



รายงานการวิจัย

เทคนิคการปรับจำนวนการซัดตัวอย่างสำหรับการหาทิศทางมาถึงของสัญญาณ

**Adaptive Snapshot for Direction of Arrival Finding**

ได้รับทุนอุดหนุนการทำวิจัยจากมหาวิทยาลัยเทคโนโลยีสุรนารี  
ผลงานวิจัยเป็นความรับผิดชอบของหัวหน้าโครงการวิจัยแต่เพียงผู้เดียว



รายงานการวิจัย

เทคนิคการปรับจำนวนการชั่งตัวอย่างสำหรับการหาทิศทางการมาถึงของสัญญาณ

Adaptive Snapshot for Direction of Arrival Finding

ผู้วิจัย

หัวหน้าโครงการ

ผู้ช่วยศาสตราจารย์ ดร. มนต์ทิพย์ภา อูทธารสกุล

สาขาวิชาวิศวกรรมโทรคมนาคม

สำนักวิชาวิศวกรรมศาสตร์

มหาวิทยาลัยเทคโนโลยีสุรนารี

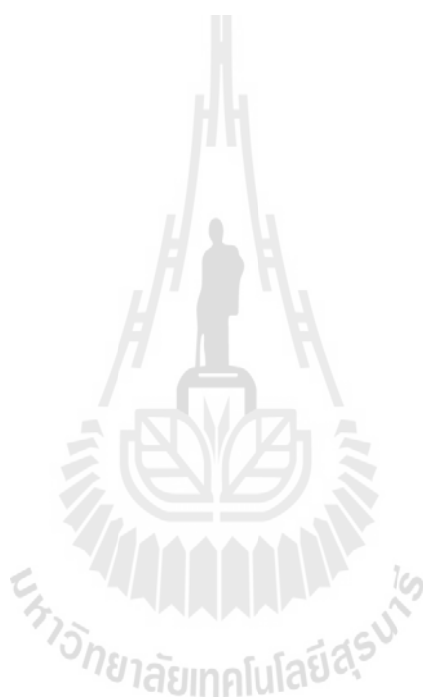
ได้รับทุนอุดหนุนการวิจัยจากมหาวิทยาลัยเทคโนโลยีสุรนารี ปีงบประมาณ พ.ศ. 2556

ผลงานวิจัยเป็นความรับผิดชอบของหัวหน้าโครงการวิจัยแต่เพียงผู้เดียว

ธันวาคม 2557

## กิตติกรรมประกาศ

ขอขอบคุณมหาวิทยาลัยเทคโนโลยีสุรนารีที่ได้ให้การสนับสนุนทุนวิจัยสำหรับ  
โครงการวิจัยนี้



ผู้วิจัย

ธันวาคม 2557

## บทคัดย่อ

ในปัจจุบันระบบเครือข่ายไร้สายนั้นได้รับความนิยมเป็นอย่างมาก เนื่องจากระบบดังกล่าวได้ถูกพัฒนาจนสามารถส่งข้อมูลได้อย่างรวดเร็วและใช้งานได้อย่างสะดวกสบายไม่ว่าจะอยู่ในบริเวณใดของเครือข่ายก็ตาม แต่ระบบเครือข่ายไร้สายนั้นก็ยังคงมีปัญหอยู่มาก เช่น สัญญาณคลื่นหลายวิถีและสัญญาณแทรกสอด เป็นต้น ดังนั้นจึงได้มีการคิดระบบที่สามารถแก้ไขปัญหาดังกล่าวนั้นก็คือ ระบบสายอากาศเก่ง เนื่องจากระบบสามารถที่จะก่อรูปลำคลื่นหลักไปยังทิศทางของสัญญาณที่ต้องการ และในขณะเดียวกันก็จะหันพู่ข้างหรือจุดศูนย์ไปยังทิศทางของสัญญาณรบกวน ซึ่งกระบวนการดังกล่าวเรียกว่า การก่อรูปลำคลื่น โดยระบบสายอากาศเก่งจะประกอบไปด้วยสองส่วนหลักๆ คือ สายอากาศแถวลำดับและระบบประมวลผลสัญญาณ ซึ่งระบบประมวลผลสัญญาณนั้นจะทำหน้าที่ในการวิเคราะห์หาทิศทางการมาถึงของสัญญาณ โดยวิธีที่เป็นที่นิยมและใช้กันอย่างแพร่หลาย ได้แก่ วิธี Multiple Signal Classification (MUSIC) algorithm เนื่องจากเป็นวิธีที่ไม่ซับซ้อนและสามารถวิเคราะห์หาทิศทางการมาถึงของสัญญาณได้อย่างถูกต้องและแม่นยำ อย่างไรก็ตามวิธี MUSIC นั้นก็ยังคงมีปัญหอยู่มากในระบบมีค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวนที่ต่ำ ประสิทธิภาพในการวิเคราะห์หาทิศทางการมาถึงของสัญญาณก็จะลดต่ำลง ดังนั้นในการแก้ปัญหานั้นจำเป็นต้องเพิ่มจำนวนการสุ่มตัวอย่างให้เหมาะสมกับค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวน ดังนั้นวิทยานิพนธ์ฉบับนี้จึงได้นำเสนอการใช้เทคนิคการปรับจำนวนการสุ่มตัวอย่างมาประยุกต์ใช้กับวิธี MUSIC ในการวิเคราะห์หาทิศทางการมาถึงของสัญญาณ โดยจะทำการปรับเลือกจำนวนการสุ่มตัวอย่างที่เหมาะสมกับค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวนในระบบ เพื่อทำการวิเคราะห์หาทิศทางการมาถึงของสัญญาณให้ได้อย่างถูกต้องและแม่นยำ นอกจากนี้ระบบยังไม่จำเป็นต้องใช้จำนวนการสุ่มตัวอย่างที่มากเกินไปจนความเหมาะสม ดังนั้นตัวประมวลผลในระบบจึงสามารถที่จะประมวลผลได้อย่างรวดเร็ว ถือเป็นการถนอมชิพของตัวประมวลผลไม่ให้งานหนักได้อีกด้วย

## Abstract

At present, wireless communication systems are very popular as these have been developed to achieve a high speed data transmission and also comfort users in everywhere. However, there are some impairment such as multipath and interference signals. A smart antenna technology is envisaged to tackle those problems because it can point main beam to desired direction while its nulls or sidelobes are steered to directions of interference signal. This process is so called beamforming. The smart antenna systems are constituted by antenna array and signal processing unit. In the signal processing unit the Directions Of Arrival (DOA) of incoming signal are predicted or estimated. The Multiple Signal Classification (MUSIC) algorithm is relatively famous among other DOA finding algorithms as it is not complex but provides high accuracy. However, in order to achieve high accuracy the algorithm needs high Signal-to-Noise Ratio (SNR) otherwise the DOA estimation may be not successful. This can be ease by utilizing more snapshot or more sampling data. Therefore, this thesis proposes adaptive snapshot technique to choose number of snapshot suitable for systems' SNR. This technique helps increasing DOA estimation accuracy. Also, in case of the system having high SNR, the technique can give small number of snapshot which is considerably able to ease the CPU of signal processing unit.

# สารบัญ

หน้า

กิตติกรรมประกาศ.....	ก
บทคัดย่อ (ภาษาไทย).....	ค
บทคัดย่อ (ภาษาอังกฤษ).....	ง
สารบัญ .....	ง
สารบัญตาราง.....	ฉ
สารบัญรูป.....	ญ
<b>บทที่</b>	
<b>1 บทนำ .....</b>	<b>1</b>
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์ของงานวิจัย.....	2
1.3 สมมติฐานของการวิจัย.....	3
1.4 ข้อตกลงเบื้องต้น .....	3
1.5 ขอบเขตการวิจัย.....	3
1.6 วิธีดำเนินการวิจัย.....	3
1.6.1 แนวทางการดำเนินงานวิจัย .....	3
1.6.2 ระเบียบวิธีวิจัย.....	4
1.6.3 สถานที่ทำการวิจัย .....	4
1.6.4 เครื่องมือที่ใช้ในการวิจัย .....	4
1.6.5 การเก็บรวบรวมข้อมูล.....	5
1.6.6 การวิเคราะห์ข้อมูล .....	5
1.7 ประโยชน์ที่คาดว่าจะได้รับ .....	5
1.8 ส่วนประกอบของงานวิจัย.....	5
<b>2 ทฤษฎีและหลักการที่เกี่ยวข้อง .....</b>	<b>7</b>
2.1 กล่าวนำ .....	7

## สารบัญ (ต่อ)

### หน้า

2.2	พื้นฐานเครือข่ายไร้สาย .....	7
2.3	สายอากาศแถวลำดับ .....	9
2.3.1	สายอากาศแถวลำดับแบบเชิงเส้น.....	10
2.3.2	สายอากาศแถวลำดับแบบเชิงระนาบ.....	13
2.4	วงจรแปลงสัญญาณความถี่ต่ำ.....	17
2.5	วงจรรองความถี่ .....	20
2.6	วงจรรขยายสัญญาณ .....	24
2.6.1	วงจรรขยายกลับขั้วสัญญาณ .....	25
2.6.2	วงจรรขยายรวมสัญญาณ .....	26
2.6.3	วงจรรขยายไม่กลับขั้วสัญญาณ .....	27
2.6.4	วงจรแรงดันไฟฟ้าตาม.....	28
2.6.5	วงจรรขยายผลต่าง .....	29
2.7	การแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล.....	30
2.7.1	การมอดูเลตทางแอมพลิจูดของพัลส์ .....	30
2.7.2	การมอดูเลตแบบรหัสพัลส์.....	31
2.7.2.1	การสุ่มตัวอย่าง .....	32
2.7.2.2	การแบ่งย่านแอมพลิจูดออกเป็นระดับต่างๆ.....	33
2.7.2.3	การเข้ารหัส .....	34
2.8	บอร์ดไมโครคอนโทรลเลอร์.....	35
2.8.1	คุณสมบัติของบอร์ด ET-AVR STAMP ATmega64/128 .....	35
2.8.2	โครงสร้างบอร์ด ET-AVR STAMP ATmega64/128 .....	37
2.8.3	โครงสร้างบอร์ด ET-AVR ISP.....	39
2.8.4	การเชื่อมต่ออุปกรณ์สำหรับ โปรแกรม hex file .....	40
2.9	ระบบสายอากาศแก่ง .....	42
2.9.1	สายอากาศแก่งแบบสลับลำคลื่น .....	43
2.9.2	สายอากาศแก่งแบบปรับลำคลื่น .....	45

## สารบัญ (ต่อ)

หน้า

2.10	วิธี Multiple Signal Classification (MUSIC) algorithm .....	47
2.11	เทคนิคการปรับจำนวนการสุ่มตัวอย่าง .....	52
2.12	สรุป .....	59
<b>3</b>	<b>การออกแบบชุดอุปกรณ์ต้นแบบหาทิศทางมาถึงของสัญญาณ .....</b>	<b>61</b>
3.1	กล่าวนำ .....	61
3.2	สายอากาศแถวลำดับ .....	62
3.3	วงจรมายสัญญาณรบกวนต่ำ .....	64
3.4	วงจรแปลงสัญญาณความถี่ต่ำ .....	69
3.5	บอร์ดไมโครคอนโทรลเลอร์ .....	73
3.5	โปรแกรมแมทแลบ .....	73
3.5	สรุป .....	75
<b>4</b>	<b>การทดสอบและวิเคราะห์ผล .....</b>	<b>77</b>
4.1	กล่าวนำ .....	77
4.2	การทดสอบวงจรแปลงสัญญาณความถี่ต่ำ .....	77
4.3	การทดสอบหาทิศทางด้วยเทคนิคการปรับจำนวนการสุ่มตัวอย่าง .....	80
<b>5</b>	<b>สรุปการวิจัยและข้อเสนอแนะ .....</b>	<b>97</b>
5.1	สรุปเนื้อหางานวิจัย .....	97
5.2	ปัญหาและข้อเสนอแนะ .....	98
5.3	แนวทางการพัฒนาในอนาคต .....	98
	รายการอ้างอิง .....	99
	ภาคผนวก	
	ภาคผนวก ก. บทความวิชาการที่ได้รับการตีพิมพ์เผยแพร่ .....	102
	ภาคผนวก ข. โปรแกรมของบอร์ดไมโครคอนโทรลเลอร์ .....	104
	ภาคผนวก ค. โปรแกรมแมทแลบ .....	121



## สารบัญตาราง

ตารางที่	หน้า
2.1	เงื่อนไขการปรับจำนวนการสุ่มตัวอย่างที่ให้ความผิดพลาดไม่เกิน 1 องศา.....57
4.1	ค่าพารามิเตอร์ต่างๆ ที่พอร์ต IF ที่ได้ทำการออกแบบ .....79
4.2	เงื่อนไขการปรับจำนวนการสุ่มตัวอย่างที่ให้ความผิดพลาดไม่เกิน 7 .....90
4.3	เงื่อนไขการปรับจำนวนการสุ่มตัวอย่างที่ให้ความผิดพลาดไม่เกิน 5 .....93
4.4	ผลการทดสอบหาทิศทางด้วยเทคนิคการปรับจำนวนการสุ่มตัวอย่าง .....94



## สารบัญรูป

รูปที่	หน้า
2.1	อุปกรณ์ส่งสัญญาณ และอุปกรณ์รับสัญญาณ .....8
2.2	การวางตัวของสายอากาศแบบเชิงเส้นจำนวน $N \times 1$ ต้น .....10
2.3	การวางตัวของสายอากาศแถวลำดับเชิงระนาบ $M \times N$ ต้น .....14
2.4	เฟสที่ต่างกันของสายอากาศแต่ละต้นสำหรับสายอากาศแถวลำดับแบบเชิงระนาบ ขนาด $2 \times 2$ เปรียบเทียบกับทิศทางมาถึงของสัญญาณตั้งแต่ $0^\circ$ ถึง $360^\circ$ .....15
2.5	วงจรผสมสัญญาณระหว่างสัญญาณ RF และ LO .....18
2.6	วงจร superheterodyne AM radio .....19
2.7	ผลตอบสนองความถี่ของวงจรกรองความถี่ทางอุดมคติ .....21
2.8	วงจรกรองสัญญาณความถี่ต่ำ .....22
2.9	วงจรกรองสัญญาณความถี่ต่ำผ่านแบบที่ .....23
2.10	วงจรกรองสัญญาณความถี่ต่ำผ่านแบบพาย .....24
2.11	สัญลักษณ์ของออปแอมป์ .....25
2.12	วงจรขยายแบบกลับขั้วสัญญาณ .....26
2.13	วงจรขยายแบบรวมสัญญาณ .....27
2.14	วงจรขยายแบบไม่กลับขั้วสัญญาณ .....27
2.15	วงจรแรงดันไฟฟ้าตาม .....28
2.16	วงจรขยายผลต่าง .....29
2.17	PAM signal ของสัญญาณ 4 ช่อง .....31
2.18	การแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัลในระบบ PCM .....32
2.19	การแบ่งย่านแอมพลิจูดออกเป็นระดับต่างๆ .....33
2.20	การเข้ารหัส .....35

