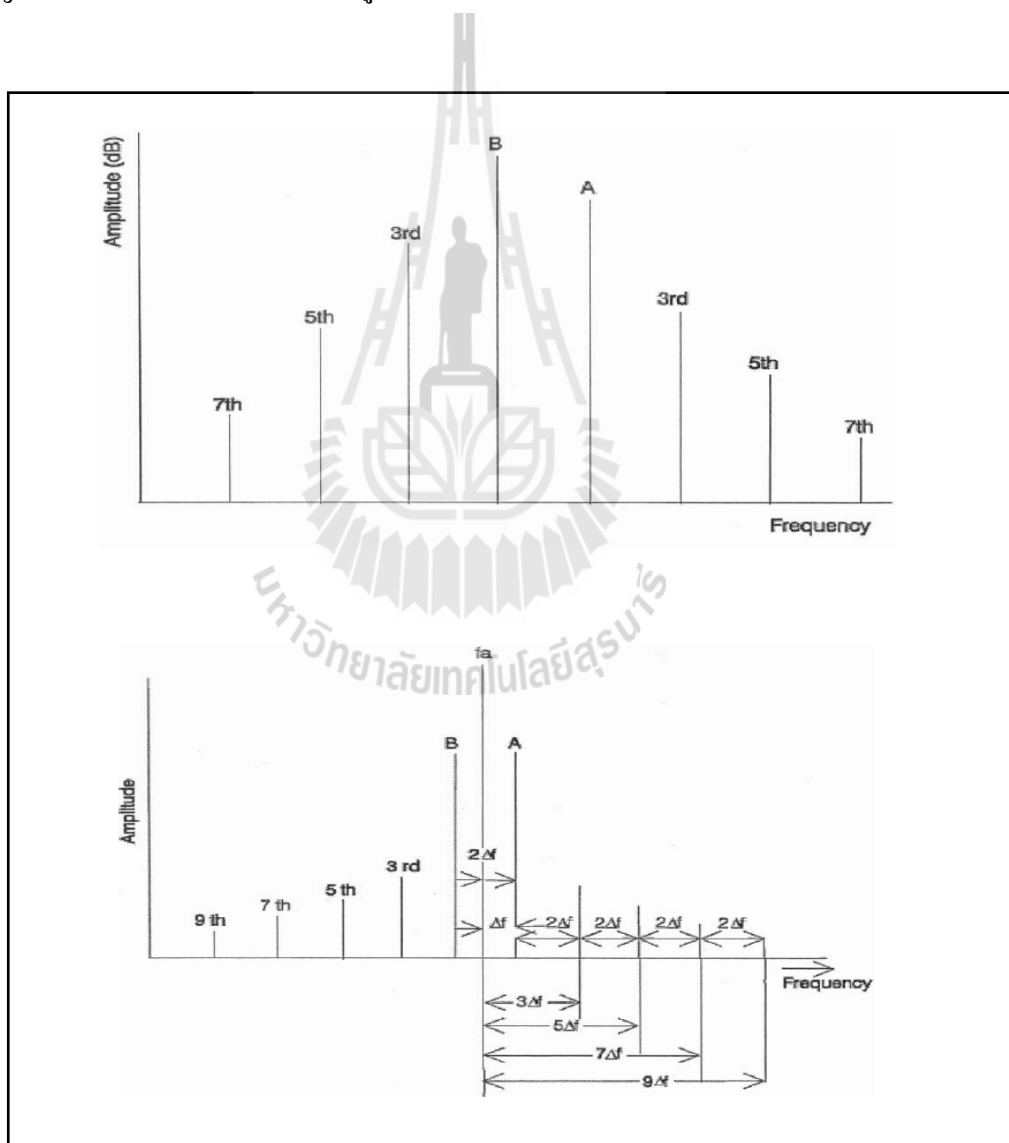
ข้อจำกัด ซิงโครนัสแอมพลิจูดมอดูเลต จะน้อยกว่าร้อยละ 2 ที่ค่าเบี่ยงเบนความถี่เท่ากับ $\pm 40 \text{ kHz}$ และความถี่เท่ากับ 500 Hz (สาเหตุสำคัญที่เกิดซิงโครนัสแอมพลิจูดมอดูเลต มาจากอุปกรณ์จ่ายไฟฟ้าไม่เรียบ ซึ่งมาตรฐานจะต้องมีการกระเพื่อมไม่เกิน 2%)

7. ข้อกำหนดด้านความปลอดภัย

- ความปลอดภัยทางไฟฟ้า (Electrical Safety Requirements)
- ความปลอดภัยเกี่ยวกับการการใช้เครื่องวิทยุคมนาคมต่อสุขภาพมนุษย์ (Radiation Exposure Requirements)

8. วิธีการทดสอบ เป็นไปตามมาตรฐานที่กำหนดข้อกำหนดด้านความปลอดภัย และวิธีการทดสอบดูรายละเอียดจากมาตรฐานและลักษณะพึงประสงค์ทางด้านเทคนิคเครื่องส่งวิทยุกระจายเสียงสำหรับชุมชนย่านความถี่วิทยุ 87.5MHz ถึง 107.5.0 MHz จะเห็นได้ว่าอุปกรณ์เครื่องส่งวิทยุชุมชนจะต้องมีการรับรองมาตรฐานเพื่อป้องกันการรบกวนและความปลอดภัยดังนั้นอุปกรณ์เครื่องส่งวิทยุชุมชนจะต้องเป็นอุปกรณ์ที่ผลิตจากต่างประเทศที่มีมาตรฐานรับรอง หรือผลิตภายในประเทศ แต่จะต้องมีหน่วยงานตรวจสอบคุณภาพรับรองคุณภาพ จึงจะนำมาใช้งานได้

ซึ่งปัจจุบันไม่การควบคุมมาตรฐานทางเทคนิค ทำให้การส่งวิทยุชุมชนมีการรบกวนสูง การรบกวนที่เกิดจากการผสมคลื่นความถี่ของเครื่องส่งวิทยุที่ออกอากาศมากกว่า 2 ความถี่ขึ้นไป (Intermodulation Product) การมอดูเลตระหว่างระหว่างกันเกิดขึ้นจากการที่เครื่องส่งจำนวนสองเครื่อง (หรือมากกว่า) ส่งสัญญาณที่ความถี่ที่มีความสัมพันธ์ในลักษณะพิเศษแล้วทำให้เกิดความถี่อื่นขึ้นมาที่ตรงกับความถี่ที่อยู่ในพิสัยของเครื่องรับที่จะรับสัญญาณนั้นซึ่งบางครั้งอาจมีระดับความแรงของสัญญาณมากกว่าระดับสัญญาณที่ต้องการเสียด้วยซ้ำ มักจะเกิดจากการที่เครื่องส่งมีการแยกสัญญาณ (isolation) ที่ไม่เพียงพอ ดังรูปที่ 2.3

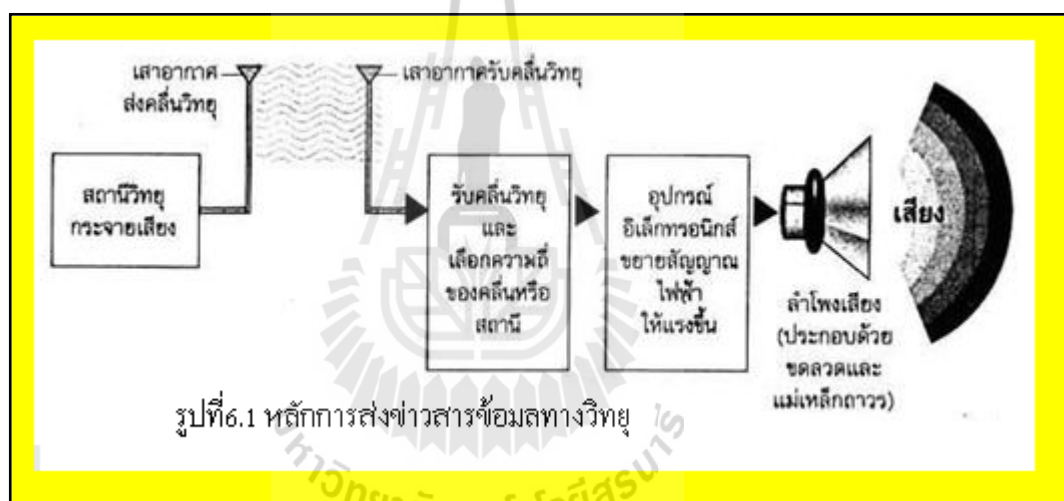


รูปที่ 2.3 Intermodulation spectrum

2.3 ระบบสื่อสารวิทยุ

การสื่อสารโทรคมนาคมนอกเหนือจากสื่อสารทางสายแล้วการสื่อสารทางคลื่นวิทยุก็นับว่าเป็นการสื่อสารที่สำคัญ ถูกนำไปใช้อย่างกว้างขวางแพร่หลาย เพราะสามารถส่งข่าวสารข้อมูลไปได้เป็นระยะทางไกลๆ ไม่สิ้นเปลืองสายช่วยแก้ปัญหาการสื่อสารทางสายในบางพื้นที่ทำไม่ได้ หรือไม่คุ้มค่าในแง่การลงทุน การดูแลรักษา และการซ่อมบำรุง ข่าวสารข้อมูลสามารถเดินทางไปได้ใน ระยะทางไกล ๆ โดยไม่เกิดการสูญหายไประหว่างการเดินทาง และสามารถส่งข่าวสารข้อมูลไปได้ครั้งละมาก ๆ ทำให้เกิดการประหยัด มีความรวดเร็ว และไม่เกิดการผิดพลาด

หลักการส่งข่าวสารข้อมูลทางคลื่นวิทยุ แสดงดังรูปที่ 2.4

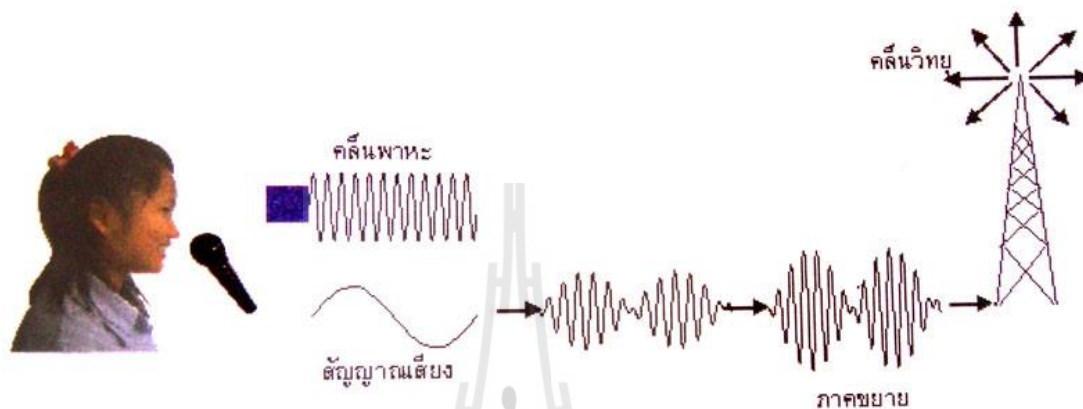


รูปที่ 2.4 แสดงหลักการส่งข่าวสารข้อมูลทางคลื่นวิทยุ

โดยการนำข่าวสารข้อมูลต่างๆ เช่นเสียงภาพ ข้อมูล และสัญญาณไฟฟ้าต่างๆ ไปเข้ากระบวนการผสมคลื่น (Modulation) กับสัญญาณคลื่นวิทยุ ส่งต่อไปให้เครื่องส่งวิทยุส่งออกอากาศแพร่กระจายคลื่นออกไปในอากาศในรูปของคลื่นแม่เหล็กไฟฟ้าทางด้านรับคลื่นไฟฟ้าส่งผ่านเข้าสายอากาศส่งต่อไปให้เครื่องรับวิทยุรับสัญญาณคลื่นเหล่านี้เข้ามา ผ่านไปเข้าขบวนการแยกคลื่น (Demodulation) แยกเอาเสียง ภาพ ข้อมูล และสัญญาณไฟฟ้าต่าง ๆ ส่งต่อไปปลายทางเพื่อใช้งาน

2.3.1 การผสมคลื่นทางความถี่หรือแบบ AM (Amplitude Modulation)

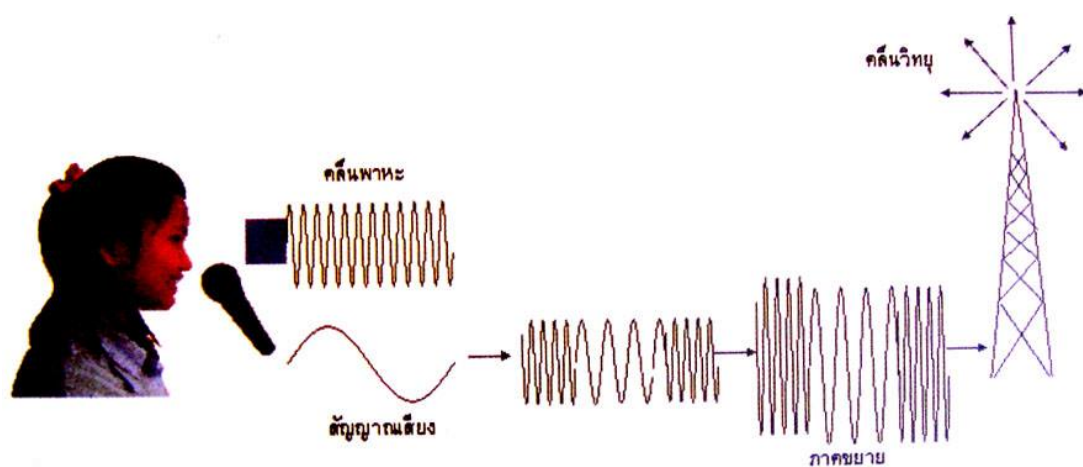
คือการนำสัญญาณเสียงหรือสัญญาณข่าวสารไปผสมกับคลื่นพาห้ความถี่สูง โดยที่สัญญาณเสียงจะไปควบคุมให้ระดับความสูง (แอมพลิจูด) ของคลื่นพาห้เกิดการเปลี่ยนแปลง แต่ความถี่ของคลื่นพาห้ยังคงเท่าเดิมไม่เปลี่ยนแปลงดังรูปที่ 2.5



รูปที่ 2.5 การผสมคลื่นทางความถี่

2.3.2 การผสมคลื่นทางความถี่หรือแบบ FM (Frequency Modulation)

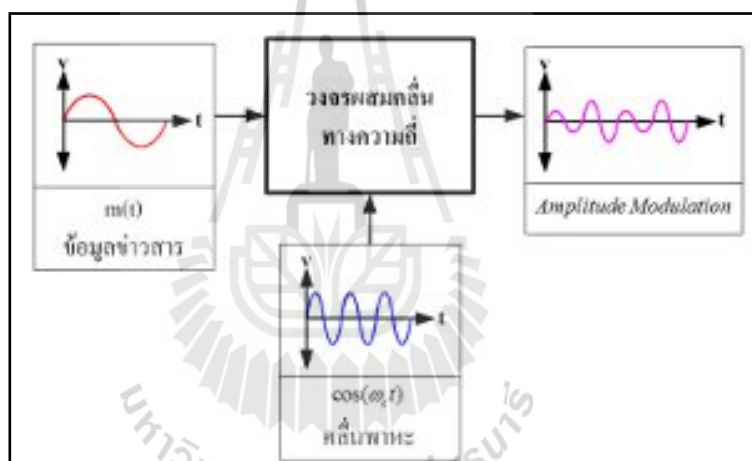
สัญญาณเอฟเอ็มที่มอดูเลตแล้วจะมีแอมพลิจูดคงที่แต่ความถี่ของสัญญาณจะเปลี่ยนแปลงไปตามแอมพลิจูดของสัญญาณข้อมูลข่าวสาร ซึ่งการมอดูเลตทางความถี่จะให้คุณภาพที่ดีกว่าการมอดูเลตทางแอมพลิจูด (AM) แต่ระบบจะมีความซับซ้อนกว่าดังรูปที่ 2.6



รูปที่ 2.6 การผสมคลื่นทางความถี่

2.3.4 การผสมคลื่นทางความถี่หรือแบบ PM (Phase Modulation)

คือการนำสัญญาณข่าวสารข้อมูลไปผสมกับคลื่นพาหะโดยเฟสของสัญญาณข่าวสารข้อมูลจะไปควบคุมกับคลื่นพาหะมีความถี่พาหะเปลี่ยนแปลงไปเพิ่มขึ้นหรือลดลงตามช่วงการเปลี่ยนแปลงของสัญญาณข่าวสารข้อมูลจากช่วงบวกเป็นช่วงลบหรือจากช่วงลบเป็นช่วงบวก คือช่วงเฟสของสัญญาณข่าวสารข้อมูลนั่นเอง โดยระดับความสูงของคลื่นพาหะยังคงเดิมไม่เปลี่ยนแปลงการผสมคลื่นแบบ (PM) นี้ก็คือการผสมคลื่นแบบ FM นั่นเอง เพียงแต่ช่วงสัญญาณข่าวสารข้อมูลที่จะไปควบคุมให้ความถี่พาหะเกิดการเปลี่ยนแปลงมีช่วงแตกต่างกัน การผสมคลื่นแบบ FM ความแรงของสัญญาณข่าวสารข้อมูลไปทำให้ความถี่พาหะเปลี่ยนแปลง การผสมคลื่นทางเฟส (PM) แสดงดังรูปที่ 2.7



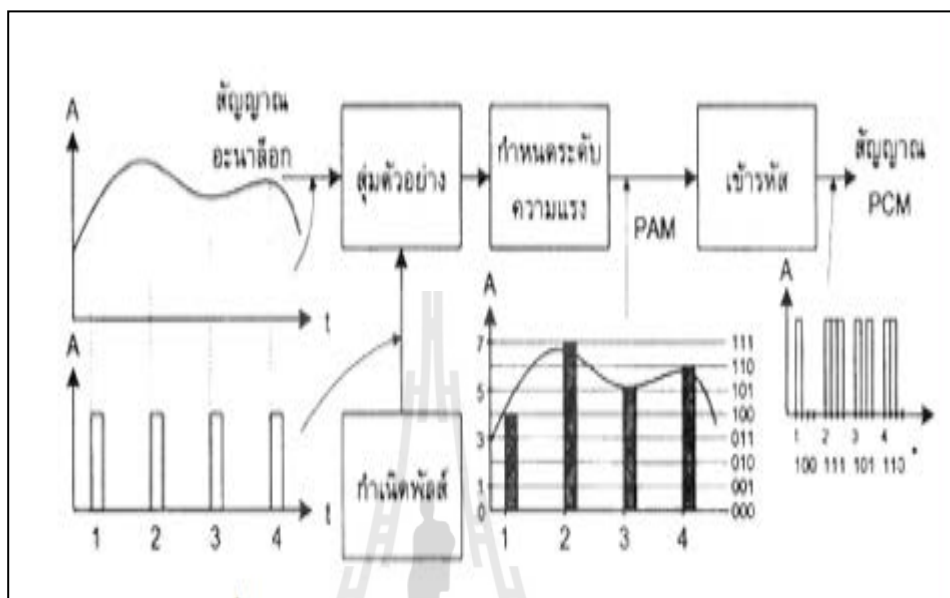
รูปที่ 2.7 ลักษณะการผสมคลื่นทางเฟส

2.3.5 การผสมสัญญาณแบบรหัสพัลส์ (PCM)

เป็นการผสมสัญญาณข่าวสารข้อมูลกับสัญญาณพัลส์ด้วยกรรมวิธีของการมัลติเพล็กซ์แบบแบ่งเวลา (TDM) เพื่อส่งข่าวสารข้อมูลแบบอะนาล็อกในลักษณะการส่งแบบดิจิทัล เรียกกรรมวิธีนี้ว่า TDM-PCM วิธีดังกล่าวนี้ถูกนำไปใช้งานอย่างแพร่หลายเพิ่มขึ้น เช่น โทรศัพท์ โทรศัพท์เคลื่อนที่ และ โทรทัศน์ เป็นต้น

หลักการทำงานของ PCM เป็นวิธีการเปลี่ยนสัญญาณอะนาล็อกให้เป็นสัญญาณดิจิทัล โดยเปลี่ยนสัญญาณอะนาล็อกไปอยู่ในรหัสเลขฐานสอง (Binary Code) การเปลี่ยนสัญญาณแบบ PCM นี้มีความสำคัญ 3 ส่วนคือการสุ่มตัวอย่าง (Sampling) การแบ่งระดับความแรงของสัญญาณ

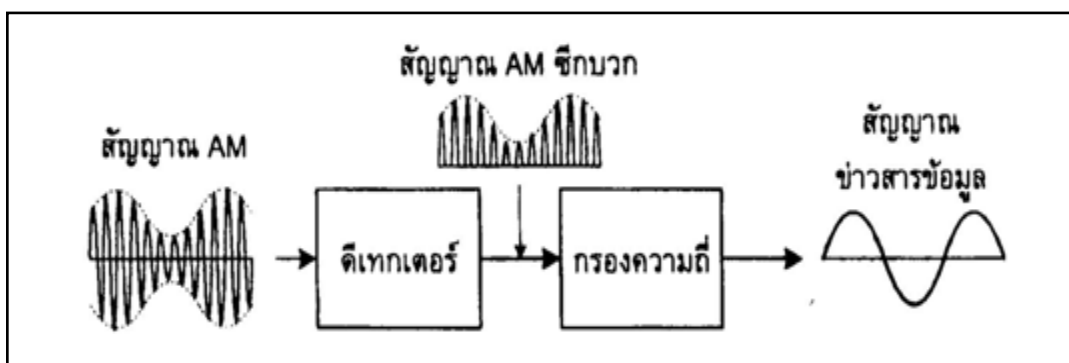
ออกเป็นระดับที่กำหนดไว้ (Quantizing) และการเข้ารหัส (Encoding) การทำงานของ PCM แสดงดังรูปที่ 2.8



รูปที่ 2.8 ลักษณะการผสมสัญญาณแบบรหัสพัลส์ (PCM)

2.3.6 การแยกคลื่นสัญญาณผสมแบบ

การแยกคลื่นสัญญาณผสมแบบ AM (AM Demodulation) สัญญาณข่าวสารข้อมูลที่ผสมกับคลื่นพาหะแบบ AM เมื่อส่งผ่านมาถึงเครื่องรับวิทยุ ก่อนที่จะส่งข่าวสารข้อมูลไปปลายทาง ต้องทำการแยกเอาข่าวสารข้อมูลออกจากคลื่นพาหะเสียก่อน โดยต้องผ่านสัญญาณ AM เพื่อกำจัดคลื่นพาหะทิ้งไป ให้เหลือเฉพาะข่าวสารข้อมูลนำไปใช้งาน ลักษณะการแยกคลื่นสัญญาณผสมแบบ AM แสดงดังรูปที่ 2.9



รูปที่ 2.9 ลักษณะการแยกคลื่นสัญญาณผสมแบบ AM

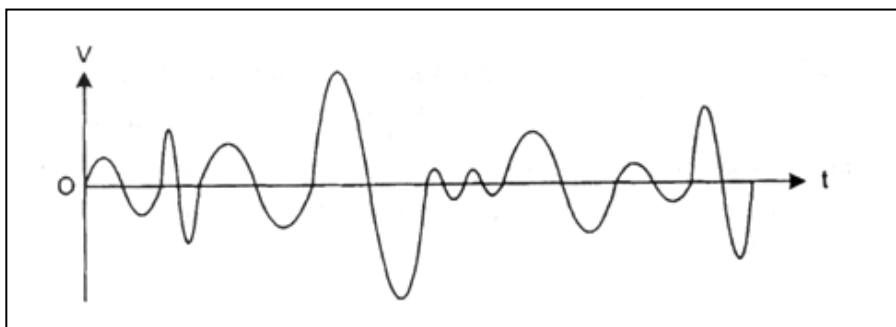
2.3.7 การแยกคลื่นสัญญาณผสมแบบ FM และ PM

การแยกคลื่นสัญญาณผสมแบบ FM และ PM (FM&PM Demodulation) สัญญาณ FM และสัญญาณ PM มีความเหมือนกันในลักษณะของคลื่นสัญญาณที่ผสมแล้วมีความถี่คลื่นพาหะเปลี่ยนแปลงไปตามสัญญาณข้อมูลที่แตกต่างกันเพียงตำแหน่งการเปลี่ยนแปลงความถี่ไปของคลื่นพาหะระหว่างสัญญาณ FM และสัญญาณ PM แตกต่างกันอยู่ที่จุดผสมดังนั้นการแยกคลื่นสัญญาณข่าวสารข้อมูลออกจากคลื่นพาหะ ในแบบ FM และแบบ PM สามารถทำงานร่วมกันได้ ในปัจจุบันการผสมคลื่นและการแยกคลื่นแบบ PM เข้ามามีบทบาทในการทำงานเพิ่มขึ้น สามารถกล่าวโดยรวมกันได้ว่าการแยกคลื่นสัญญาณผสมแบบ FM และ PM ทำได้หลายชนิดเช่น เรโซดิเทกเตอร์ (Radio Detector) ควอดราเจอร์ดิเทกเตอร์ (Quadrature Detector) และเฟสล็อกลูป (Phase Locked Loop ; PLL) เป็นต้น

2.3.8 คลื่นเสียงคลื่นวิทยุ

สัญญาณข้อมูลข่าวสารที่ใช้ในระบบสื่อสารโทรคมนาคมนั้นส่วนมากประกอบด้วยสัญญาณเสียง สัญญาณภาพ และสัญญาณข้อมูลต่างๆ สัญญาณเหล่านี้จะอยู่ในรูปของคลื่นไฟฟ้า คลื่นไฟฟ้าที่ถูกนำมาใช้งานแบ่งออกได้เป็น 2 ชนิด คือ คลื่นเสียง (Audio Wave) และคลื่นวิทยุ (Radio Wave) คลื่นทั้งสองชนิดแตกต่างกันในช่วงความยาวของคลื่น โดยคลื่นเสียงที่มีความถี่ต่ำส่วนคลื่นวิทยุมีความถี่สูง รายละเอียดของคลื่นแต่ละชนิดมีดังนี้

1. คลื่นเสียงเป็นคลื่นที่มีความถี่ต่ำอยู่ในย่านความถี่ประมาณ 20 ถึง 20,000 Hz คลื่นเสียงนี้เป็นคลื่นที่มนุษย์ทุกคนรับฟังได้ ใช้ในการสื่อสารข่าวสารและข้อมูลซึ่งกันและกัน คลื่นเสียงอาจกำเนิดขึ้นจากการเปล่งเสียงของมนุษย์ สัตว์และสิ่งมีชีวิต หรืออาจกำเนิดขึ้นจากเครื่องกำเนิดสัญญาณเสียงต่างๆ คลื่นเสียงเนื่องจากมีความถี่ต่ำจึงไม่สามารถเดินทางไปได้ไกล เกิดการจางหายของคลื่นได้ง่ายคลื่นเสียงที่มนุษย์สามารถได้ยินเกิดจากการสั่นสะเทือนของอากาศโดยรอบเคลื่อนที่ไปกระทบเยื่อแก้วหูผู้ฟังให้สั่นสะเทือนตามลักษณะดังรูปที่ 2.10



รูปที่ 2.10 ลักษณะคลื่นเสียงที่หูเราสามารถได้ยินได้

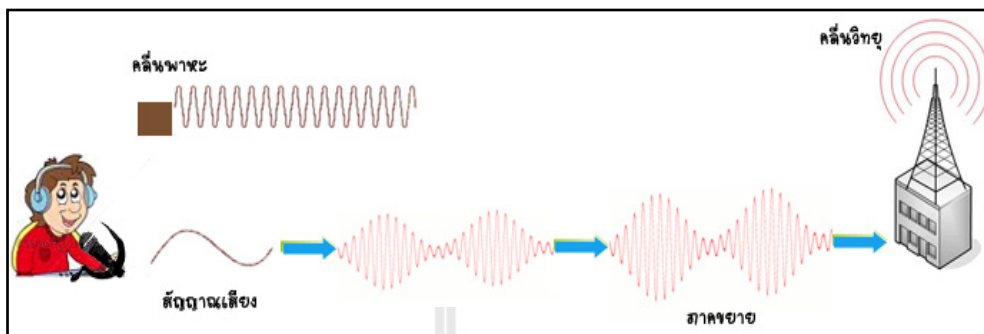
คลื่นเสียงที่มนุษย์ได้ยินจะมีระดับความถี่ของเสียงแตกต่างกันแยกออกเป็นเสียงทุ้มมีความถี่ต่ำ (Low Frequency) มีความถี่ประมาณ 20 ถึง 500 Hz เสียงกลางมีความถี่ปานกลาง (Middle Frequency) มีความถี่ประมาณ 500 ถึง 5,000 Hz และเสียงแหลมมีความถี่สูง (High Frequency) มีความถี่ประมาณ 5,000 ถึง 20,000 Hz คลื่นเสียงเดินทางไปได้ไม่ไกลการเดินทางจะได้ไกลแค่ไหนนั้นขึ้นอยู่กับความแรงของคลื่นเสียงคลื่นเสียงมีความแรงน้อยคลื่นเสียงเดินทางไปได้ไกลคลื่นเสียงมีความแรงมากขึ้นคลื่นเสียงเดินทางไปได้ไกลมากขึ้น

2.3.9 คลื่นวิทยุ

เป็นคลื่นที่มีความถี่สูงมากอยู่ในย่านความถี่ประมาณ 10 KHz (10,000 Hz) ถึง 300 GHz (300,000,000,000 Hz) ความถี่ในย่านนี้หูมนุษย์ไม่สามารถรับฟังได้คลื่นวิทยุสามารถเดินทางไปได้ไกลมากด้วยความเร็วเท่ากับคลื่นแสงเดินทางมีความเร็วโดยประมาณ 3×10^8 เมตรต่อวินาที (m/s) เพราะคลื่นวิทยุเดินทางเคลื่อนที่ไปในรูปคลื่นแม่เหล็กไฟฟ้า (Electromagnetic Wave) สามารถเดินทางผ่านไปได้ในทุกหนทุกแห่งโดยไม่ต้องอาศัยตัวกลางพาคลื่นวิทยุไปในสุญญากาศคลื่นวิทยุก็สามารถเดินทางไปได้ความถี่ของคลื่นวิทยุมีความสัมพันธ์กับความเร็วในการเคลื่อนที่ของคลื่นวิทยุและความยาวคลื่นคลื่นวิทยุมีความถี่ช่วง $10^4 - 10^9$ Hz (เฮิรตซ์) ใช้ในการสื่อสารคลื่นวิทยุมีการส่งสัญญาณ 2 ระบบคือ

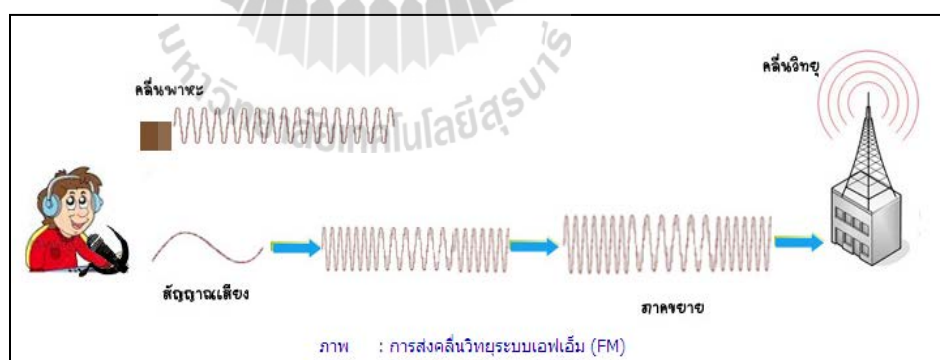
1. ระบบเอเอ็ม (A.M. = amplitude modulation) ระบบเอเอ็ม มีช่วงความถี่ 530 - 1600 kHz (กิโลเฮิรตซ์) สื่อสารโดยใช้คลื่นเสียงผสมเข้าไปกับคลื่นวิทยุเรียกว่า "คลื่นพาหะ" โดยแอมพลิจูดของคลื่นพาหะจะเปลี่ยนแปลงตามสัญญาณคลื่นเสียงในการส่งคลื่นระบบ A.M. สามารถส่งคลื่นได้ทั้งคลื่นดินเป็นคลื่นที่เคลื่อนที่ในแนวเส้นตรงขนานกับผิวโลกและคลื่นฟ้าโดยคลื่นจะไป

สะท้อนที่ชั้นบรรยากาศไอโอโนสเฟียร์แล้วสะท้อนกลับลงมา จึงไม่ต้องใช้สายอากาศตั้งสูงรับดังแสดงในรูปที่ 2.11



รูปที่ 2.11 การส่งคลื่นวิทยุเอเอ็ม

2.ระบบเอฟเอ็ม (F.M. = frequency modulation) ระบบเอฟเอ็ม มีช่วงความถี่ 88 - 108 MHz (เมกะเฮิรตซ์) สื่อสารโดยใช้คลื่นเสียงผสมเข้ากับคลื่นพาหะโดยความถี่ของคลื่นพาหะจะเปลี่ยนแปลงตามสัญญาณคลื่นเสียงในการส่งคลื่นระบบ F.M. ส่งคลื่นได้เฉพาะคลื่นดินอย่างเดียว ถ้าต้องการส่งให้กลุ่มพื้นที่ต้องมีสถานีถ่ายทอดและเครื่องรับต้องตั้งเสาอากาศดังแสดงในรูปที่ 2.12

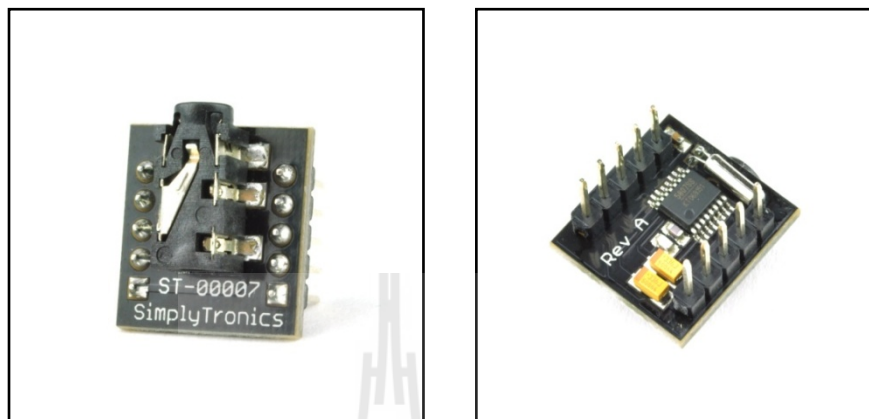


รูปที่ 2.12 การส่งคลื่นระบบเอฟเอ็ม

2.4 โมดูลเครื่องรับวิทยุ FM สำหรับไมโครคอนโทรลเลอร์

เป็นอุปกรณ์ต่อพ่วงกับไมโครคอนโทรลเลอร์เพื่อช่วยให้สามารถสร้างเครื่องรับวิทยุ FM สเตอริโอในยุคใหม่ที่ควบคุมการทำงานด้วยไมโครคอนโทรลเลอร์หัวใจหลักของโมดูลนี้คือชิปรับคลื่น วิทยุ FM เบอร์ RDA5707SS

จาก RDA Microelectronics คือ โมดูลมีสายอากาศในตัวพร้อมแจ็คหูฟังเพียงจ่ายไฟและส่งคำสั่งให้ ถูกต้องก็จะได้รับฟังรายการวิทยุตามต้องการดังรูปที่ 2.13



รูปที่ 2.13 FM Module เบอร์ RDA5707SS ด้านหน้า-ด้านหลัง

2.4.1 คุณสมบัติที่สำคัญมีดังนี้

- ติดต่อกับไมโครคอนโทรลเลอร์ผ่านระบบบัส I²C ใช้กับไมโครคอนโทรลเลอร์ทุกตระกูล
- รองรับความถี่ 76 ถึง 108 MHz นั่นคือเต็มย่านความถี่ของวิทยุ FM
- ไม่ต้องใช้สายอากาศภายนอก
- ไฟเลี้ยงย่านกว้าง +2.7 ถึง +5.5V ต้องการกระแสไฟฟ้า 25 mA
- ตัวโมดูลได้รับการจัดการให้อยู่ในรูปตัวถัง DIP 10 ขา ทำให้ง่ายต่อการนำไปใช้งานจะเสียบบอร์ดเพื่อต่อทดลองแบบชั่วคราวก็สะดวกหรือจะติดตั้งลงบนแผ่นวงจรพิมพ์ก็ทำได้ง่าย ใช้กับแผ่นวงจรพิมพ์เอนกประสงค์ได้สบายๆ
- ขนาด 0.70 x 0.65 x 0.34 นิ้ว (1.78 x 1.65 x 0.87 ซม.)
- อุณหภูมิใช้งาน 0 ถึง 50 องศาเซลเซียส

ขาที่	ชื่อขา	การกำหนด
1	SDA	ขาข้อมูลอนุกรมของระบบบัส I ² C
2	SCL	ขาสัญญาณนาฬิกาอนุกรมของระบบบัส I ² C
3	ANT	ขาต่อสายอากาศภายนอกเพิ่มเติม
4	ROUT	ขาเอาต์พุตเสียงช่องขวา
5	LOUT	ขาเอาต์พุตเสียงช่องซ้าย
6	GPIO1	ขาพอร์ตอินพุตเอาต์พุตบนพินประสงค์ 1
7	GPIO2	ขาพอร์ตอินพุตเอาต์พุตบนพินประสงค์ 2
8	GPIO3	ขาพอร์ตอินพุตเอาต์พุตบนพินประสงค์ 3
9	VDD	ขาต่อไฟเลี้ยง +2.7 ถึง +5.5V
10	GND	ขาต่อกราวด์

รูปที่ 2.14 หน้าตาและการจัดขาและหน้าที่ของขาใช้งานทั้งหมดของโมดูลวิทยุ FM

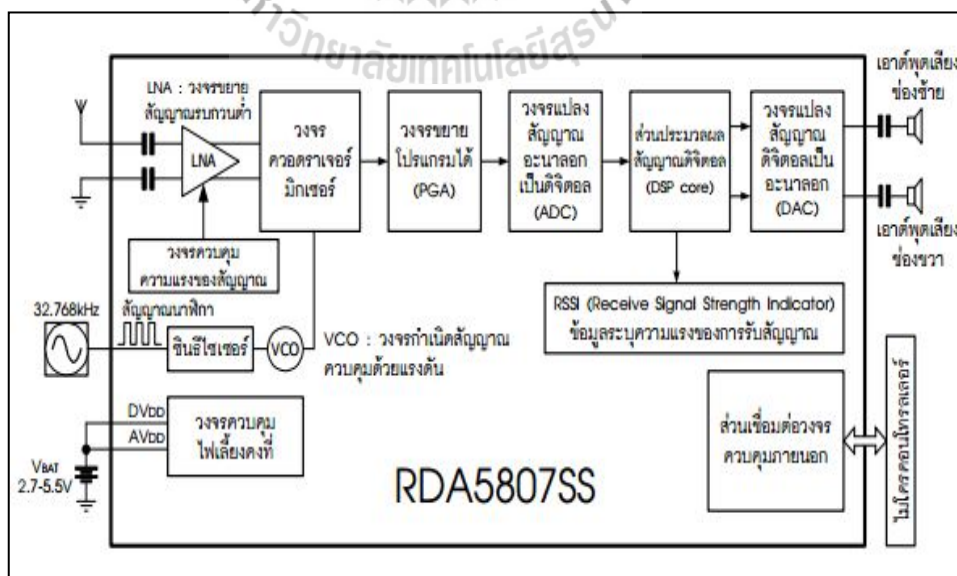
2.4.2 การทำงาน

1. ส่วนรับวิทยุ FM สัญญาณคลื่นวิทยุจะถูกเหนี่ยวนำเข้าสู่วงจรด้วยสายอากาศ จากนั้นวงจรขยายสัญญาณรบกวนต่ำจะทำการขยายสัญญาณให้แรงขึ้นเพื่อป้อนเข้าสู่วงจรควอดราเจอร์มิกเซอร์โดยมีวงจรควบคุมความแรงของสัญญาณช่วยควบคุมไม่ให้สัญญาณมีความแรงเกินไปจนเกิดความเพี้ยนวงจรควอดราเจอร์มิกเซอร์ทำหน้าที่ในการแปลงสัญญาณคลื่นวิทยุเป็นสัญญาณความถี่กลางหรือ IF (Intermediate Frequency) เพื่อส่งต่อไปยังวงจรขยายแบบโปรแกรมได้หรือ PGA เพื่อกำหนดอัตราความแรงของสัญญาณ IF ให้เหมาะสมจากนั้นจึงส่งต่อไปยังวงจรแปลงสัญญาณ อนาลอกเป็นดิจิตอลเพื่อให้ไมโครคอนโทรลเลอร์ภายนอกสามารถส่งข้อมูลเข้ามาควบคุมได้โดยผ่านทางส่วนประมวลผลสัญญาณดิจิตอลที่ส่วนประมวลผลสัญญาณดิจิตอลจะทำหน้าที่เลือกช่องความถี่ของวิทยุ FM คีมอดูเลชั่น ถอดรหัส FM สเตอริโอมัลติเพล็กซ์จนได้เป็นข้อมูลเสียงทั้งช่องซ้ายและขวาก่อนส่งต่อไปยังวงจรแปลงสัญญาณดิจิตอลเป็นอนาลอก ทำการแปลงเป็นเสียงเพื่อขับออกหูฟังหรือต่อเข้ากับวงจรขยายกำลังต่อไปโดยผู้ใช้งานสามารถปรับระดับสัญญาณได้ด้วยผ่านการเขียนโปรแกรมติดต่อกับรีจิสเตอร์ควบคุมการทำงาน

2. ซินธิไซเซอร์ในส่วนนี้ทำหน้าที่ในการสร้างสัญญาณนาฬิกาสำหรับกำหนดจังหวะการทำงานของวงจรควอดราเจอร์มิกเซอร์โดยเลือกได้ 7 ค่าคือ 32.768 KHz, 12M, 24M, 13M, 26M, 19.2M, 38.4MHz เสถียรภาพของสัญญาณวิทยุ FM ที่รับได้จะขึ้นกับส่วนนี้เป็นสำคัญด้วยการใช้วงจรซินธิไซเซอร์จะทำให้ความถี่ที่ใช้ในการผสมสัญญาณเพื่อแปลงคลื่นวิทยุเป็นคลื่นความถี่กลางหรือ IF มีความแน่นอนสูงทำให้ความสามารถในการรับสัญญาณของโมดูลรับวิทยุ FM มีเสถียรภาพ

3. ภาคจ่ายไฟ ในตัวไอซี RDA5807SS มี วงจรควบคุมไฟเลี้ยงคงที่แบบแรงดันตกคร่อมต่ำจึงทำให้สามารถใช้งานกับไฟเลี้ยงในย่านกลางตั้งแต่ +2.7 ถึง +5.5V จึงทำให้ใช้งานกับไมโครคอนโทรลเลอร์ได้ทุกตระกูลที่ใช้ไฟเลี้ยงทั้ง +3.3V และ +5V

4. ส่วนเชื่อมต่อวงจรควบคุมภายนอกทำหน้าที่จัดการสัญญาณต่างๆที่จำเป็นสำหรับติดต่อกับไมโครคอนโทรลเลอร์ภายนอกผ่านทางบัส I²C และ I²S โดยทั่วไปมักใช้บัส I2C มากกว่าเพราะใช้ได้กับไมโครคอนโทรลเลอร์ได้ทุกตระกูลหากเป็นบัส I2S มีไมโครคอนโทรลเลอร์สมัยใหม่บางตระกูลเท่านั้นที่มี โมดูลการติดต่อผ่านบัส I2S สัญญาณจากไมโครคอนโทรลเลอร์จะถูกส่งผ่านไปยังส่วนประมวลสัญญาณดิจิทัล, ซินธิไซเซอร์จะแปลงสัญญาณดิจิทัลเป็นอนาลอกเพื่อติดต่อกับรีจิสเตอร์ที่ทำหน้าที่ควบคุมการทำงานของส่วนต่าง ๆ นั้นดังรูปที่ 2.15



รูปที่ 2.15 โค้ดแแกรมการทำงานภายในของไอซี RDA5807SS ที่ใช้ในโมดูลวิทยุ FM

2.4.3 รีจิสเตอร์ควบคุมการทำงานของโมดูลวิทยุเอฟเอ็ม

เนื่องจากโมดูลรับวิทยุ FM ต้องทำงานร่วมกับไมโครคอนโทรลเลอร์ดังนั้นจึงต้องมีรีจิสเตอร์สำหรับควบคุมการทำงานซึ่งจะรับข้อมูลคำสั่งจากไมโครคอนโทรลเลอร์เพื่อส่งต่อไปยังวงจรประมวลผลสัญญาณดิจิทัลทำการประมวลผลและนำไปควบคุมการทำงานของวงจรต่างๆ ภายในไอซี RDA5807SS เพื่อให้ได้สัญญาณเสียงจากการรับคลื่นวิทยุ FM ออกมาให้ได้ฟังกันในที่สุรหัทสแอดเดรสประจำตัวแบบ 7 บิต ตามข้อกำหนดของการเป็นอุปกรณ์ระบบบัส I2C ของโมดูลรับวิทยุ FM คือ 0010000 สำหรับบิตสุดท้าย (บิต 0) จะเป็นตัวกำหนดว่าต้องการอ่านหรือเขียนข้อมูลกับโมดูลนี้ หากต้องการอ่านค่าแอดเดรสจะเป็น 00100001 หรือ 0x21 หากต้องการเขียนข้อมูลหรือคำสั่งแอดเดรสของโมดูลจะเปลี่ยนเป็น 00100000 หรือ 0x20 โดยคำสั่งที่ใช้ควบคุมประกอบด้วยคำสั่งเพิ่ม-ลดระดับเสียง, คำสั่งค้นหาสถานี คำสั่งปรับเลื่อนค่าความถี่ เป็นต้น รีจิสเตอร์ที่ใช้กำหนดการทำงานของโมดูลรับวิทยุ FM และเข้าถึงได้ด้วยผู้ใช้งานมีขนาด 16 บิต จำนวน 5 ตัว คือรีจิสเตอร์ 0x02, 0x03, 0x04, 0x05 และ 0x06 ดังมี ข้อมูลโดยสรุปแสดงในตารางที่

1-6



รีจิสเตอร์ 0x02 : รีจิสเตอร์ตั้งค่าการทํางานหลัก			
บิตที่	ชื่อ	การทํางาน	ค่าเริ่มต้น
15	DHIZ	บิตควบคุมสถานะขาเอาต์พุตสัญญาณเสียง "0" - อิมพีแดนซ์สูง "1" - ทํางานปกติ	0
14	DMUTE	บิตควบคุมการตัดสัญญาณเสียง "0" - ตัดเสียง "1" - ทํางานปกติ	0
13	MONO	บิตเลือกระบบเสียง "0" - สเตอริโอมีลติเพล็กซ์ "1" - โมโน	0
12	BASS	บิตเพิ่มเสียงทุ้ม "0" - ไม่เพิ่ม "1" - เพิ่ม	0
11	RESERVED	สำรองไว้ - กำหนดเป็น "0"	0
10	CLK32_INPUT_ENABLE	บิตเอ็นเอเบิลวงจรสัญญาณนาฬิกาของชิป "0" - เอ็นเอเบิล "1" - ดิสเอเบิล	0
9	SEEKUP	บิตเลือกโหมดการค้นหาสถานี "0" - ค้นหาสถานีความถี่ที่ต่ำลง (seek down) "1" - ค้นหาสถานีความถี่สูงขึ้น (seek up)	0
8	SEEK	บิตเอ็นเอเบิลการค้นหาสถานี "0" - เอ็นเอเบิล "1" - ดิสเอเบิล	0
7	RESERVED	สำรองไว้ - กำหนดเป็น "0"	0
6 ถึง 4	CLK_MODE	บิตเลือกความถี่สัญญาณนาฬิกา ต้องมีค่า = 000 เพื่อใช้ความถี่ 32.768kHz	000
3 และ 2	RESERVED	สำรองไว้ - กำหนดเป็น "0"	00
1	SOFT_RESET	บิตรีเซ็ต กำหนดเป็น "1" เมื่อต้องการรีเซ็ตไอซี	0
0	ENABLE	บิตเอ็นเอเบิลการทํางานของโมดูล "0" - ดิสเอเบิล "1" - เอ็นเอเบิล	0

ตารางที่ 1 แสดงข้อมูลของรีจิสเตอร์ 0x02

รีจิสเตอร์ 0x04 : รีจิสเตอร์ตัวค่าระบบ			
บิตที่	ชื่อ	การทำงาน	ค่าเริ่มต้น
15	RESERVED	สำรองไว้ - กำหนดเป็น '0'	0
14	STCEIN	บิตเอ็นแอสเบิลการอินเทอร์รัปต์เมื่อการค้นหาลงานเสร็จสิ้น โดยจะส่งพัลส์ '0' ออกไปทางขา GPIO2 '0' - ดิสเอบิล '1' - เอ็นแอสเบิล	0
13 และ 12	RESERVED	สำรองไว้ - กำหนดเป็น '0'	00
11	DE	บิตเลือกค่าคาบเวลาของการดีเิมฟ้าชิส '0' - 75µs '1' - 50µs	0
10	AGCD	บิตดิสเอบิลการควบคุมอัตราขยายอัตโนมัติ '0' - เอ็นแอบิล AGC '1' - ดิสเอบิล AGC	0
9 ถึง 7	RESERVED	สำรองไว้ - กำหนดเป็น '0'	000
6	I2S_ENABLED	บิตเอ็นแอสเบิลการติดต่อผ่านบัส I ² S '0' - ดิสเอบิล '1' - เอ็นแอบิล	0
5 และ 4	GPIO3 (1:0)	บิตกำหนดการทำงานของขา GPIO3 '00' - อิมพีแดนซ์สูง '01' - แสดงสถานะโมโน/สเตอริโอ '10' - เอาต์พุตชจิก '0' '11' - เอาต์พุตชจิก '1'	00
3 และ 2	GPIO2 (1:0)	บิตกำหนดการทำงานของขา GPIO2 '00' - อิมพีแดนซ์สูง '01' - เอาต์พุตอินเทอร์รัปต์ '10' - เอาต์พุตชจิก '0' '11' - เอาต์พุตชจิก '1'	00
1 และ 0	GPIO1 (1:0)	บิตกำหนดการทำงานของขา GPIO1 '00' - อิมพีแดนซ์สูง '01' - ไม่ใช้งาน '10' - เอาต์พุตชจิก '0' '11' - เอาต์พุตชจิก '1'	00

ตารางที่ 3 ข้อมูลของรีจิสเตอร์ 0x04

รีจิสเตอร์ 0x0B : รีจิสเตอร์สถานะ RSSI		
บิต	ชื่อ	การทำงาน
15 ถึง 9	RSSI (6:0)	บิตแสดงค่าระดับความแรงของสัญญาณ (เป็นแบบลอการิทึม) '00000' - ระดับความแรงต่ำสุด '11111' - ระดับความแรงสูงสุด
8	FM_TRUE	บิตแสดงสถานะของช่องความถี่ '0' - ช่องความถี่ปัจจุบันไม่ใช่สถานี '1' - ช่องความถี่ปัจจุบันเป็นสถานี
7	FM_READY	บิตแสดงสถานะของการค้นหาสถานี '0' - ยังค้นหาอยู่ '1' - ค้นหาเสร็จแล้ว พร้อมให้อ่านค่า
6 ถึง 0	ไม่ใช้งาน ทำให้เป็น 0 หลังจากอ่านค่าแล้ว	

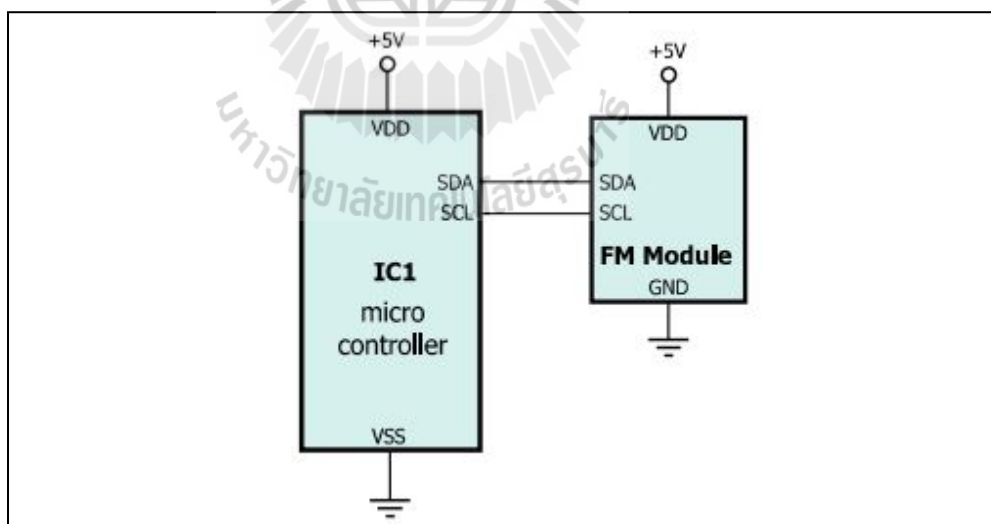
รีจิสเตอร์ 0x0C : รีจิสเตอร์สถานะการรับคลื่นจากสถานีวิทยุ		
บิต	ชื่อ	การทำงาน
15	RESERVED	สำรองไว้ - อ่านค่าเป็น '0'
14	STC	บิตแจ้งสถานะการค้นหาสถานี '0' - ยังไม่เสร็จ '1' - ค้นหาเสร็จแล้ว
13	SF	บิตแจ้งสถานะสถานี '0' - พบสถานี '1' - ไม่พบสถานีวิทยุ
12 และ 11	RESERVED	สำรองไว้ - กำหนดเป็น '0'
10	ST	บิตแจ้งสถานะสเตอริโอ/โมโน '0' - โมโน '1' - สเตอริโอ
9 ถึง 0	READCHAN (9:0)	บิตแสดงค่าความถี่ของสถานีวิทยุ ถ้าบิต BAND (ในรีจิสเตอร์ 0x03) = '0' ความถี่ = (ช่วงความถี่ในหน่วย kHz (กำหนดจากบิต SPACE ในรีจิสเตอร์ 0x03) x READCHAN) + 87MHz ถ้าบิต BAND (ในรีจิสเตอร์ 0x03) = '1' ความถี่ = (ช่วงความถี่ในหน่วย kHz (กำหนดจากบิต SPACE ในรีจิสเตอร์ 0x03) x READCHAN) + 76MHz

ตารางที่ 6 ข้อมูลของรีจิสเตอร์ 0x0A และ รีจิสเตอร์ 0x0B

ดังนั้นในการเขียนโปรแกรมเพื่อติดต่อกับโมดูลรับวิทยุ FM นี้ จะต้องเริ่มจากติดต่อกับตัวอุปกรณ์ให้ได้ก่อนด้วยการใช้ ☐ รหัสแอดเดรสประจำตัวของโมดูลรับวิทยุ FM นี้ก่อน เมื่อมีการตอบสนองแล้วจึงทำการติดต่อกับปริจิสเตอร์ทั้ง 5 ตัวเพื่อควบคุมการทำงานต่อไป

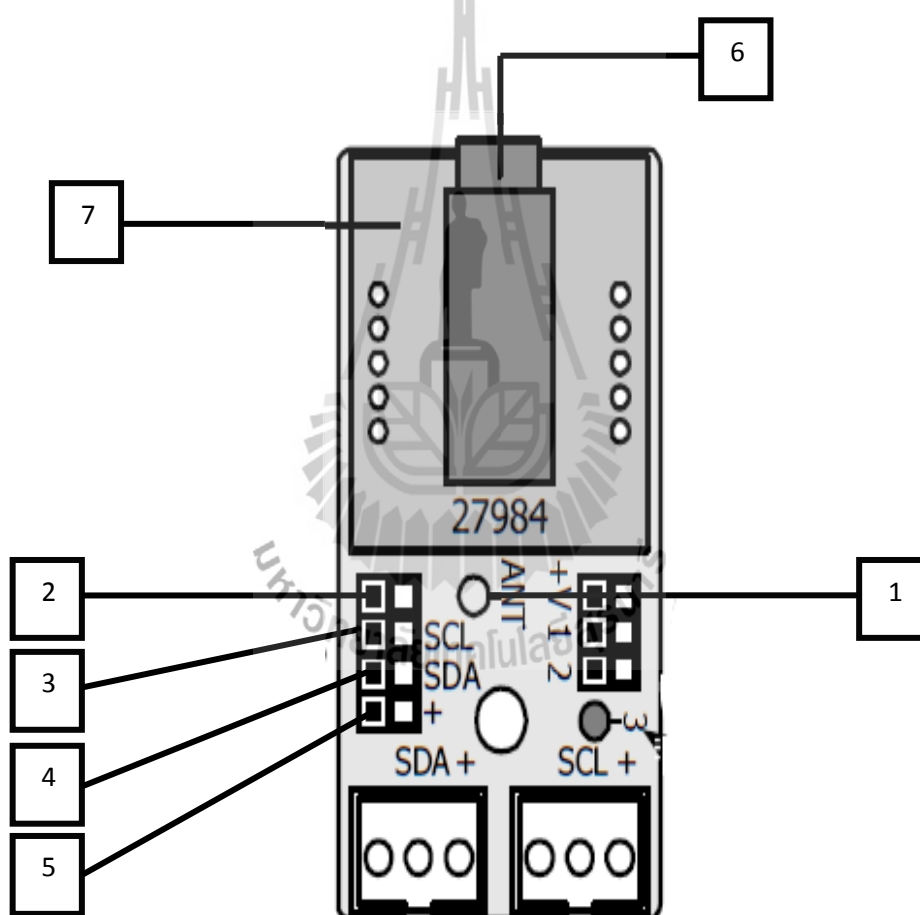
2.4.4 การติดต่อกับบอร์ดไมโครคอนโทรลเลอร์

เนื่องจากโมดูลวิทยุ FM ต ☐ องการการติดต่อผ่านระบบบัส I²C ซึ่งใช้สายสัญญาณ 2 เส้น คือ SDA และ SCL ดังนั้นในการติดต่อกับไมโครคอนโทรลเลอร์ ☐ จึงใช้ขาพอร์ ☐ ตเพียง 2 เส้น โดยไม่ต้องต่อตัวต้านทานพูลอัพที่ขาสัญญาณ SDA และ SCL เนื่องจากมีการต่อตัวต้านทานนี้ไว้ภายในโมดูลแล้ววงจรใช้งานเบื้องต้น ☐ นจึง ☐ ยามากและมีอุปกรณ์ ☐ ต่อภายนอกน้อยหากใช้ ☐ ไมโครคอนโทรลเลอร์ที่มีวงจรกำเนิดสัญญาณนาฬิกาอยู่ภายในตัวดังรูปที่ 19 ด้วยความสะดวกและ ☐ งานได้ ☐ ไม่ ☐ ยากทำให้ความคิดในการในสร้างโครงงานที่เกี่ยวกับเครื่องรับวิทยุ FM เอาไว้ ☐ งานเองทำได้ ☐ ง ☐ ยขึ้นเพียงมีความรู้ด้านการเขียนโปรแกรมไมโครคอนโทรลเลอร์ ☐ และการต่อ ☐ วงจรไม่ ☐ ต้องปรับแต่งขดลวดหรือสรั ☐ วงจร FM ดิมอดูเลเตอร์ ☐ วงจรถอดรหัส FM สเตอริโอมีัลติเพล็กซ์ ☐ ให้ ☐ อยู่ ☐ ยากอีกต่อไป



รูปที่ 19 ตัวอย่างวงจรเชื่อมต่อกับไมโครคอนโทรลเลอร์อย่าง ☐ ยที่สุดเพื่อใช้ ☐ งานโมดูลวิทยุ FM

2.4.5 การใช้งานของ FM Radio Module มีส่วนประกอบหลักๆดังนี้



รูปภาพที่ 20 ภาพแสดงส่วนประกอบของ FM Radio Module

FM Radio Module

หมายเลข 1 สายอากาศ

หมายเลข 2 สาย GND

หมายเลข 3 จุดต่อ I²C Bus แบบ IDC

หมายเลข 4 จุดต่อ I²C Bus แบบ IDC

หมายเลข 5 ไฟเลี้ยง 5 โวลต์

หมายเลข 6 สเตอริโอขนาด 3.5 mm

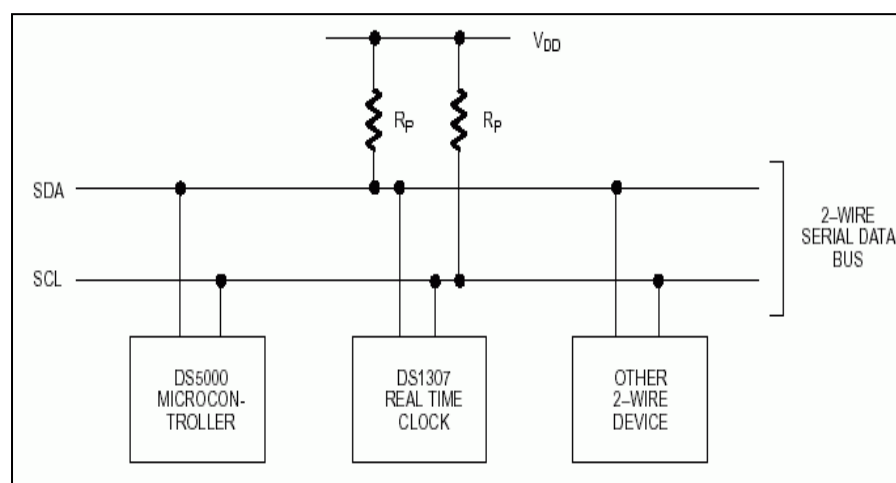
หมายเลข 7 โมดูลวิทยุ FM

2.5 การเชื่อมต่ออุปกรณ์แบบ I²C

I²C Bus ย่อมาจาก Inter Integrate Circuit Bus (IIC) นิยมเรียกสั้นๆว่า BUS (ไอ-แอสคว-ซี-บัส) เป็นการสื่อสารอนุกรมแบบซิงโครนัส (Synchronous) เพื่อใช้ติดต่อสื่อสารระหว่างไมโครคอนโทรลเลอร์ (MCU) กับอุปกรณ์ภายนอกซึ่งถูกพัฒนาขึ้นโดยบริษัท Philips Semiconductors โดยใช้สายสัญญาณเพียง 2 เส้นเท่านั้น คือ serial data (SDA) และสาย serial clock (SCL) ซึ่งสามารถเชื่อมต่ออุปกรณ์ จำนวนหลายๆตัว เข้าด้วยกันได้ทำให้ MCU ใช้พอร์ตเพียง 2 พอร์ตเท่านั้น

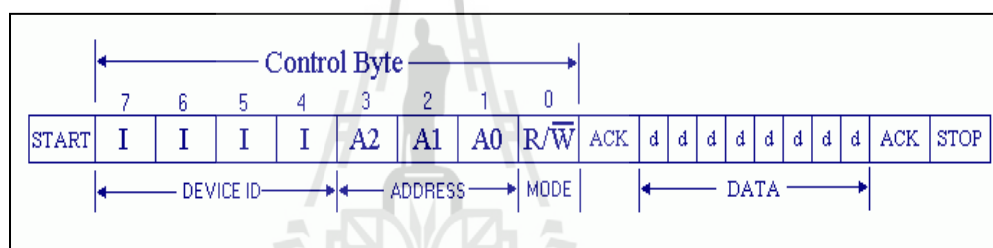
2.5.1 การเชื่อมต่ออุปกรณ์แบบ I²C BUS

I²C BUS ใช้สายสัญญาณ 2 เส้น คือ SCL ,SDA สำหรับติดกับอุปกรณ์แบบ 2 ทิศทาง โดยที่สายสัญญาณทั้ง 2 จะต้องต่อกับตัวต้านทานแบบ pull up 2-10K เนื่องจากเอาต์พุตมีลักษณะเป็นแบบ Open Darin หรือเป็นแบบ Open Collector เพื่อให้เอาต์พุตเชื่อมต่อกันได้หลายตัว