## สารบัญรูป (ต่อ)

รูปที่	หน้า
2.21	โครงสร้างของบอร์ด ET-AVR STAMP ATmega64/128.....37
2.22	การจัดเรียงขาของพอร์ต C ในหมายเลข 5..... 38
2.23	โครงสร้างของบอร์ด ET-AVR ISP..... 39
2.24	การจัดเรียงขาของพอร์ต ETPSPL.....40
2.25	บอร์ด ET-CAP10P โดย (ซ้าย) คือ ET-CAP10P V2.0 และ (ขวา) คือ ET-CAP10P V1.0.....41
2.26	การต่อ ET-AVR ISP เข้ากับ ET-AVR STAMP ATmega64/128.....41
2.27	ระบบสายอากาศเก้ง.....42
2.28	ส่วนประกอบของระบบสายอากาศเก้ง..... 43
2.29	โครงสร้างพื้นฐานของระบบสายอากาศเก้งแบบสลับลำคลื่น.....44
2.30	โครงสร้างพื้นฐานของสายอากาศเก้งแบบปรับตัวได้.....46
2.31	การวิเคราะห์หาทิศทางการมาถึงของสัญญาณแบบวิธี MUSIC.....50
2.32	เปรียบเทียบการวิเคราะห์หาทิศทางการมาถึงของสัญญาณระหว่างวิธี Delay and Sum วิธี Capon's และวิธี MUSIC (ก) เมื่อมีสัญญาณเข้ามาที่ 90 และ 100 องศา (ข) เมื่อมีสัญญาณเข้ามาที่ 90 และ 95 องศา.....51
2.33	กราฟค่าความผิดพลาดเฉลี่ยในการวิเคราะห์หาทิศทาง เมื่อมีสัญญาณ เข้ามาในทิศทาง 30 องศา.....54
2.34	กราฟค่าความผิดพลาดเฉลี่ยในการวิเคราะห์หาทิศทาง กรณีแย่งสุดของ แต่ละการสุ่มตัวอย่าง.....55
2.35	แผนภาพการทำงานของโปรแกรมตามเงื่อนไขดังตารางที่ 2.1.....57
2.36	การปรับเลือกจำนวนการสุ่มตัวอย่าง เมื่อมีสัญญาณเข้ามาที่ 10 องศา.....58
2.37	การปรับเลือกจำนวนการสุ่มตัวอย่าง เมื่อมีสัญญาณเข้ามาที่ 40 องศา.....58
3.1	ส่วนประกอบของชุดอุปกรณ์ต้นแบบหาทิศทางการมาถึงของสัญญาณ .....61
3.2	สายอากาศแถวลำดับเชิงเส้นขนาด 4×1 ที่ใช้ในงานวิจัยนี้ .....63
3.3	แบบรูปการแผ่พลังงานของสายอากาศแถวลำดับเชิงระนาบขนาด 4×1 มีระยะห่างระหว่างสายอากาศเท่ากับ $\lambda/2$ .....63

## สารบัญญรูป (ต่อ)

รูปที่	หน้า
3.4	วงจรถายสัญญาณรบกวนต่ำแบบสำเร็จรูป.....64
3.5	รูปแบบและขนาดของโครงสร้างวงจรถายสัญญาณรบกวนต่ำ.....65
3.6	ค่าเจาะจงทางไฟฟ้า.....65
3.7	ข้อมูลประสิทธิภาพของวงจรถายสัญญาณรบกวนต่ำ.....66
3.8	ความสัมพันธ์ระหว่างความถี่และอัตราการจัดสัญญาณ.....66
3.9	ความสัมพันธ์ระหว่างความถี่และค่า directivity.....67
3.10	ความสัมพันธ์ระหว่างความถี่และค่า VSWR.....67
3.11	ความสัมพันธ์ระหว่างความถี่และค่ากำลังของสัญญาณ.....68
3.12	ความสัมพันธ์ระหว่างความถี่และสัญญาณรบกวน.....68
3.13	ไอซีของวงจรแปลงสัญญาณความถี่ต่ำ.....69
3.14	วงจรแปลงสัญญาณความถี่ต่ำ.....70
3.15	ค่าจริงและจินตภาพเพื่อนำไปคำนวณเมทริกซ์ที่พอร์ต RF.....71
3.16	ค่าจริงและจินตภาพเพื่อนำไปคำนวณเมทริกซ์ที่พอร์ต IF.....71
3.17	ค่าพารามิเตอร์ต่างๆ ที่พอร์ต RF.....72
3.18	ค่าพารามิเตอร์ต่างๆ ที่พอร์ต IF.....72
3.19	ค่าสัมประสิทธิ์การสะท้อนที่พอร์ต LO.....73
3.20	สัญญาณที่รับได้จากบอร์ดไมโครคอนโทรลเลอร์ที่ความถี่ (ก) 100Hz (ข) 200Hz (ค) 300Hz และ (ง) 500Hz.....74
4.1	การวิเคราะห์หาทิศทางการมาถึงของสัญญาณเมื่อเลือกใช้ความถี่ (ก) 50Hz (ข) 100Hz (ค) 200Hz.....78
4.2	วงจรแปลงสัญญาณความถี่ต่ำที่สร้างขึ้นจริง.....79
4.3	ผลที่ได้จากวงจรแปลงสัญญาณความถี่ต่ำที่พอร์ต IF.....80
4.4	ชุดอุปกรณ์ต้นแบบวิเคราะห์หาทิศทางการมาถึงของสัญญาณในการทดสอบจริง.....81
4.5	สัญญาณที่รับได้จากบอร์ดไมโครคอนโทรลเลอร์ เมื่อมีสัญญาณ เข้ามาในทิศทาง 90 องศา.....82

## สารบัญรูป (ต่อ)

รูปที่	หน้า
4.6 ระบบสามารถหาทิศทางได้จริงที่ 90 องศา .....	82
4.7 สัญญาณที่รับได้จากบอร์ดไมโครคอนโทรลเลอร์ เมื่อมีสัญญาณ เข้ามาในทิศทาง 30 องศา.....	83
4.8 ระบบสามารถหาทิศทางได้จริงที่ 30 องศา .....	83
4.9 กราฟค่าความผิดพลาดเฉลี่ยที่ได้จากผลทดสอบวัดจริง เมื่อมีสัญญาณ เข้ามาในทิศทาง 30 องศา.....	85
4.10 แผนภาพการทำงานของโปรแกรมในกรณีนำค่ากำลังเฉลี่ยของสัญญาณ มาเปรียบเทียบกับเงื่อนไข.....	86
4.11 การหาค่า $P_k - P_k$ ของสัญญาณที่บอร์ดไมโครคอนโทรลเลอร์รับได้จากรูปที่ 4.7.....	87
4.12 กราฟค่าความผิดพลาดเฉลี่ยที่ได้จากผลทดสอบวัดจริง ในกรณีแย่งสุดของ แต่ละจำนวนการสุ่มตัวอย่าง.....	89
4.13 การกำหนดค่าความผิดพลาดเฉลี่ยให้ไม่เกิน 7 องศา.....	89
4.14 การปรับเลือกจำนวนการสุ่มตัวอย่างที่ได้จากผลวัดจริง เมื่อมีสัญญาณ เข้ามาที่ 30 องศา.....	90
4.15 ภาพขยายของรูปที่ 4.14 .....	91
4.16 กราฟค่าความผิดพลาดเฉลี่ยในกรณีแย่งสุดที่ได้จากการวัดจริง เมื่อทำการประมาณ กราฟในรูปแบบของเอ็กซ์โปเนนเชียล.....	92
4.17 ภาพขยายกราฟค่าความผิดพลาดเฉลี่ยที่ได้จากผลวัดจริงของรูปที่ 4.14.....	93
4.18 แบบจำลองทิศทางของสัญญาณจากภาคส่งที่ตั้งอยู่ในระยะต่างๆ.....	94
4.19 การปรับเลือกจำนวนการสุ่มตัวอย่างที่ได้จากผลวัดจริง เมื่อมีสัญญาณ เข้ามาในทิศทางต่างๆ ที่ได้จากรายที่ 4.4.....	95

# บทที่ 1

## บทนำ

เนื้อหาในบทนี้เป็นการกล่าวถึงความเป็นมาและเหตุจูงใจในการทำงานวิจัยฉบับนี้ ซึ่งประกอบด้วย ความเป็นมาและความสำคัญของปัญหา วัตถุประสงค์ของงานวิจัย แนวทางการดำเนินงานวิจัย ประโยชน์ที่คาดว่าจะได้รับ และส่วนประกอบของงานวิจัย

### 1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันระบบเครือข่ายไร้สายนั้นได้รับความนิยมเป็นอย่างมาก และมีการให้บริการอยู่ในหลายย่านความถี่ เนื่องจากระบบดังกล่าวได้ถูกพัฒนาจนสามารถส่งข้อมูลได้อย่างรวดเร็วและใช้งานได้อย่างสะดวกสบาย ไม่ว่าจะอยู่ในบริเวณใดของเครือข่ายก็ตาม แต่ระบบเครือข่ายไร้สายนั้นก็ยังคงมีปัญหาอยู่มาก เช่น สัญญาณคลื่นหลายวิถี (multipath signal) และสัญญาณแทรกสอด (interference signal) เป็นต้น ดังนั้นจึงได้มีการคิดระบบที่สามารถแก้ไขปัญหาดังกล่าวนั้นก็คือระบบเสาอากาศแก๊ง (smart antenna systems) เนื่องจากระบบสามารถที่จะก่อรูปลำคลื่นหลัก (main beam) ไปยังทิศทางของสัญญาณที่ต้องการ และในขณะที่เดียวกันก็จะหันพู่ข้าง (sidelobes) หรือจุดศูนย์ (nulls) ไปยังทิศทางของสัญญาณรบกวน กระบวนการดังกล่าวเรียกว่า การก่อรูปลำคลื่น (beamforming) ซึ่งกระบวนการนี้จะช่วยลดปัญหาของสัญญาณคลื่นหลายวิถีและสัญญาณแทรกสอดของช่องสัญญาณได้

ระบบเสาอากาศแก๊งนั้นจะประกอบไปด้วยสองส่วนหลักๆ คือ เสาอากาศแถวลำดับ (array antennas) และระบบประมวลผลสัญญาณ (signal processing systems) ซึ่งในส่วนของเสาอากาศแถวลำดับจะทำหน้าที่รับหรือส่งสัญญาณ และในส่วนของระบบการประมวลผลสัญญาณนั้นจะทำหน้าที่วิเคราะห์หาทิศทางการมาถึงของสัญญาณ (Direction-Of-Arrival: DOA) ซึ่งข้อมูลที่ได้มานั้นจะถูกนำมาใช้ในการคำนวณเพื่อก่อรูปลำคลื่นต่อไป โดยเทคนิคที่ใช้ในการวิเคราะห์หาทิศทางการเข้ามาของสัญญาณนั้นมีหลายวิธี เช่น Multiple Signal Classification (MUSIC) algorithm และ Estimation of Signal Parameters via Rotational Invariance Technique (ESPRIT) เป็นต้น แต่วิธีที่เป็นที่นิยมและถูกใช้ในการวิเคราะห์หาทิศทางการมาถึงของสัญญาณมากที่สุด ได้แก่ วิธี MUSIC เนื่องจากเป็นวิธีที่ไม่ซับซ้อนและสามารถวิเคราะห์หาทิศทางการมาถึงของสัญญาณได้อย่างถูกต้องและแม่นยำ อย่างไรก็ตามวิธี MUSIC นั้นก็ยังคงมีปัญหาคือ จากการศึกษาปริทัศน์วรรณกรรมที่ผ่านมาได้มีการนำเสนองานวิจัยของ Wei, Z., and Xiaoli, X. (2007); Ye, Z., Xiang, L., and Xu, X.

(2007) ได้แสดงถึงปัญหาการวิเคราะห์หาทิศทางการมาถึงของสัญญาณวิธี MUSIC เมื่อระบบมีค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวน (Signal to Noise Ratio: SNR) ที่ต่ำ ประสิทธิภาพในการวิเคราะห์หาทิศทางการมาถึงของสัญญาณก็จะลดต่ำลง ดังนั้นจากงานวิจัยของ Kaveh, M., and Barabell, A.J. (1986); Xin, J., and Sano, A. (2004) ได้นำเสนอวิธีการแก้ปัญหา โดยจำเป็นจะต้องเพิ่มจำนวนการสุ่มตัวอย่าง (snapshot) ให้เหมาะสมกับค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวน ซึ่งถ้าเราใช้จำนวนการสุ่มตัวอย่างน้อยไป ระบบก็ไม่สามารถวิเคราะห์หาทิศทางการมาถึงของสัญญาณได้อย่างถูกต้องและแม่นยำ แต่ถ้าใช้จำนวนการสุ่มตัวอย่างมากเกินไป หน่วยประมวลผลกลางหรือซีพียู (CPU: Central Processing Unit) ของตัวประมวลผลก็จะทำงานหนักขึ้น เนื่องจากระบบยังมีจำนวนการสุ่มตัวอย่างมากขึ้นเท่าไร เวลาที่ใช้ในการประมวลผลก็จะนานมากขึ้นเท่านั้น ส่งผลให้ซีพียูของตัวประมวลผลทำงานหนักขึ้นตามไปด้วย

ดังนั้นงานวิจัยฉบับนี้จึงนำเสนอการใช้เทคนิคการปรับจำนวนการสุ่มตัวอย่าง (adaptive snapshot technique) มาประยุกต์ใช้กับวิธี MUSIC ในการวิเคราะห์หาทิศทางการมาถึงของสัญญาณในระบบสายอากาศแก่ง โดยจะทำการปรับเลือกจำนวนการสุ่มตัวอย่างให้เหมาะสมกับค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวนในระบบ เพื่อทำการวิเคราะห์หาทิศทางการมาถึงของสัญญาณให้ได้ อย่างถูกต้องและแม่นยำ นอกจากนี้ระบบยังไม่จำเป็นต้องใช้จำนวนการสุ่มตัวอย่างที่มากเกินไป ความเหมาะสม ดังนั้นตัวประมวลผลในระบบจึงสามารถที่จะประมวลผลได้อย่างรวดเร็ว ถือเป็น การลดข้อเสียของตัวประมวลผลไม่ให้ทำงานหนักได้อีกด้วย

## 1.2 วัตถุประสงค์ของงานวิจัย

งานวิจัยนี้กล่าวถึงการวิเคราะห์หาทิศทางการมาถึงของสัญญาณ ที่มีการใช้เทคนิคการปรับจำนวนการสุ่มตัวอย่างมาประยุกต์ใช้กับวิธี MUSIC โดยรายละเอียดสามารถอธิบายได้ดังนี้

1.2.1 เพื่อศึกษาวิธีการวิเคราะห์หาทิศทางการมาถึงของสัญญาณด้วยวิธี MUSIC ในระบบสายอากาศแก่ง

1.2.2 เพื่อศึกษาและออกแบบสร้างชุดอุปกรณ์ที่ใช้ในการวิเคราะห์หาทิศทางการมาถึงของสัญญาณ โดยการใช้เทคนิคการปรับจำนวนการสุ่มตัวอย่างที่ได้นำเสนอในงานวิจัยนี้ มาประยุกต์ใช้ร่วมกับวิธี MUSIC

### 1.3 สมมติฐานของการวิจัย

1.3.1 ชุดอุปกรณ์การวิเคราะห์หาทิศทางการมาถึงของสัญญาณ โดยใช้เทคนิคการปรับจำนวนการสุ่มตัวอย่าง สามารถที่จะวิเคราะห์หาทิศทางได้อย่างถูกต้องและแม่นยำ

1.3.2 การใช้เทคนิคการปรับจำนวนการสุ่มตัวอย่าง สามารถช่วยให้ระบบสามารถประมวลผลได้เร็วขึ้น

### 1.4 ข้อตกลงเบื้องต้น

1.4.1 วิธีที่ใช้ในการวิเคราะห์หาทิศทางการมาถึงของสัญญาณ คือ วิธี MUSIC

1.4.2 การใช้เทคนิคการปรับจำนวนการสุ่มตัวอย่างจะนำไปประยุกต์ใช้กับการวิเคราะห์หาทิศทางการมาถึงของสัญญาณด้วยวิธี MUSIC

1.4.3 ใช้โปรแกรมแมทแลบ (Matlab) ในการจำลองผล

### 1.5 ขอบเขตการวิจัย

1.5.1 จำลองผลการวิเคราะห์หาทิศทางการมาถึงของสัญญาณ โดยใช้เทคนิคการปรับจำนวนการสุ่มตัวอย่างด้วยโปรแกรมแมทแลบ

1.5.2 ออกแบบและสร้างชุดอุปกรณ์ต้นแบบเพื่อใช้ในการวิเคราะห์หาทิศทางการมาถึงของสัญญาณ โดยใช้เทคนิคการปรับจำนวนการสุ่มตัวอย่าง