รูปที่ 21 ลักษณะการเชื่อมต่ออุปกรณ์แบบ I²C BUS

2.5.2 การเขียน-อ่านข้อมูลกับอุปกรณ์แบบ I²C BUS



รูปที่ 22 รูปแบบการเขียน/อ่านข้อมูลแบบ I²C BUS

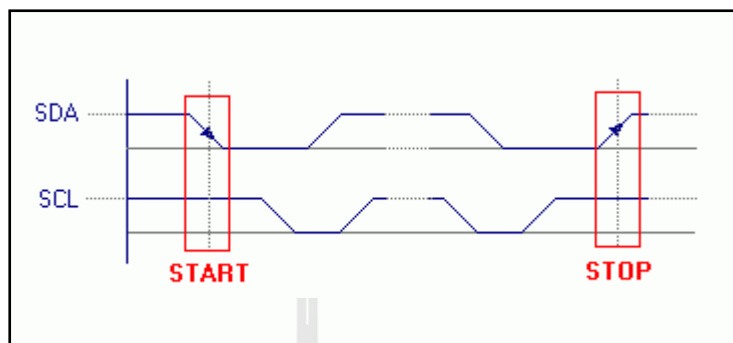
2.5.3 การรับ-ส่งข้อมูลแบบ I²C BUS MCU จะเริ่มต้นการส่งข้อมูลด้วยการ

- ส่งสถานะเริ่มต้น (START Conditions) เพื่อแสดงการขอใช้บัส
- แล้วตามด้วย รหัสควบคุม (Control Byte) ซึ่งประกอบ ด้วยรหัส ประจำตัวอุปกรณ์ Device ID ,Device Address ,และ Mode ในการเขียนหรืออ่านข้อมูล
- เมื่ออุปกรณ์ รับทราบว่า MCU ต้องการ จะติดต่อด้วยก็ต้องส่งสถานะรับรู้ (Acknowledge) หรือแจ้งให้ MCU รับรู้ว่าข้อมูลที่ได้ส่งมามีความถูกต้อง
- และเมื่อสิ้นสุดการส่งข้อมูล MCU จะต้องส่ง สถานะสิ้นสุด (STOP Conditions) เพื่อบอกกับอุปกรณ์ว่า สิ้นสุดการใช้บัส

2.5.4 การกำหนดสถานะเริ่มต้นและสถานะสิ้นสุดของ I²C BUS

(START

and STOP Conditions)



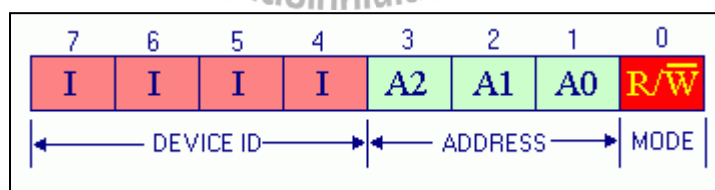
รูปที่ 22 I²C BUS START and STOP Conditions

สถานะบัสว่างคือเมื่อบัสไม่ได้ถูกใช้งาน ทั้ง SCL และ SDA จะเป็น 1 ทั้งคู่

ลักษณะการกำหนดสถานะเริ่มต้นและสถานะสิ้นสุดของ BUS

- เมื่อต้องการส่งข้อมูล MCU จะต้องส่งสถานะเริ่มต้น (START Conditions) คือให้ SDA เปลี่ยนจาก 1 มาเป็น 0 ในขณะที่ SCL มีค่าเป็น 1
- เมื่อสิ้นสุดการการใช้บัส MCU จะต้องส่งสถานะสิ้นสุด (STOP Conditions) คือให้ SDA เปลี่ยนจาก 0 มาเป็น 1 ในขณะที่ SCL มีค่าเป็น 1

2.5.5 รหัสควบคุมของ I²C BUS (Control Byte)

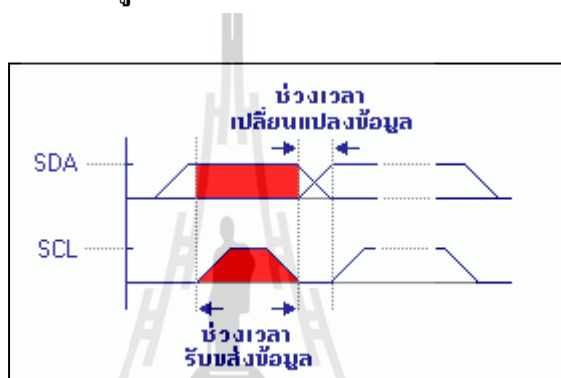


รูปที่ 23 I²C BUS (Control Byte)

รหัสควบคุมของ I²C BUS ประกอบด้วยรหัสประจำตัวของอุปกรณ์ (Device ID) ประกอบด้วยบิต 1-7 และบิต 0 เป็นบิตควบคุมการเขียนอ่าน

- รหัสประจำตัวของอุปกรณ์ ประกอบด้วยรหัสประจำตัวจากผู้ผลิต Product ID 4 บิต (บิต 4-7) ที่เปลี่ยนแปลงแก้ไขไม่ได้และ Device Address 3 บิต (บิต 1-3) ซึ่งผู้ใช้สามารถ กำหนด เองได้ รวมแล้วเป็นรหัส 7 บิต ใช้ระบุตัวอุปกรณ์ ที่ต่ออยู่บนบัส จะมีค่าซ้ำกันไม่ได้
- บิตควบคุมการเขียนอ่าน (Mode) บิต 0 เมื่อ MCU ต้องการเขียนข้อมูลไปยังอุปกรณ์ก็กำหนดให้บิตนี้เป็น 0 และเมื่อต้องการ อ่านข้อมูล จากอุปกรณ์ ก็กำหนดให้บิตนี้เป็น 1

2.5.6 ช่วงเวลารับส่งบิตข้อมูลของ I²C BUS

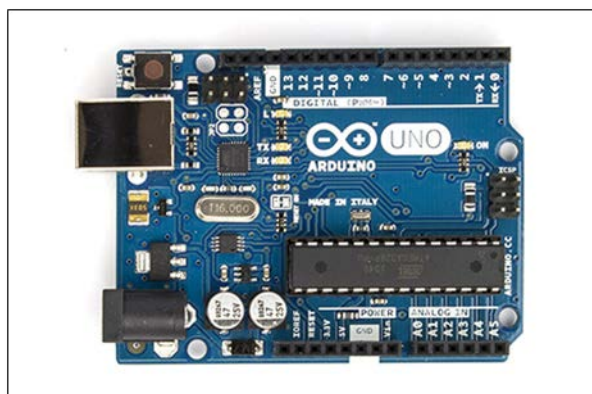


รูปที่ 24 การรับส่งบิตข้อมูลของ I²C BUS

- สถานะการรับ-ส่งข้อมูล จะกระทำในขณะที่ขา SCL เป็น 1
- สถานะการเปลี่ยนแปลงข้อมูล จะกระทำในขณะที่ขา SCL เป็น 0

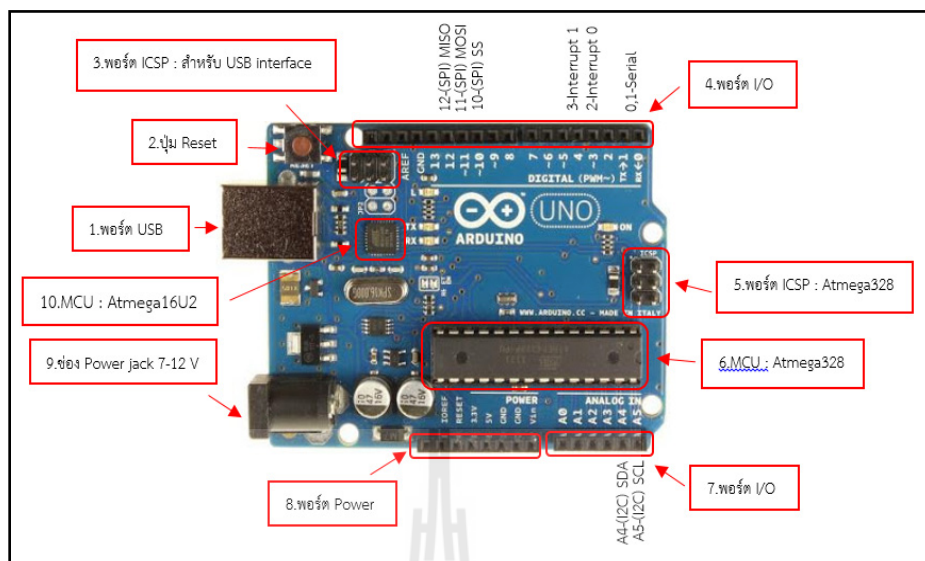
2.6 Arduino UNO R3

Arduino เป็นบอร์ดไมโครคอนโทรลเลอร์ที่ใช้ AVR ขนาดเล็กเป็นตัวประมวลผล เหมาะสำหรับการศึกษาเรียนรู้ไมโครคอนโทรลเลอร์ และการนำไปประยุกต์ใช้งานในการควบคุมอุปกรณ์ ทั้งอินพุตและเอาต์พุต ต่างๆได้ โปรแกรมภาษาของ Arduino จะใช้รูปแบบของภาษา C++ Arduino Uno R3 ตัวนี้ใช้ ATmega328 เป็นแบบ PDIP สามารถถอดออก เสียบก็สามารถเปลี่ยนใหม่ได้ มี Digital Input / Output 14 พอร์ต สามารถทำงานแบบ PWM (pulse width modulate) ทั้งหมด 6 พอร์ต มีส่วนแปลงสัญญาณอนาล็อกเป็นดิจิทัล (Analog to Digital Converter) ทั้งหมด 6 พอร์ต บนบอร์ดติดตั้งสัญญาณนาฬิกาภายนอก ความถี่ 16 MHz พร้อมด้วยพอร์ต USB และ หัวเสียบไฟเลี้ยงบอร์ด บนบอร์ดยังสามารถดาวน์โหลดโปรแกรมโดยตรงได้โดยผ่านหัวเสียบ ICSP และยังมีปุ่ม Reset ให้บนบอร์ดอีกด้วย บนบอร์ดติดตั้งไมโครคอนโทรลเลอร์ ATmega16U เพื่อช่วยในการส่งข้อมูลแบบอนุกรมผ่านทาง USB (Usb to Serial Converter) การส่งผ่านข้อมูล สามารถทำได้รวดเร็ว อีกทั้งไม่จำเป็นต้องติดตั้งไดรฟ์เวอร์สำหรับการใช้งานซอฟต์แวร์ในการดาวน์โหลดโปรแกรมรองรับระบบปฏิบัติการ Windows, Linux และ Mac



รูปที่ 25 บอร์ด Arduino UNO R3





รูปที่ 26 ส่วนประกอบของบอร์ด Aduino UNO R3

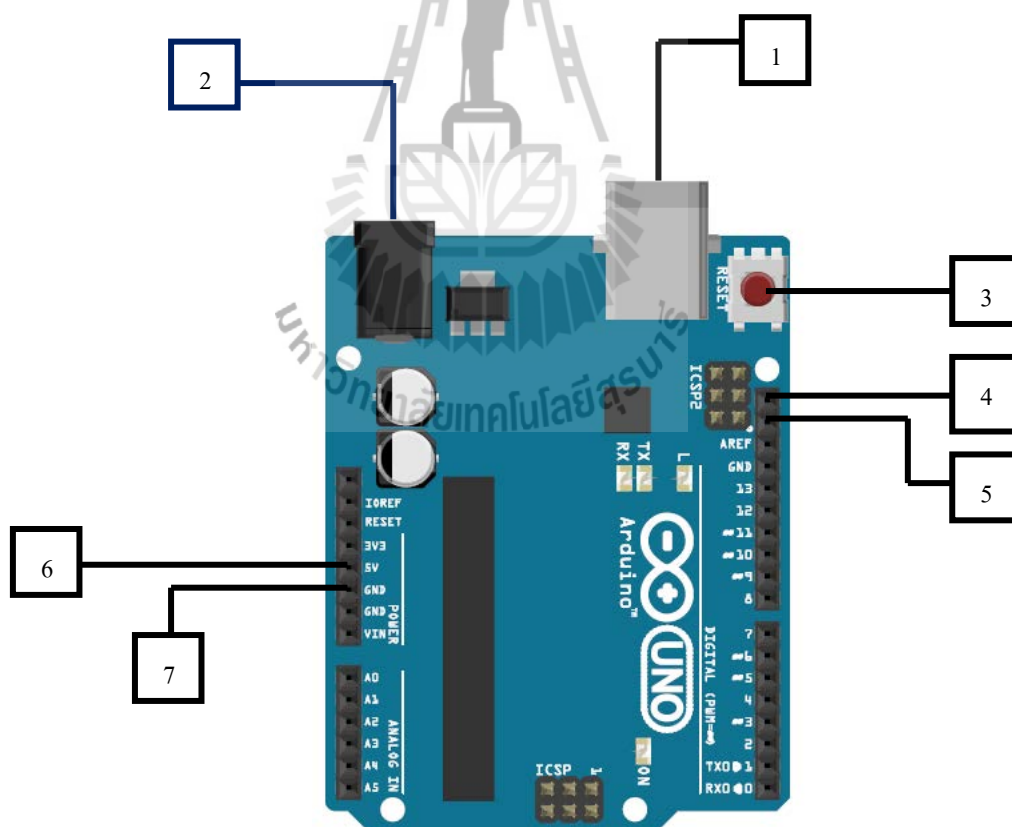
- 1.USB Port : ใช้สำหรับต่อกับ Computer เพื่ออัปโหลดโปรแกรมเข้า MCU และจ่ายไฟให้กับบอร์ด
- 2.Reset Button : เป็นปุ่ม Reset ใช้กดเมื่อต้องการให้ MCU เริ่มการทำงานใหม่
- 3.ICSP Port : ของ Atmega16U2 เป็นพอร์ตที่ใช้โปรแกรม Visual Com port บน Atmega16U2
4. I/O Port : Digital I/O ตั้งแต่ขา D0 ถึง D13 นอกจากนี้ บาง Pin จะทำหน้าที่อื่นๆ เพิ่มเติมด้วย เช่น Pin0,1 เป็นขา Tx, Rx Serial, Pin3,5,6,9,10 และ 11 เป็นขา PWM
- 5.ICSP Port: Atmega328 เป็นพอร์ตที่ใช้โปรแกรม Bootloader
6. MCU: Atmega328 เป็น MCU ที่ใช้นบนบอร์ด Arduino
7. I/O Port: นอกจากจะเป็น Digital I/O แล้วยังเปลี่ยนเป็นช่องรับสัญญาณอนาล็อกตั้งแต่ ขา A0 - A5
- 8.Power Port: ไฟเลี้ยงของบอร์ดเมื่อต้องการจ่ายไฟให้กับวงจรภายนอก ประกอบด้วยขา ไฟเลี้ยง +3.3 V, +5V, GND, V_{in}
9. Power Jack: รับไฟจาก Adapter โดยที่แรงดันอยู่ระหว่าง 7-12 V

10.MCU ของ Atmega16U2 เป็น MCU ที่ทำหน้าที่เป็น USB to Serial โดย Atmega328 จะติดต่อกับ Computer ผ่านAtmega16U2

2.6.1 คุณสมบัติของบอร์ด

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Clock Speed	16 MHz
Length	68.6 mm
Width	53.4 mm
Weight	25 g

2.6.2 การใช้งานของบอร์ด Arduino UNO R3 ที่ใช้ในการออกแบบโดยเราใช้
เพียงไม่กี่พอร์ตในการทำงานดังรูปภาพที่ 27



รูปที่ 27 การใช้งานของบอร์ด Arduino UNO R3

หมายเลข 1 ไว้สำหรับสาย USB เชื่อมกับคอมพิวเตอร์

หมายเลข 2 ต่อเข้ากับ Power

หมายเลข 3 ปุ่มกด Reset

หมายเลข 4 สาย SDA

หมายเลข 5 สาย SCL

หมายเลข 6 ต่อกับ Volt

หมายเลข 7 ต่อกับ GND

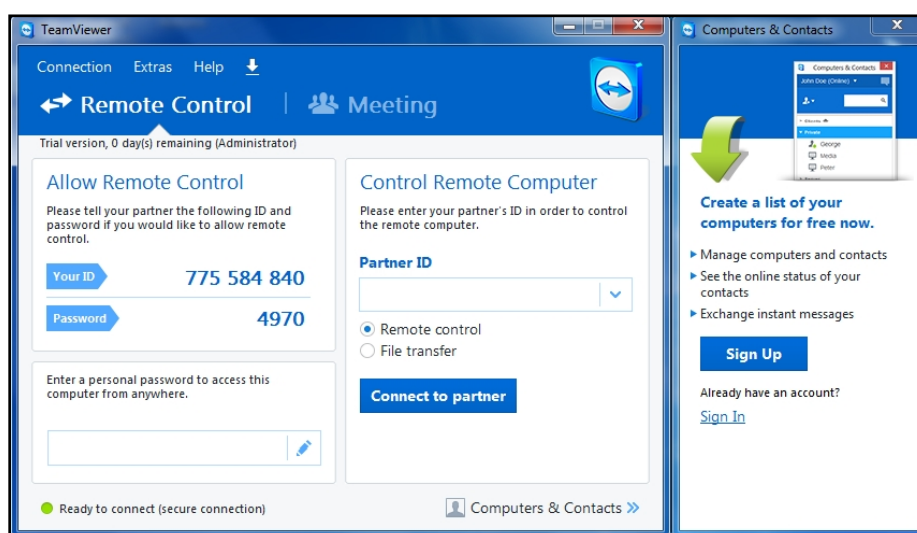
2.7 โปรแกรม Visual Basic 2010



รูปที่ 28 แสดงโปรแกรม Visual Basic 2010

โปรแกรม Microsoft Visual Basic 2010 Express Edition หรือเรียกโดยย่อว่า Visual Basic Express หรือ VB Express เป็นโปรแกรมที่ใช้สำหรับสร้างหรือพัฒนาโปรแกรมใช้งานบนวินโดวส์ โดยใช้ภาษา Visual Basic กับระบบปฏิบัติการ Windows ของบริษัทไมโครซอฟต์ Visual Basic พัฒนามาจากภาษา BASIC (ย่อมาจาก Beginner's All purpose Symbolic Instruction Code) ซึ่งเป็นภาษาคอมพิวเตอร์ระดับสูงที่ใช้งานง่าย เหมาะสำหรับผู้เริ่มใช้คอมพิวเตอร์เพราะใช้คำในภาษาอังกฤษที่เข้าใจง่ายและเมื่อเป็น Visual Basic ซึ่งใช้ลักษณะของการมองเห็นได้(Visual) ที่เป็นการติดต่อกับผู้ใช้ด้วย กราฟิกหรือรูปภาพ (Graphical User Interface – GUI) จึงทำให้การพัฒนาโปรแกรมใช้งานทำได้สะดวกและรวดเร็วขึ้น ถึงแม้จะใช้งานง่ายแต่ก็มีความสามารถสูง เหมาะสำหรับการพัฒนาโปรแกรมใช้งานได้หลายด้าน เช่น งานคำนวณทั่วไปงานด้านฐานข้อมูล เกม และอื่นๆ บริษัทไมโครซอฟต์ได้เปิดตัวโปรแกรม Visual Studio 2010 ซึ่งเป็นชุดเครื่องมือ พัฒนาซอฟต์แวร์เพื่อช่วยเหลือนักพัฒนา (ไม่ว่าจะเป็นนักพัฒนามือใหม่หรือนักพัฒนามืออาชีพ) ในการปรับปรุงขั้นตอนการพัฒนาและช่วยให้การแก้ปัญหาที่ซับซ้อนทำได้ง่ายขึ้น และน่าพอใจมากขึ้นกว่าเดิม ซึ่งมีส่วนประกอบหนึ่งคือ Visual Basic รุ่นใหม่ล่าสุด ใช้ชื่อ ว่า Microsoft Visual Basic 2010 โดยปรับปรุงจาก Visual Basic 2008 เดิมที่ใช้กันอยู่ เพิ่มขีดความสามารถหลายอย่างเพื่อการพัฒนาโปรแกรมใช้งานทำได้สะดวกรวดเร็วขึ้น และได้ออก Visual Basic 2010 Express ซึ่งใช้งานง่าย เรียนรู้ง่ายและน่าสนุก ทำโปรแกรมใช้งานได้เร็ว เหมาะสำหรับผู้ที่ต้องการศึกษาและพัฒนาโปรแกรมใช้งานที่ไม่จำเป็นต้องใช้ความสามารถทุกอย่างของ Visual Basic 2010

2.8 โปรแกรม Team Viewer



รูปที่ 29 ลักษณะหน้าต่างของโปรแกรม Team Viewer

โปรแกรม Team Viewer เป็นโปรแกรมการรีโมต (Remote) หรือ โปรแกรมควบคุมเครื่องระยะไกล เราสามารถใช้โปรแกรมนี้เข้าไปใช้งาน หรือควบคุมเครื่องอื่นที่เราต้องการ โดยจะแบ่งการทำงาน เป็น 2 ฟังก์ชัน คือเครื่องต้นทาง (เครื่องที่ทำการควบคุม) กับ เครื่องปลายทาง (เครื่องที่ถูกควบคุม) โดยเครื่องต้นทาง จะเห็นหน้าจอของเครื่องปลายทาง และสามารถโต้ตอบแก้ไขเปลี่ยนแปลง โอนย้ายข้อมูล หรือติดตั้งโปรแกรม ลักษณะเหมือนกับเราไปนั่งอยู่ที่หน้าจอของเครื่องปลายทางนั้น

ประเภทการใช้งาน โปรแกรม Team Viewer

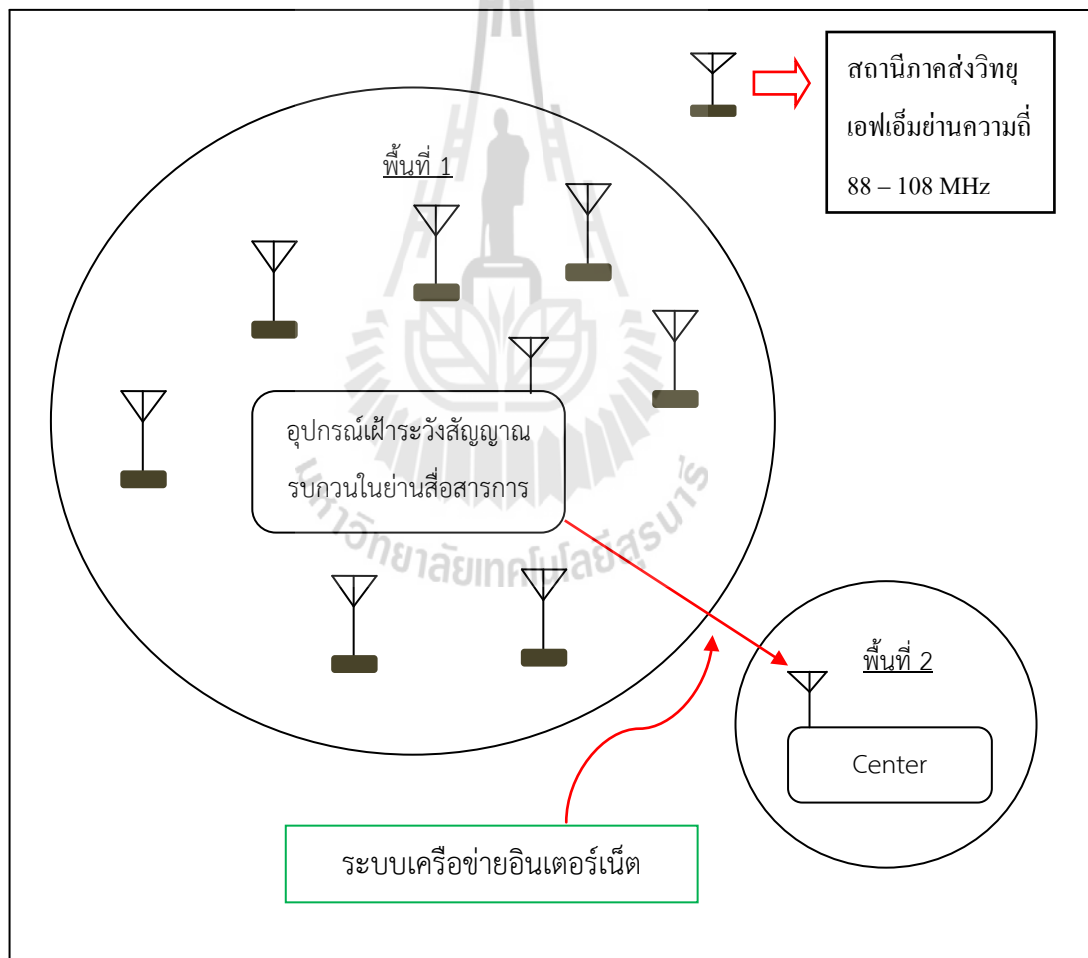
- Remote Support เป็นการรีโมตไปควบคุมหน้าจอของเครื่องปลายทาง สามารถควบคุมคอมพิวเตอร์จากระยะไกลที่ใดก็ได้ผ่านโครงข่ายอินเทอร์เน็ต
- Remote presentation สามารถนำเสนองานต่างๆ จากเครื่องต้นทางไปยังเครื่องปลายทางได้
- File transfer สามารถทำการรับส่งไฟล์ระหว่างเครื่องปลายทางกับเครื่องต้นทาง ถึงแม้ว่าจะมี Firewall กันอยู่ก็ตาม โดยที่ไม่จำเป็นต้องตั้งค่า Firewall ใหม่
- VPNเป็นการรีโมตเพื่อเชื่อมต่อแบบ VPN กับเครื่องปลายทาง เหมือนกับที่เราอยู่ในวงแลนเดียวกันกับเครื่องปลายทาง

บทที่ 3

การออกแบบอุปกรณ์เฝ้าระวังสัญญาณรบกวนในย่านสื่อสารการบิน

3.1 กล่าวนำ

ในบทนี้จะกล่าวถึงขั้นตอนการออกแบบและการสร้างอุปกรณ์เฝ้าระวังสัญญาณรบกวนในย่านสื่อสารการบินโดยลักษณะการทำงานนี้สามารถติดตั้งอุปกรณ์ไว้ตามพื้นที่ต่างๆและยังสามารถตรวจสอบค้นหาสถานีที่กำลังออกอากาศอยู่ในพื้นที่นั้น ซึ่งผู้ตรวจสอบไม่จำเป็นต้องอยู่ในพื้นที่ที่ติดตั้งอุปกรณ์สามารถตรวจสอบผ่านทางระบบเครือข่ายอินเทอร์เน็ตทำให้สามารถเพิ่มประสิทธิภาพในการตรวจสอบ ซึ่งระบบการทำงานจะแสดงดังรูปต่อไปนี้



รูปที่ 3.1 ภาพรวมของระบบเฝ้าระวังสัญญาณรบกวนในย่านสื่อสารการบิน

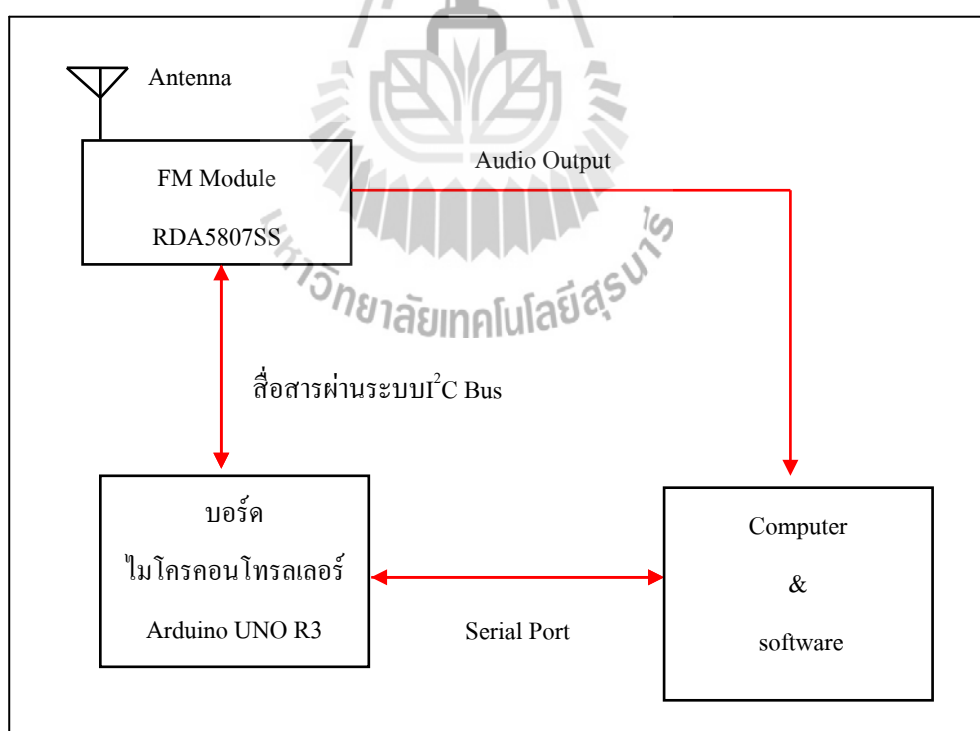


ลักษณะการทำงานรูปที่ 1 หากต้องการทราบว่าพื้นที่นั้นๆ มีช่องความถี่อะไรบ้างที่กำลังออกอากาศจะนำตัวอุปกรณ์เฟ้าะวังสัญญาณรบกวนในย่านสื่อสารการบินไปติดตั้งในพื้นที่นั้น โดยเครื่องนี้จะทำการค้นหาและตรวจสอบว่ามีคลื่นความถี่อะไรบ้างที่กำลังออกอากาศซึ่งตัวอุปกรณ์สามารถระบุคลื่นวิทยุที่ออกอากาศในขณะนั้นได้หรือต้องการตรวจสอบคลื่นวิทยุในพื้นที่อื่นสามารถทำการเชื่อมต่อผ่านระบบเครือข่ายอินเทอร์เน็ตทั้งนี้ยังช่วยเพื่อลดทรัพยากรบุคคล ลดระยะเวลาในการทำงานและเพิ่มประสิทธิภาพในการค้นหาอีกด้วย

ซึ่งอุปกรณ์เฟ้าะวังสัญญาณรบกวนในย่านสื่อสารการบินที่ทำการพัฒนาขึ้นจะถูกติดตั้งและใช้งานในพื้นที่ 1 ซึ่งมีองค์ประกอบและหลักการทำงานที่สำคัญดังนี้

3.2 องค์ประกอบและหลักการทำงานของอุปกรณ์เฟ้าะวังสัญญาณรบกวนในย่านสื่อสารการบิน

จากที่กล่าวมาข้างต้นอุปกรณ์เฟ้าะวังสัญญาณรบกวนในย่านสื่อสารการบินทำงานเปรียบเสมือนเครื่องรับวิทยุจะถูกติดตั้งไว้ตามพื้นที่ต่างๆตามที่ต้องการซึ่งจะมีโปรแกรมคอมพิวเตอร์สำหรับควบคุมจะถูกพัฒนาขึ้นเพื่อให้สามารถควบคุมเครื่องคอมพิวเตอร์ภากรับ ซึ่งมีองค์ประกอบหลักดังนี้