1.5.3 วิเคราะห์ผลโดยเปรียบเทียบระหว่างผลที่ได้จากการจำลองแบบและผลที่ได้จากการวัดจริงของระบบต้นแบบ

### 1.6 วิธีดำเนินการวิจัย

1.6.1 แนวทางการดำเนินงาน

- 1) สํารวจปริทัศน์วรรณกรรม และงานวิจัยที่เกี่ยวข้องกับงานวิจัย
- 2) ศึกษาทฤษฎีที่เกี่ยวข้องกับระบบสายอากาศแก่ง
- 3) ศึกษาการวิเคราะห์หาทิศทางการมาถึงของสัญญาณ
- 4) ใช้โปรแกรมแมทแลบในการจำลองผลการวิเคราะห์หาทิศทางการมาถึงของสัญญาณด้วยวิธี MUSIC



5) ใช้โปรแกรมแมทแลบในการจำลองผลการวิเคราะห์หาทิศทางการมาถึงของสัญญาณด้วยการใช้เทคนิคการปรับจำนวนการสุ่มตัวอย่าง โดยประยุกต์ใช้ร่วมกับวิธี MUSIC

6) ออกแบบและสร้างชุดอุปกรณ์ต้นแบบที่ใช้ในการวิเคราะห์หาทิศทางการมาถึงของสัญญาณเพื่อนำไปใช้ทดสอบจริง

7) วิเคราะห์ผลโดยเปรียบเทียบระหว่างผลที่ได้จากการจำลองแบบและผลที่ได้จากการวัดจริงของระบบต้นแบบ

1.6.2 ระเบียบวิธีวิจัย เป็นงานวิจัยประยุกต์ ซึ่งดำเนินการตามกรอบงานดังต่อไปนี้

1) การศึกษาและเก็บรวบรวมข้อมูล โดยการสำรวจปริทัศน์วรรณกรรมและงานวิจัยที่เกี่ยวข้องกับงานวิจัย

2) ใช้โปรแกรมแมทแลบในการจำลองผลการวิเคราะห์หาทิศทางการมาถึงของสัญญาณด้วยการใช้เทคนิคการปรับจำนวนการสุ่มตัวอย่าง โดยประยุกต์ใช้ร่วมกับวิธี MUSIC

3) สร้างชุดอุปกรณ์ต้นแบบการวิเคราะห์หาทิศทางการมาถึงของสัญญาณเพื่อนำไปใช้ทดสอบจริง

4) วิเคราะห์ผลโดยเปรียบเทียบระหว่างผลที่ได้จากการจำลองแบบและผลที่ได้จากการวัดจริงของระบบต้นแบบ

5) รวบรวมข้อมูลที่ได้มาเขียนงานวิจัย

1.6.3 สถานที่ทำการวิจัย

ห้องปฏิบัติการวิจัยกลุ่มวิศวกรรมโทรคมนาคมและปฏิบัติการสื่อสารไร้สายอาคารเครื่องมือ 4 มหาวิทยาลัยเทคโนโลยีสุรนารี 111 ถ.มหาวิทยาลัย ต.สุรนารี อ.เมือง จ. นครราชสีมา 30000

1.6.4 เครื่องมือที่ใช้ในการวิจัย

- 1) เครื่องคอมพิวเตอร์ส่วนบุคคล (personal computer)
- 2) โปรแกรมแมทแลบ
- 3) เครื่องวิเคราะห์วงจรรข่าย (network analyzer)
- 4) เครื่องกำเนิดสัญญาณ (signal generator)
- 5) ออสซิลโลสโคป

### 1.6.5 การเก็บรวบรวมข้อมูล

- 1) เก็บผลจากการจำลองผลการวิเคราะห์หาทิศทางการมาถึงของสัญญาณด้วยการใช้เทคนิคการปรับจำนวนการสุ่มตัวอย่าง โดยประยุกต์ใช้ร่วมกับวิธี MUSIC
- 2) เก็บผลการทดสอบที่ได้จากการจำลองแบบและผลที่ได้จากการวัดจริงของระบบต้นแบบ

### 1.6.6 การวิเคราะห์ข้อมูล

นำผลที่ได้จากการจำลองแบบโดยโปรแกรมแมทแลบมาเปรียบเทียบกับผลที่ได้จากการวัดจริงของระบบต้นแบบ เพื่อทำการวิเคราะห์และสรุปผล

## 1.7 ประโยชน์ที่คาดว่าจะได้รับ

1.7.1 การใช้เทคนิคการปรับจำนวนการสุ่มตัวอย่าง สามารถช่วยให้ระบบมีการประมวลผลได้เร็วขึ้น ถือเป็นการถนอมชีพียูของตัวประมวลผลไม่ให้งานหนักอีกด้วย

1.7.2 สามารถนำชุดอุปกรณ์การวิเคราะห์หาทิศทางการมาถึงของสัญญาณ โดยการใช้เทคนิคการปรับจำนวนการสุ่มตัวอย่าง ไปวิเคราะห์หาทิศทางของสัญญาณไร้สาย

## 1.8 ส่วนประกอบของงานวิจัย

งานวิจัยฉบับนี้ประกอบด้วย 5 บท

**บทที่ 1** เป็นบทนำ กล่าวถึงความจำเป็นมาและความสำคัญของปัญหา วัตถุประสงค์ของงานวิจัย สมมติฐานของการวิจัย ข้อตกลงเบื้องต้น ขอบเขตการวิจัย วิธีดำเนินการวิจัยและประโยชน์ที่คาดว่าจะได้รับ

**บทที่ 2** กล่าวถึงภาคส่ง-รับในระบบสื่อสาร อุปกรณ์ต่างๆ ในระบบภาครับสัญญาณ ระบบสายอากาศแก่ การวิเคราะห์การมาถึงของสัญญาณแบบวิธี MUSIC และเทคนิคการปรับจำนวนการสุ่มตัวอย่าง

**บทที่ 3** กล่าวถึงการออกแบบอุปกรณ์ภาครับ และสร้างโปรแกรมควบคุมบอร์ดไมโครคอนโทรลเลอร์ในการแปลงข้อมูลอนาล็อกเป็นดิจิทัลและส่งไปยังคอมพิวเตอร์ โดยใช้โปรแกรมแมทแลบในการรับข้อมูลและทำการหาทิศทางต่อไป

**บทที่ 4** กล่าวถึงการทดสอบวงจรแปลงสัญญาณความถี่ต่ำ และการทดสอบชุดอุปกรณ์ต้นแบบ ได้แก่ การวิเคราะห์หาทิศทางการมาถึงของสัญญาณ รวมไปถึงการทดสอบหาทิศทางด้วยเทคนิคการปรับจำนวนการสุ่มตัวอย่าง

**บทที่ 5** กล่าวถึงการสรุปผลของการวิจัย ปัญหาและข้อเสนอแนะแนวทางแก้ไขปัญหา และ  
แนวทางการพัฒนาต่อในอนาคต



## บทที่ 2

### ทฤษฎีและหลักการที่เกี่ยวข้อง

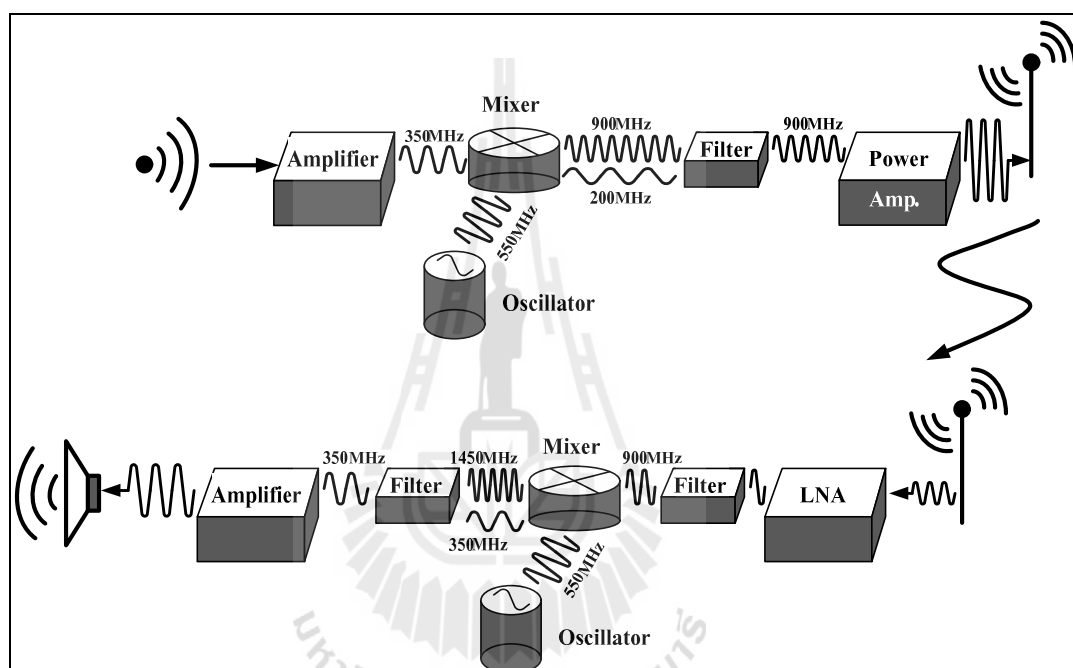
#### 2.1 กล่าวนำ

ในการสร้างชุดอุปกรณ์ต้นแบบตัวหาทิศทางการมาถึงของสัญญาณแบบวิธี MUSIC ร่วมกับเทคนิคการปรับจำนวนการสุ่มตัวอย่างนั้นประกอบไปด้วย 2 ส่วนหลักๆ คือ อุปกรณ์ภาครับสัญญาณและตัวประมวลผลสัญญาณ ซึ่งทั้งหมดนี้จำเป็นต้องมีการศึกษาในส่วนของทฤษฎีที่เกี่ยวข้อง ดังนั้นบทนี้จะเป็นการกล่าวถึงทฤษฎีภาครับสัญญาณในระบบสื่อสาร ซึ่งจะได้นำกล่าวถึงหน้าที่ของอุปกรณ์แต่ละส่วน เช่น สายอากาศแถวลำดับ (antenna array) วงจรแปลงสัญญาณความถี่ต่ำ (down converter) วงจรขยายสัญญาณ (amplifier) วงจรกรองสัญญาณ (filter) เป็นต้น ทฤษฎีของระบบสายอากาศจะเน้นไปในส่วนของการวิเคราะห์หาทิศทางแบบวิธี MUSIC ร่วมกับเทคนิคการปรับจำนวนการสุ่มตัวอย่างซึ่งเป็นส่วนประกอบที่สำคัญที่สุดของงานวิจัยฉบับนี้และสุดท้ายเป็นการสรุปเนื้อหาทั้งหมดของบทนี้

#### 2.2 พื้นฐานเครือข่ายไร้สาย

การสื่อสารด้วยการสนทนาพูดคุยจัดเป็นกิจกรรมส่วนหนึ่งในชีวิตประจำวันของมนุษย์เรา ซึ่งหากพิจารณาในรายละเอียดของการพูดคุยสนทนากันนั้น จะประกอบไปด้วยคู่สนทนาตั้งแต่สองคนขึ้นไป แต่ละคนก็จะสามารถเป็นได้ทั้งผู้พูดและผู้ฟังเพื่อแลกเปลี่ยนความคิดเห็นซึ่งกันและกัน การสื่อสารในลักษณะนี้จัดเป็นการสื่อสารในระยะทางใกล้ๆเท่านั้น แต่เมื่อในกรณีที่ต้องการสื่อสารบนระยะทางที่ห่างไกลกันในรูปแบบของอิเล็กทรอนิกส์ มนุษย์จึงจำเป็นต้องใช้เทคโนโลยีการสื่อสาร หรือเทคโนโลยีโทรคมนาคมเข้ามาช่วยเพื่อตอบสนองความต้องการดังกล่าว ซึ่งการสื่อสารทางอิเล็กทรอนิกส์นั้นจะเป็นกระบวนการถ่ายโอนข้อมูลหรือข่าวสารจากต้นทาง (source) ไปยังปลายทาง (destination) อาจมีทั้งสื่อแบบมีสาย (guided media) และสื่อแบบไร้สาย (unguided media: wireless) ก็ได้ ยกตัวอย่างเช่น การส่งข้อมูลคอมพิวเตอร์ เสียงโทรศัพท์ การส่งสัญญาณโทรทัศน์และวิทยุ ซึ่งอาจจะโอนถ่ายกันบนสายสื่อสาร เช่น สายทองแดง (copper cable) เส้นใยแก้วนำแสง (fiber optic cable) หรือสื่อไร้สายอย่างคลื่นวิทยุ (radio wave) และคลื่นแม่เหล็กไฟฟ้า (microwave) เป็นต้น โดยข้อมูลจากต้นทางที่ส่งไปนั้น ซึ่งเดิมอยู่ในรูปแบบของสัญญาณเสียงหรือ

สัญญาณอนาล็อก (analog signal) ก็อาจจะถูกแปลงให้เป็นสัญญาณดิจิทัล (digital signal) เพื่อให้เกิดประสิทธิภาพและความปลอดภัยต่อการเดินทางในระยะไกลๆ บนสายดิจิทัลความเร็วสูง และท้ายสุดเมื่อสัญญาณเดินทางไปถึงปลายทางก็จะมีการเปลี่ยนแปลงกลับมาเป็นสัญญาณอนาล็อกเพื่อให้เหมือนกับข้อมูลต้นฉบับที่ส่งมา โดยอ้างอิงจากหนังสือของลัญจกร วุฒิสัทติกุลกิจ. (2542); สุขสันต์ เรือนแก้ว. (2548)



รูปที่ 2.1 อุปกรณ์ส่งสัญญาณ และอุปกรณ์รับสัญญาณ

รูปที่ 2.1 แสดงการสื่อสารไร้สายจากอุปกรณ์ส่งสัญญาณไปยังอุปกรณ์รับสัญญาณ โดยในขั้นตอนแรกสัญญาณที่ถูกส่งจะถูกสร้างจากอุปกรณ์ที่สร้างสัญญาณ เช่น คอมพิวเตอร์ สถานีวิทยุ หรือโทรศัพท์มือถือ เป็นต้น สัญญาณที่ถูกสร้างขึ้นนั้นจะต้องนำไปเข้าวงจรขยายสัญญาณ (amplifier) ก่อนที่จะทำการส่งสัญญาณที่ขยายแล้วไปผสมกับสัญญาณที่ออสซิลเลเตอร์ (oscillator) สร้างขึ้น ด้วยวงจรผสมสัญญาณ (mixer) ตัวอย่างเช่น สัญญาณจากวงจรขยายสัญญาณมีความถี่ 350MHz แต่เนื่องจากความถี่ของสัญญาณที่จะส่งโดยอุปกรณ์ส่งสัญญาณนั้นต้องการอยู่ที่ 900MHz ดังนั้นออสซิลเลเตอร์จะสร้างสัญญาณที่มีความถี่ 550MHz เข้าไปในวงจรผสมสัญญาณเพื่อรวมสัญญาณทั้งสองเข้าด้วยกัน โดยสัญญาณที่ผสมแล้วจะถูกส่งออกมาอยู่สองส่วนคือ ส่วนที่ได้จากการบวกกันคือ 900MHz (350+550) และส่วนที่ได้จากการลบกันคือ 200MHz (550-350) เป็นต้น

เห็นได้ว่าสัญญาณที่ได้นั้นจะมีสัญญาณที่ไม่ต้องการรวมอยู่ด้วย ดังนั้นจึงจำเป็นต้องหาวิธีการที่จะกำจัดคลื่นที่ไม่ต้องการนั้นออกไปด้วยวงจรกรองสัญญาณ (filter) แต่อย่างไรก็ตามสัญญาณที่ได้ยังอ่อนอยู่มากที่จะถูกส่งไปยังที่ไกลๆ จึงต้องมีตัวขยายสัญญาณอีกตัวหนึ่ง ซึ่งมีกำลังขยายที่คิดว่าตัวแรกเรียกว่า วงจรขยายสัญญาณกำลังสูง (high power amplifier) เพื่อเพิ่มความเข้มของสัญญาณแล้วจึงจะส่งสัญญาณนี้ออกไปใช้ในการสื่อสารต่อไป

จากนั้นอุปกรณ์รับสัญญาณจะมีสายอากาศ (antenna) ไว้คอยรับสัญญาณ และจะใช้วงจรขยายสัญญาณเพื่อเพิ่มความเข้มของสัญญาณเนื่องจากสัญญาณที่รับมานั้นจะอ่อนมาก ซึ่งตัวขยายในอุปกรณ์รับสัญญาณนี้เรียกว่า วงจรขยายสัญญาณรบกวนต่ำ (low noise amplifier) และเนื่องจากสายอากาศนั้นอาจจะรับสัญญาณข้างเคียงที่ไม่ต้องการเข้ามา วงจรกรองสัญญาณก็จะทำการกรองสัญญาณที่ไม่จำเป็นออกไป ซึ่งจากตัวอย่างตัวกรองจะกำจัดสัญญาณทุกตัวออกไปยกเว้นที่มีความถี่ 900MHz จากนั้นวงจรผสมสัญญาณจะทำหน้าที่แยกสัญญาณออกมา ตัวอย่างเช่น สัญญาณที่มีความถี่ 350MHz และสัญญาณทั้งหมดที่ส่งมานั้นมีความถี่อยู่ที่ 900MHz ดังนั้นออสซิลเลเตอร์จำเป็นต้องสร้างสัญญาณความถี่ที่ 550MHz จะได้สัญญาณที่เดินทางออกจากวงจรผสมสัญญาณออกมาสองความถี่คือ 1450MHz ( $900+550$ ) และ 350MHz ( $900-550$ ) เป็นต้น ดังนั้นระบบจึงจำเป็นต้องมีวงจรกรองสัญญาณอีกตัวหนึ่งเพื่อกรองสัญญาณที่ไม่ต้องการออกไปหรือในที่นี้คือความถี่ 1450MHz นั่นเอง หลังจากผ่านกระบวนการของวงจรกรองสัญญาณ และวงจรผสมสัญญาณแล้วจะทำให้สัญญาณที่ได้นั้นอ่อนลง ดังนั้นจำเป็นต้องทำให้สัญญาณแรงขึ้นโดยการส่งไปยังวงจรขยายสัญญาณตัวที่สอง สัญญาณที่ออกมาก็สามารถนำไปใช้ได้แล้ว เช่น รับฟังเสียงจากลำโพงหรือโทรศัพท์มือถือ เป็นต้น ซึ่งส่วนประกอบต่างๆ ของอุปกรณ์รับสัญญาณจะได้นำเสนอโดยละเอียดในหัวข้อถัดไป

### 2.3 สายอากาศแถวลำดับ

สายอากาศแถวลำดับ คือ การนำสายอากาศมาเรียงตัวกันในรูปแบบต่างๆ โดยมีระยะห่างระหว่างสายอากาศแต่ละตัวที่เท่ากันหรือไม่ก็ได้ ซึ่งสายอากาศแต่ละตัวที่นำมาจัดเรียงให้เป็นแถวลำดับนั้น เรียกว่า องค์ประกอบ (element) สายอากาศแถวลำดับนั้นถือว่าเป็นส่วนประกอบที่สำคัญมากในระบบสายอากาศเก่งที่ทำให้สามารถหันลำคลื่นหลัก (main lobes) ไปยังทิศทางของสัญญาณที่ต้องการ และสามารถหันพู่ข้างหรือจุดศูนย์ (nulls) ไปยังทิศทางของสัญญาณแทรกสอด ส่งผลให้ระบบสามารถเพิ่มค่าสภาพเจาะจงทิศทางและค่าอัตราขยายของสายอากาศได้ โดยทั่วไปการวางตัวของสายอากาศแถวลำดับมีอยู่หลายรูปแบบด้วยกัน แต่วิธีที่เป็นที่นิยมใช้กันอย่างแพร่หลายนั้นก็คือ

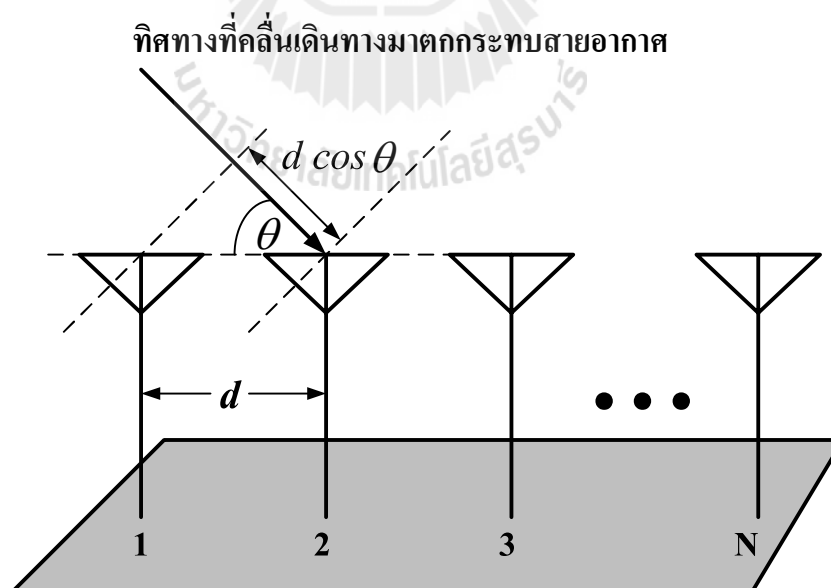
สายอากาศแถวลำดับแบบเชิงเส้น (linear arrays) และสายอากาศแถวลำดับแบบเชิงระนาบ (planar arrays) ซึ่งมีรายละเอียดดังต่อไปนี้

### 2.3.1 สายอากาศแถวลำดับแบบเชิงเส้น

สายอากาศแถวลำดับแบบเชิงเส้นเป็นสายอากาศแถวลำดับที่มีการเรียงตัวกันของสายอากาศแต่ละต้นเป็นแนวเส้นตรง ซึ่งถือว่าเป็นสายอากาศแถวลำดับแบบพื้นฐาน และมีโครงสร้างที่เรียบง่ายที่สุด การวางตัวสายอากาศของสายอากาศแถวลำดับจำเป็นต้องคำนึงถึงระยะห่างระหว่างสายอากาศ ( $d$ ) ขององค์ประกอบแต่ละองค์ประกอบนั้นด้วย เนื่องจากระยะห่างของแต่ละองค์ประกอบนั้นจะส่งผลต่อการแผ่กระจายคลื่นของสายอากาศ โดยทั่วไประยะห่างที่ใช้ของสายอากาศแต่ละต้นจะวางตัวห่างกันเป็นระยะครึ่งความยาวคลื่น ซึ่งสามารถคำนวณได้จากสมการที่ (2.1) ดังนี้

$$d = \frac{\lambda}{2} \quad (2.1)$$

เมื่อ  $\lambda$  คือ ความยาวคลื่น



รูปที่ 2.2 การวางตัวของสายอากาศแบบเชิงเส้นจำนวน  $N \times 1$  ต้น

รูปที่ 2.2 แสดงสายอากาศแถวลำดับแบบเชิงเส้นจำนวน  $N$  ต้นหรือ  $N \times 1$  ต้น กำหนดให้ระยะห่างของสายอากาศแต่ละต้นมีระยะที่เท่ากัน และมีแอมพลิจูดเท่ากัน โดยที่สายอากาศแต่ละต้นที่ถัดมาจากต้นแรกหรือต้นอ้างอิงจะมีค่าเฟสที่มากกว่าเมื่อเปรียบเทียบกับต้นก่อนหน้า รูปแบบดังกล่าวนี้เรียกว่า แถวลำดับสม่ำเสมอ (uniform array) โดยอ้างอิงจากหนังสือของ Frank, B.G., Ph.D. (2005) ซึ่งเราสามารถหาค่าพลังงานของสายอากาศแถวลำดับนี้ได้จากการคูณกันของค่าพลังงานของสายอากาศต้นเดียวที่จุดอ้างอิงกับตัวประกอบแถวลำดับ (Array Factor:  $AF$ ) ในการหาตัวประกอบแถวลำดับนั้นสามารถหาได้จากสมการที่ (2.2)

$$AF = 1 + e^{+j(kd \cos \theta + \beta)} + e^{+j2(kd \cos \theta + \beta)} + \dots + e^{j(N-1)(kd \cos \theta + \beta)} \quad (2.2)$$

$$AF = \sum_{n=1}^N e^{j(n-1)(kd \cos \theta + \beta)} \quad (2.3)$$

ซึ่งสามารถเขียนได้อีกรูปหนึ่งดังสมการที่ (2.4)

$$AF = \sum_{n=1}^N e^{j(n-1)\psi} \quad (2.4)$$

เมื่อ  $\psi = kd \cos \theta + \beta$  โดยที่  $k$  คือ หมายเลขคลื่น (wave number) ซึ่งมีค่าเท่ากับ  $2\pi / \lambda$ ,  $d$  คือ ระยะห่างระหว่างสายอากาศแต่ละต้น,  $\theta$  คือ มุมของคลื่นที่เดินทางมาตกกระทบที่สายอากาศแต่ละต้น และ  $\beta$  คือ ความต่างเฟสของสายอากาศแต่ละต้น ในกรณีที่ เป็นแถวลำดับไม่สม่ำเสมอ (nonuniform array) นั้น ก็ยังสามารถใช้สมการหาตัวประกอบแถวลำดับที่กล่าวมาข้างต้นได้เช่นกัน และจากสมการ (2.4) เราสามารถลดรูปสมการได้ดังนี้

$$(AF)e^{j\psi} = e^{j\psi} + e^{j2\psi} + e^{j3\psi} + \dots + e^{j\psi(N-1)} + e^{jN\psi} \quad (2.5)$$

แทนสมการ (2.4) ลงใน (2.5) จะสามารถลดรูปสมการลงเหลือ

$$(AF)(e^{j\psi} - 1) = (-1 + e^{jN\psi}) \quad (2.6)$$



ทำการย้ายข้างและจัดรูปสมการจะได้

$$\begin{aligned}
 AF &= \left[ \frac{e^{jN\psi} - 1}{e^{j\psi} - 1} \right] \\
 &= e^{j[(N-1)/2]\psi} \left[ \frac{e^{j(N/2)\psi} - e^{-j(N/2)\psi}}{e^{j(1/2)\psi} - e^{-j(1/2)\psi}} \right] \\
 &= e^{j[(N-1)/2]\psi} \left[ \frac{\sin\left(\frac{N}{2}\psi\right)}{\sin\left(\frac{1}{2}\psi\right)} \right] \tag{2.7}
 \end{aligned}$$

เมื่อกำหนดให้จุดอ้างอิงอยู่ตรงจุดศูนย์กลางของสายอากาศแถวลำดับ จะได้ระยะห่างของสายอากาศ  $d=0$  และ  $\beta=0$  จะได้  $\psi = kd \cos \theta + \beta = 0$  ดังนั้นสมการที่ (2.7) จะสามารถลดรูปลงได้เท่ากับ

$$AF = \left[ \frac{\sin\left(\frac{N}{2}\psi\right)}{\sin\left(\frac{1}{2}\psi\right)} \right] \tag{2.8}$$

จะเห็นได้ว่า  $\psi$  ถือว่ามีค่าน้อยมาก ๆ ดังนั้นเราสามารถประมาณค่าสมการได้เท่ากับ

$$AF \cong \left[ \frac{\sin\left(\frac{N}{2}\psi\right)}{\frac{N}{2}\psi} \right] \tag{2.9}$$

เมื่อพิจารณาค่าสูงสุดของสมการที่ (2.8) และ (2.9) จะมีค่าเท่ากับ  $N$  เพื่อกำหนดให้ค่าสูงสุดของแต่ละสมการมีค่าเท่ากับหนึ่ง ดังนั้นสมการมาตรฐานของตัวประกอบแถวลำดับคือ

$$(AF)_n \cong \frac{1}{N} \left[ \frac{\sin\left(\frac{N}{2}\psi\right)}{\sin\left(\frac{1}{2}\psi\right)} \right] \tag{2.10}$$

หรือ

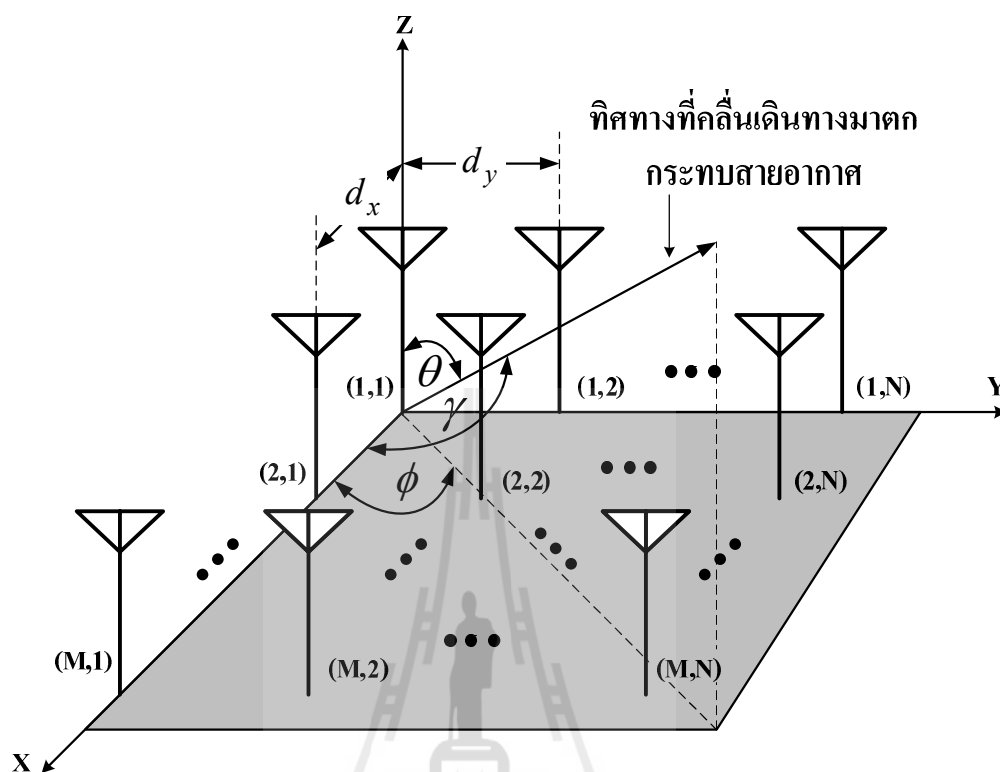
$$(AF)_n \cong \left[ \frac{\sin\left(\frac{N}{2}\psi\right)}{\frac{N}{2}\psi} \right] \quad (2.11)$$

ดังนั้นเมื่อเราทราบค่าตัวประกอบแฉวลำดับแล้วเราก็สามารถหาค่าพลังงานของสายอากาศแบบเชิงเส้นได้จากสมการนี้

$$E (\text{ผลรวม}) = [E (\text{ของสายอากาศต้นเดียว ณ จุดอ้างอิง})] \times [\text{ตัวประกอบแฉวลำดับ}] \quad (2.12)$$