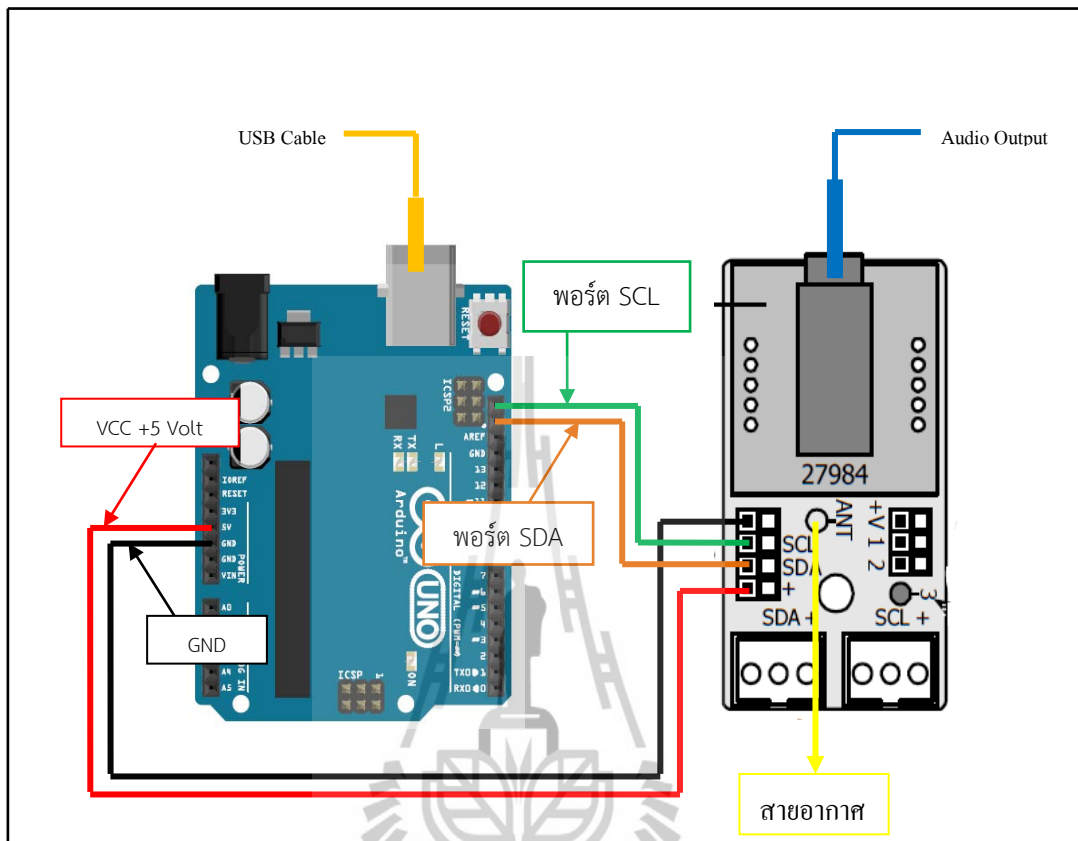


รูปที่ 3.2 แสดงส่วนองค์ประกอบของอุปกรณ์เฟ้าะวังสัญญาณรบกวนในย่านสื่อสารการบิน

ลักษณะการทำงานขององค์ประกอบและหลักการทำงานของอุปกรณ์เฝ้าระวังสัญญาณรบกวนในย่านสื่อสารการบินรูปที่2 FM Module RDA5807SS ทำหน้าที่ในการรับคลื่นวิทยุโดย FM Module จะทำงานควบคู่กับไมโครคอนโทรลเลอร์ผ่านทางบัส I²C Bus เพื่อควบคุมการรับคลื่นวิทยุบอร์ดไมโครคอนโทรลเลอร์ที่ใช้ในการทำงานคือ Arduino UNO R3 เพื่อทำการเขียนคำสั่งควบคุมให้ FM Module เพื่อให้ FM Module ทำงานและรับคลื่นวิทยุได้ นอกจากนี้ยังมีโปรแกรมคอมพิวเตอร์ที่ได้รับการพัฒนาเพื่อให้การใช้งานและตรวจสอบคลื่นวิทยุได้สะดวกยิ่งขึ้น โดยโปรแกรมคอมพิวเตอร์จะรับคำสั่งควบคุมและทำการประมวลผลและส่งโปรแกรมคำสั่งกลับไปยังบอร์ดไมโครคอนโทรลเลอร์อีกครั้งผ่าน Serial Port เมื่อบอร์ดไมโครคอนโทรลเลอร์ได้รับคำสั่งกลับมา จะทำการส่งคำสั่งไปยัง FM Module และทำการแสดงผลคำสั่งควบคุมและสัญญาณ FM Module ที่ได้รับจะถูกเชื่อมต่อไปยังคอมพิวเตอร์เพื่อตรวจสอบสัญญาณเสียงที่ได้รับนั้นเป็นสัญญาณเสียงจริงหรือไม่

3.2.1 การควบคุมการทำงานของ FM Module RDA5807SS

FM Module RDA5807SS ทำหน้าที่ในการรับคลื่นวิทยุและรองรับความถี่คลื่นวิทยุ FM ตลอดย่านตั้งแต่ 87.50 ถึง 107.50 MHz โดยที่ไม่ต้องต่อสายอากาศภายนอกและยังได้รับการออกแบบให้ทำงานกับไมโครคอนโทรลเลอร์ผ่านทางบัส I²C เพื่อควบคุมการรับสัญญาณวิทยุโดยการสื่อสารจะต้องมีบอร์ดไมโครคอนโทรลเลอร์ Arduino ในการควบคุมซึ่งบอร์ดไมโครคอนโทรลเลอร์ที่ใช้งาน คือ Arduino UNO R3 โดยใช้พอร์ตเพียง 2 เส้น คือ serial data (SDA) และสาย serial clock (SCL) ซึ่งจะแสดงการเชื่อมต่อดังรูปที่ 3.3

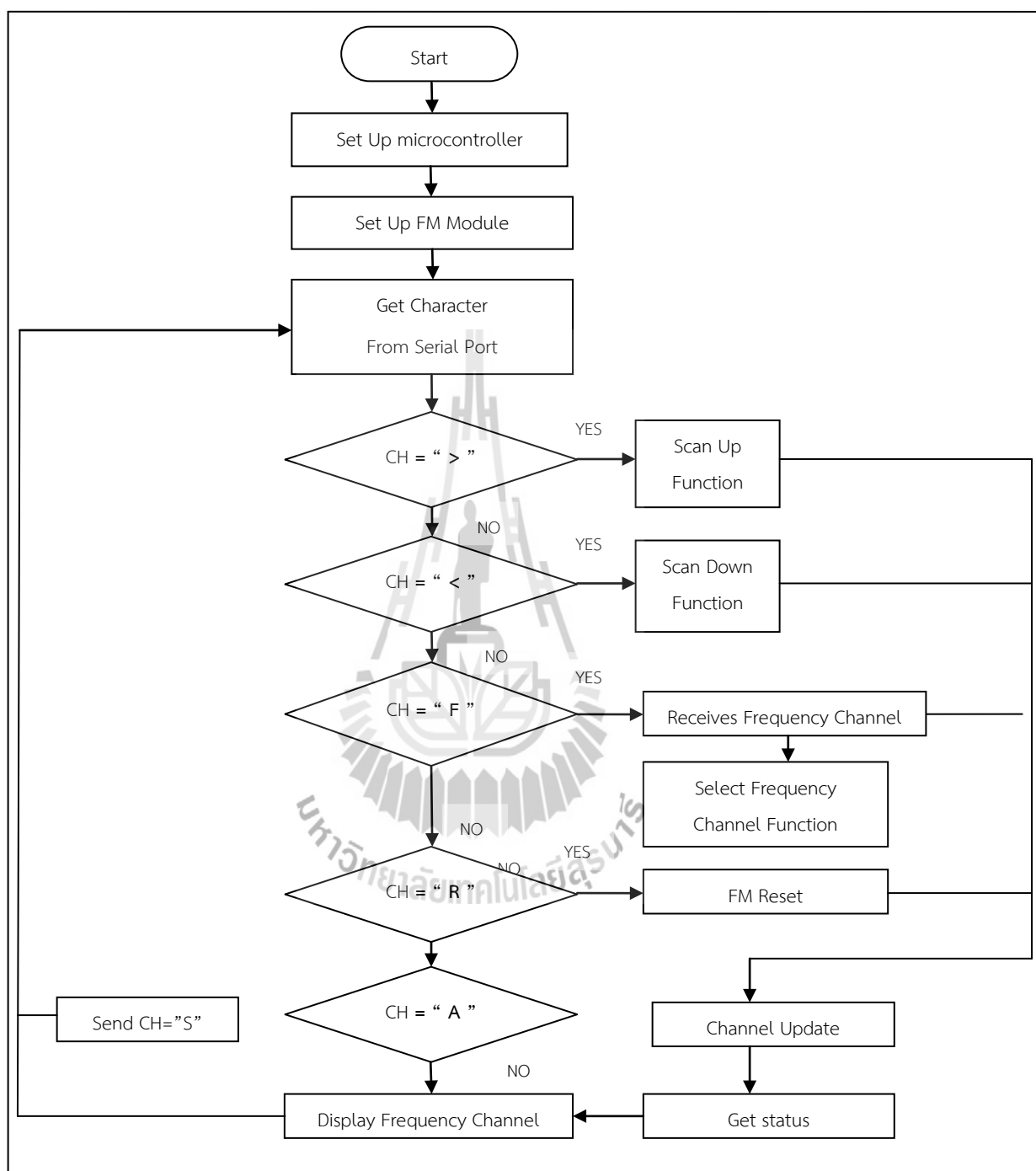


รูปที่ 3.3 แสดงการควบคุมการทำงานของ FM Module RDA5807SS

จากรูปที่ 3.3 บอร์ดไมโครคอนโทรลเลอร์ Arduino UNO R3 จะมีสายนำสัญญาณประกอบด้วย 4 เส้น มีดังนี้ สายนำสัญญาณแรกพอร์ต SCL ของบอร์ดไมโครคอนโทรลเลอร์ Arduino UNO R3 จะอยู่ที่ Pin ADC5 ซึ่งจะทำการเชื่อมต่อกับพอร์ต SCL ของ FM Module RDA5807SS สายนำสัญญาณที่สองพอร์ต SDA ของบอร์ดไมโครคอนโทรลเลอร์ Arduino UNO R3 จะอยู่ที่ Pin ADC4 ซึ่งจะทำการเชื่อมต่อกับ พอร์ต SDA ของ FM Module RDA5807SS โดยสายนำสัญญาณที่สามไฟเลี้ยงบวกของวงจรบอร์ดไมโครคอนโทรลเลอร์ Arduino UNO R3 จะทำการเชื่อมต่อกับ ไฟเลี้ยงบวกของ FM Module RDA5807SS และสายนำสัญญาณสุดท้ายกราวด์ของบอร์ดไมโครคอนโทรลเลอร์ Arduino UNO R3 จะทำการเชื่อมต่อกับกราวด์ของ FM Module RDA5807SS นอกจากนี้ USB Cable เป็นช่องเสียบของบอร์ดไมโครคอนโทรลเลอร์ Arduino UNO R3 ใช้สำหรับเชื่อมต่อเข้ากับคอมพิวเตอร์เพื่อควบคุมและเขียนโปรแกรมและ Audio Output ของ FM Module RDA5807SS สำหรับไว้ส่งสัญญาณเสียงที่โมดูลรับได้

3.2.2 การออกแบบคำสั่งสำหรับควบคุม

หลังจากที่เราทำการเชื่อมต่อบอร์ดไมโครคอนโทรลเลอร์และ FM Module เบอร์ RDA5807SS เรียบร้อยแล้วจากนั้นจะทำการเขียนคำสั่งโดยที่ระบบควบคุมทั้งหมด ลักษณะการออกแบบคำสั่งสำหรับควบคุมและการติดต่อระหว่างบอร์ดไมโครคอนโทรลเลอร์และ FM Module เบอร์ RDA5807SS สามารถสรุปเป็นแผนผังการทำงาน ดังรูปที่ 3.4



รูปที่ 3.4 แผนผังแสดงการออกแบบคำสั่งสำหรับควบคุม

คำสั่ง setup()

เมื่อโปรแกรมเริ่มทำงานในฟังก์ชัน `setup()` โดยกำหนดให้ขาพอร์ตที่เชื่อมต่อสวิตช์ควบคุมทั้งหมดเป็นพอร์ตอินพุตดิจิตอลจากนั้นติดต่อกับบัส I²C ด้วยคำสั่ง `Wire.begin()` เพื่อทำการรีเซ็ตโมดูลวิทยุ FM จากนั้นอ่านค่าสถานีล่าสุดออกมาแล้วทำการแสดงผลด้วยฟังก์ชัน `chan_print()`

คำสั่ง loop()

เมื่อเข้าสู่ฟังก์ชัน `loop()` อันเป็นส่วนหลักของโปรแกรมหลักจะมีการตรวจสอบการกดสวิตช์ต่างๆที่ใช้ควบคุมการทำงานแล้วทำการแสดงสถานะการทำงานเมื่อมีการเพื่อเปลี่ยนลดหรือเพิ่มค่าความถี่จะมีการเรียกใช้ฟังก์ชัน `chan_update()` เพื่อปรับค่าความถี่ล่าสุดที่เกิดขึ้นแล้วเรียกใช้ฟังก์ชัน `tune_enable()` เพื่อทำให้โมดูลวิทยุ FM เลือกรับสัญญาณจากสถานีที่มีความถี่ตรงกับที่ต้องการ (ส่วนจะรับชัดหรือไม่และรับได้หรือไม่ไม่สนใจ) โดยจะเรียกใช้คำสั่ง `TUNE` ที่ใช้โดยตัวแปร `Config` ข้อมูลสำหรับกำหนดการทำงานทั้งหมดจะถูกเขียนไปยังโมดูลวิทยุ FM ด้วยฟังก์ชัน `setConfig()` โดยจะเขียนรีจิสเตอร์หลักทั้ง 5 ตัวในคราวเดียวกันจากนั้นทำการอ่านสถานะการค้นหาสถานีและค่าความถี่ปัจจุบันจากโมดูลวิทยุ FM ด้วยฟังก์ชัน `getStatus()` แล้วหยุดการค้นหาสถานีทำการปรับปรุงค่าความถี่ปัจจุบันแล้วส่งคำสั่งไปยังโมดูลวิทยุ FM อีกครั้งเพื่อให้โมดูลวิทยุ FM ทำงานอยู่กับสถานีวิทยุล่าสุดที่เลือกไว้พร้อมกันนั้นกำหนดให้แสดงค่าสถานีที่เลือกไว้ด้วยฟังก์ชัน `chan_print()` ที่ถูกเรียกให้ทำงานภายในฟังก์ชัน `chan_update()`

ส่วนของฟังก์ชันย่อยที่ทำงานในโปรแกรมหลักมีทั้งสิ้น 8 ฟังก์ชันคือ

1. `fm_reset()` ทำหน้าที่รีเซ็ตการทำงานของโมดูลวิทยุ FM ให้กลับไปยังค่าตั้งต้น
2. `chan_update()` เป็นฟังก์ชันปรับปรุงค่าความถี่ปัจจุบันสำหรับโมดูลวิทยุ FM ด้วยการนำค่าของตัวแปร `fm_seek` มาบวกกับค่าความถี่ล่าสุดที่กำหนดไว้ในตัวแปร `MINB_FREQ` โดยจะเตรียมข้อมูลของความถี่เพื่อเขียนลงในรีจิสเตอร์ `0x03` ของโมดูลวิทยุ FM ด้วยฟังก์ชัน `setConfig()` ต่อไป
3. `setConfig()` เป็นฟังก์ชันที่มีความสำคัญมากที่สุดตัวหนึ่งทำหน้าที่ในการเขียนข้อมูลคำสั่งลง

ในรีจิสเตอร์หลักของโมดูลวิทยุ FM 5 ตัวคือรีจิสเตอร์ `0x02`, `0x03`, `0x04`, `0x05` และ `0x06` ในฟังก์ชันนี้มีการเรียกใช้คำสั่งและฟังก์ชันของไลบรารี `Wire` ซึ่งใช้ในการติดต่อกับอุปกรณ์ระบบบัส I²C ของ Arduino

ในรีจิสเตอร์ 0x02 จะกำหนดค่าการทำงานของบิตทั้ง 16 บิต ของรีจิสเตอร์ดังนี้

D001 => 1101 0000 0000 0001 (16 : 0)

- บิต 0 บิตนี้จะกำหนดเป็น “1” ให้เป็นเอ็นเอเบิลเพื่อเปิดการทำงานของโมดูล
- บิต 1 กำหนดเป็น “0” เนื่องจากไม่ต้องการรีเซต
- บิต 2 และ 3 กำหนดเป็น “00” เพื่อสำรองไว้
- บิต 4 ถึง 6 กำหนดเป็น “000” เพื่อเลือกความถี่สัญญาณนาฬิกา เพื่อใช้ความถี่ 32.768 kHz
- บิต 7 กำหนดเป็น “0” เพื่อสำรองไว้
- บิต 8 กำหนด “0” เพื่อเปิดเอ็นเอเบิลของการค้นหาสถานี
- บิต 9 กำหนด “0” เพื่อเลือกโหมดค้นหาสถานีความถี่ที่ต่ำลง (SEEKDOWN)
- บิต 10 กำหนด “0” เพื่อเปิดเอ็นเอเบิลวงจรสัญญาณนาฬิกาของชิป
- บิต 11 กำหนดเป็น “0” เพื่อสำรองไว้
- บิต 12 กำหนด “1” เพื่อเพิ่มเสียงทุ้ม (BASS)
- บิต 13 กำหนด “0” เพื่อเลือกระบบเสียงเป็นระบบสเตอริโอ มัลติเพล็กซ์
- บิต 14 กำหนด “1” เพื่อให้การควบคุมการตัดสัญญาณเสียงทำงานปกติ
- บิต 15 กำหนด “1” เพื่อให้ควบคุมสถานะขาเอาต์พุตสัญญาณเสียงให้ทำงานปกติ

ในรีจิสเตอร์ 0x03 จะกำหนดค่าการทำงานของบิตทั้ง 16 บิต ของรีจิสเตอร์ดังนี้

0002 => 0000 0000 0000 0010 (16 : 0)

- บิต 1 และ 0 กำหนด “10” เพื่อเลือกช่วงความถี่ในการเปลี่ยนสถานีครั้งละ 50 kHz
- บิต 3 และ 2 กำหนด “00” เพื่อเลือกระบบความถี่ของวิทยุ FM ที่ความถี่ 88.0 – 108.0 MHz
- บิต 4 กำหนด “0” เพื่อไม่ให้ปรับความถี่
- บิต 5 กำหนดเป็น “0” สำรองไว้
- บิต 15 ถึง 6 กำหนด “0” เลือกความถี่ของสถานีวิทยุถ้าบิต Band = “0”
ความถี่ = (ช่วงความถี่ในหน่วย kHz (กำหนดจากบิต SPACE) x CHAN) + 87 MHz

ในรีจิสเตอร์ 0x04 จะกำหนดค่าการทำงานของบิตทั้ง 16 บิต ของรีจิสเตอร์ดังนี้

044 => 00000 0001 1011 1000 (16 : 0)

- บิต 1 และ 0 กำหนด “0” เพื่อกำหนดการทำงานของขา GPO1 ให้มีอิมพีแดนซ์ที่สูงขึ้น

- บิต 2 และ 3 กำหนด “10” เพื่อกำหนดการทำงานของขา GPIO2 ที่เอาต์พุตลอจิก “0”
- บิต 5 และ 4 กำหนด “11” เพื่อกำหนดการทำงานของขา GPIO3 ที่เอาต์พุตลอจิก “1”
- บิต 6 กำหนด “0” เพื่อปิดเอ็นเอเบิลการติดต่อผ่านบัสด I²S
- บิต 8 และ 7 กำหนด “11” ไม่ต้องสำรอง
- บิต 9 กำหนด “0” เพื่อสำรองไว้
- บิต 10 กำหนด “0” เพื่อปิดเอ็นเอเบิลการควบคุมอัตราขยายอัตโนมัติ(AGCD)
- บิต 11 กำหนด “0” เพื่อเลือกค่าดีเอ็มฟาซิสที่เท่ากับ 75 μ s
- บิต 13 และ 12 กำหนดเป็น “00” เพื่อสำรองไว้
- บิต 14 กำหนด “0” เพื่อปิดเอ็นเอเบิลการอินเตอร์รัปต์ เมื่อการค้นหาสถานีเสร็จสิ้นโดยจะส่งพัลส์ “0” ออกไปทางขา GPIO2
- บิต 15 กำหนดเป็น “0” เพื่อสำรองไว้

ในรีจิสเตอร์ 0x05 จะกำหนดค่าการทำงานของบิตทั้ง 16 บิต ของรีจิสเตอร์ดังนี้

86D3 =>1000 0110 1101 0011 (16 : 0)

- บิต 3 ถึง 0 กำหนด “0011” เพื่อปรับความดังของเสียง (สเกลลอกลีเนียร์) โดย “0000” ต่ำสุด “1111” สูงสุด
- บิต 5 และ 4 กำหนด “00” เพื่อเลือกกระแสไฟฟ้าอินพุตที่ 2.1 mA ของวงจร LNA
- บิต 7 และ 6 กำหนด “00” เพื่อเลือกทั้งอินพุตบวกและอินพุตลบของวงจร LNA
- บิต 14 ถึง 8 กำหนด “0000110” เพื่อกำหนดความแรงของสัญญาณ RSSI ที่ใช้ในการค้นหาสถานี(ค่าต่ำสุด = 0)
- บิต 15 กำหนด “1” เพื่อกำหนดโหมดการทำงานของอินเตอร์รัปต์ให้อ่านค่าจากรีจิสเตอร์ 0x0c

4. getStatus() เป็นฟังก์ชันอ่านค่าสถานะการค้นหาสถานีและค่าความถี่ของสถานีปัจจุบันของโมดูลวิทยุ FM โดยเก็บสถานะการค้นหาสถานีไว้ที่ตัวแปร fm_stc และค่าของสถานีไว้ที่ตัวแปร fm_seek

5. tune_enable() เป็นฟังก์ชันเอ็นเอเบิลหรือควบคุมให้โมดูลวิทยุ FM ทำการรับคลื่น

6. tune_disable() เป็นฟังก์ชันหยุดการรับคลื่นของโมดูลวิทยุ FM

7. seek_disable() เป็นฟังก์ชันหยุดการค้นหาช่องความถี่ของสถานี

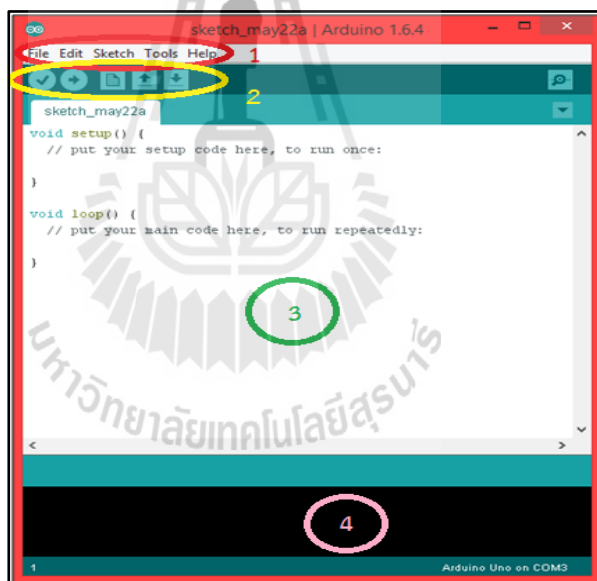
3.2.3 การเขียนโปรแกรมคำสั่งควบคุม

การเขียนโปรแกรมคำสั่งควบคุมสำหรับ Arduino จำเป็นต้องมีการกำหนดค่าเริ่มต้นต่างๆ มีขั้นตอนพัฒนาโปรแกรมดังนี้

1. ทำการติดตั้งโปรแกรม Arduino หลังติดตั้งเสร็จจะมีไอคอนปรากฏขึ้นดังรูปทำการคลิกไปที่ไอคอนตามรูป

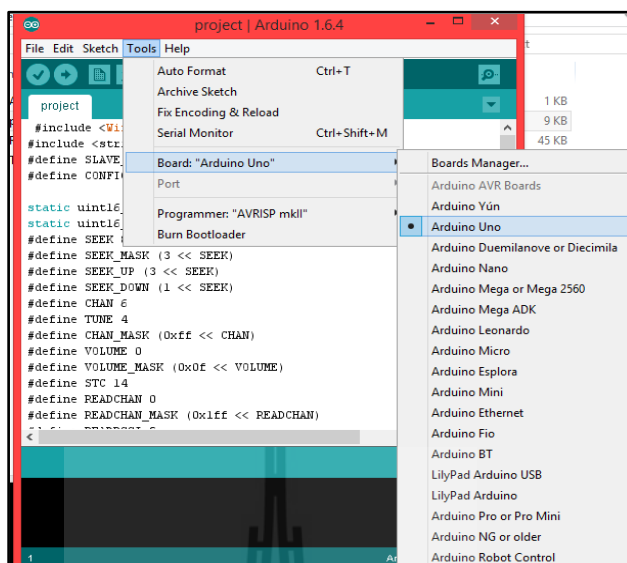


2. เมื่อเปิดโปรแกรมแล้วจะพบกับหน้าต่างของ IDE ซึ่งจะประกอบด้วยส่วนต่างๆ ดังนี้
ส่วนที่หนึ่งเป็น Menu การใช้งานของโปรแกรม Arduino ส่วนที่สองเป็น Shortcut สำหรับทำการ Compile, Upload ส่วนที่สามเป็น Editor ไว้สำหรับเขียนคำสั่งและส่วนที่สี่เป็น Error List สำหรับดูข้อผิดพลาดของโปรแกรม



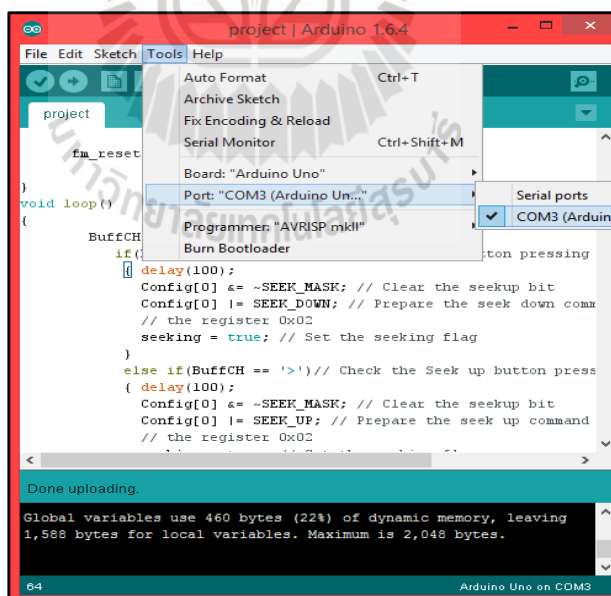
รูปที่ 3.5 แสดงหน้าต่างการใช้งานของโปรแกรม Arduino

3. การใช้งานร่วมกับ Arduino UNO จำเป็นต้องมีการตั้งค่าโปรแกรมเริ่มต้นก่อนการใช้งาน ดังนี้
 - การตั้งค่า Board โดยให้เลือกที่ Tools => Board => Arduino UNO



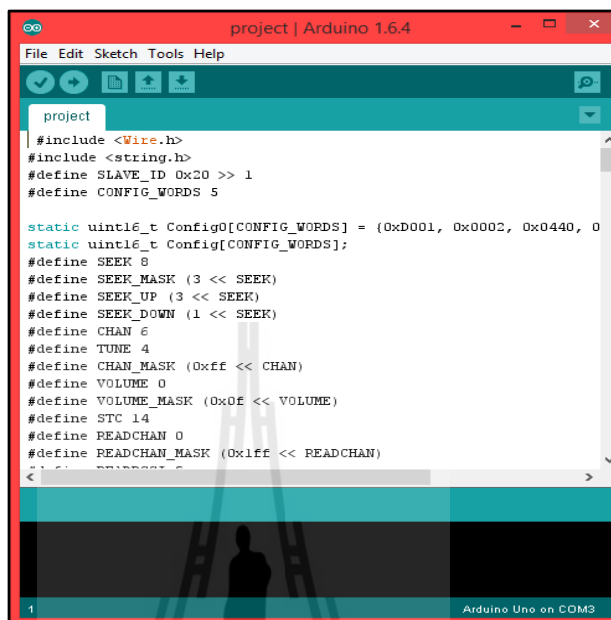
รูปที่ 3.6 แสดงการตั้งค่า Board

4. จากนั้นทำการตั้งค่า Port ให้ไปที่ Tools => Serial Port และเลือกให้ตรงกับบอร์ด Arduino UNO ที่ใช้งาน (สำหรับบอร์ด Arduino UNO R3 โปรแกรมจะเลือกให้อัตโนมัติ) ดังรูปที่ 3.7



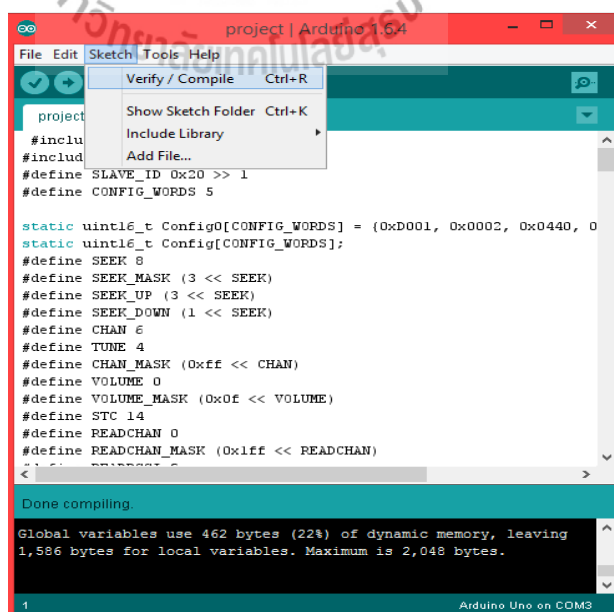
รูปที่ 3.7 แสดงการตั้งค่า Port

5. เมื่อทำการตั้งค่า Board และ Port เสร็จเรียบร้อยแล้วหลังจากนั้นทำการเขียนคำสั่งควบคุมลงบนหน้าต่าง IDE ดังรูปที่ 3.8



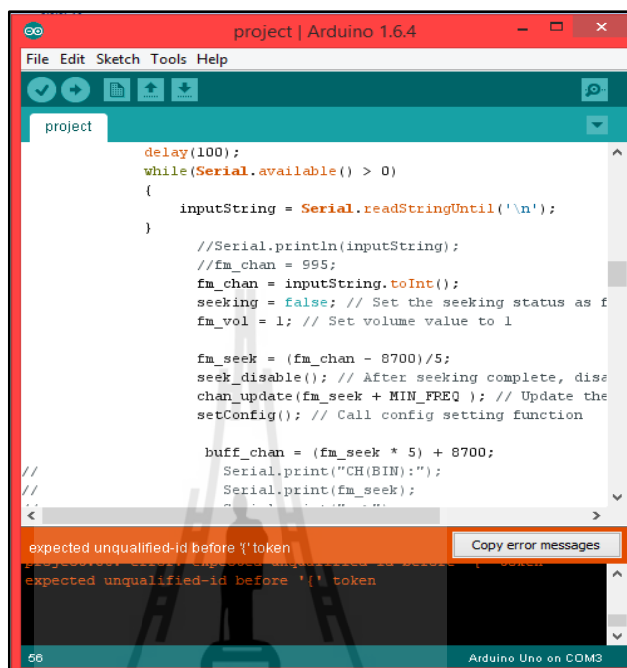
รูปที่ 3.8 แสดงการเขียนโปรแกรมลงบนหน้าต่าง IDE

6. จากนั้นทำการคอมไพล์โปรแกรมไปที่ Sketch => Verify / Compile เมื่อคอมไพล์เรียบร้อยแล้วจะปรากฏข้อความ “Done Compiling” ดังรูปที่ 3.9



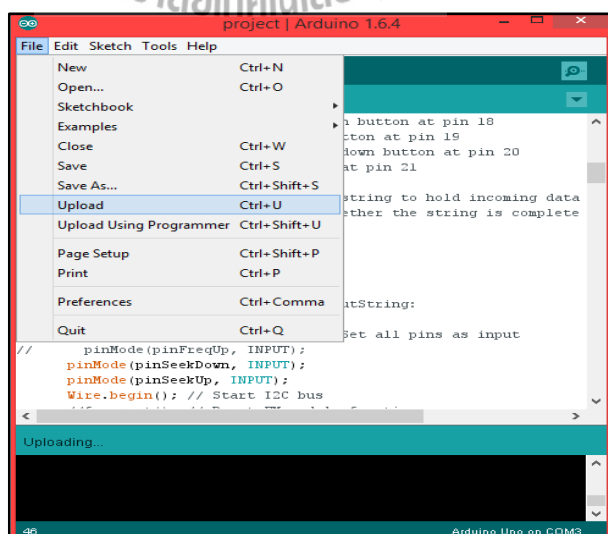
รูปที่ 3.9 แสดงการคอมไพล์โปรแกรม

- แต่ในกรณีที่คอมไพล์ไม่สำเร็จจะปรากฏข้อความ “expected Unqualified-id before ‘{’ token” ดังรูปที่ 3.10