### 2.3.2 สายอากาศแฉวลำดับแบบเชิงระนาบ

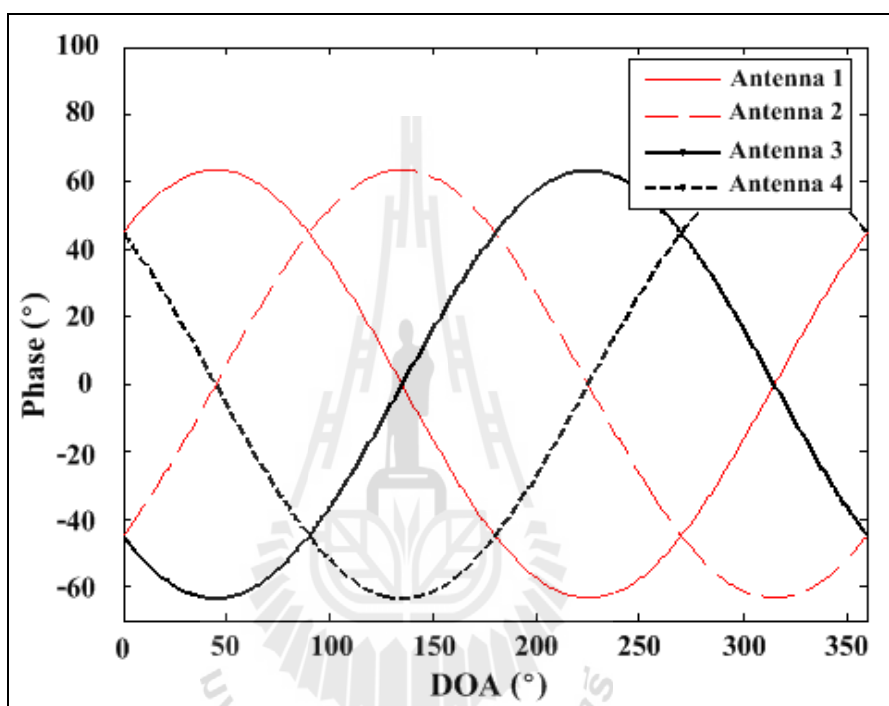
สายอากาศแฉวลำดับแบบเชิงระนาบเป็นรูปแบบของสายอากาศแต่ละตัวถูกจัดเรียงตัวกันเป็นสี่เหลี่ยมมุมฉาก ซึ่งเป็นรูปแบบที่ได้มีการประยุกต์มาจากรูปแบบสายอากาศแฉวลำดับแบบเชิงเส้น แต่การจัดเรียงตัวกันของสายอากาศในลักษณะนี้จะมีแบบรูปการแผ่พลังงาน (radiation pattern) ที่ยืดหยุ่นกว่าแบบเชิงเส้น ก็คือสามารถที่จะควบคุมและเปลี่ยนแปลงแบบรูปการแผ่พลังงานได้ ดังนั้นสายอากาศแฉวลำดับแบบเชิงระนาบจึงมีความเอนกประสงค์มากและสามารถให้แบบรูปการแผ่พลังงานที่มีความสมดุลและยังมีระดับพู่ข้างที่ต่ำอีกด้วย นอกจากนี้ระบบยังสามารถที่จะหันพู่หลักไปในมุมเงยทุกๆ ทิศทางรอบตัว 360° ดังนั้นสายอากาศที่มีการเรียงตัวแบบเชิงระนาบนี้จึงเป็นสายอากาศที่สามารถนำไปประยุกต์ใช้ประโยชน์ได้อย่างมากมาย เช่น การใช้งานในระบบเรดาร์ (radar) การชี้ทิศทางระยะไกล (remote sensing) รวมไปถึงการสื่อสารไร้สายอื่น ๆ เช่น ในระบบสายอากาศแฉว ซึ่งจะถูกรายละเอียดในส่วนถัดไป



รูปที่ 2.3 การวางตัวของสายอากาศแถวลำดับเชิงระนาบ  $M \times N$  ต้น

รูปที่ 2.3 แสดงการวางตัวของสายอากาศแถวลำดับเชิงระนาบแบบสี่เหลี่ยมบนระนาบ  $x-y$  โดยที่สายอากาศ  $M$  ต้นจะวางตัวอยู่ในตำแหน่งทิศทางของแกน  $x$  ซึ่งมีระยะห่างระหว่างสายอากาศคือ  $d_x$  และสายอากาศ  $N$  ต้นจะวางตัวอยู่ในตำแหน่งทิศทางของแกน  $y$  ที่มีระยะห่างระหว่างสายอากาศคือ  $d_y$  ดังนั้นเราจะได้ขนาดของสายอากาศแถวลำดับเชิงระนาบแบบสี่เหลี่ยม  $M \times N$  ต้น และมีค่าสัมประสิทธิ์การถ่วงน้ำหนักของสัญญาณทั้งระบบคือ  $W_{mn}$  ที่สายอากาศต้น  $(m,n)$  จากที่เราได้กล่าวไว้ข้างต้นว่าสายอากาศแถวลำดับแบบเชิงระนาบนั้นมีพื้นฐานและแนวคิดมาจากสายอากาศแถวลำดับเชิงเส้น ก็คือในสายอากาศแถวลำดับเชิงระนาบนั้นเราสามารถมองได้ว่าที่สายอากาศ  $M$  ต้น ในแต่ละต้นจะมีสายอากาศ  $N$  ที่มีการเรียงตัวแบบแถวลำดับเชิงเส้นเป็นองค์ประกอบอยู่ด้วย หรือในแบบเดียวกันเมื่อมีสายอากาศ  $N$  ต้น ในแต่ละต้นจะมีสายอากาศ  $M$  ที่มีการเรียงตัวแบบแถวลำดับเชิงเส้นเป็นองค์ประกอบเช่นเดียวกัน และเมื่อทำการพิจารณาสัญญาณเข้ามาที่มุม  $\theta$  หรือ มุมในแนวระนาบ (azimuth plane) สายอากาศแต่ละต้นนั้นจะสามารถรับสัญญาณจากแหล่งกำเนิดสัญญาณได้เหมือนกันทุกต้น แต่จะแตกต่างกันที่เวลาของสายอากาศแต่ละต้นขณะที่สัญญาณลงมา

ตกกระทบที่สายอากาศ โดยอ้างอิงจากหนังสือของ Allen, B., and Ghavami, M. (2005) ดังนั้นเราจึงจะได้กราฟของเฟสที่แตกต่างกันของสายอากาศแถวลำดับเชิงระนาบขนาด  $2 \times 2$  ที่มีระยะห่างระหว่างสายอากาศแต่ละต้นเท่ากับ  $\lambda/4$  ที่ตำแหน่งมุม  $\theta$  ตั้งแต่  $0^\circ$  ถึง  $360^\circ$  ของสายอากาศแต่ละต้น ได้ดังที่แสดงในรูปที่ 2.4



รูปที่ 2.4 เฟสที่ต่างกันของสายอากาศแต่ละต้นสำหรับสายอากาศแถวลำดับแบบเชิงระนาบขนาด  $2 \times 2$  เปรียบเทียบกับทิศทางการมาถึงของสัญญาณตั้งแต่  $0^\circ$  ถึง  $360^\circ$

เมื่อเราต้องการคำนวณหาค่าพลังงานของสายอากาศแถวลำดับแบบเชิงระนาบก็จะสามารถหาค่าได้จากสมการที่ (2.12) เช่นเดียวกันกับสายอากาศแถวลำดับแบบเชิงเส้น แต่จะแตกต่างกันตรงที่ค่าของตัวประกอบแถวลำดับ ซึ่งสามารถหาได้โดยเริ่มจากการพิจารณามุม  $\gamma$  ดังนี้

$$\begin{aligned} \cos \gamma &= \hat{a}_x \cdot \hat{a}_r = \hat{a}_x \cdot \left( \hat{a}_x \sin \theta \cos \phi + \hat{a}_y \sin \theta \sin \phi + \hat{a}_z \cos \theta \right) \\ &= \sin \theta \cos \phi \end{aligned} \quad (2.13)$$

เมื่อ  $\hat{a}_x$ ,  $\hat{a}_y$ ,  $\hat{a}_z$  และ  $\hat{a}_r$  คือเวกเตอร์หนึ่งหน่วยของแกน  $x$ ,  $y$ ,  $z$  และ  $r$  ตามลำดับ โดยที่  $\theta$  คือ มุมในแนวระนาบ และ  $\phi$  คือ มุมในแนวตั้ง เมื่อพิจารณาเฉพาะในแนวแกน  $x$  เราสามารถหาค่าตัวประกอบแถวลำดับของสายอากาศได้ดังนี้

$$\begin{aligned} AF_x &= \sum_{m=1}^M I_{m1} e^{j(m-1)(kd_x \cos \gamma + \beta_x)} \\ &= \sum_{m=1}^M I_{m1} e^{j(m-1)(kd_x \sin \theta \cos \phi + \beta_x)} \end{aligned} \quad (2.14)$$

เมื่อ  $I_{m1}$  คือค่าสัมประสิทธิ์กระแสกระตุ้นของสายอากาศแต่ละต้น,  $d_x$  คือระยะห่างของสายอากาศแต่ละต้นในแนวแกน  $x$  และ  $\beta_x$  คือค่าความต่างเฟสของสายอากาศแต่ละต้นในแนวแกน  $x$  เมื่อพิจารณาเฉพาะในแนวแกน  $y$  เช่นเดียวกันกับที่ได้พิจารณาในแนวแกน  $x$  เราจะได้ค่าตัวประกอบแถวลำดับของสายอากาศเท่ากับ

$$AF_y = \sum_{n=1}^N I_{1n} e^{j(n-1)(kd_y \sin \theta \sin \phi + \beta_y)} \quad (2.15)$$

เมื่อ  $I_{1n}$  คือค่าสัมประสิทธิ์กระแสกระตุ้นของสายอากาศแต่ละต้น,  $d_y$  คือระยะห่างของสายอากาศแต่ละต้นในแนวแกน  $y$  และ  $\beta_y$  คือค่าความต่างเฟสของสายอากาศแต่ละต้นในแนวแกน  $y$  ดังนั้น เราสามารถหาค่าตัวประกอบแถวลำดับของสายอากาศทั้งในแนวแกน  $x$  และ  $y$  รวมกัน หรือที่เรียกว่าแบบระนาบ ได้ด้วยการคูณค่าตัวประกอบแถวลำดับเข้าด้วยกันจะได้

$$AF = \sum_{n=1}^N I_{1n} \left[ \sum_{m=1}^M I_{m1} e^{j(m-1)(kd_x \sin \theta \cos \phi + \beta_x)} \right] e^{j(n-1)(kd_y \sin \theta \sin \phi + \beta_y)} \quad (2.16)$$

กำหนดให้แอมพลิจูดของสายอากาศแต่ละต้นทั้งในแนวแกน  $x$  และ  $y$  มีค่าเท่ากันจะได้

$$I_{mn} = I_{m1} I_{1n} \quad (2.17)$$

และกำหนดให้แอมพลิจูดมีค่าเท่ากับหนึ่งจะได้  $I_{mn} = I_0$  ดังนั้นเราสามารถลดรูปสมการ (2.16) ลงได้เท่ากับ

$$AF = I_0 \sum_{m=1}^M e^{j(m-1)(kd_x \sin \theta \cos \phi + \beta_x)} \sum_{n=1}^N e^{j(n-1)(kd_y \sin \theta \cos \phi + \beta_y)} \quad (2.18)$$

เช่นเดียวกันกับสายอากาศแถวลำดับแบบเชิงเส้น เราสามารถทำสมการค่าตัวประกอบแถวลำดับของสายอากาศให้อยู่ในรูปแบบมาตรฐานได้โดยใช้ฟังก์ชันไซน์ตามที่แสดงในสมการที่ (2.10) และ (2.11) จะได้

$$AF_n(\theta, \phi) = \left\{ \frac{1}{M} \frac{\sin\left(\frac{M}{2}\psi_x\right)}{\sin\left(\frac{\psi_x}{2}\right)} \right\} \left\{ \frac{1}{N} \frac{\sin\left(\frac{N}{2}\psi_y\right)}{\sin\left(\frac{\psi_y}{2}\right)} \right\} \quad (2.19)$$

เมื่อ

$$\psi_x = kd_x \sin \theta \cos \phi + \beta_x \quad (2.20)$$

$$\psi_y = kd_y \sin \theta \cos \phi + \beta_y \quad (2.21)$$

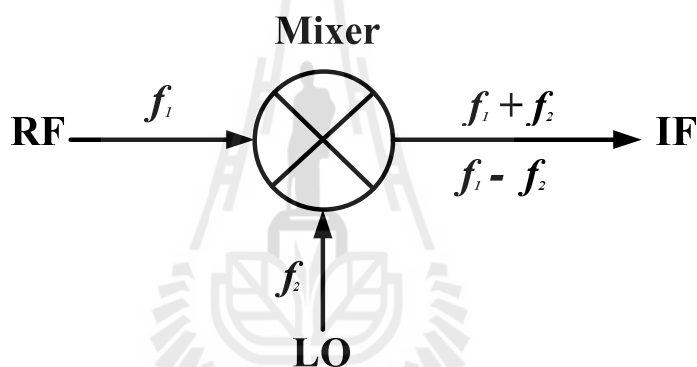
ดังนั้นเมื่อเราได้ค่าตัวประกอบแถวลำดับแล้วนั้นก็จะสามารถทำการหาค่าพลังงานของสายอากาศแถวลำดับแบบเชิงระนาบได้จากสมการที่ (2.12)

## 2.4 วงจรแปลงสัญญาณความถี่ต่ำ

โดยทั่วไปในภาครับของระบบสื่อสารเมื่อสัญญาณที่รับได้มีความถี่ที่สูงมาก ระบบจะไม่สามารถส่งสัญญาณเหล่านั้นผ่านสายนำสัญญาณไปได้ไม่ไกลมากนัก เพราะสัญญาณจะสูญเสียไปในสายจนหมด ดังนั้นภาครับจึงจำเป็นต้องมีวงจรแปลงสัญญาณความถี่ต่ำเพื่อแปลงความถี่ของสัญญาณที่รับได้ให้มีค่าที่ต่ำลงเพียงพอที่จะส่งไปตามสายนำสัญญาณโดยไม่ให้เกิดการสูญเสียภายในสายมากนัก ซึ่งความถี่ที่ถูกแปลงให้ต่ำลงนี้เรียกว่า ความถี่ IF (Intermediate Frequency)

จากนั้นก็ทำการส่งสัญญาณที่แปลงแล้ว ไปขยายให้มีระดับแรงดันสูงขึ้นเพื่อส่ง ไปใช้งานในขั้นถัดไป โดยอ้างอิงจากบทความของจารุพงศ์ จินาพันธ์. (2551)

ในการแปลงความถี่ให้ต่ำลงนั้นจะใช้หลักการเช่นเดียวกันกับเครื่องรับวิทยุคือ จะมีวงจรกำเนิดความถี่ภายใน (Local Oscillator: LO) โดยความถี่ที่ได้จาก LO นั้นจะถูกส่งไปยังวงจรผสมสัญญาณ (mixer) เพื่อทำการผสมสัญญาณกับสัญญาณความถี่สูงที่รับได้ จากรูปที่ 2.5 วงจรผสมสัญญาณจะทำงานโดยจะรับสัญญาณเข้ามา 2 สัญญาณ ได้แก่ สัญญาณ RF จากภาคขยาย RF และสัญญาณ LO จากวงจรกำเนิดความถี่ภายใน เพื่อทำการผสมสัญญาณจะได้สัญญาณผลบวกและผลต่างดังนี้



รูปที่ 2.5 วงจรผสมสัญญาณระหว่างสัญญาณ RF และ LO

นำสัญญาณทั้งสองมารวมกันจะได้ผลบวกดังนี้

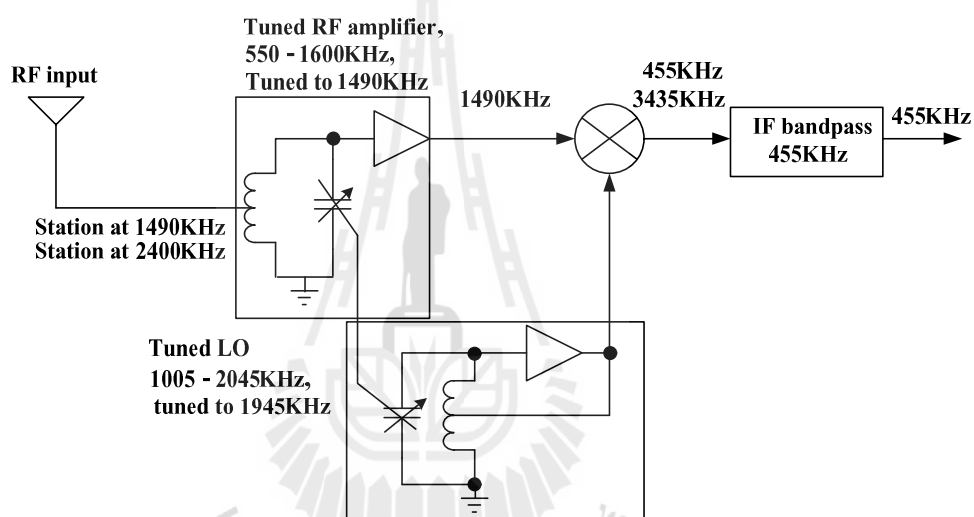
$$f_{LO} + f_{RF} = f_{IF} \quad \text{หรือ} \quad f_1 + f_2 = f_{IF} \quad (2.22)$$

และนำสัญญาณทั้งสองมาหักล้างกันจะได้ผลต่างดังนี้

$$f_{LO} - f_{RF} = f_{IF} \quad \text{หรือ} \quad f_1 - f_2 = f_{IF} \quad (2.23)$$

เมื่อดำเนินการเสร็จสมบูรณ์วงจรกรองความถี่จะทำการกรองเฉพาะความถี่ที่มีการลบกันหรือความถี่ผลต่างมาใช้งานเท่านั้น ความถี่ดังกล่าวนี้คือ ความถี่ IF จากนั้นก็จะทำการส่งไปยังภาคขยายต่อไป

จากรูปที่ 2.6 จะทำการยกตัวอย่างวงจรวิทยุ AM แบบ superheterodyne ที่มีการใช้วงจรผสมสัญญาณความถี่เช่นเดียวกัน และการคำนวณความถี่ขาออกของ superheterodyne radio จากสมการที่ (2.22) จะเป็นดังนี้



รูปที่ 2.6 วงจร superheterodyne AM radio

แทนค่าจะได้

$$1,945\text{KHz} + 1,490\text{KHz} = 3,435\text{KHz}$$

และจากสมการที่ (2.23) แทนค่าจะได้

$$1,945\text{KHz} - 1,490\text{KHz} = 455\text{KHz}$$



เมื่อได้ความถี่ที่มีการผสมสัญญาณแล้วก็จะมีการเลือกใช้ความถี่ที่เราต้องการ ซึ่งในที่นี้ก็คือความถี่ 455KHz เพราะเป็นความถี่ที่อยู่ในช่วงของความถี่วิทยุรับได้ และความถี่ที่ 3,435KHz ก็จะมีการกรองความถี่เพื่อกำจัดออกไป

## 2.5 วงจรกรองความถี่

วงจรกรองความถี่นั้นจะใช้สำหรับกำหนดให้ความถี่ผ่านไปหรือผ่านไปไม่ได้ อาจจะเป็นเฉพาะช่วงใดช่วงหนึ่งหรือช่วงกว้างๆ ก็ได้ ซึ่งจะประกอบไปด้วยตัวต้านทาน (resistor: R), ขดลวดเหนี่ยวนำ (inductors: L) และตัวเก็บประจุ (capacitors: C) โดยเอาคุณสมบัติประจำตัวของอุปกรณ์แต่ละชนิด คือ L จะยอมให้ความถี่ต่ำผ่านได้ง่าย ความถี่สูงผ่านยาก, C ความถี่ต่ำผ่านได้ยาก ความถี่สูงผ่านง่าย และ R จะมีความต้านทานทุกความถี่ให้มีระดับสัญญาณลดลง

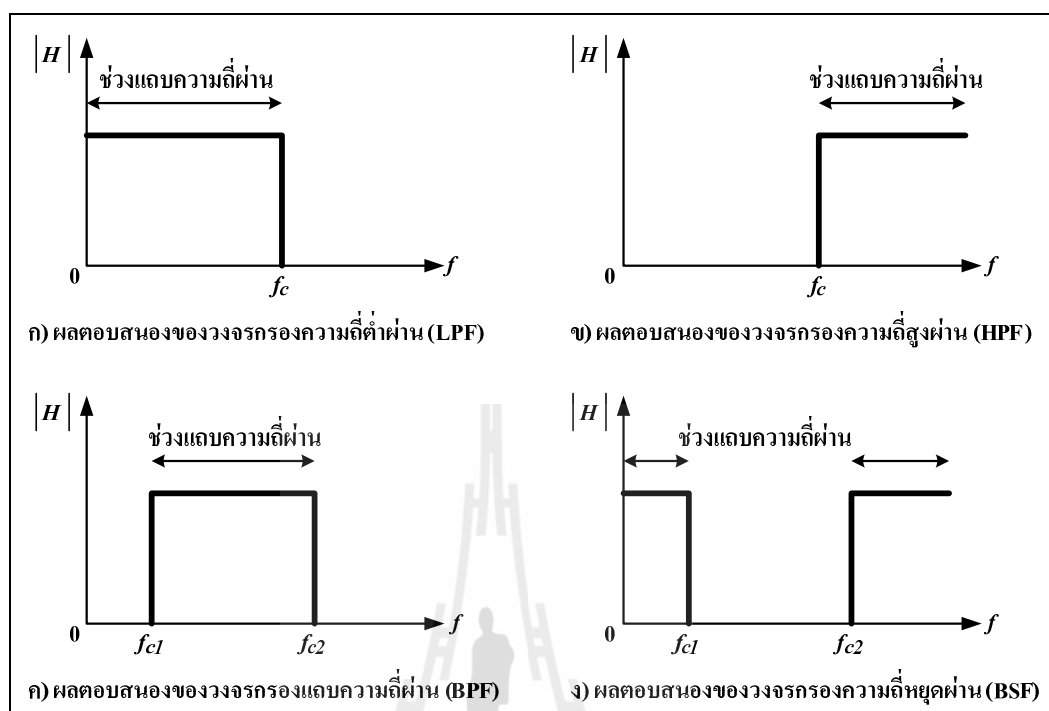
วงจรกรองความถี่มีด้วยกัน 2 แบบ คือ

1. แบบพาสซีฟ (passive) คือ วงจรที่ประกอบขึ้นด้วยอุปกรณ์แบบพาสซีฟ ซึ่งหมายถึงอุปกรณ์ที่สามารถทำงานได้โดยไม่ต้องมีการกระตุ้นด้วยไฟฟ้าเพื่อให้ทำงาน ซึ่งได้แก่อุปกรณ์ประเภท R, L และ C

2. แบบแอคทีฟ (active) คือ วงจรที่ประกอบขึ้นด้วยอุปกรณ์ที่ต้องการไฟฟ้าเพื่อกระตุ้นการทำงานของตัวอุปกรณ์ ซึ่งได้แก่อุปกรณ์ประเภททรานซิสเตอร์ หรือ ไอซี

วงจรกรองความถี่แบ่งได้ออกเป็น 4 ประเภทคือ

- ก. วงจรกรองสัญญาณความถี่ต่ำผ่าน (Low Pass Filter: LPF)
- ข. วงจรกรองสัญญาณความถี่สูงผ่าน (High Pass Filter: HPF)
- ค. วงจรกรองสัญญาณแถบความถี่ผ่าน (Band Pass Filter: BPF)
- ง. วงจรกรองสัญญาณความถี่หยุดผ่าน (Band Stop Filter: BSF)

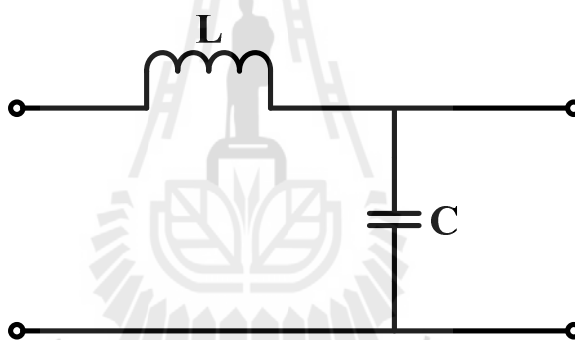


รูปที่ 2.7 ผลตอบสนองความถี่ของวงจรกรองความถี่ทางอุดมคติ

จากผลตอบสนองความถี่ของวงจรกรองความถี่ทางอุดมคติในรูปที่ 2.7 เมื่อให้  $|H|$  คือขนาดของแรงดันทางด้านเอาต์พุต วงจรกรองความถี่ต่ำผ่านจะยอมให้ความถี่ตั้งแต่ 0Hz ถึงความถี่  $f_c$  ผ่านไปยังขั้วเอาต์พุตของวงจรได้ ส่วนความถี่ที่สูงกว่าความถี่  $f_c$  จะไม่ผ่านไปยังขั้วเอาต์พุตของวงจร สำหรับวงจรกรองความถี่สูงผ่านจะยอมให้ความถี่สูงกว่าความถี่  $f_c$  ผ่านไปยังขั้วเอาต์พุตของวงจรได้ ส่วนความถี่ตั้งแต่ 0Hz ถึงความถี่  $f_c$  จะไม่ผ่านไปยังขั้วเอาต์พุตของวงจร สำหรับวงจรกรองแถบความถี่ผ่านจะยอมให้ความถี่ตั้งแต่  $f_{c1}$  ไปจนถึงความถี่  $f_{c2}$  ผ่านไปยังขั้วเอาต์พุตของวงจร ส่วนความถี่ตั้งแต่ 0Hz ถึงความถี่  $f_{c1}$  กับความถี่ที่สูงกว่า  $f_{c2}$  จะไม่ผ่านไปยังขั้วเอาต์พุตของวงจร และวงจรกรองความถี่หยุดผ่านจะไม่ยอมให้ช่วงความถี่  $f_{c1}$  ถึงความถี่  $f_{c2}$  ผ่านไปยังขั้วเอาต์พุตของวงจร ส่วนความถี่อื่นๆ วงจรก็จะยอมให้ผ่านไปยังขั้วเอาต์พุตได้ สำหรับงานวิจัยฉบับนี้จะขอกล่าวถึงเพียงวงจรกรองสัญญาณความถี่ต่ำผ่านเท่านั้น ซึ่งมีรายละเอียดดังนี้

### วงจรกรองสัญญาณความถี่ต่ำผ่าน

วงจรกรองสัญญาณความถี่ต่ำผ่านหรือบางครั้งอาจจะเรียกว่า วงจร high-cut filter สำหรับความถี่วิทยุ และ treble cut filter สำหรับวงจรขยายเสียง ซึ่งมีลักษณะการต่อคือ ใช้ขดลวดเหนี่ยวนำอนุกรมกับวงจร และตัวเก็บประจุขนานกับวงจรดังรูปที่ 2.8 คุณสมบัติของวงจรก็คือ เมื่อเราป้อนความถี่ต่ำเข้าวงจรขดลวดเหนี่ยวนำจะมีค่า  $X_L$  ต่ำและตัวเก็บประจุจะมีค่า  $X_C$  สูง ทำให้ความถี่ต่ำผ่านขดลวดเหนี่ยวนำได้สะดวกระดับสัญญาณเอาต์พุตจึงผ่านได้มาก แต่เมื่อความถี่สูงกว่าจุดที่กำหนดค่า  $X_L$  จะมากขึ้นและค่า  $X_C$  จะลดลงทำให้ความถี่ผ่านขดลวดได้ลดลง โดยบางส่วนของสัญญาณที่ได้ก็จะถูกตัวเก็บประจุดึงลงกราวด์ ดังนั้นระดับสัญญาณเอาต์พุตจึงผ่านได้น้อยมาก โดยอ้างอิงจากหนังสือของ Lutovac, and Miroslav, D. (2001); Schanmann, and Rolf. (1941)



รูปที่ 2.8 วงจรกรองสัญญาณความถี่ต่ำ

ซึ่งค่า  $X_C$  หมายถึง ค่า capacitive reactance คือค่าความต้านทานของตัวเก็บประจุที่มีต่อไฟกระแสสลับ โดยที่ความถี่สูงขึ้นความต้านทานนี้จะลดลง สามารถหาได้จากสูตร

$$X_C = \frac{1}{2\pi fC} \quad (2.24)$$

- $X_C$  คือ ค่า capacitive reactance มีหน่วยเป็น โอห์ม (ohms)
- $f$  คือ ค่าความถี่มีหน่วยเป็นเฮิรตซ์ (hertz)
- $C$  คือ ค่าความจุมีหน่วยเป็นฟารัด (farads)
- $\pi$  มีค่าเท่ากับ 3.1416

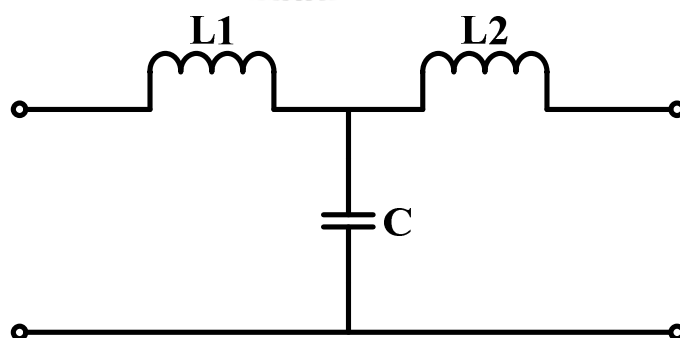
และค่า  $X_L$  หมายถึง ค่า inductive reactance คือความต้านทานของขดลวดเหนี่ยวนำที่มีต่อไฟ  
กระแสสลับ สามารถหาได้จากสูตร

$$X_L = 2\pi fL \quad (2.25)$$

- $X_L$  คือ ค่า inductive reactance มีหน่วยเป็น โอห์ม (ohms)
- $f$  คือ ค่าความถี่มีหน่วยเป็นเฮิรตซ์ (hertz)
- $L$  คือ ค่าความเหนี่ยวนำมีหน่วยเป็นเฮนรี (henrys)
- $\pi$  มีค่าเท่ากับ 3.1416

โดยที่วงจรกรองสัญญาณความถี่ต่ำผ่านยังสามารถแบ่งออกเป็น 2 แบบคือ

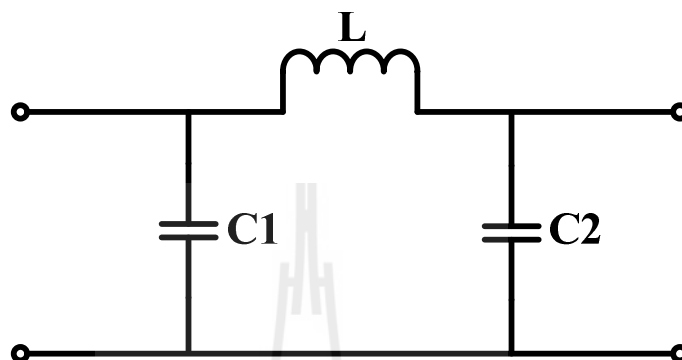
**1. T Type low pass filter** การใช้ขดลวดเหนี่ยวนำหรือตัวเก็บประจุเพียงตัวเดียว  
ไม่สามารถกำจัดสัญญาณความถี่สูงได้ทั้งหมด เมื่อที่จุด cut off ทำให้ความถี่สูงผ่านไป  
เราจึงมีวิธีแก้ปัญหาโดยการเพิ่มขดลวดเหนี่ยวนำเข้าไปในวงจรอีกหนึ่งชุด เมื่อต่อเข้าไป  
แล้วจะมีลักษณะวงจรคล้ายตัว T จึงเรียกว่า วงจรกรองสัญญาณความถี่ต่ำผ่านแบบที ซึ่ง  
แสดงได้ดังรูปที่ 2.9 และถ้าหากต้องการเพิ่มประสิทธิภาพให้กับวงจรกรองความถี่เราอาจจะ  
เพิ่มวงจรนี้เป็นหลายชุดก็ได้



รูปที่ 2.9 วงจรกรองสัญญาณความถี่ต่ำผ่านแบบที

**2. Pi Type low pass filter** วงจรนี้จะใช้ตัวเก็บประจุ 2 ตัวและขดลวดเหนี่ยวนำ 1  
ตัวต่อกันดังรูปที่ 2.10 ซึ่งมีรูปร่างคล้ายตัว Pi (พาย:  $\pi$ ) จึงเรียกว่า วงจรกรองสัญญาณ