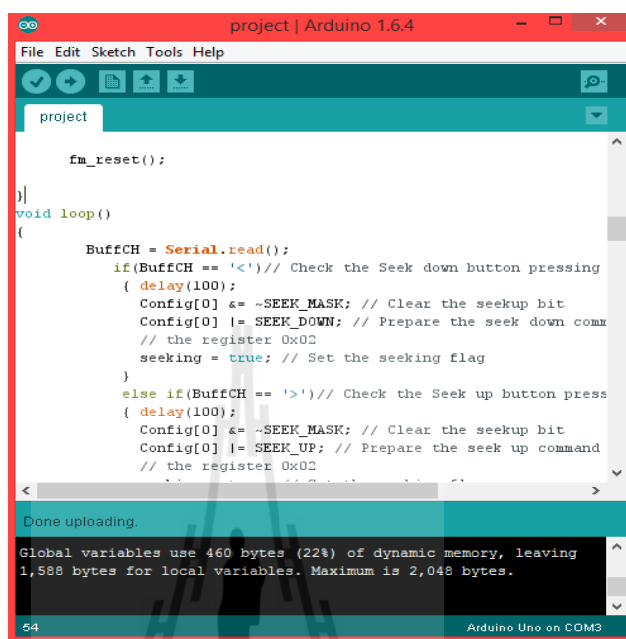
รูปที่ 3.10 แสดงข้อความการคอมไพล์ไม่สำเร็จ

7. หลังจากทำการคอมไพล์โปรแกรมเรียบร้อยแล้ว จึงทำการ UPLOAD โปรแกรมให้กับบอร์ด Arduino UNO สามารถทำได้โดยเลือกที่ File => Upload (หรือกด Ctrl+U) ดังรูปที่ 3.11



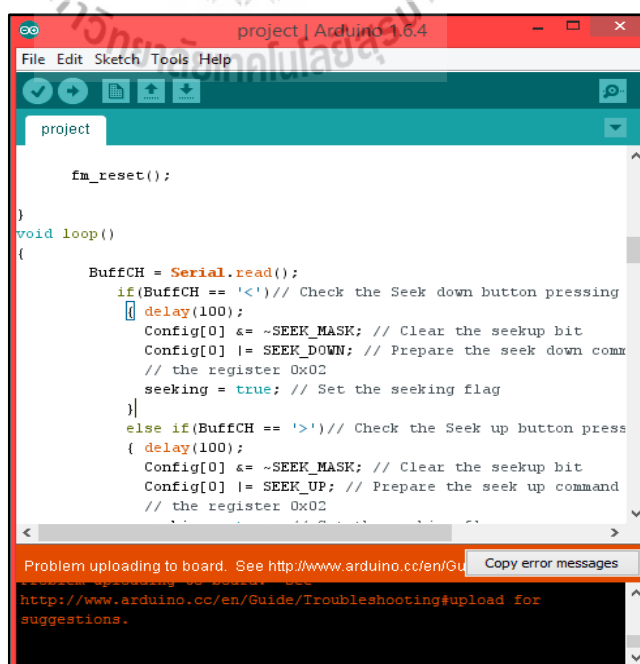
รูปที่ 3.11 แสดงข้อความการ UPLOAD โปรแกรมไม่สำเร็จ

- เมื่อ UPLOAD โปรแกรมให้กับบอร์ด Arduino UNO สำเร็จแล้วจะปรากฏข้อความ “Done uploading.” ดังรูปที่ 3.12



รูปที่ 3.12 แสดงการ UPLOAD โปรแกรม

- แต่ในกรณีที่ทำการ UPLOAD โปรแกรมให้กับบอร์ด Arduino UNO ไม่สำเร็จจะปรากฏข้อความ “Problem uploading to board.” ดังรูปที่ 3.13



รูปที่ 3.13 แสดงข้อความการ UPLOAD โปรแกรมไม่สำเร็จ

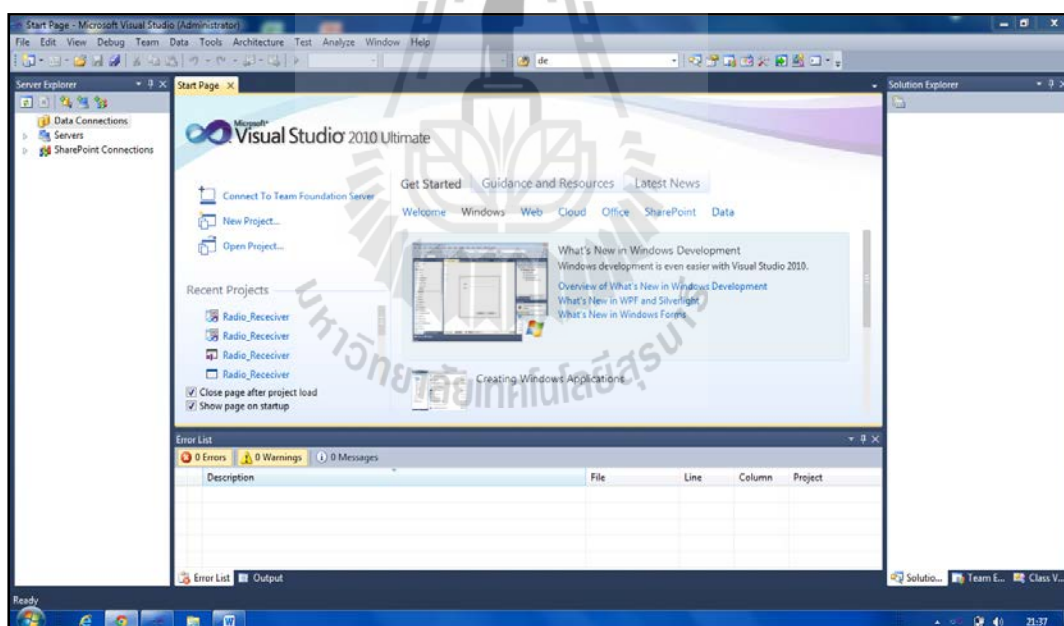
3.2.4 การออกแบบโปรแกรมคำสั่งควบคุม

การออกแบบโปรแกรมคำสั่งควบคุมสำหรับ Microsoft Visual Studio ประกอบด้วยเครื่องมือที่ช่วยให้พัฒนาแอปพลิเคชันที่หลากหลายและสะดวกสบายต่อการใช้งานซึ่งก่อนที่จะได้เรียนรู้การเขียนโปรแกรม ควรศึกษาองค์ประกอบต่างๆที่มีในโปรแกรมเสียก่อนโดยมีขั้นตอนการออกแบบโปรแกรมดังนี้

1. ทำการติดตั้งโปรแกรม Visual Basic ก็จะมีไอคอนอันหนึ่งปรากฏขึ้นมาที่ Desktop คลิกไปที่ไอคอนตามรูป



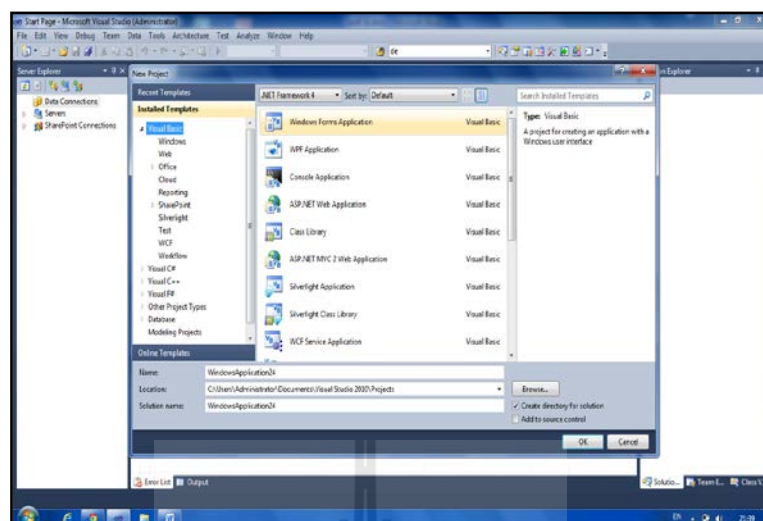
2. เปิดโปรแกรม Microsoft Visual Studio 2010 จะปรากฏหน้า Start Page ในการสร้าง Project ใหม่ให้ Click ที่ New Project หรือ Click ที่เมนู File => New Project



รูปที่ 3.14 แสดงหน้าต่างการใช้งานโปรแกรม

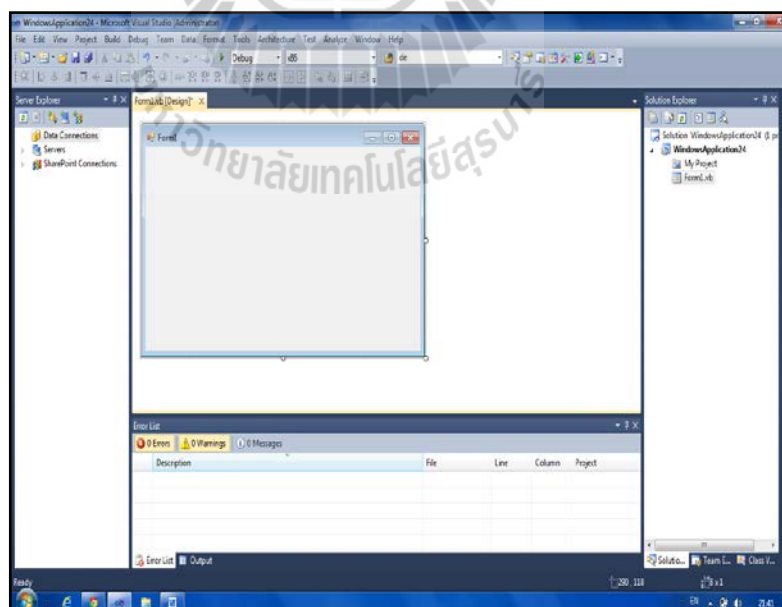
3. จะปรากฏกรอบ New Project

- ให้เลือก Visual Basic => Windows
- เลือก Windows Forms Application



รูปที่ 3.15 แสดงฟังก์ชันการสร้างโปรแกรม

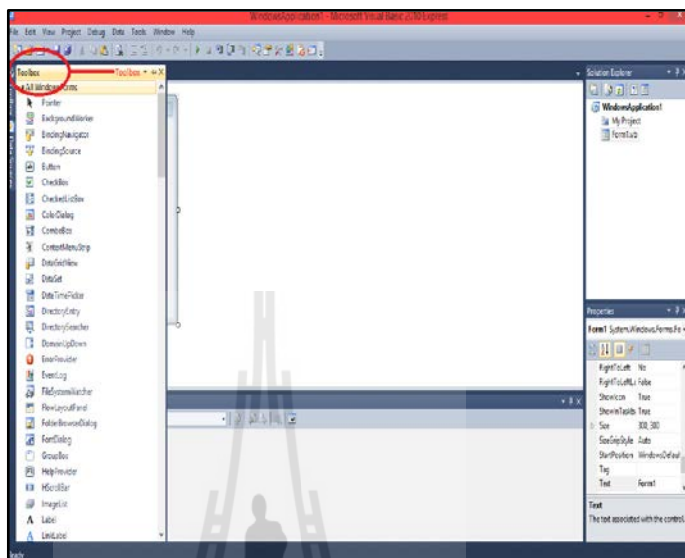
- ตั้งชื่อ Project ในช่อง Name
- กำหนดตำแหน่งในการจัดเก็บ Project ในช่อง Location
- Click ปุ่ม Ok ดังรูปที่ 3.16



รูปที่ 3.16 แสดงหน้าต่างการใช้งานของโปรแกรม

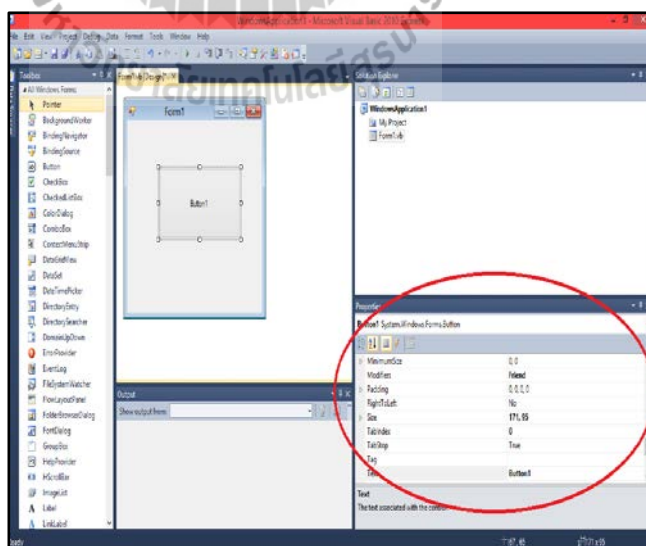
4. ขั้นตอนการเขียนโปรแกรม

- สร้าง Object ต่างๆที่ต้องการใช้งาน ลงบนฟอร์ม เช่น Progressbar, Button และ Timer เป็นต้น



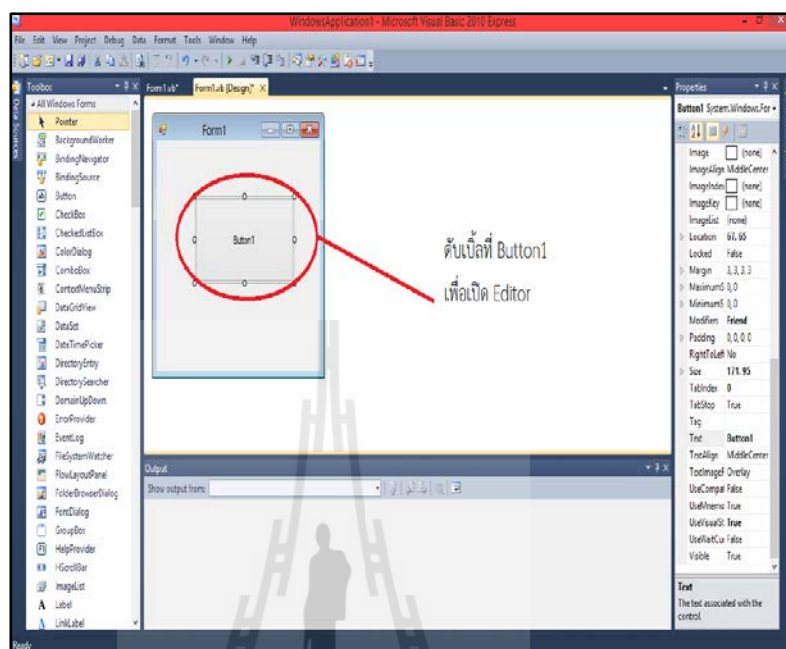
รูปที่ 3.17 การสร้าง Object

- กำหนดพรีอเพอร์ติวของฟอร์มและออบเจ็คต่างๆบนฟอร์มโดยใช้วินโดว์ Properties
- ปรับแต่งรายละเอียดของ GUI ย่อ-ขยาย-ย้าย หรือปรับเปลี่ยน ขนาด-สี-ตัวอักษรตัวเลข และสัญลักษณ์ต่างๆตามสะดวก



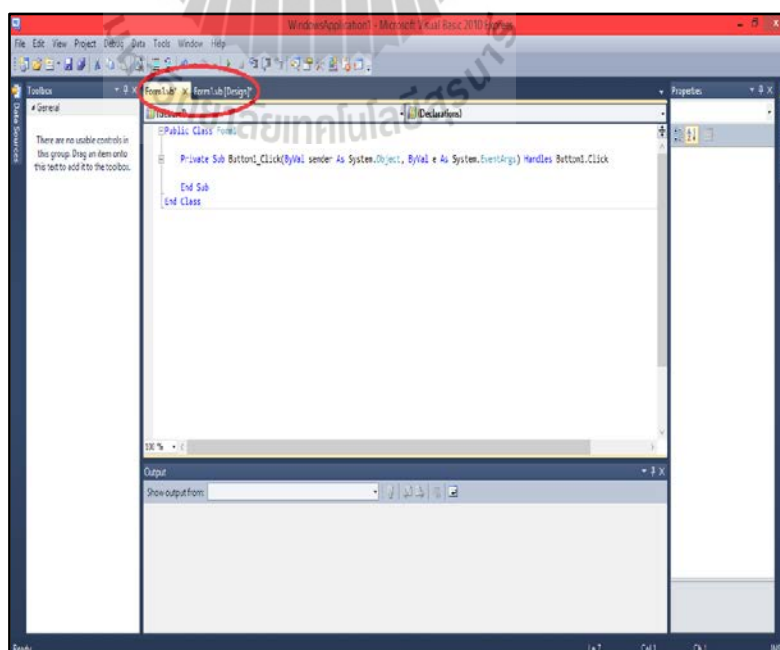
รูปที่ 3.18 แสดงรายละเอียดของฟอร์ม

- เขียนโค้ดควบคุมการทำงานของโปรแกรมโดยดับเบิลที่ Button1 เพื่อเปิด Editor ดังรูปที่ 3.19



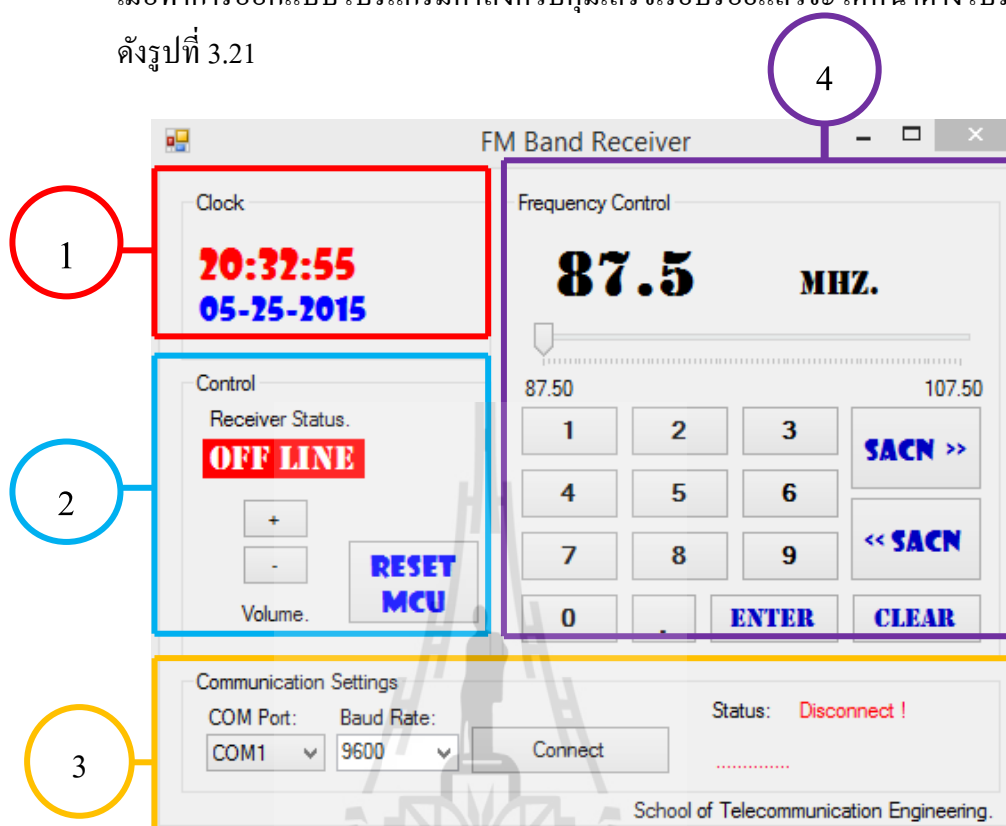
รูปที่ 3.19 แสดงหน้าต่างการเขียนโค้ดควบคุม

- จะปรากฏหน้าต่างโค้ด Button1 ที่คลิกเมื่อสักรู้และทำการกำหนดคำสั่งที่ต้องการลงไปดังรูปที่ 3.20



รูปที่ 3.20 แสดงหน้าต่างการกำหนดหรือแก้ไขโค้ดคำสั่ง

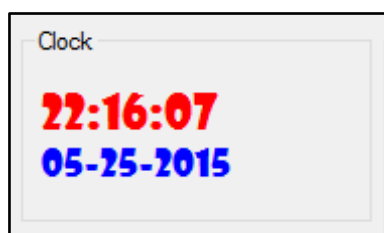
- เมื่อทำการออกแบบโปรแกรมคำสั่งควบคุมเสร็จเรียบร้อยแล้วจะได้หน้าต่างโปรแกรมดังรูปที่ 3.21



รูปที่ 3.21 แสดงหน้าต่างการออกแบบโปรแกรมคำสั่งควบคุม

การออกแบบหน้าต่างโปรแกรมคำสั่งควบคุมถูกแบ่งเป็นส่วนที่สำคัญดังนี้

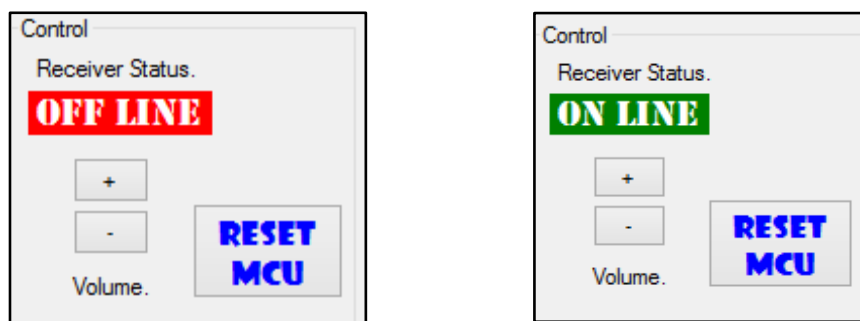
1. หน้าต่างแสดงผล Clock



รูปที่ 3.22 แสดงหน้าต่างแสดงผลส่วนที่หนึ่ง

ลักษณะการทำงาน Clock เป็นการบอกเวลาและวันที่ของโปรแกรมคำสั่งควบคุม

2. การควบคุมส่วน Control

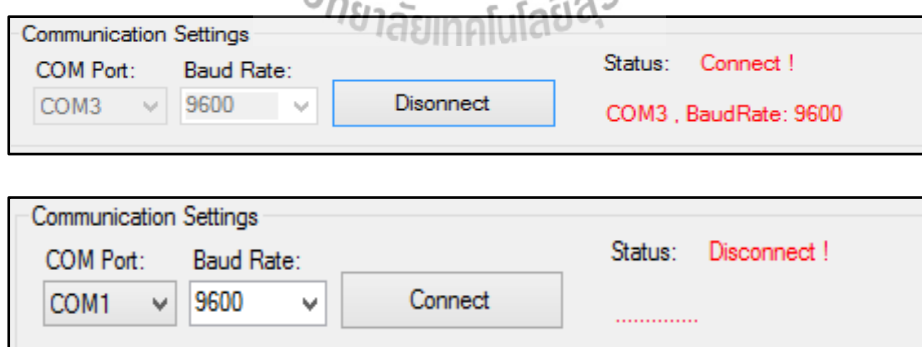


รูปที่ 3.23 แสดงหน้าต่างโปรแกรมคำสั่งควบคุมส่วนที่สอง

ลักษณะการทำงาน Control

เมื่อยังไม่มีการเชื่อมต่อผ่านระบบอินเทอร์เน็ตหรือการเชื่อมต่อล้มเหลวจะขึ้นสถานะ OFF LINE โดยพื้นหลังยังคงแสดงสีแดงแต่เมื่อใดที่มีการเชื่อมต่อผ่านระบบอินเทอร์เน็ตและมีการเชื่อมต่อสำเร็จจะขึ้นสถานะ ON LINE โดยพื้นหลังจะขึ้นเป็นสีเขียว เมื่อโปรแกรมการควบคุมเกิดข้อผิดพลาดหรือไม่ได้ยินคลื่นวิทยุ จะทำการกด RESET MCU เพื่อ Reset โปรแกรมให้กลับมายังความถี่เริ่มต้น 87.5 MHz หากต้องการเพิ่มเสียงหรือลดเสียง โปรแกรมซอฟต์แวร์จะทำการส่ง U (UP) และ D (DOWN) ไปยังโปรแกรมควบคุมเพื่อทำการเพิ่มเสียงและลดเสียง

3. การควบคุมส่วน Communication Setting

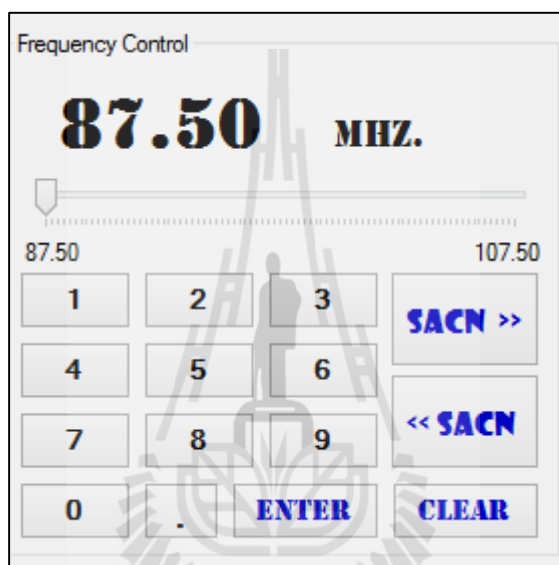


รูปที่ 3.24 แสดงหน้าต่างโปรแกรมคำสั่งควบคุมส่วนที่สาม

ลักษณะการทำงาน Communication Setting

ทำการเลือก Com Port ที่ใช้อยู่ในขณะนั้นและเลือก Baud Rate ที่ 9600 เพื่อต้องการส่งข้อมูลด้วยความเร็ว 9600 บิตต่อวินาที หลังจากนั้นทำการกด Connect เพื่อเชื่อมต่อกับระบบเครือข่าย อินเทอร์เน็ตถ้ามีการเชื่อมต่อสำเร็จจะปรากฏข้อความ “Status Connect! Com3, Baud Rate : 9600” แต่ถ้ามีการเชื่อมต่อไม่สำเร็จจะปรากฏข้อความ “Status Disconnect”

4. การควบคุมส่วนที่สี่ Frequency Control



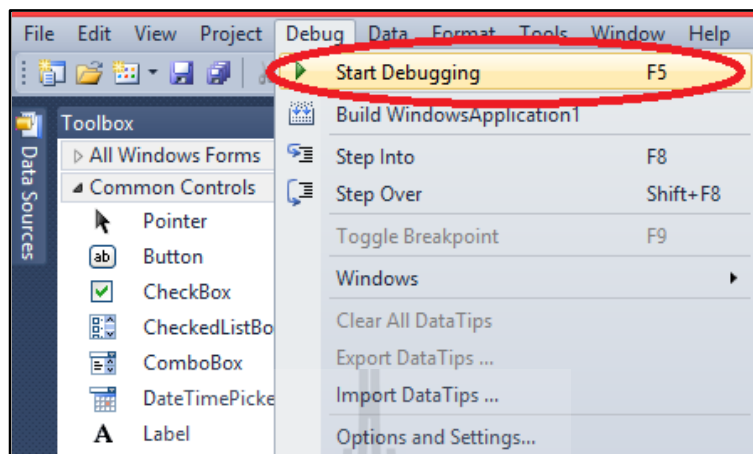
รูปที่ 3.25 แสดงหน้าต่างโปรแกรมคำสั่งควบคุมส่วนที่สี่

ลักษณะการทำงาน Frequency Control

เมื่อต้องการป้อนความถี่ให้ทำการกดตัวเลขลงไปซึ่งการป้อนความถี่จะต้องอยู่ในย่านความถี่ตั้งแต่ 87.50 MHz – 107.5 MHz เมื่อป้อนเสร็จตัวเลขที่ถูกป้อนจะโชว์บนหน้าจอ หลังจากนั้นทำการกดปุ่ม ENTER เพื่อยืนยันความถี่ที่ป้อนเข้ามา ในกรณีที่เกิดป้อนความถี่ผิดพลาดสามารถกดปุ่ม CLEAR เพื่อกลับไปป้อนความถี่ใหม่อีกครั้งได้และเพื่อความสะดวกในการหาค้นหาสามารถกดปุ่ม SCANUP เป็นการค้นหาสถานีวิทยุที่ออกอากาศอยู่ใน ณ ขณะนั้นแต่ในขณะค้นหาสถานีจะมีความถี่เพิ่มขึ้นจากสถานีเดิมขึ้นไปเรื่อยๆ ส่วน SCAN Down เป็นการค้นหาสถานีวิทยุที่ออกอากาศอยู่ใน ณ ขณะนั้น แต่ต่างตรงที่ความถี่ที่กำลังค้นหานั้นจะลดลงจากสถานีเดิมไปเรื่อยๆแต่เพื่อความสะดวกยิ่งขึ้นสามารถใช้แถบเลื่อนหาค้นหาวิทยุได้เช่นกัน

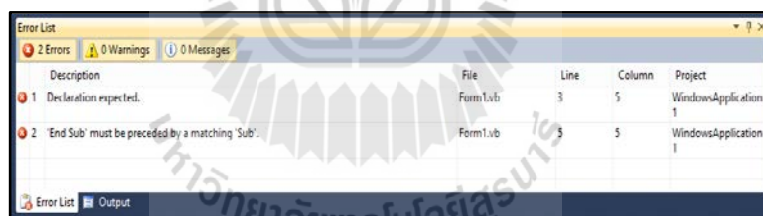
- ทำการรันโปรแกรมเพื่อทดสอบว่าโปรแกรมทำงานได้ตามที่ต้องการหรือไม่

โดยการเลือกที่ Debug => Start Debugging ดังรูปที่ 3.26



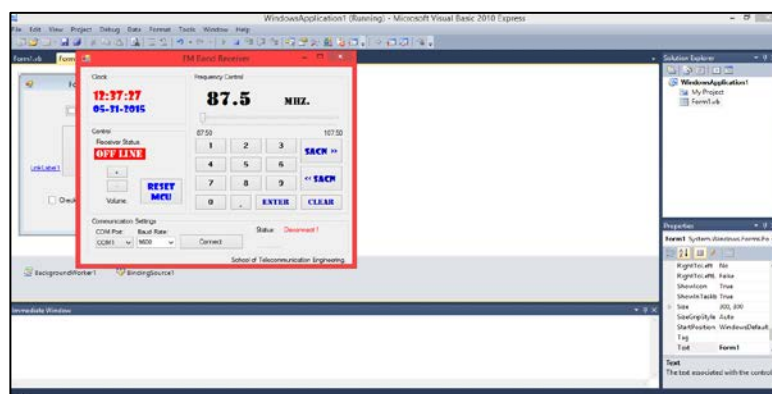
รูปที่ 3.26 แสดงการรันโปรแกรมเพื่อทดสอบ

- ถ้าเขียนโปรแกรมคำสั่งได้ถูกต้องและไม่มี Error ก็จะปรากฏข้อความ“Immediate Window” แต่ถ้าเขียนโปรแกรมคำสั่งไม่ถูกต้องและ Error ก็จะปรากฏข้อความ “Error List” ดังรูปที่ 3.27



รูปที่ 3.27 การรันโปรแกรมเพื่อทดสอบเมื่อเกิดการ Error

5. เมื่อทำการรันโปรแกรมเรียบร้อยแล้วจะได้หน้าต่างดังรูปที่ 28



รูปที่ 3.28 แสดงหน้าต่างเมื่อรันโปรแกรมเสร็จเรียบร้อยแล้ว

บทที่ 4

การใช้งานและการทดสอบอุปกรณ์ต้นแบบ

4.1 กล่าวนำ

ในบทนี้จะกล่าวถึงการใช้งานและผลการทดสอบของอุปกรณ์ต้นแบบที่สร้างขึ้น ซึ่งอุปกรณ์ต้นแบบนี้จะประกอบไปด้วย FM Module ที่ทำหน้าที่รับสัญญาณวิทยุในย่านความถี่ 87.50-107.50 MHz บอร์ดไมโครคอนโทรลเลอร์ Arduino ทำหน้าที่ในการควบคุมการทำงานของ FM Module โดยมีโปรแกรมคอมพิวเตอร์ที่ได้รับการพัฒนาขึ้นทำหน้าที่ในการสั่งงานให้อุปกรณ์ต้นแบบทำการรับความถี่วิทยุหากต้องการควบคุมและสั่งงานอุปกรณ์ต้นแบบนี้ในระยะไกลสามารถทำได้โดยใช้โปรแกรม Team Viewer ในการเชื่อมต่อผ่านระบบเครือข่ายอินเทอร์เน็ต การทดสอบอุปกรณ์ต้นแบบทางผู้จัดทำจะทำการทดสอบ โดยแบ่งการทดสอบออกเป็น 3 กรณี คือ 1.ทดสอบรับคลื่นสัญญาณวิทยุของอุปกรณ์ต้นแบบเปรียบเทียบกับเครื่องรับวิทยุทั่วไป 2.การทดสอบการค้นหาสถานีแบบอัตโนมัติของอุปกรณ์ต้นแบบเปรียบเทียบกับวิทยุทั่วไป 3.การทดสอบการใช้งานควบคุมระยะไกลโดยควบคุมสั่งงานผ่านทางโปรแกรม Team Viewer โดยการติดตั้งและการทดสอบจะกล่าวถึงในหัวข้อต่อไป

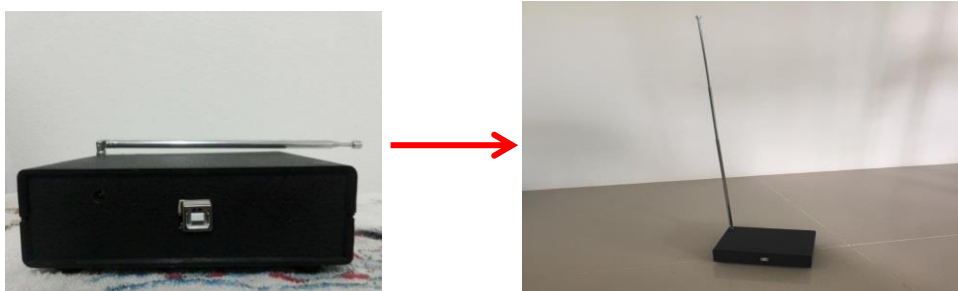
4.2 การติดตั้งและการใช้งานอุปกรณ์รับสัญญาณวิทยุ

ก่อนจะใช้งานต้องมีการติดตั้งอุปกรณ์ต่างๆ โดยอุปกรณ์ต้นแบบนี้สามารถรับคลื่นความถี่วิทยุความถี่ตั้งแต่ 87.50-107.50 MHz และจะทดสอบว่าในบริเวณนั้นๆสามารถรับความถี่อะไรได้บ้าง โดยอุปกรณ์ที่จะใช้งานรวมกันทั้งหมดมีดังนี้ 1.เครื่องรับวิทยุสื่อสารต้นแบบ 2.เครื่องคอมพิวเตอร์ที่มีโปรแกรมที่ได้พัฒนาขึ้นสำหรับสั่งงานเครื่องรับ 3.สาย USB สำหรับเชื่อมต่อ 4.สาย Audio Output สำหรับรับสัญญาณเสียงที่ได้จากเครื่องรับ ซึ่งขั้นตอนการติดตั้งและการใช้งานอุปกรณ์ต้นแบบมีดังต่อไปนี้

1.เลือกพื้นที่ที่ต้องการตรวจสอบซึ่งพื้นที่นี้ ต้องเป็นลักษณะพื้นที่โล่งไม่อับสัญญาณ ไม่อยู่ในตึกหรืออาคาร



2. ยืดสายอากาศที่ติดอยู่กับกล่องอุปกรณ์ต้นแบบขึ้น เพื่อให้สามารถรับสัญญาณได้ดีขึ้น



รูปที่ 4.1 การยืดสายอากาศ

3. เชื่อมต่อสาย Audio Output จากอุปกรณ์ต้นแบบเข้ากับเครื่องคอมพิวเตอร์ ดังรูปที่ 4.2



(ก)

(ข)

รูปที่ 4.2 (ก) การเชื่อมต่อสาย Audio Output ที่อุปกรณ์ต้นแบบ

(ข) การเชื่อมต่อสาย Audio Output เข้ากับช่อง line-in ของคอมพิวเตอร์

4. เชื่อมต่อสาย USB จากอุปกรณ์ต้นแบบเข้ากับเครื่องคอมพิวเตอร์ ดังรูปที่ 4.3



(ค)

(ง)

รูปที่ 4.3

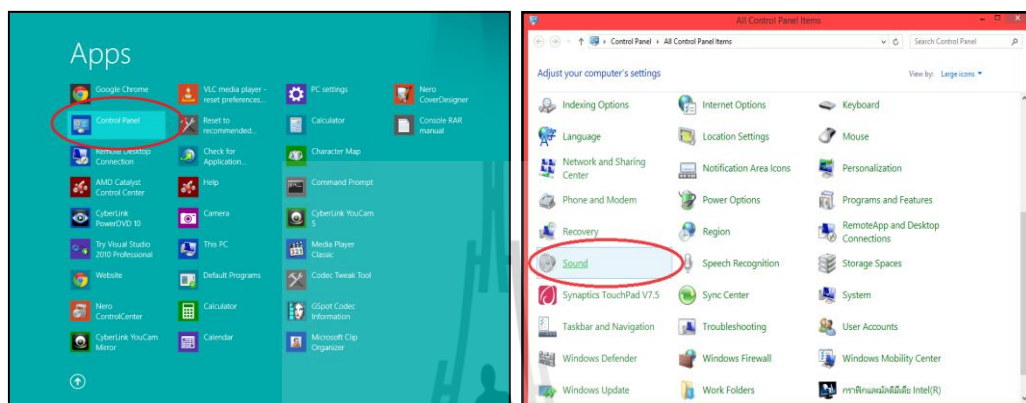
(ค) การเชื่อมต่อสาย USB ที่อุปกรณ์ต้นแบบ

(ง) การเชื่อมต่อสาย USB เข้ากับเครื่องคอมพิวเตอร์

5.ทำการต่อสาย Audio Output จากอุปกรณ์ต้นแบบเข้ากับช่อง line-in ของคอมพิวเตอร์เพื่อให้เครื่องคอมพิวเตอร์รับเสียงจาก FM Module ได้

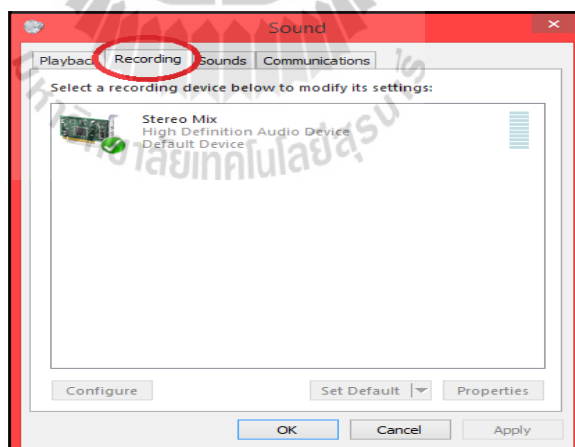
การตั้งค่าให้เครื่องคอมพิวเตอร์สามารถรับสัญญาณเสียงจาก FM Module ได้โดยตั้งค่าดังนี้

- คลิก Start จากนั้นคลิก Control Panel แล้วทำการค้นหาโหมด Sound



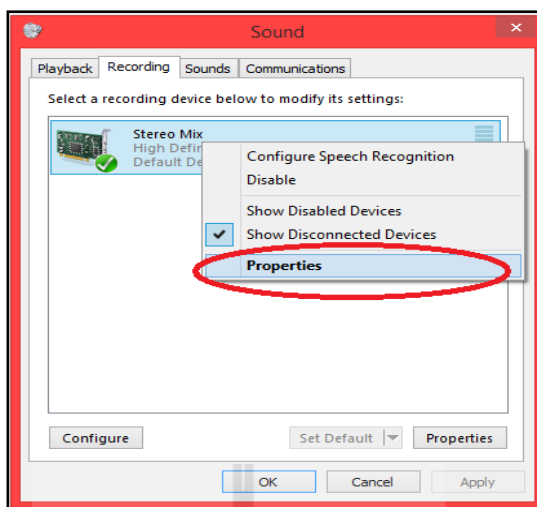
รูปที่ 4.4 แสดงการเลือกโหมด

- เมื่อคลิกเข้ามาจะปรากฏหน้าต่างโหมด Sound ให้เลือกแถบ Recording จะปรากฏชื่ออุปกรณ์เสียง (ไอคอนอาจแตกต่างกันไปจากนี้อยู่ที่การตั้งค่าของเครื่องคอมพิวเตอร์)



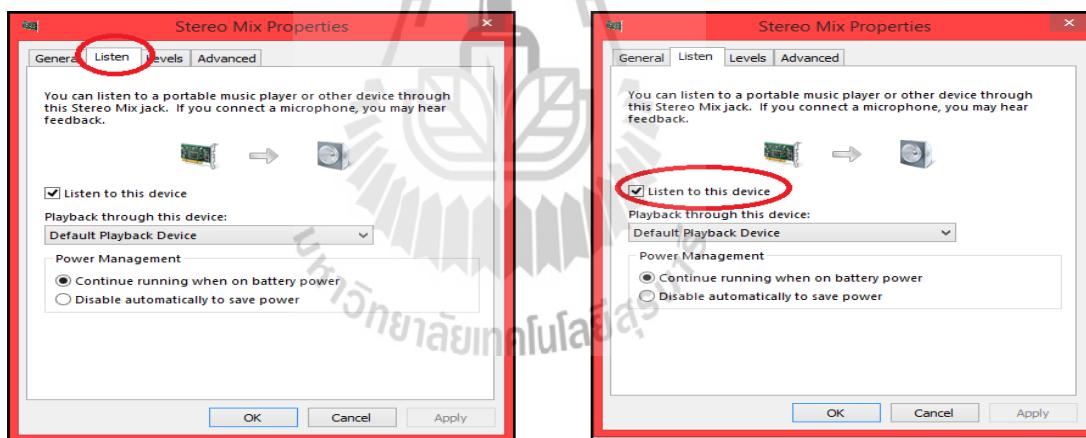
รูปที่ 4.5 แสดง Stereo Mix

- หลังจากนั้นทำการคลิกขวาที่ชื่ออุปกรณ์จากนั้นเลือก Properties (คุณสมบัติ) เพื่อต้องการตั้งค่าให้มีเสียงออกมาจากข้างใน Computer



รูปที่ 4.6 การตั้งค่าเสียงของคอมพิวเตอร์

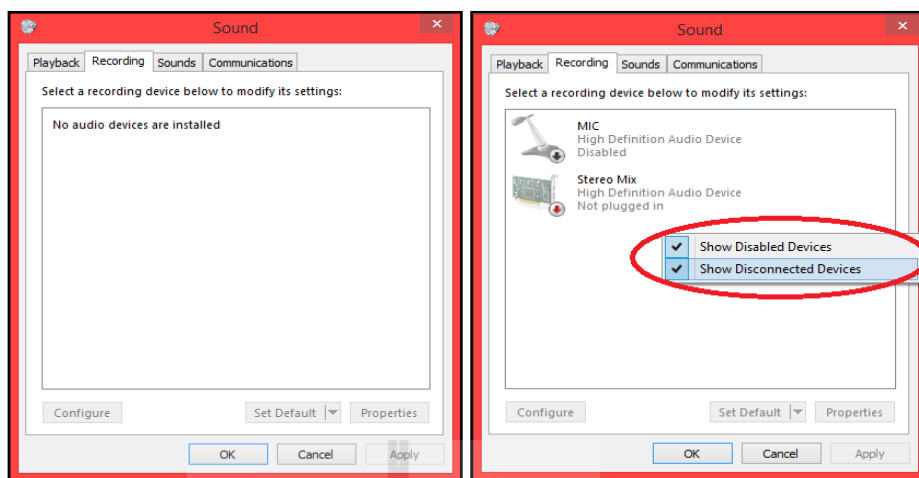
- เมื่อทำการเลือก Properties (คุณสมบัติ) แล้วให้เลือกแถบ Listen หลังจากนั้นใส่เครื่องหมายถูกหน้าข้อความ Listen to this Device เพื่อเป็นการบอกว่าต้องการให้มีเสียงออกมาจากข้างใน Computer



รูปที่ 4.7 การตั้งค่าเสียงให้ดัง

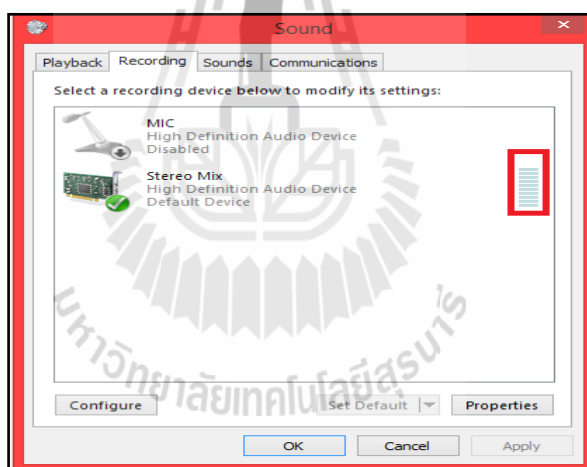
ในกรณีที่ชื่ออุปกรณ์เสียงไม่ปรากฏ ให้ตั้งค่านั้นต่อไปนี้

- ให้ทำการคลิกขวาแล้วเลือก Show Disabled Devices และ Show Disconnected Devices ดังรูปที่ 4.8



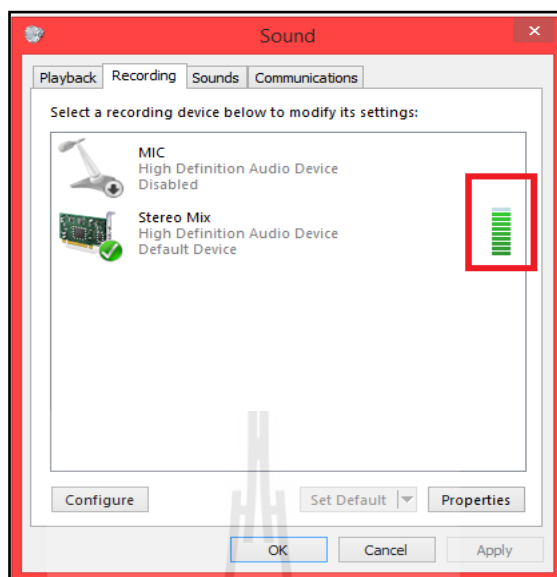
รูปที่ 4.8 ตั้งค่าเสียงในกรณีที่ชื่ออุปกรณ์ไม่ปรากฏ

- เมื่อชื่ออุปกรณ์เสียงปรากฏให้ทำการทดสอบ line-in โดยส่งสัญญาณไปยังช่อง line-in จากแหล่งที่มาของอุปกรณ์เสียง ให้ดูที่แถบสีเทาข้างชื่ออุปกรณ์



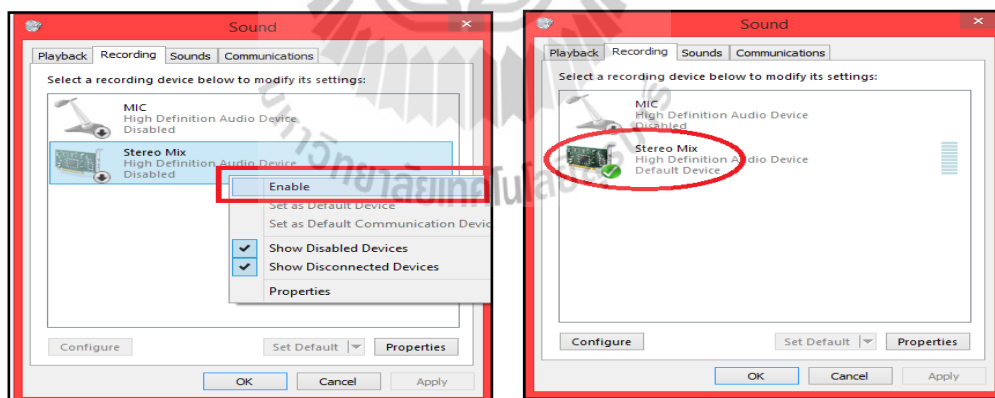
รูปที่ 4.9 แสดงการทดสอบ line-in

- ถ้าแถบสีเทาข้างชื่ออุปกรณ์เปลี่ยนเป็นสีเขียว นั่นหมายความว่าอุปกรณ์นี้ทำงานแล้วและสัญญาณเสียงถูกส่งไปยังช่อง line-in



รูปที่ 4.10 การเลือก line-in

- ถ้าแถบสีเทายังคงเป็นสีเทาไม่ทำงานนั้นหมายถึงอุปกรณ์จับสัญญาณไม่ได้คลิกขวาที่ชื่ออุปกรณ์เสียงจากนั้นเลือก Enable (เปิดใช้งาน) โดยชื่ออุปกรณ์เสียงจะปรากฏเป็นเครื่องหมายถูกสีเขียว



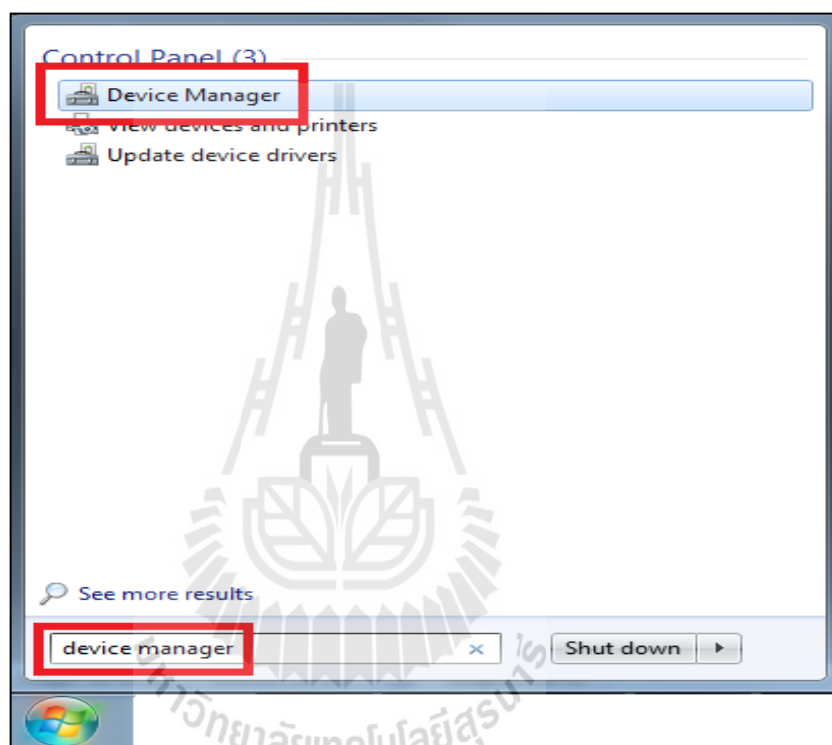
รูปที่ 4.11 เลือก line-in ไม่ได้

6.ทำการเชื่อมต่อสาย USB เข้ากับเครื่องคอมพิวเตอร์

ในการสั่งงานบอร์ด Arduino จำเป็นจะต้องสื่อสารทาง Serial Port ผ่านทางสาย USB ซึ่งการเชื่อมต่อ Serial Port จะแบ่งเป็น 2 กรณี

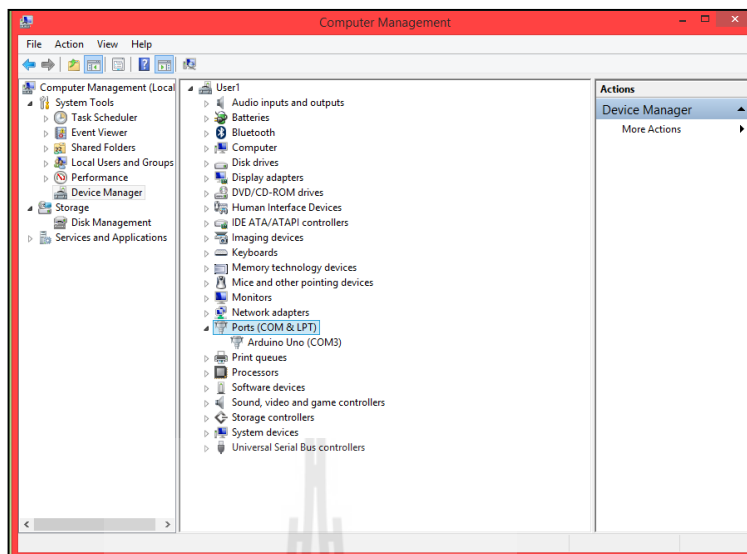
กรณีที่ 1 เครื่องคอมพิวเตอร์ที่มี Serial Port หลายช่อง ในขั้นตอนที่ต้องเลือก Port สำหรับเชื่อมต่อ Arduino Uno จะปรากฏ Com Port ขึ้นมาหลายอัน ถ้าหากเป็นกรณีนี้ให้ตรวจสอบว่า Arduino Uno ของเรากำลังอยู่กับ Com Port ไหน

- ให้ไปที่ Device Manager โดยพิมพ์คำว่า“device manager” ลงในช่องค้นหาของ window-7 หรือ สามารถเข้าได้จาก Control Panel ก็ได้เช่นกัน



รูปที่ 4.12 วิธีการติดตั้ง Device (1)

- เมื่อเข้ามาที่ Device Manager ให้เลือกหา Ports (COM & LPT)
- เมื่อเลือกที่ Ports (COM & LPT) ให้มองหา USB Serial Ports แล้วสังเกตว่า USB Serial Ports เป็น COM หมายเลขอะไร (ในรูปเป็น COM3)
- เมื่อรู้ว่า Arduino Uno ของเรากำลังอยู่กับ Ports อะไรให้กลับไปเลือก Ports ให้ตรงกัน เช่น Tools => Port => COM3 เป็นต้น



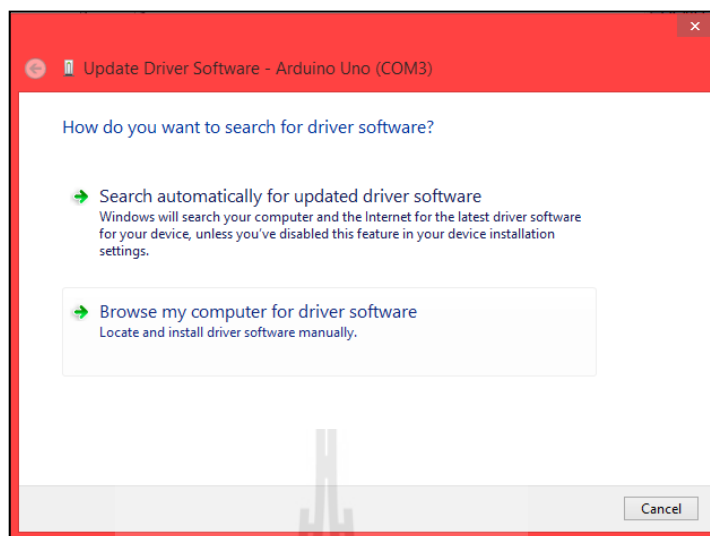
รูปที่ 4.13 วิธีการติดตั้ง Device (2)

กรณีที่ 2 Serial Port ไม่สามารถเชื่อมต่อกับคอมพิวเตอร์ได้

อาจจะเกิดจากหลายสาเหตุ ยกตัวอย่างเช่น ยังไม่ได้ทำการลง Device ของ Serial Port ชนิดนั้นๆ หรือสาย USB อาจจะเกิดการหลวมหรือเชื่อมต่อไม่แน่น วิธีการจัดการขั้นต้นคือ

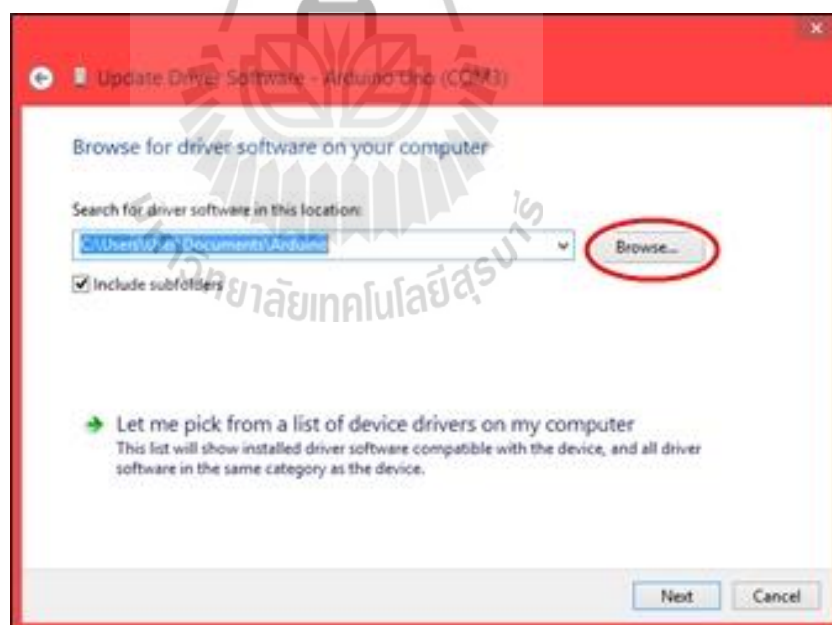
- ตรวจสอบว่าสาย USB แน่นหรือไม่
- ตรวจสอบที่เครื่องอุปกรณ์ว่ามีไฟ LED แสดงสถานะ ว่าบอร์ด Arduino ทำงานหรือไม่
- ทำการลง Device โดยมีขั้นตอนการลงดังนี้

- 1.เข้ามาที่ Device Manager ให้เลือกหา Ports (COM&LPT)
- 2.ทำการหา Ports (COM & LPT) => คลิกขวาเลือก Update Driver Software
- 3.จากนั้นให้คลิกขวาที่ Arduino จะพบกับ "Update Driver Software"
- 4.แล้วให้เลือก "Browse my computer for Driver software"



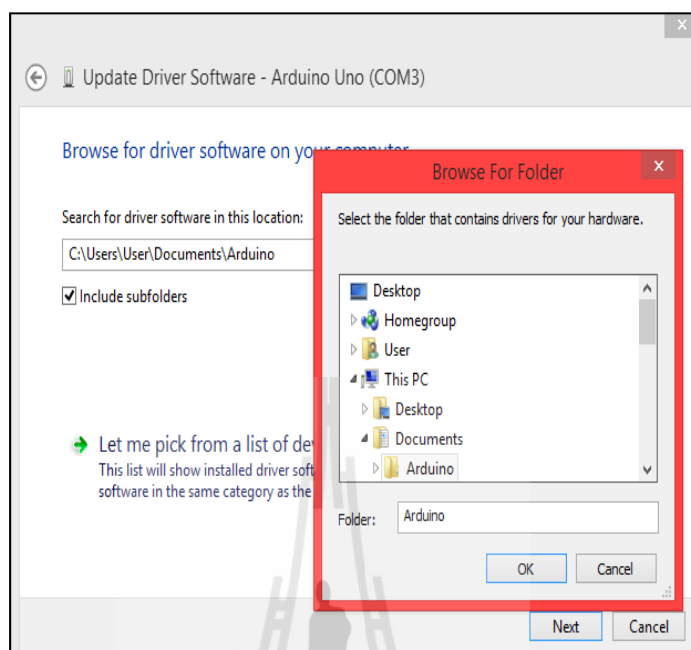
รูปที่ 4.14 ทำการลง Device

5.เลือกตำแหน่ง Device แล้วคลิก Browse ตำแหน่งของ File Device



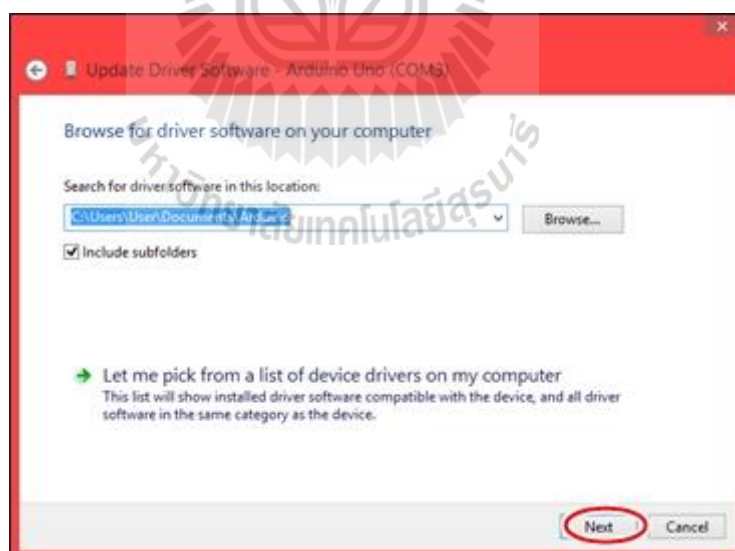
รูปที่ 4.15 การเลือกตำแหน่ง File

6.ตำแหน่ง Device จะอยู่ในโฟลเดอร์โปรแกรม Arduino ดังรูป แล้วกด OK



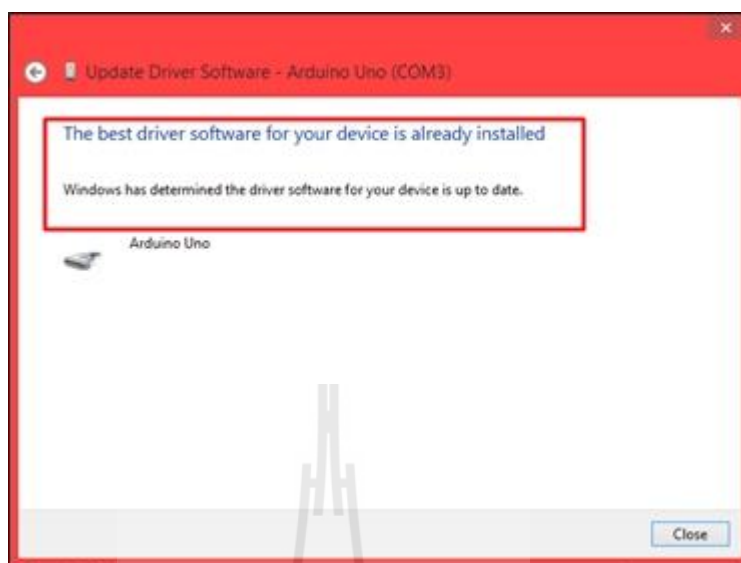
รูปที่ 4.16 ปรากฏตำแหน่ง File

7.เมื่อทำการเลือก File Device แล้ว ให้คลิกที่ Next



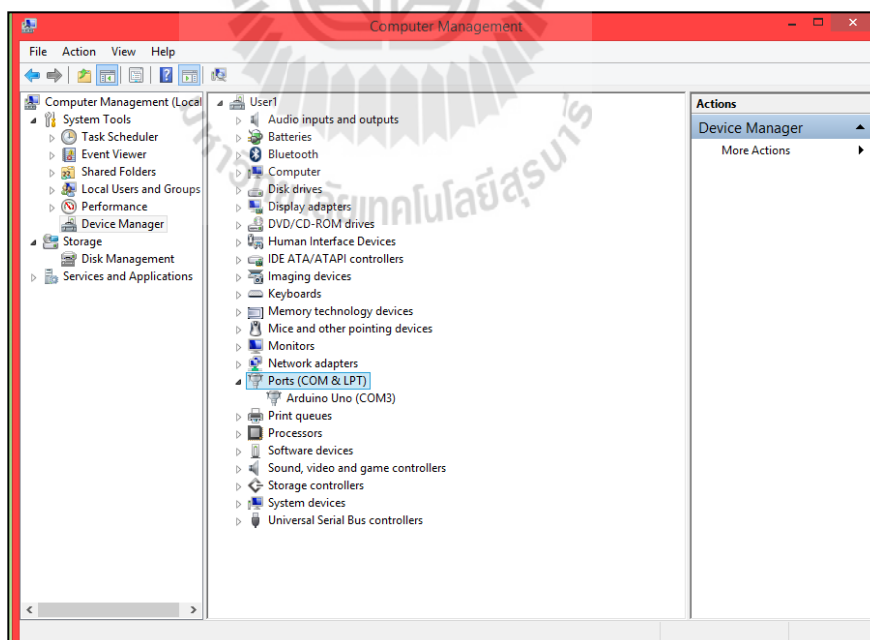
รูปที่ 4.17 กดเลือกคำสั่งดำเนินการต่อ

8.จากนั้นกรุณารอสักครู่ เมื่อเสร็จสิ้นการติดตั้งแล้วจะปรากฏข้อความ ดังนี้



รูปที่ 4.18 แสดงการติดตั้งสำเร็จ

9.เมื่อทำการติดตั้งเสร็จสมบูรณ์แล้ว สามารถไปดูใน Device Manager จะเห็นรายการ Arduino UNO พร้อมหมายเลข Com Port ที่ใช้ในการเชื่อมต่อได้ ดังรูป