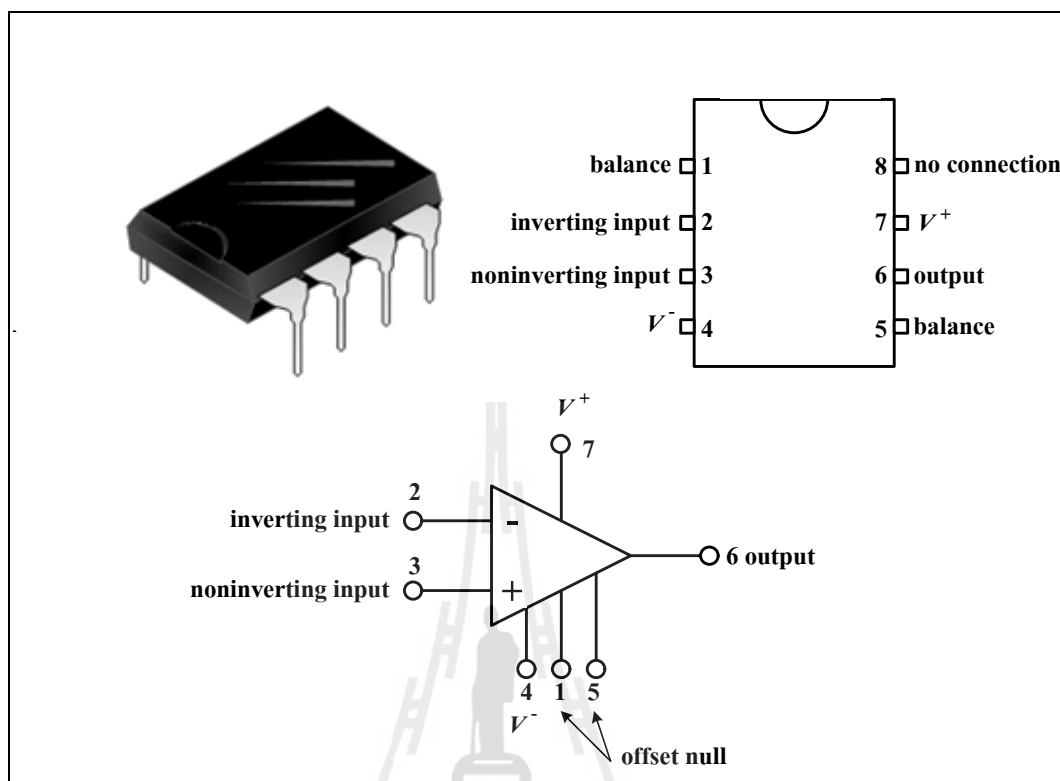
ความถี่ต่ำผ่านแบบพาส วงจรแบบนี้จะนิยมใช้ในภาคจ่ายไฟและวงจรเรกูเลเตอร์ (Regulator)



รูปที่ 2.10 วงจรกรองสัญญาณความถี่ต่ำผ่านแบบพาส

## 2.6 วงจรขยายสัญญาณ

ออปแอมป์ (op-amp) เป็นชื่อย่อสำหรับเรียกวงจรขยายที่มาจาก operating amplifier เป็นวงจรขยายแบบต่อตรงที่มีอัตราขยายสูงมาก โดยใช้การป้อนกลับแบบลบไปควบคุมลักษณะการทำงาน ทำให้ผลการทำงานของวงจรมีไม่ขึ้นกับพารามิเตอร์ภายในของออปแอมป์ ซึ่งวงจรภายในประกอบไปด้วยวงจรขยายที่ต่อเนื่องกันก็คือ วงจรขยายดิฟเฟอเรนเชียลด้านทางเข้า, วงจรขยายดิฟเฟอเรนเชียลภาคที่สอง, วงจรเลื่อนระดับ และวงจรขยายกำลังด้านทางออก สัญลักษณ์ที่ใช้แทนออปแอมป์จะเป็นรูปสามเหลี่ยม โดยจะมีขั้วอินพุตสองขั้วคือ ขั้วอินพุตบวก (non-inverting input) และขั้วอินพุตลบ (inverting input) ส่วนขั้วเอาต์พุตจะมีเพียงขาเดียวเท่านั้น

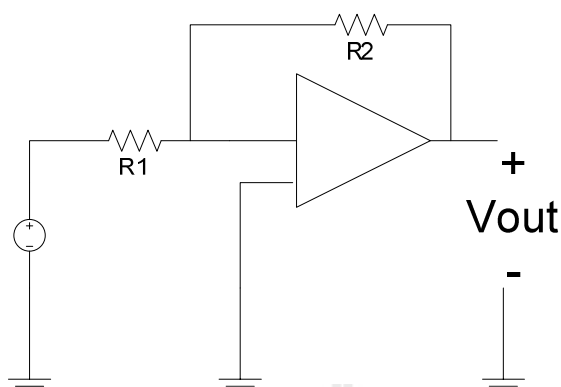


รูปที่ 2.11 สัญลักษณ์ของออปแอมป์

วงจรรขยายสัญญาณ โดยทั่วไปนิยมใช้วงจรรขยายหลายภาคที่เชื่อมต่อระหว่างภาคด้วย RC (RC coupled amplifiers) เนื่องจากสามารถออกแบบได้ง่ายและราคาถูก โดยอ้างอิงจากหนังสือของ Graeme, and Jerald, G. (1999) และปฏิบัติการ การวิเคราะห์วงจรไฟฟ้า 1 ซึ่งวงจรรขยายสัญญาณจะถูกแบ่งออกเป็น 5 ประเภทดังนี้

### 2.6.1 วงจรรขยายกลับขั้วสัญญาณ

วงจรรขยายกลับขั้วสัญญาณ (inverting amplifier) แสดงได้ดังรูปที่ 2.12



รูปที่ 2.12 วงจรขยายแบบกลับขั้วสัญญาณ

วงจรถ่ายกลับขั้วสัญญาณนี้จะมีแรงดันทางค่านเอาต์พุตเป็น

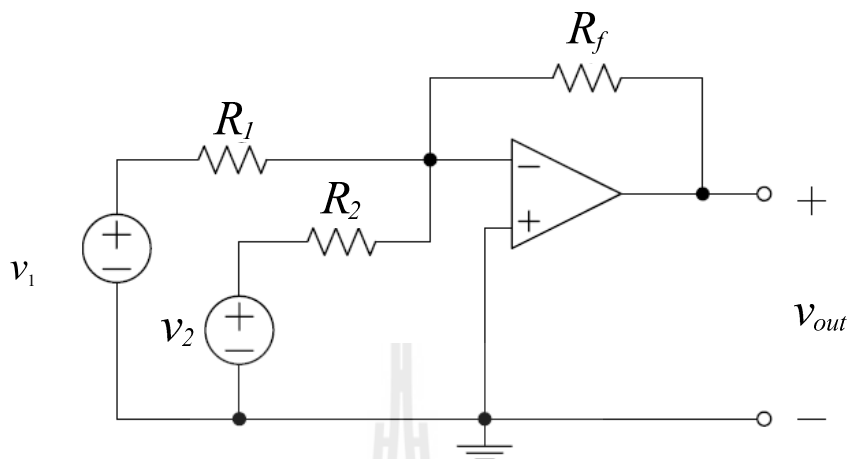
$$v_{out} = -\frac{R_f}{R_1} v_{in} \quad (2.26)$$

ดังนั้นวงจรมีอัตราขยายเป็น  $-R_f/R_1$  ซึ่งเราสามารถปรับอัตราขยายได้ตามค่าของ  $R_1$  และ  $R_f$  นั่นเอง ส่วนเครื่องหมายลบแสดงถึงการกลับขั้วของสัญญาณ ก็คือถ้าแรงดันอินพุตมีค่าเป็นบวกแรงดันเอาต์พุตจะมีค่าเป็นลบ ในทางกลับกันถ้าแรงดันอินพุตมีค่าเป็นลบแรงดันเอาต์พุตจะมีค่าเป็นบวก

### 2.6.2 วงจรขยายรวมสัญญาณ

วงจรถ่ายรวมสัญญาณ (summing amplifier) แสดงได้ดังรูปที่ 2.13 โดยมีแรงดันทางค่านเอาต์พุตเป็น

$$v_{out} = -\left(\frac{R_f}{R_1} v_1 + \frac{R_f}{R_2} v_2\right) \quad (2.27)$$

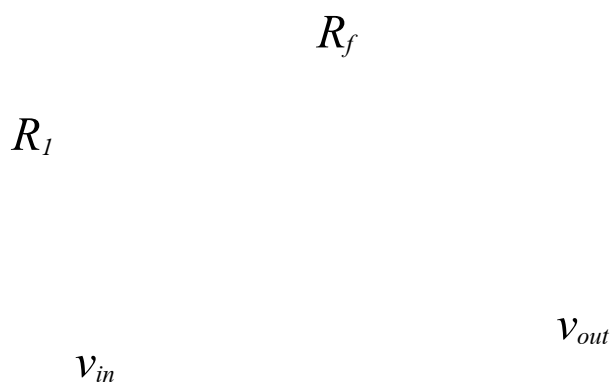


รูปที่ 2.13 วงจรขยายแบบรวมสัญญาณ

ดังนั้นวงจรนี้จะมีอัตราขยายสำหรับ  $v_1$  เป็น  $-R_f/R_1$  และมีอัตราขยายสำหรับ  $v_2$  เป็น  $-R_f/R_2$  ซึ่งเราสามารถปรับอัตราขยายได้ตามค่าของ  $R_1$ ,  $R_2$  และ  $R_f$  นั่นเอง โดยแรงดันทางด้าเอาต์พุตจะเป็นการบวกกันของแรงดันทั้งสอง

### 2.6.3 วงจรขยายไม่กลับขั้วสัญญาณ

วงจรขยายไม่กลับขั้วสัญญาณ (non-inverting amplifier) แสดงได้ดังรูปที่ 2.14



รูปที่ 2.14 วงจรขยายแบบไม่กลับขั้วสัญญาณ



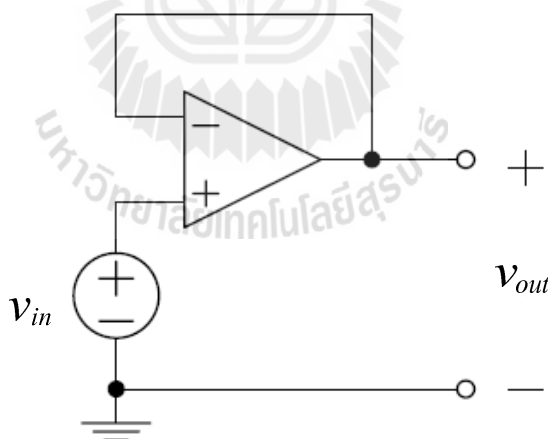
วงจรรขยายไม่กลับขั้วสัญญาณนี้จะมีแรงดันทางด้านเอาต์พุตเป็น

$$v_{out} = \left(1 + \frac{R_f}{R_1}\right)v_{in} \quad (2.28)$$

ดังนั้นวงจรมีอัตราขยายเป็น  $1 + R_f/R_1$  ซึ่งเราสามารถปรับอัตราขยายได้ตามค่าของ  $R_1$  และ  $R_f$  นั้นเอง โดยอัตราขยายที่เป็นบวกแสดงถึงการไม่กลับขั้วของสัญญาณ ก็คือถ้าแรงดันอินพุตมีค่าเป็นบวกแรงดันเอาต์พุตจะมีค่าเป็นบวก ในทางกลับกันถ้าแรงดันอินพุตมีค่าเป็นลบแรงดันเอาต์พุตก็จะมีค่าเป็นลบ

#### 2.6.4 วงจรแรงดันไฟฟ้าตาม

เมื่อพิจารณาวงจรรขยายไม่กลับขั้วสัญญาณดังรูปที่ 2.14 เมื่อ  $R_f$  มีค่าเป็นศูนย์ และ  $R_1$  มีค่าเป็นอนันต์ ( $\alpha$ ) จะได้วงจรดังรูปที่ 2.15



รูปที่ 2.15 วงจรแรงดันไฟฟ้าตาม

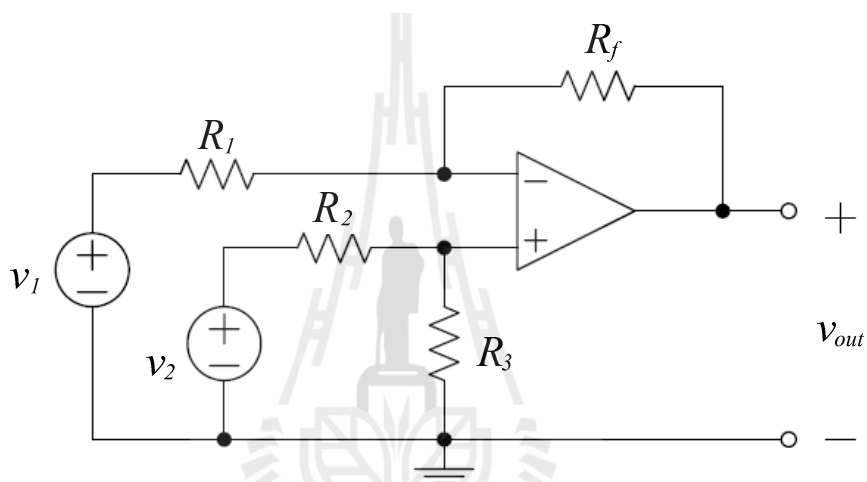
ซึ่งถ้าพิจารณาสมการที่ 2.28 โดยกำหนดค่าตัวต้านทาน  $R_f = 0$  และ  $R_1 = \alpha$  แล้ววงจรมีแรงดันทางด้านเอาต์พุตเป็น

$$v_{out} = v_{in} \quad (2.29)$$

ดังนั้นวงจรนี้จะมีอัตราขยายเป็น 1 หรือก็คือไม่มีการขยายสัญญาณนั่นเอง (แรงดันทางด้านเอาต์พุตมีค่าเท่ากับแรงดันทางด้านอินพุต) เราจึงเรียกวงจรดังรูป 2.15 นี้ว่า วงจรแรงดันไฟฟ้าตาม (voltage follower)

### 2.6.5 วงจรขยายผลต่าง

วงจรรขยายผลต่าง (difference amplifier) แสดงได้ดังรูปที่ 2.16



รูปที่ 2.16 วงจรขยายผลต่าง

วงจรรขยายผลต่างนี้จะมีแรงดันทางด้านเอาต์พุตเป็น

$$v_{out} = \left( \frac{R_f}{R_1} \right) \left( \frac{1 + R_1 / R_f}{1 + R_2 / R_3} \right) (v_2 - v_1) \quad (2.30)$$

ถ้ากำหนดให้อัตราส่วนของ  $R_1 / R_f$  และ  $R_2 / R_3$  มีค่าเท่ากันคือ

$$\frac{R_1}{R_f} = \frac{R_2}{R_3} \quad (2.31)$$

จะได้สมการ (2.30) เป็น

$$v_{out} = \frac{R_f}{R_1} (v_2 - v_1) \quad (2.32)$$

ดังนั้นวงจรนี้จะมีอัตราขยายเป็น  $R_f/R_1$  โดยทำการขยายผลต่างระหว่างสัญญาณแรงดันอินพุตทั้งสอง ( $v_2 - v_1$ )