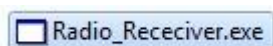


รูปที่ 4.19 แสดง Com Port

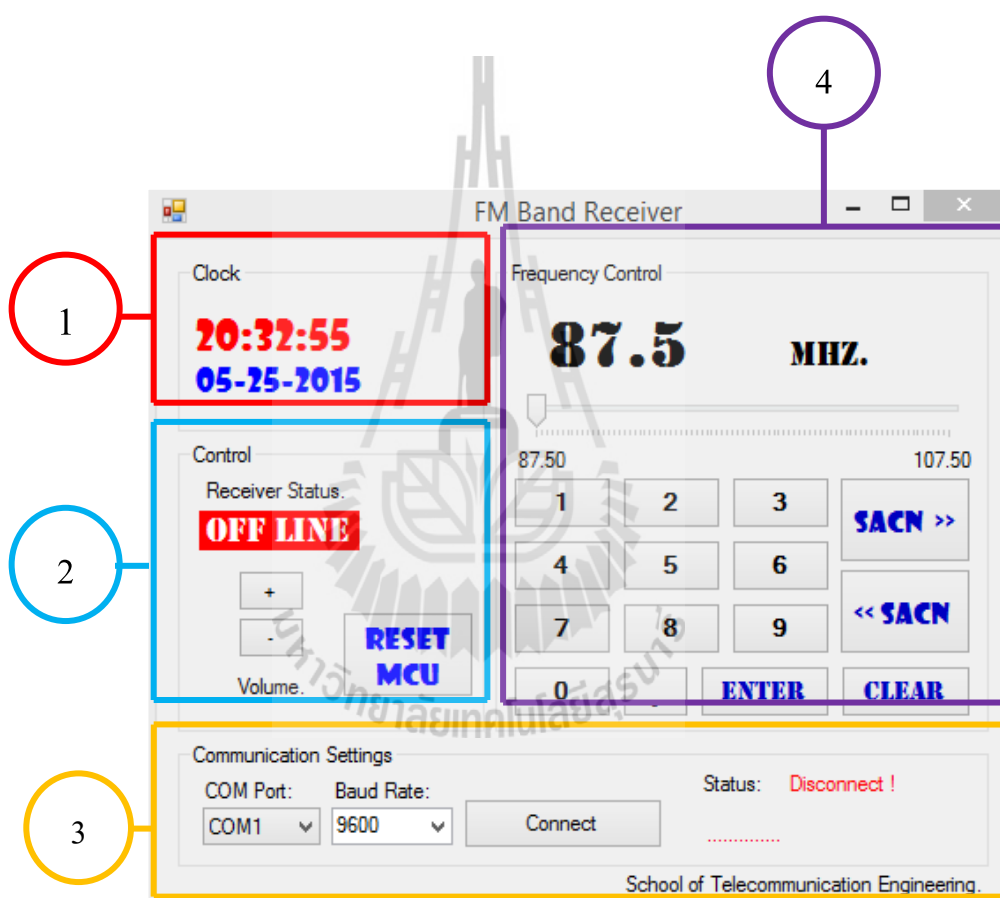
4.3 การใช้งาน

การใช้งานในการสั่งงานควบคุมอุปกรณ์ต้นแบบจำเป็นต้องมีโปรแกรมคอมพิวเตอร์เพื่อใช้ในการสั่งงาน โดยมีขั้นตอนการใช้งานดังต่อไปนี้

1.เปิดโปรแกรมที่ Radio Rcceiver ได้พัฒนาขึ้นจากไอคอนนี้



2.เมื่อทำการเปิดโปรแกรมจะเห็นหน้าต่างแสดงผลที่ได้สร้างขึ้น ดังรูป



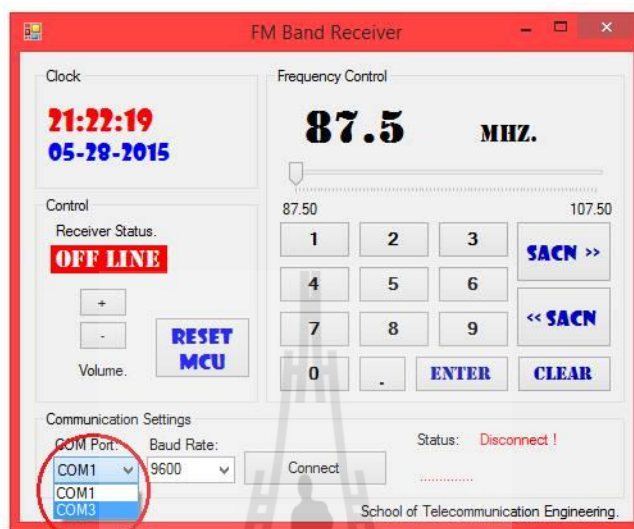
รูปที่ 4.20 แสดงหน้าต่างการออกแบบโปรแกรมคำสั่งควบคุม

โดยมีฟังก์ชันแสดงหน้าต่างการใช้งานดังนี้

- หน้าต่างแสดงผลส่วนที่หนึ่ง Clock
- การควบคุมส่วนที่สอง Control
- การควบคุมส่วนที่สาม Communication Setting
- การควบคุมส่วนที่สี่ Frequency Control

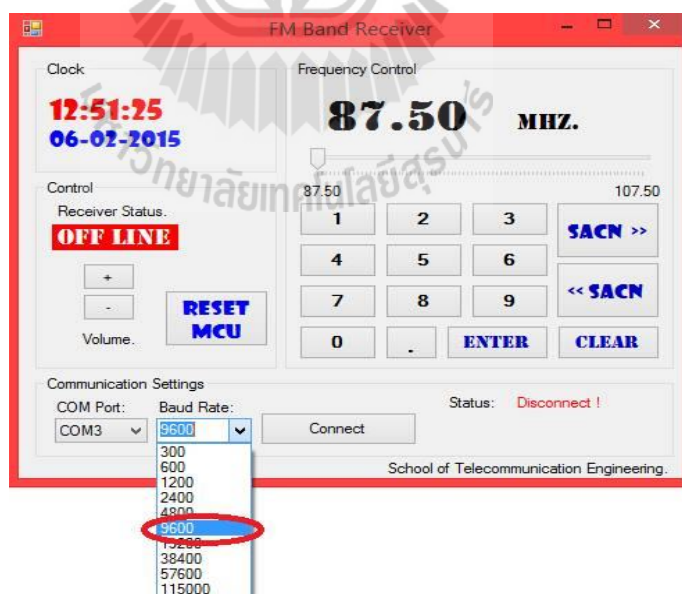
3.ทำการเชื่อมต่อโปรแกรม Radio_Receiver กับ โปรแกรม Arduino

โดยกดเลือก COM Port => COM 3 (โดย COM Port ที่แสดงใน Serial Port ของคอมพิวเตอร์จากข้อ 4)



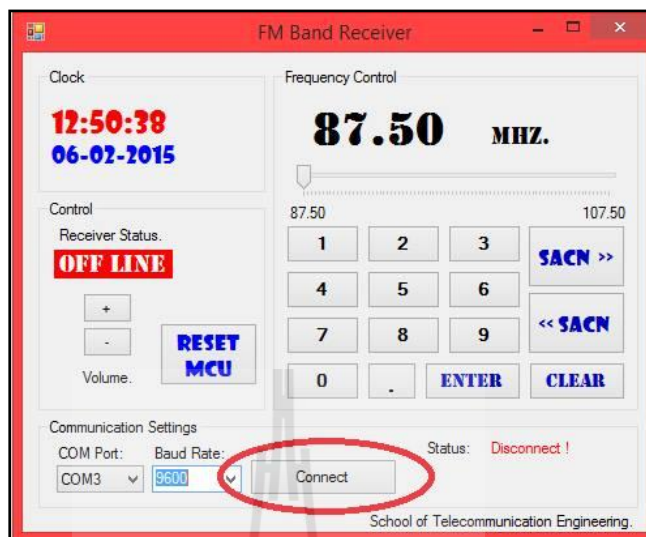
รูปที่ 4.21 การเลือก COM Port

4.เลือก Baud Rate ที่ 9600 ดังรูป



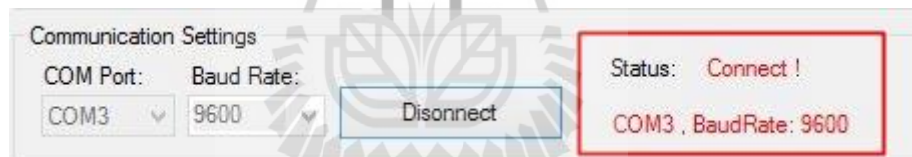
รูปที่ 4.22 การเลือก Baud Rate

5.แล้วทำการเลือก Connect



รูปที่ 4. 23 การเลือก Connect

6.ถ้าเชื่อมต่อสำเร็จจะปรากฏข้อความ “Status Connect! Com3, Baud Rate : 9600” ดังรูป



รูปที่ 4.24 แสดง Status การเชื่อมต่อ

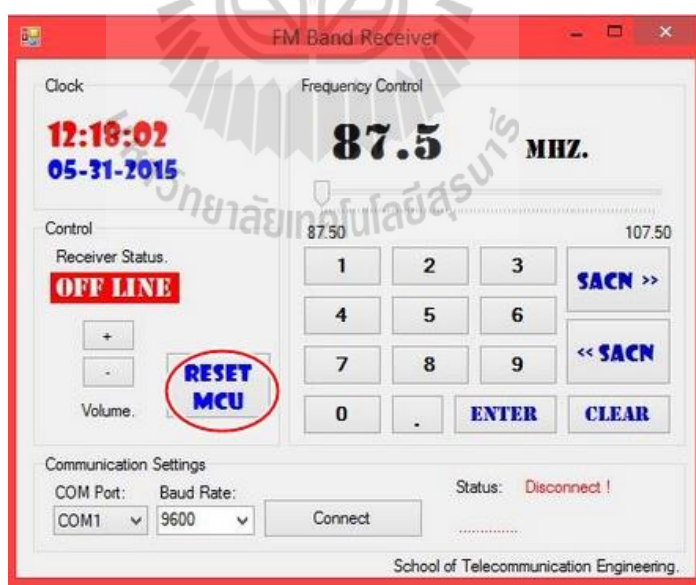
12.เมื่อทำการเชื่อมต่อกับอุปกรณ์เครื่องรับวิทยุได้จะขึ้นสถานะ Receiver Status เป็น “ON LINE” แสดงว่าสามารถติดต่อสื่อสารกับเครื่องรับวิทยุได้



รูปที่ 4.25 แสดงสถานะ Receiver Status

กรณีที่ติดต่อสื่อสารกับเครื่องรับวิทยุไม่สำเร็จจะขึ้นสถานะของ Receiver Status เป็น “OFF LINE”

ในกรณีที่เครื่องรับทำงานไม่ถูกต้องและเมื่อต้องการที่จะเริ่มระบบใหม่ ให้ทำการกด RESET MCU หรือให้ทำการเชื่อมต่อ Serial Port ใหม่อีกครั้ง

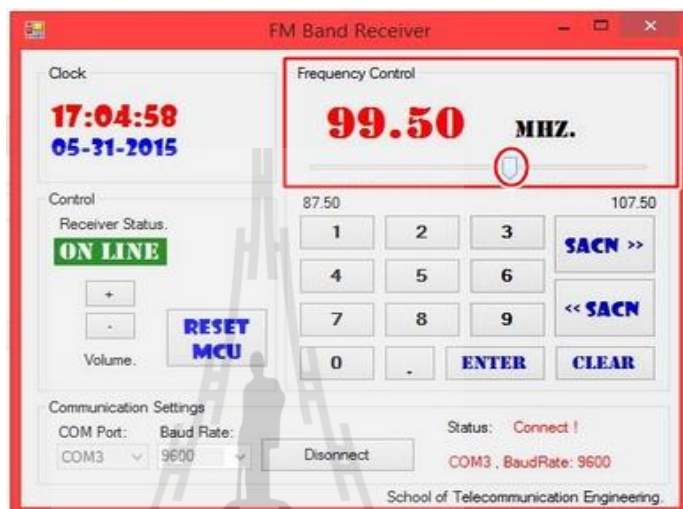


รูปที่ 4.26 การเริ่มระบบใหม่

4.4 การใช้งานเครื่องรับวิทยุ

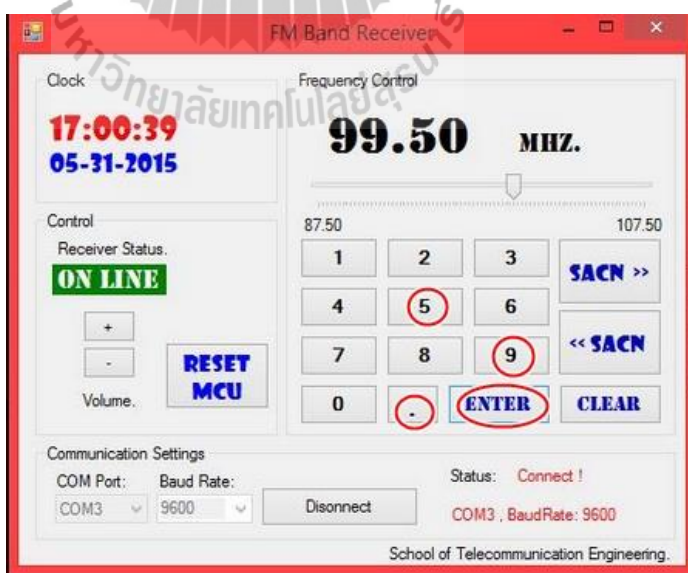
การใช้งานเครื่องรับวิทยุโดยสั่งงานผ่าน โปรแกรมที่ได้พัฒนาขึ้นโดยจะมีรูปแบบการเลือกคลื่นความถี่หลายรูปแบบ ซึ่งในตัวโปรแกรมนี้อาจแบ่งการเลือกเป็น 3 แบบ คือ

1.สามารถเลื่อนแทบความถี่ที่ต้องการรับสัญญาณได้โดยเลือกจากแทบความถี่ ซึ่งแทบความถี่จะเปลี่ยนไปตามแทบความถี่ที่ต้องการ



รูปที่ 4.27 แสดงการเลื่อนแทบความถี่

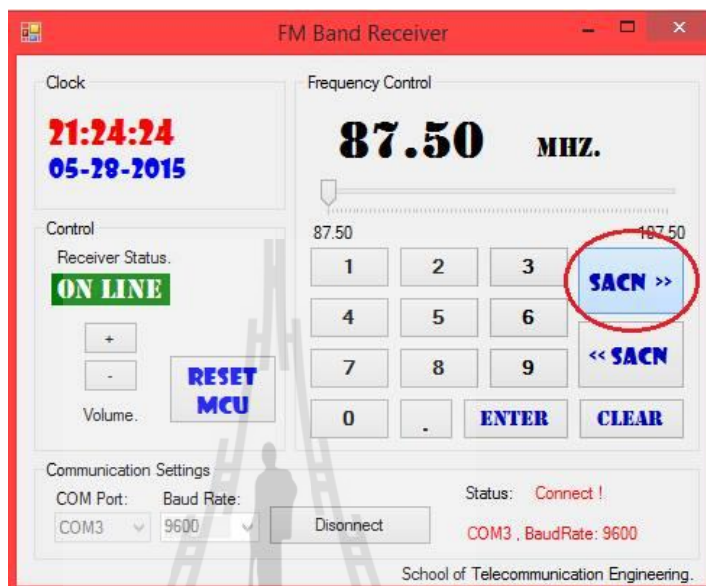
2.สามารถระบุความถี่ที่ต้องการรับสัญญาณได้โดยการป้อนตัวเลขความถี่



รูปที่ 4.28 แสดงการป้อนตัวเลขความถี่

3.สามารถเลือกค้นหาสถานีวิทยุแบบอัตโนมัติ

โดยการเลือกแบบอัตโนมัติทำได้ 2 วิธีคือ การค้นหาช่องความถี่ถัดไปจากความถี่ปัจจุบัน คือ SCAN UP หรือ การค้นหาสถานีก่อนที่จะถึงสถานีปัจจุบัน คือ SCAN DOWN



รูปที่ 4.29 แสดง SCAN UP



รูปที่ 4.30 แสดง SCAN DOWN

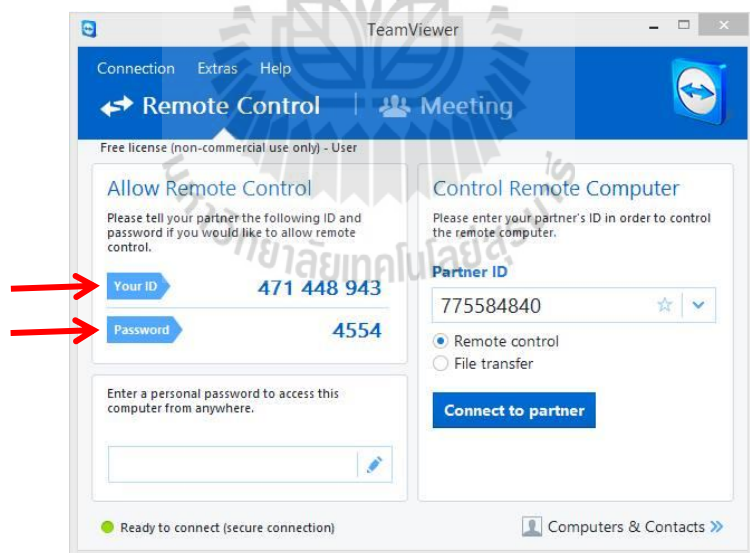
4.5 การใช้งานโปรแกรม Team Viewer

โปรแกรม Team Viewer เป็นโปรแกรมสำหรับการใช้ในการเชื่อมต่อคอมพิวเตอร์จากระยะไกลที่มีความสำคัญอีกโปรแกรมเนื่องจากโปรแกรมที่ดีและมีระบบความปลอดภัยที่น่าเชื่อถือและที่สำคัญอีกอย่างคือสามารถใช้งานผ่านทางอินเทอร์เน็ตโดยไม่ต้องทำการคอนฟิกโปรแกรมอื่นๆให้ยุ่งยากและยังสามารถใช้โปรแกรมตัวนี้เข้าไปควบคุมเครื่องที่กำลังติดต่อกับกับเราได้อีกซึ่งมีวิธีการใช้งานดังนี้

1. ทำการติดตั้งโปรแกรม team viewer หลังติดตั้งเสร็จจะมีไอคอนปรากฏไอคอนขึ้นดังรูปทำการคลิกไปที่ไอคอนตาม  รูปโดยที่เครื่องคอมพิวเตอร์นี้ต้องมีการเชื่อมต่อผ่านเครือข่ายอินเทอร์เน็ต

2. เมื่อเปิดโปรแกรมขึ้นมาจะมีหน้าต่างโปรแกรมตามรูปที่ 4.3.1

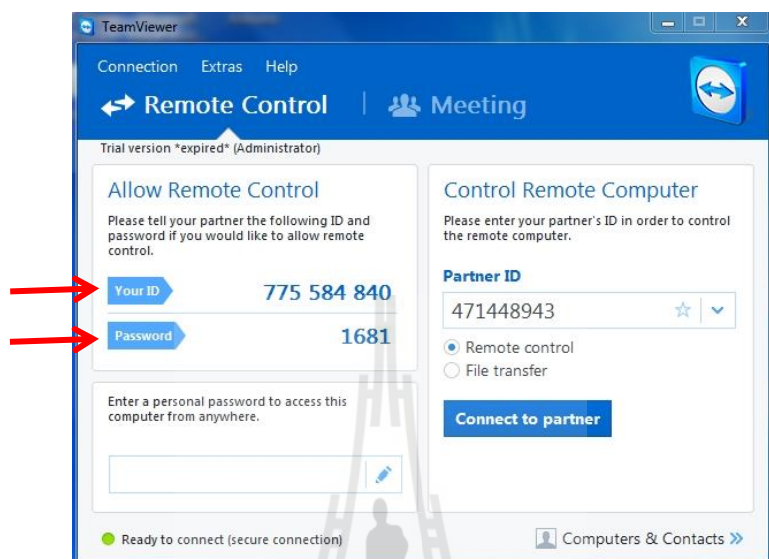
- ส่วนแรกผู้ใช้งานคอมพิวเตอร์ที่ติดตั้งอยู่กับเครื่องรับวิทยุสื่อสารให้สังเกตที่ Allow Remote Control รอจนกว่าระบบจะทำการรันรหัสประจำเครื่อง และ รหัสผ่าน มาให้ดังรูปที่ 31



รูปที่ 4.31 หน้าต่างโปรแกรม Team Viewer (1)

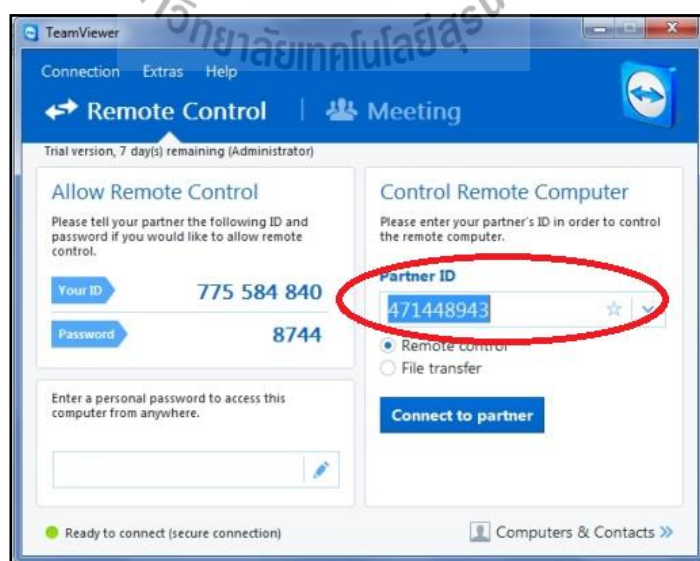
- ส่วนที่ 2 เครื่องคอมพิวเตอร์สำหรับที่จะไปควบคุมเครื่องรับวิทยุสื่อสารนั้นจำเป็นต้องลงโปรแกรม Team Viewer และทำการเปิดโปรแกรมเช่นเดียวกับส่วนที่หนึ่งโดยจะปรากฏ รหัสประจำเครื่อง และ รหัสผ่าน ที่ Allow Remote Control เพื่อบ่งบอกว่า

เครื่องคอมพิวเตอร์คนละเครื่องกับเครื่องคอมพิวเตอร์ที่ติดตั้งอยู่กับเครื่องรับวิทยุสื่อสาร



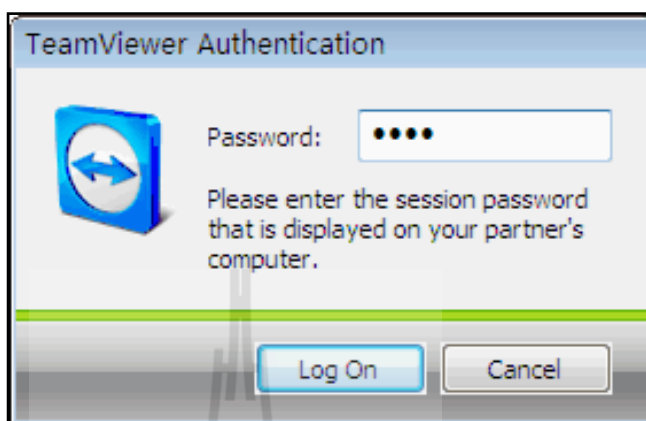
รูปที่ 4.32 หน้าต่างโปรแกรม Team Viewer (2)

3. หากต้องการควบคุมเครื่องคอมพิวเตอร์ที่ติดตั้งอยู่กับเครื่องรับวิทยุสื่อสารจำเป็นต้องป้อนรหัสประจำเครื่องของเครื่องคอมพิวเตอร์ที่ติดตั้งอยู่กับเครื่องรับวิทยุสื่อสารให้ใส่ลงในช่อง Partner จากนั้นคลิกที่ Remote Support และทำการคลิกปุ่ม Connect to partner รอให้โปรแกรมทำการเชื่อมต่อดังรูปที่ 4.33



รูปที่ 4.33 แสดงหน้าต่างการใช้งาน Control Remote Computer

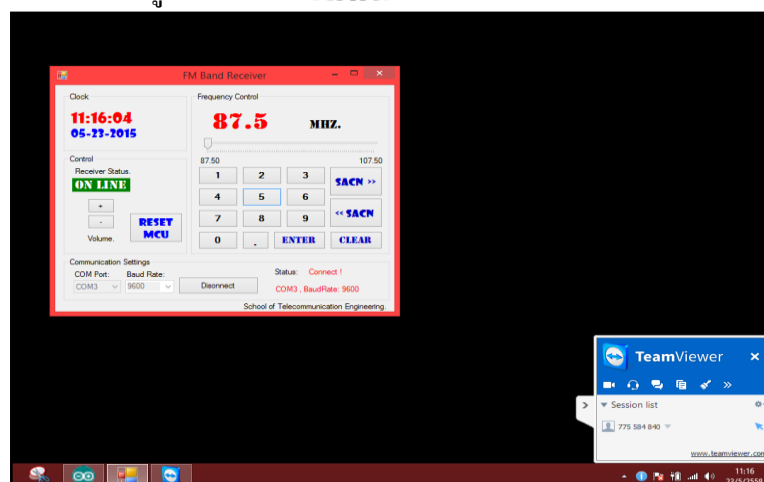
4. หากเครื่องคอมพิวเตอร์ที่ติดตั้งอยู่กับเครื่องรับวิทยุสื่อสารมีระบบความปลอดภัยจะให้ใส่รหัสผ่าน โดยจะขึ้นหน้าต่าง Team Viewer Authentication จึงทำการใส่รหัสผ่านหลังจากนั้นคลิกปุ่ม Log On เพื่อเชื่อมต่อดังรูปที่ 4.34



รูปที่ 4.34 แสดงหน้าต่าง Team Viewer Authentication

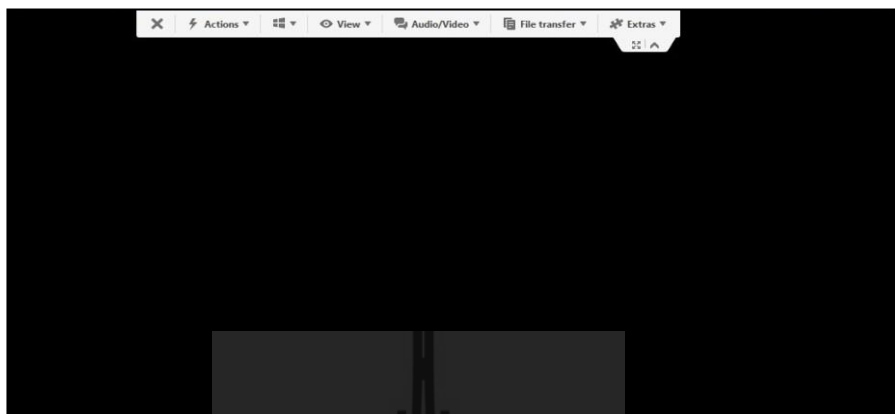
หลังจากทำการเชื่อมต่อเสร็จก็สามารถใช้งานต่างๆเหมือนกับการนั่งทำงานหน้าเครื่องที่ติดตั้งอยู่กับเครื่องรับวิทยุสื่อสารโดยจะมีลักษณะการใช้งานและการควบคุมดังนี้

5. เมื่อทำการเชื่อมต่อเสร็จหน้าจอของเครื่องคอมพิวเตอร์ที่ติดตั้งอยู่กับเครื่องรับวิทยุสื่อสารจะสังเกตได้ว่าพื้นหลัง (Background) จะเปลี่ยนเป็นสีดำนั้นแสดงว่าเครื่องที่กำลังควบคุมเชื่อมต่อสำเร็จจะมีหน้าต่างด้านล่าง แสดงสถานะการเชื่อมต่อโดยเครื่องที่ควบคุมจะสามารถควบคุมการทำงานเหมือนเครื่องคอมพิวเตอร์นั้นและยังสามารถได้ยินเสียงออกซิโจากเครื่องที่ติดตั้งอยู่กับเครื่องรับวิทยุสื่อสารรับได้ รูปที่ 4.35



รูปที่ 4.35 แสดงการเชื่อมต่อเสร็จหน้าจอของเครื่องคอมพิวเตอร์ที่ติดตั้งอยู่กับเครื่องรับวิทยุสื่อสาร

6.ทางด้านบนจอคอมพิวเตอร์ที่ถูกควบคุมจะมีเมนู View สามารถปรับขนาดหน้าจอของเครื่องที่กำลังควบคุมอยู่ได้หากต้องการเลิกใช้งานกดปุ่ม X เพื่อออกจากการควบคุมรูปที่ 4.36



รูปที่ 4.36 แสดงเมนู View

4.6 การทดสอบการรับสัญญาณวิทยุในช่องความถี่ต่างๆ

ในการทดสอบอุปกรณ์ต้นแบบนี้ทางผู้จัดทำได้เลือกใช้บริเวณอาคารสุรนิทัศน์ เป็นบริเวณทดสอบเพื่อตรวจสอบความถี่วิทยุที่ได้รับได้ จากเครื่องรับวิทยุต้นแบบเปรียบเทียบกับเครื่องรับวิทยุทั่วไป(ยี่ห้อ GRUNDIG รุ่น G6 AVIATOR)

- จุดการทดสอบคือ บริเวณอาคารสุรนิทัศน์



รูปที่ 4.37 แสดงตำแหน่งของการทดสอบในพื้นที่บริเวณอาคารสุรนิทัศน์

จากรูปที่ 4.37 จะบอกถึงตำแหน่งในการทดสอบการรับคลื่นความถี่วิทยุของอุปกรณ์ต้นแบบเปรียบเทียบกับเครื่องรับวิทยุทั่วไป

4.6.1 ทดสอบรับคลื่นสัญญาณวิทยุของอุปกรณ์ต้นแบบเปรียบเทียบกับเครื่องรับวิทยุทั่วไป

ตารางการทดสอบรับคลื่นสัญญาณวิทยุของอุปกรณ์ต้นแบบเปรียบเทียบกับเครื่องรับวิทยุทั่วไป

โดยจะแบ่งระดับการรับสัญญาณวิทยุเป็น 3 ระดับคือ

ระดับที่ 0 คือ รับสัญญาณไม่ได้

ระดับที่ 1 คือ รับสัญญาณได้แต่มีสัญญาณรบกวน

ระดับที่ 2 คือ รับสัญญาณได้ชัดเจน

ตารางที่ 4.1 ทดสอบรับคลื่นสัญญาณวิทยุของอุปกรณ์ต้นแบบเปรียบเทียบกับเครื่องรับวิทยุทั่วไป

ความถี่วิทยุ	ระดับสัญญาณที่รับได้					
	อุปกรณ์ต้นแบบ			เครื่องรับวิทยุทั่วไป		
	0	1	2	0	1	2
87.50			√	√		
87.75	√				√	
88.00	√			√		
88.25	√				√	
88.50	√				√	
88.75	√			√		
89.00	√				√	
89.25	√				√	
89.50		√				√
89.75	√					√
90.00	√				√	
90.25	√					√
90.50	√				√	
90.75	√			√		
91.00			√	√		
91.25	√				√	

ความถี่วิทยุ	ระดับสัญญาณที่ได้รับได้					
	อุปกรณ์ต้นแบบ			เครื่องรับวิทยุทั่วไป		
	0	1	2	0	1	2
91.50	√			√		
91.75		√				√
92.00	√				√	
92.25	√			√		
92.50			√			√
92.75	√				√	
93.00	√					√
93.25	√			√		
93.50	√					√
93.75		√			√	
94.00		√				√
94.25	√				√	
94.50		√			√	
94.75		√				√
95.00	√				√	
95.25	√					√
95.50	√					√
95.75	√			√		
96.00	√					√
96.25	√					√
96.50	√			√		
96.75	√					√
97.00	√			√		
97.25	√			√		
97.50	√					√
97.75	√					√

ความถี่วิทยุ	ระดับสัญญาณที่ได้รับ					
	อุปกรณ์ต้นแบบ			เครื่องรับวิทยุทั่วไป		
	0	1	2	0	1	2
98.00	✓			✓		
98.25		✓				✓
98.50	✓			✓		
98.75	✓			✓		
99.00			✓	✓		
99.25	✓				✓	
99.50			✓			✓
99.75	✓				✓	
100.00	✓					✓
100.25	✓					✓
100.50	✓					✓
100.75	✓					✓
101.00	✓					✓
101.25			✓	✓		
101.50	✓				✓	
101.75	✓					✓
102.00			✓		✓	
102.25	✓			✓		
102.50	✓				✓	
102.75	✓				✓	
103.00	✓				✓	
103.25			✓			✓
103.50	✓			✓		
103.75			✓			✓
104.00	✓					✓
104.25	✓					✓

ความถี่วิทยุ	ระดับสัญญาณที่ได้รับ					
	อุปกรณ์ต้นแบบ			เครื่องรับวิทยุทั่วไป		
	0	1	2	0	1	2
104.50			√			√
104.75	√				√	
105.00	√			√		
105.25			√			√
105.50	√				√	
105.75	√			√		
106.00	√			√		
106.25			√			√
106.50		√				√
106.75		√				√
107.00	√			√		
107.25			√			√
107.50	√					√

จากตารางผลการทดสอบพบว่าอุปกรณ์ต้นแบบเปรียบเทียบกับเครื่องรับวิทยุทั่วไป พบว่าอุปกรณ์ต้นแบบสามารถรับสัญญาณความถี่วิทยุได้ตรงกับเครื่องรับวิทยุทั่วไปและจำนวนระดับสัญญาณวิทยุที่ได้รับได้ผลการทดสอบเป็นดังนี้

	อุปกรณ์ต้นแบบ	เครื่องรับวิทยุทั่วไป
จำนวนสถานีที่รับสัญญาณได้แต่มีสัญญาณรบกวน	9	23
จำนวนที่รับสัญญาณได้	13	35

ตารางที่ 4.2 แสดงสรุปผลการทดสอบอุปกรณ์ต้นแบบเปรียบเทียบกับเครื่องรับวิทยุทั่วไป

4.6.2 การทดสอบการค้นหาสถานีแบบอัตโนมัติของอุปกรณ์ต้นแบบเปรียบเทียบกับวิทยุทั่วไป

ตารางที่ 4.3 การทดลองการค้นหาสถานีแบบอัตโนมัติของวิทยุทั่วไป(MHz)

ครั้งที่ 1	ครั้งที่ 2	ครั้งที่ 3	ครั้งที่ 4	ครั้งที่ 5
87.50	87.50	87.50	87.50	87.50
89.20	89.20	89.20	89.20	89.20
-	89.50	89.50	89.50	-
89.70	89.70	89.70	-	89.70
90.00	90.00	90.00	90.00	90.00
90.40	90.40	90.40	90.40	90.40
-	91.00	-	-	-
92.00	92.00	92.00	92.00	92.00
92.80	92.80	-	-	-
93.20	93.20	93.20	93.20	93.20
93.70	93.70	93.70	93.70	93.70
94.20	94.20	94.20	94.20	-
94.50	-	-	-	-
-	-	95.00	95.50	95.50
95.80	95.80	95.80	95.80	95.80
96.20	96.20	-	-	-
96.50	96.50	96.50	96.50	96.50
97.00	97.00	97.00	97.00	97.00
97.50	97.50	97.50	97.50	-
-	98.30	98.30	98.30	-
99.00	99.00	99.00	99.00	99.00
99.50	99.50	99.50	99.50	99.50
99.80	99.80	99.80	99.80	-
100.00	100.00	100.00	100.00	100.00
-	100.20	100.20	100.20	100.20
100.50	100.50	100.50	-	100.50

ครั้งที่ 1	ครั้งที่ 2	ครั้งที่ 3	ครั้งที่ 4	ครั้งที่ 5
100.80	100.80	100.80	-	-
101.00	101.00	101.00	101.00	101.00
101.20	101.20	101.20	101.20	101.20
102.00	102.00	102.00	102.00	102.00
-	102.80	-	102.80	-
103.30	103.30	103.30	103.30	103.30
103.70	103.70	-	103.70	103.70
104.00	104.00	104.00	104.00	104.00
-	104.50	104.50	104.50	-
105.20	105.20	105.20	105.20	105.20
106.50	106.50	106.50	106.50	106.50
-	106.80	106.80	106.80	106.80
107.20	107.20	107.20	107.20	107.20
107.50	107.50	107.50	107.50	107.50

จากตารางตารางการทดสอบการค้นหาสถานีแบบอัตโนมัติของวิทยุทั่วไป จะพบว่าคลื่นความถี่ที่รับได้มีจำนวนช่องจากการตรวจสอบทั้งหมด 5 ครั้ง เป็นดังนี้

	ครั้งที่ 1	ครั้งที่ 2	ครั้งที่ 3	ครั้งที่ 4	ครั้งที่ 5
จำนวนช่องที่รับได้	32	38	34	33	28
ช่องเดิม	-	31	33	31	27
ช่องใหม่	-	7	1	2	1

ตารางที่ 4.4 แสดงสรุปผลการทดสอบการค้นหาสถานีแบบอัตโนมัติของวิทยุทั่วไป

ตารางที่ 4.5 การทดสอบการค้นหาสถานีแบบอัตโนมัติของอุปกรณ์ต้นแบบ(MHz)

ครั้งที่ 1	ครั้งที่ 2	ครั้งที่ 3	ครั้งที่ 4	ครั้งที่ 5
87.50	87.50	87.50	87.50	87.50
89.25	-	-	-	-
89.50	-	-	-	-

ครั้งที่ 1	ครั้งที่ 2	ครั้งที่ 3	ครั้งที่ 4	ครั้งที่ 5
92.00	92.00	-	92.00	-
-	94.25	-	94.25	-
96.00	96.00	96.00	96.00	96.00
-	-	96.50	96.50	96.50
99.00	99.00	99.00	99.00	99.00
99.50	99.50	99.50	99.50	99.50
-	-	101.25	-	-
102.00	102.00	102.00	102.00	102.00
103.25	103.25	103.25	103.25	103.25
104.50	104.50	104.50	104.50	104.50
105.25	105.25	105.25	105.25	105.25
106.25	106.25	106.25	106.25	106.25
-	-	106.50	-	106.50
-	106.75	106.75	-	106.75
107.25	107.25	107.25	107.25	107.25

จากตารางตารางการทดลองใช้อุปกรณ์ต้นแบบในการค้นหาสถานีวิทยุแบบอัตโนมัติ จะพบว่าคลื่นความถี่ที่ได้รับได้มีจำนวนช่องจากการตรวจสอบทั้งหมด 5 ครั้ง เป็นดังนี้

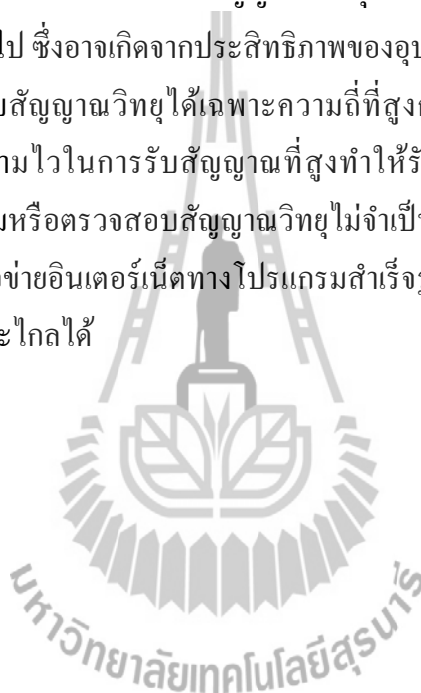
	ครั้งที่ 1	ครั้งที่ 2	ครั้งที่ 3	ครั้งที่ 4	ครั้งที่ 5
จำนวนช่องที่ได้รับได้	13	13	14	13	13
ช่องเดิม	-	11	11	11	11
ช่องใหม่	-	2	3	2	2

ตารางที่ 4.6 แสดงสรุปการทดลองใช้อุปกรณ์ต้นแบบในการค้นหาสถานีวิทยุแบบอัตโนมัติ

4.7 กล่าวสรุป

อุปกรณ์ต้นแบบนี้ มีส่วนประกอบของ FM Module ที่ทำหน้าที่รับความถี่วิทยุตั้งแต่คลื่นความถี่ 87.50-107.50 MHz และอุปกรณ์นี้ได้มีการส่งงานผ่านทางโปรแกรมคอมพิวเตอร์ที่ได้พัฒนาขึ้นมาจากการนำอุปกรณ์ต้นแบบมาทำการทดสอบใช้งานในสภาพแวดล้อมจริง และได้บันทึกผลการทดสอบเอาไว้ ทำให้สามารถทำการทดสอบได้บรรลุจุดประสงค์

นอกจากนี้ได้รู้ว่าอุปกรณ์ต้นแบบสามารถทำการรับสัญญาณวิทยุในย่านความถี่ 87.50-107.50 MHz ได้ตรงกับเครื่องรับวิทยุทั่วไป และสามารถรับสัญญาณวิทยุที่ออกอากาศใน ขณะนั้นได้ ถึงแม้ว่าความสามารถในการรับสัญญาณของอุปกรณ์ต้นแบบจะรับคลื่นความถี่วิทยุได้น้อยกว่าเครื่องรับวิทยุทั่วไป ซึ่งอาจเกิดจากประสิทธิภาพของอุปกรณ์ต้นแบบที่มีความไวในการรับสัญญาณที่ต่ำ จึงทำให้รับสัญญาณวิทยุได้เฉพาะความถี่ที่สูงกว่าเครื่องรับวิทยุทั่วไป เนื่องจากเครื่องรับวิทยุทั่วไปมีความไวในการรับสัญญาณที่สูงทำให้รับสัญญาณได้มากกว่า และหากผู้ตรวจสอบต้องการควบคุมหรือตรวจสอบสัญญาณวิทยุไม่จำเป็นต้องอยู่ในพื้นที่นั้นๆ โดยสามารถตรวจสอบผ่านระบบเครือข่ายอินเทอร์เน็ตทางโปรแกรมสำเร็จรูป(โปรแกรม Team Viewer) ที่สามารถควบคุมได้ในระยะไกลได้



บทที่ 5

ข้อสรุปของโครงการ

5.1 กล่าวนำ

เนื้อหาในบทนี้เป็นการกล่าวถึงบทสรุปของโครงการอุปกรณ์เฝ้าระวังสัญญาณรบกวนในย่านสื่อสารการบิน ซึ่งประกอบไปด้วยปัญหาที่พบในขณะดำเนินการ วิธีการแก้ปัญหา ข้อเสนอแนะ และวิธีการพัฒนาโครงการต่อไป

5.2 ปัญหาที่พบและแนวทางในการแก้ไข

ในการทำโครงการอุปกรณ์เฝ้าระวังสัญญาณรบกวนในย่านสื่อสารการบิน ที่ปรากฏปัญหาต่างๆ ดังแสดงไว้ในตารางที่ 5.1 ซึ่งประกอบไปด้วยปัญหาที่พบ สาเหตุของปัญหารวมทั้งวิธีการแก้ไข

ตารางที่ 5.1 ปัญหาและสาเหตุที่พบในขณะดำเนินงานและวิธีการแก้ไข

ปัญหาที่พบในการปฏิบัติงาน	สาเหตุและวิธีการแก้ไข
เรื่องโปรแกรมการใช้งาน	สาเหตุ : โปรแกรมที่ใช้ ไม่เคยใช้งานมาก่อน วิธีแก้ไข: ศึกษาวิธีการเขียนโปรแกรมในอินเทอร์เน็ต แล้วนำมาเป็นแนวทางในการเขียนโปรแกรม
บอร์ดชำรุด	สาเหตุ : การเชื่อมต่อกับพอร์ตการใช้งานหลุดบ่อย วิธีแก้ไข: เปลี่ยนบอร์ดชนิดอื่น มาใช้ในการทำงานแทนบอร์ดเดิม จึงทำให้เสียเวลาในการทำงาน
ประสิทธิภาพเครื่องรับวิทยุ	สาเหตุ : ความไวในการรับสัญญาณต่ำลงส่งผลให้การค้นสถานีแบบอัตโนมัติได้จำนวนช่องสัญญาณน้อยกว่าเครื่องรับวิทยุทั่วไป วิธีแก้ไข : ต้องปรับความไวในการรับสัญญาณให้กับเครื่องรับวิทยุ

ปัญหาที่พบในการปฏิบัติงาน	สาเหตุและวิธีการแก้ปัญหา
ประสิทธิภาพเครื่องรับวิทยุ	สาเหตุ : ความไวในการรับสัญญาณต่ำลงส่งผลให้การค้นสถานีแบบอัตโนมัติได้จำนวนช่องสัญญาณน้อยกว่าเครื่องรับวิทยุทั่วไป วิธีแก้ไข : ต้องปรับความไวในการรับสัญญาณให้กับเครื่องรับวิทยุ
	สาเหตุ : เมื่อทำการค้นหาสัญญาณจะมีสัญญาณรบกวนเข้ามาแทรกทำให้ได้รับสัญญาณที่ไม่ชัดเจน วิธีแก้ไข : ทดสอบในบริเวณพื้นที่โล่ง ไม่อับสัญญาณ
	สาเหตุ : เครื่องรับวิทยุใน MCU เกิดการ Error รับสัญญาณบางสถานีไม่ได้ วิธีแก้ไข : กดปุ่ม RESET MCU

5.3 ข้อเสนอแนะ

5.3.1 โปรแกรม Arduino1.0.6 การใช้งานจะใกล้เคียงกับภาษาซี แต่ในความจริงแล้วจะมีรายละเอียดบางส่วนที่ต้องใช้คำสั่งเฉพาะของโปรแกรม Arduino1.0.6 ทำให้บุคคลที่สนใจใช้บอร์ด Arduino ในการทำงานนั้นควรที่จะศึกษาโปรแกรม Arduino1.0.6 ให้ละเอียดเสียก่อนซึ่งได้กล่าวถึงตัวอย่างโค้ดคำสั่งในการใช้กับชุดอุปกรณ์ต้นแบบแล้วในบทที่ 2 (ดูตัวอย่างหน้า 33)

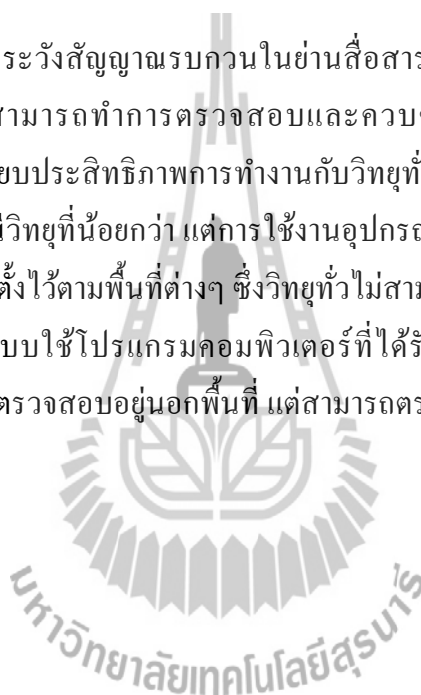
5.3.2 FM Radio Module มีหลายรุ่นมีทั้งแบบที่ใช้งานเป็นทั้งตัวส่งสัญญาณและตัวรับสัญญาณ ดังนั้นควรเลือกงานให้เหมาะสมกับการใช้งาน แต่ถ้าผู้สนใจจะใช้งานควรหาข้อมูลเพิ่มเติมเพื่อความเหมาะสมในการใช้งาน

5.4 แนวทางในการพัฒนาต่อไป

เนื่องจากอุปกรณ์ต้นแบบสามารถรับคลื่นความถี่วิทยุผ่านการควบคุมจากเครื่องคอมพิวเตอร์ในอนาคตอาจมีการพัฒนาโดยการนำโปรแกรมการควบคุมนี้ไปประยุกต์ใช้กับโทรศัพท์มือถือหรืออุปกรณ์ที่จะสามารถควบคุมการค้นหาสถานีวิทยุที่มีขนาดเล็ก ให้สามารถค้นหาสถานีได้แม้กระทั่งอยู่นอกพื้นที่ควบคุม

5.5 บทสรุปของโครงการ

โครงการอุปกรณ์เฝ้าระวังสัญญาณรบกวนในย่านสื่อสารการบิน ได้จัดทำอุปกรณ์ต้นแบบออกมา และอุปกรณ์นี้สามารถทำการตรวจสอบและควบคุมคลื่นวิทยุที่ใช้งานได้จริงตามวัตถุประสงค์ถ้าเปรียบเทียบกับประสิทธิภาพการทำงานกับวิทยุทั่วไป อาจจะมีประสิทธิภาพในด้านการรับสัญญาณจากสถานีวิทยุที่น้อยกว่า แต่การใช้งานอุปกรณ์ต้นแบบสะดวกกว่าวิทยุทั่วไปคือสามารถนำอุปกรณ์ไปติดตั้งไว้ตามพื้นที่ต่างๆ ซึ่งวิทยุทั่วไปไม่สามารถทำได้ และข้อได้เปรียบในการทำงานของอุปกรณ์ต้นแบบใช้โปรแกรมคอมพิวเตอร์ที่ได้รับการพัฒนาใช้ในการตรวจสอบสัญญาณหรือแม้กระทั่งผู้ตรวจสอบอยู่นอกพื้นที่ แต่สามารถตรวจสอบและควบคุมผ่านทางระบบเครือข่ายอินเทอร์เน็ตได้



บรรณานุกรม

- [1] www.mec.ac.th/m56/mit_std/image/file/sak/unicon10-209226.pdf
- [2] www.thaieasyelec.com/products/.../arduino/.../arduino-uno-r3-detail.html
- [3] <http://www.senate.go.th/w3c/senate/spaw2/uploads/files/18-09-56>
- [4] http://www.prd.go.th/ewt_dl_link.php?nid=24876&filename=expert
- [5] <https://www.teamviewer.com/th/>
- [6] www.cdd.go.th/support/program/TeamViewer.pdf
- [7] ชีรวัฒน์ ประกอบผล. (2554) .การเขียนแอปพลิเคชันด้วย Visual Basic 2010 .สำนักพิมพ์ Simplify







คำสั่งในการเขียนโปรแกรมควบคุม

```
#include <Wire.h>
#include <string.h>
#define SLAVE_ID 0x20
#define CONFIG_WORDS
static uint16_t = Config0[CONFIG_WORDS]
Config0[CONFIG_WORDS] = {0xD001, 0x0002, 0x0440, 0x86D3, 0x4000};
static uint16_t Config[CONFIG_WORDS];
#define SEEK
#define SEEK_MASK
#define SEEK_UP
#define SEEK_DOWN
#define CHAN
#define TUNE
#define CHAN_MASK
#define VOLUME
#define VOLUME_MASK
#define STC
#define READCHAN
#define READCHAN_MASK
#define READRSSI
#define MAX_FREQ 1080
#define MIN_FREQ 870
uint32_t fm_vol = 1
uint32_t fm_chan = 875;
uint32_t fm_stc,
char BuffCH;
String inputString = "";
int buff_chan;
```

กำหนดตัวแปร

```

void setup()
{

    Serial.begin(9600); // เริ่มต้นการใช้งานที่บอดเรทที่ 9600
    Wire.begin(); // เริ่มต้นติดต่อกับ i2c ด้วยฟังก์ชัน Wire.begin()
    fm_reset(); // ส่งคำสั่งเพื่อรีเซ็ตโมดูลเอฟเอ็ม

}

void loop()
{

    BuffCH = Serial.read(); // กำหนดให้ BuffCH = Serial.read();
    if(BuffCH == '<') {
        // ถ้ากดเครื่องหมายน้อยกว่าโปรแกรมจะทำการหาสถานีที่มีความถี่ต่ำกว่าจากสถานีเดิม
        delay(100);
        เลือกข้อมูลคำสั่งค้นหาสถานีที่ช่องความถี่ต่ำกว่า โดยกำหนดคำสั่งลงในตัวแปร Config[0]
        Config[0] &= ~SEEK_MASK; // เริ่มต้นการกด
        Config[0] |= SEEK_DOWN; // เริ่มการค้นหาสถานีที่มีความถี่ต่ำกว่าจากสถานีเดิม
        seeking = true; // เตรียมค้นหาสถานีด้วยคำสั่ง seeking = true;
    } else if(BuffCH == '>') {
        // ถ้าเป็นเครื่องหมายมากกว่าโปรแกรมจะทำการหาสถานีที่มีความถี่สูงกว่าจากสถานีเดิมโดยส่ง
        ข้อมูลไปยังรีจิสเตอร์ 0x02
        delay(100);
        เลือกข้อมูลคำสั่งค้นหาสถานีที่ช่องความถี่ต่ำกว่าโดยกำหนดคำสั่งลงในตัวแปร Config[0]
        Config[0] &= ~SEEK_MASK; // เริ่มต้นการกด
        Config[0] |= SEEK_UP;
        // เริ่มการค้นหาสถานีที่มีความถี่สูงกว่าจากสถานีเดิมโดยส่งข้อมูลไปยังรีจิสเตอร์ 0x02
        seeking = true; // เตรียมค้นหาสถานีด้วยคำสั่ง seeking = true;
    }
}

```

```

} else if (BuffCH == 'F') // ถ้าต้องการป้อนค่าให้ใส่ F นำหน้า
    delay(100);
    while(Serial.available() > 0) // โดยความถี่นั้นจะต้องมีความถี่มากกว่าศูนย์
    {
        inputString = Serial.readStringUntil('\n');
        // อ่านค่าตัวอักษรและตัวเลขที่ป้อนเข้ามาด้วยฟังก์ชัน Serial.readStringUntil
    }
    fm_chan = inputString.toInt();
    // ทำการแปลงตัวอักษรและตัวเลขให้เป็นคลื่นความถี่ที่ต้องการด้วยฟังก์ชัน
    seeking = false; // เตรียมหยุดการค้นหาสถานีด้วยคำสั่ง seeking = false;
    fm_vol = 1; // กำหนดค่าระดับเสียงตั้งต้นเป็น 1
    fm_seek = (fm_chan - 8700)/5; // เก็บค่าความถี่ของสถานี
    seek_disable(); // หยุดการค้นหาช่องความถี่ของสถานี
    chan_update(fm_seek + MIN_FREQ);
    // ปรับปรุงค่าความถี่ปัจจุบันสำหรับโมดูลวิทยุ FM ด้วยการนำค่าของตัวแปร
    fm_seek มาบวกกับค่าความถี่ต่ำสุดที่กำหนดไว้ในตัวแปร MIN_FREQ
    setConfig(); // เตรียมข้อมูลของความถี่เพื่อเขียนลงในรีจิสเตอร์ 0x03 ของ โมดูล
    วิทยุ FM ด้วยฟังก์ชัน setConfig();
    buff_chan = (fm_seek * 5) + 8700;
    Serial.print((buff_chan)/100);
    Serial.print(".");
    Serial.println((buff_chan)% 100);
} else if (BuffCH == 'U') { // ถ้าทำการกด U ให้เพิ่มเสียง
    fm_vol++;
    Config[3] &= ~VOLUME_MASK;
    Config[3] |= fm_vol << VOLUME;
    setConfig();
    // เตรียมข้อมูลของความถี่เพื่อเขียนลงในรีจิสเตอร์ 0x05 ของโมดูลวิทยุ FM ด้วย
    ฟังก์ชัน setConfig();

```

```

}else if (BuffCH == 'D') { // ถ้าทำการกด D ให้ลดเสียง
    fm_vol--;
    Config[3] &= ~VOLUME_MASK;
    Config[3] |= fm_vol << VOLUME;
    // แล้วทำการเก็บข้อมูลลงในรีจิสเตอร์ 0x05
    setConfig(); // เตรียมข้อมูลของความถี่เพื่อเขียนลงในรีจิสเตอร์ 0x05 ของโมดูล
    วิทยุ FM ด้วยฟังก์ชัน setConfig();
}
}else if (BuffCH == 'S') { // ถ้าทำการกด S
    Serial.print("A"); // ทำการแสดง A ออกหน้าจอ
}
}else if (BuffCH == 'R') {
    // ถ้ากด R จะทำการรีเซ็ตโปรแกรมให้กลับไปเริ่มต้นที่ความถี่เดิมคือ 87.5 MHz
    Config[0] = 0xD003; // แล้วทำการเก็บข้อมูลลงในรีจิสเตอร์ 0x05
    setConfig();
    // เตรียมข้อมูลของความถี่เพื่อเขียนลงในรีจิสเตอร์ 0x05 ของโมดูลวิทยุ FM ด้วย
    ฟังก์ชัน setConfig();
    delay(200);
    Config[0] = 0xD001 ; // แล้วทำการเก็บข้อมูลลงในรีจิสเตอร์ 0x01
    setConfig();
    // เตรียมข้อมูลของความถี่เพื่อเขียนลงในรีจิสเตอร์ 0x01 ของโมดูลวิทยุ FM ด้วย
    ฟังก์ชัน setConfig();
    fm_reset(); // รีเซ็ตการทำงานของโมดูลวิทยุ FM ให้กลับไปยังค่าตั้งต้น
}

if (seeking) {
    setConfig(); // // เตรียมข้อมูลของความถี่
    delay(100);
    getStatus(); // อ่านค่าสถานะการค้นหาสถานี และค่าความถี่ของสถานีปัจจุบันของ
    โมดูลวิทยุ FM
}

if (seeking && fm_stc) {
    // ตรวจสอบว่าการค้นหาสถานีเสร็จสิ้นหรือยังด้วยการ ตรวจสอบสถานะของตัวแปร
    fm_stc ด้วยคำสั่ง (seeking && fm_stc)
}

```

```

seek_disable();
// เมื่อค้นหาสถานีเสร็จสิ้นทำการหยุดการค้นหาสถานีด้วยฟังก์ชันseek_disable
chan_update(fm_seek + MIN_FREQ);
// จากนั้นปรับปรุงค่าความถี่ด้วยฟังก์ชัน chan_update() แล้วส่งคำสั่งไปยังโมดูล
วิทยุ FM ด้วยฟังก์ชัน setConfig()

setConfig();
buff_chan = (fm_seek * 5) + 8700;
Serial.print((buff_chan)/100);
Serial.print(".");
Serial.println((buff_chan)% 100);
}
tune_disable(); // หยุดการเลือกความถี่
}
}

void setConfig() // กำหนดคำสั่งให้กับโมดูลวิทยุ FM
{
    Wire.beginTransmission(SLAVE_ID); // ติดต่อกับโมดูลวิทยุ FM ผ่านฟังก์ชัน
    for(int i = 0 ; i < CONFIG_WORDS ; i++) // เตรียมค่าของคำสั่งที่ต้องการ
        เขียนไปยังโมดูลวิทยุ FM
    {
        เขียนข้อมูลไปยังโมดูลวิทยุ FM ด้วยคำสั่ง
        Wire.write((int)Config[i] >> 8);
        Wire.write((int)(Config[i] & 0xff));
    }
    Wire.endTransmission(); // สิ้นสุดการติดต่อกับโมดูลวิทยุ FM
}

void fm_reset() // รีเซ็ตการทำงานของโมดูลวิทยุ FM ให้กลับไปยังค่าตั้งต้น
{
    seeking = false; // เตรียมหยุดการค้นหาสถานีด้วยคำสั่ง seeking = false;
    fm_vol = 1; // กำหนดค่าระดับเสียงตั้งต้นเป็น 1
}

```

```

memcpy(Config,Config0,CONFIG_WORDS * sizeof(uint16_t));
Config[1] &= ~CHAN_MASK;
Config[1] |= 1 << CHAN;
Config[3] &= ~VOLUME_MASK;
Config[3] |= fm_vol << VOLUME;
tune_enable(); // ให้โมดูลวิทยุ FM ทำการรับคลื่น
setConfig(); // กำหนดคำสั่งให้กับโมดูลวิทยุ FM
tune_disable(); // หยุดการรับคลื่นของโมดูลวิทยุ FM
}

void tune_enable() // ให้โมดูลวิทยุ FM ทำการรับคลื่น
{
    Config[1] |= (1 << TUNE); // กำหนด TUNE ใน โมดูลวิทยุที่รีจิสเตอร์ 0x03
}

void tune_disable() // หยุดการรับคลื่นของโมดูลวิทยุ FM
{
    Config[1] &= ~(1 << TUNE); // รีเซ็ต TUNE ใน โมดูลวิทยุที่รีจิสเตอร์ 0x03
}

void seek_disable() // หยุดการค้นหาช่องความถี่ของสถานีด้วยฟังก์ชัน seek_disable()
{
    seeking = false; // เตรียมหยุดการค้นหาสถานีด้วยคำสั่ง seeking = false;
    Config[0] &= ~SEEK_MASK;
}

void chan_update(int val) // ปรับปรุงค่าความถี่ด้วยฟังก์ชัน chan_update()
{
    if (val >= MIN_FREQ || val <= MAX_FREQ)
        // ตรวจสอบค่าความถี่ว่าเกินขอบเขตที่ตั้งไว้หรือไม่(ต้องอยู่ในช่วง 870 ถึง
        // 1080 ซึ่งตรงกับค่าความถี่ 87.0 ถึง 108.0MHz)
    {
        fm_chan = val;
        Config[1] &= ~CHAN_MASK;
    }
}

```

```

    Config[1] |= (fm_chan - MIN_FREQ) << CHAN;
    tune_enable(); // ให้โมดูลวิทยุ FM ทำการรับคลื่น
    chan_print(); // แสดงค่าสถานีปัจจุบันด้วยฟังก์ชัน chan_print()
}

}

void chan_print() // แสดงค่าสถานีปัจจุบันด้วยฟังก์ชัน chan_print()
{
    int chan = ((Config[1] & CHAN_MASK) >> CHAN) + MIN_FREQ;
}

void getStatus()
    // อ่านค่าสถานะการค้นหาสถานีและค่าความถี่ของสถานีปัจจุบันของโมดูลวิทยุ
    FM โดยเก็บสถานะการค้นหาสถานีไว้ที่ตัวแปร fm_stc และค่าของสถานีไว้ที่ตัวแปร
    fm_seek
    {
        uint16_t Status[1];
        Wire.requestFrom(SLAVE_ID, 2);
        for(int i = 0 ; i < 1 ; i++)
        {
            Status[i] = Wire.read();
            Status[i] = Status[i] << 8 | Wire.read();
        }

        fm_seek = Status[0] & READCHAN_MASK;
        fm_stc = Status[0] >> STC & 1;
    }

```

ประวัติผู้เขียน



นางสาวอรปภา บุญเพชร เกิดวันที่ 14 ตุลาคม 2535 ภูมิลำเนาเดิมอยู่ที่ บ้านเลขที่ 422 ถนนมหาจักรพรรดิ ตำบลในเมือง อำเภอเมือง จังหวัดพิษณุโลก จบการศึกษาลาย โรงเรียนเตรียมอุดมศึกษา ภาคเหนือ ปีการศึกษา 2554 ปัจจุบันกำลังศึกษาอยู่ชั้นปีที่ 4 สาขาวิชา โทรคมนาคม มหาวิทยาลัยเทคโนโลยีสุรนารี จังหวัดนครราชสีมา



นางสาวอรทัย อารมณเพียร เกิดวันที่ 5 กันยายน 2535 ภูมิลำเนาเดิมอยู่ที่ บ้านเลขที่ 187 หมู่ 5 ตำบลโนนสมบูรณ์ อำเภอนาจะหลวย จังหวัด อุบลราชธานี จบการศึกษาด้านและปลาย โรงเรียนเดชอุดม ปีการศึกษา 2554 ปัจจุบันกำลังศึกษาอยู่ชั้นปีที่ 4 สาขาวิชาโทรคมนาคม มหาวิทยาลัยเทคโนโลยีสุรนารี จังหวัดนครราชสีมา



นางสาววิภัทรวีร์ โสมจุมจัง เกิดวันที่ 21 ธันวาคม 2535 ภูมิลำเนาเดิม อยู่ที่ บ้านเลขที่ 21 หมู่ 12 ตำบลเชียงใหม่ อำเภอโพธิ์ชัย จังหวัด ร้อยเอ็ด จบการศึกษาลาย โรงเรียนร่องคำ ปีการศึกษา 2554 ปัจจุบัน กำลังศึกษาอยู่ชั้นปีที่ 4 สาขาวิชาโทรคมนาคม มหาวิทยาลัยเทคโนโลยี สุรนารี จังหวัดนครราชสีมา