## 2.7 การแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล

ในกรณีระบบเครือข่ายเป็นแบบดิจิทัลคือ สามารถส่งผ่านสัญญาณดิจิทัลสู่ช่องทางการสื่อสารดิจิทัลได้โดยตรง เช่น ในระบบเครือข่าย ISDN หรือการส่งไปรษณีย์อิเล็กทรอนิกส์ (Electronic mail: E-mail) เป็นต้น เราสามารถที่จะส่งสัญญาณดิจิทัลออกจากคอมพิวเตอร์สู่เครือข่ายได้โดยตรงแบบไม่ต้องผ่านโมเด็ม และในขณะเดียวกันเราก็สามารถที่จะส่งสัญญาณอนาล็อกผ่านเข้าไปในระบบเครือข่ายดิจิทัลได้โดยการเปลี่ยนแปลงสัญญาณอนาล็อกให้เป็นสัญญาณดิจิทัลเสียก่อน โดยใช้อุปกรณ์ที่ทำงานตรงกันข้ามกับโมเด็ม นั่นก็คือ โคเดก (CODEC หรือ coder / decoder) โดยอ้างอิงจากหนังสือของธนวัฒน์ ศรีสอ้าน, และสุทธิชัย มณีรัตนรุ่งโรจน์. (2542)

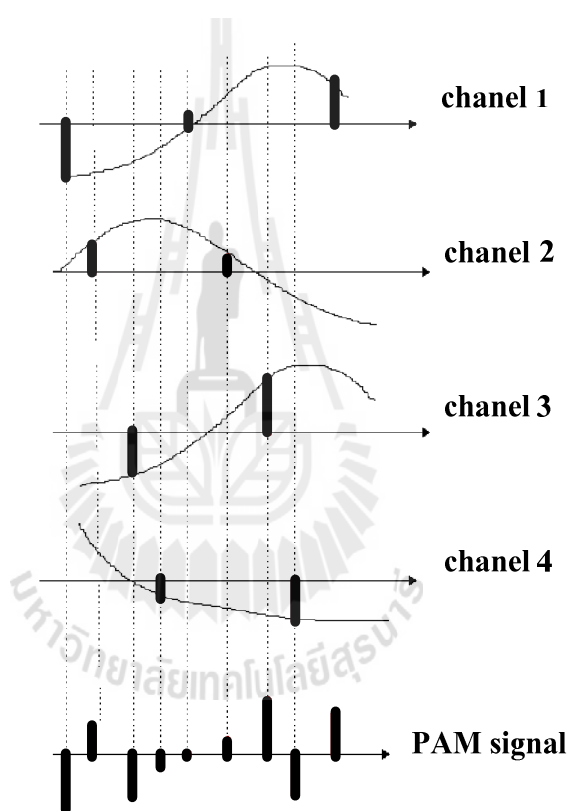
เทคนิคการแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล แบ่งออกเป็น 2 วิธี คือ

### 2.7.1. การมอดูเลตทางแอมพลิจูดของพัลส์

เป็นการใช้รูปคลื่นของสัญญาณอนาล็อกมาแปลงเป็นข้อมูลดิจิทัลโดยการวัดความสูงของลูกคลื่นจากหลายตำแหน่ง เรียกว่า วิธีสุ่มตัวอย่าง (sampling) โดยจะสุ่มตัวอย่างเป็นสองเท่าของความถี่สูงสุดของสัญญาณที่ใช้ เนื่องจากเสียงสนทนาที่มีความถี่สูงสุด 4,000Hz ดังนั้นค่าความถี่ของการสุ่มตัวอย่างจึงเท่ากับ 8,000 ครั้งต่อวินาที แล้วจึงสร้างสัญญาณดิจิทัลจากความสูงที่วัดได้ วิธีการนี้เป็นเทคนิคแบบแรกที่ใช้ในการแปลงเสียงสนทนา (voice) ให้เป็นข้อมูลดิจิทัล เรียกว่า “voice digitization” โดยการมอดูเลตทางแอมพลิจูดของพัลส์ (Pulse Amplitude Modulation: PAM) นั้นจะถูกนำไปใช้กับระบบการมอดูเลตแบบรหัสพัลส์ (Pulse Code Modulation: PCM) และถือเป็นกรรมวิธีขั้นแรกของระบบ PCM อีกด้วย ซึ่งสัญญาณโทรศัพท์ที่มีแอมพลิจูดแปรผันต่อเนื่องกันกับเวลาจะถูกสุ่มตัวอย่างทำให้ได้กระบวนของพัลส์ (pulse train) ซึ่งเรียกว่าตัวอย่าง (samples) และ

แอมพลิจูดของแต่ละตัวอย่างที่สุ่มมาได้นั้นจะเป็นสัดส่วน โดยตรงกับแอมพลิจูดชั่วขณะของสัญญาณที่ต่อเนื่อง ณ เวลาที่ได้มีการสุ่มตัวอย่างนั้น

รูปที่ 2.17 แสดง PAM signal ของสัญญาณ 4 ช่อง โดยแต่ละตัวอย่างที่ได้ทำการสุ่มมาของสัญญาณแต่ละช่องนั้น เรียกว่า PAM sample และเมื่อนำ PAM sample ทั้งหมดของทุกสัญญาณมารวมกันจะเรียกว่า PAM signal และช่วงเวลาของการสุ่มตัวอย่างแต่ละครั้งในสัญญาณเดียวกันเรียกว่า เฟรม (frame)

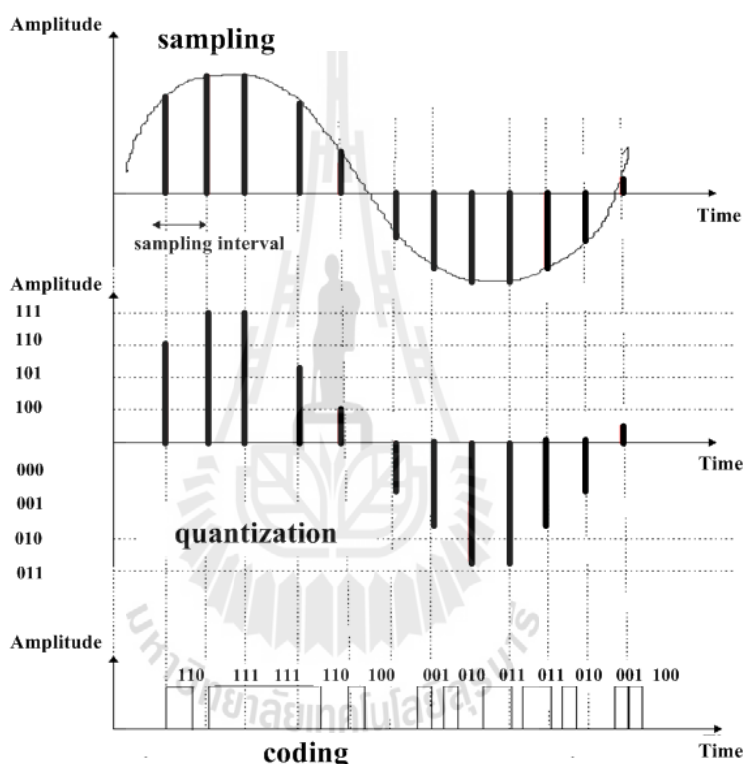


รูปที่ 2.17 PAM signal ของสัญญาณ 4 ช่อง

### 2.7.2. การมอดูเลตแบบรหัสพัลส์

หลักการขั้นต้นของระบบ PCM ก็คือ การแปลงสัญญาณอนาล็อกให้เป็นสัญญาณดิจิทัล สำหรับระบบ โทรศัพท์จะเป็นการแปลงสัญญาณ โทรศัพท์ไปเป็นกระบวนการพัลส์ ในรูปแบบของรหัสฐานสอง (binary code) แล้วส่งไปในตัวกลางที่ปลายทางฝั่งรับ โดยกระบวนการพัลส์ในรูปแบบของรหัสฐานสองดังกล่าวจะถูกแปลงกลับ ไปเป็น

สัญญาณอนาล็อกดั้งเดิม และการส่งนั้นจะเป็นการส่งสัญญาณแบบดิจิทัล (digital transmission) ซึ่งมีข้อได้เปรียบเหนือกว่าการส่งสัญญาณแบบอนาล็อก (analog transmission) ในเรื่องของการลดทอนสัญญาณจากสิ่งรบกวนและมีความเพี้ยนของสัญญาณต่ำ โดยอ้างอิงจากหนังสือของ Waggener, and Bill. (1995) ซึ่งจะแสดงได้ดังรูปที่ 2.18 และมีหลักการทำงาน 3 ขั้นตอนคือ

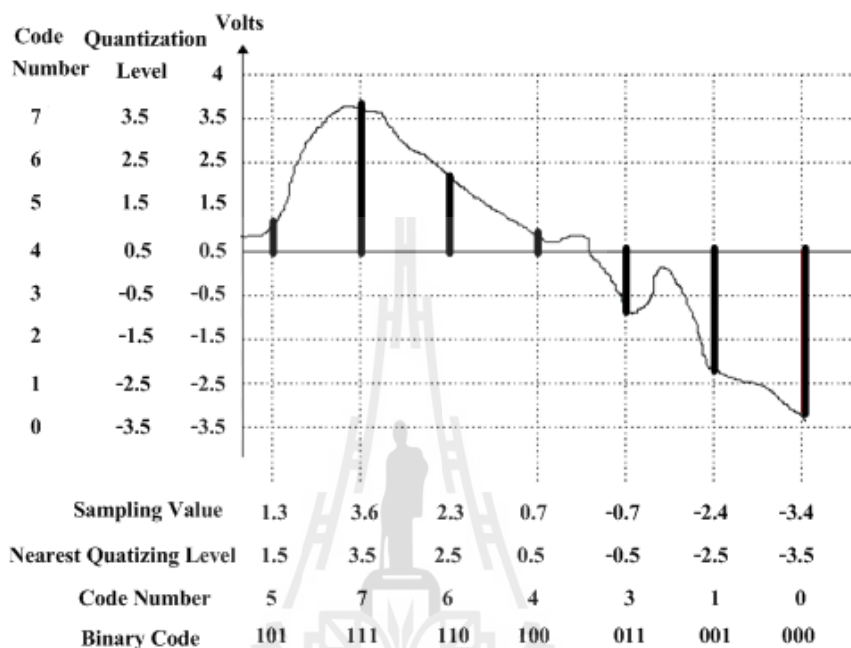


รูปที่ 2.18 การแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัลในระบบ PCM

### 2.7.2.1 การสุ่มตัวอย่าง

การสุ่มตัวอย่าง หมายถึง การเลือกเอาค่าแอมพลิจูดที่จุดใดๆ ของสัญญาณอนาล็อก ที่มีช่วงเวลาที่เท่ากัน ตัวอย่างที่สุ่มมาได้ก็คือ กระบวนของพัลส์ หรือเรียกว่า PAM sample และการสุ่มตัวอย่างต่อวินาที เรียกว่า sampling rate ซึ่งจาก sampling theorem ได้กล่าวไว้ว่า ถ้าทำการสุ่มตัวอย่างสัญญาณอนาล็อกด้วยช่วงเวลาที่สม่ำเสมอในอัตราส่วนอย่างน้อย 2 เท่าของความถี่สูงสุดของสัญญาณนั้นๆ แล้วตัวอย่างที่สุ่มมาได้จะบรรจุข่าวสารของสัญญาณเดิมได้ครบถ้วน และในระบบ PCM สัญญาณโทรศัพท์จะถูกสุ่มตัวอย่างด้วย

sampling rate 8000 ครั้งต่อวินาทีหรือถูกสุ่มตัวอย่างทุกๆ 125 ไมโครวินาที ซึ่งเรียกว่า sampling interval ดังแสดงตามรูปที่ 2.19



รูปที่ 2.19 การแบ่งย่านแอมพลิจูดออกเป็นระดับต่างๆ (quantizing)

### 2.7.2.2 การแบ่งย่านแอมพลิจูดออกเป็นระดับต่าง ๆ

การสุ่มตัวอย่างสัญญาณ โทรศัพท์ในอัตราที่สม่ำเสมอจะทำให้ได้ PAM signal ที่แอมพลิจูดของมันเป็นส่วนหนึ่งของระดับของสัญญาณ ณ เวลาที่ได้ทำการสุ่มนั้น ซึ่งแอมพลิจูดดังกล่าวอาจมีค่าได้มากมายไม่จำกัด การกำหนดรหัสฐานสองให้กับการสุ่มตัวอย่างที่มีค่าแอมพลิจูดมากให้ได้ครบทุกตัวนั้นก็จำเป็นที่จะต้องใช้จำนวนบิตในกลุ่มรหัส (code word) ตามไปด้วยทำให้ไม่เหมาะสมในทางปฏิบัติ อย่างไรก็ตามการแบ่งย่านแอมพลิจูดออกเป็นระดับต่างๆ ด้วยจำนวนที่จำกัด เราก็สามารถที่จะแทนแอมพลิจูดค่าต่างๆ ของสัญญาณที่สุ่มมาด้วยจำนวนจำกัดของระดับที่ได้ทำการแบ่งไว้ ซึ่งอาจจะมีผลคลาดเคลื่อนไปได้บ้าง และการให้รหัสฐานสองก็ควรกำหนดเอาค่าที่ตรงกันหรือใกล้เคียงที่สุดกับระดับที่ได้ทำการแบ่งไว้ วิธีในการแบ่งย่านแอมพลิจูดของตัวอย่างที่สุ่มมาได้ ออกเป็นระดับต่างๆ ที่มีจำนวนจำกัดและกำหนดระดับที่แน่นอนให้กับ PAM signal นั้น

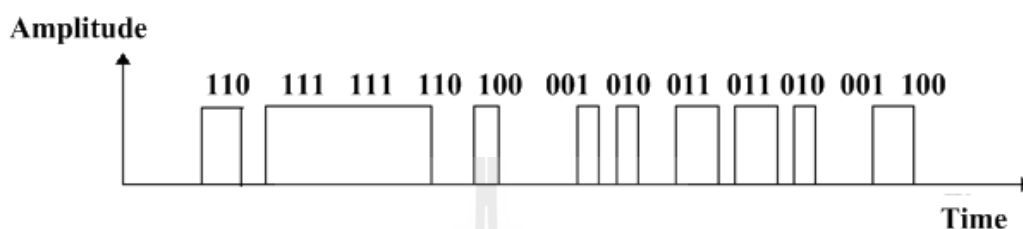
เรียกว่า quantizing โดยเรียกระดับหนึ่งๆ ที่แบ่งได้ว่า quantizing level และจะเรียกช่วงห่างระหว่าง quantizing level ว่า quantizing interval หรือ quantum step.

รูปที่ 2.19 แสดงสัญญาณที่จะทำการสุ่มตัวอย่างมีย่านแอมพลิจูดอยู่ระหว่าง +4 ถึง -4 โวลต์ โดยสมมติว่าเราแบ่งย่านแอมพลิจูดนี้ออกเป็น 8 ระดับคือ -3.5 -2.5 -1.5 -0.5 0.5 1.5 และ 2.5 โวลต์ (quantizing interval = 1 โวลต์) การสุ่มตัวอย่างครั้งแรกตรงกับแอมพลิจูด 1.3 โวลต์เรากำหนดให้อยู่ใน quantizing level ที่ 1.5 โวลต์ เพราะเป็นระดับที่ใกล้เคียงที่สุด และการสุ่มตัวอย่างครั้งที่สองตรงกับแอมพลิจูด 3.6 โวลต์ เรากำหนดให้อยู่ใน quantizing level ที่ 3.5 โวลต์ (ระดับที่ใกล้เคียงที่สุด) โดยการสุ่มตัวอย่างครั้งต่อไปนั้นก็จะเป็นเช่นเดียวกันจะเห็นได้ว่า quantizing level ที่เรากำหนดให้ นั้นเป็นเพียงค่าที่ใกล้เคียงกับค่าของแอมพลิจูดจริงที่ได้มาจากการสุ่มตัวอย่าง ซึ่งความคลาดเคลื่อนจากการ quantizing ย่อมจะต้องเกิดขึ้นได้บ้างเช่น จากการสุ่มตัวอย่างในครั้งแรก quantizing level ที่กำหนดให้จะคลาดเคลื่อนไป 0.2 โวลต์ เป็นต้น ความคลาดเคลื่อนนี้เกิดขึ้นในลักษณะที่ไม่แน่นอนและเราเรียกความคลาดเคลื่อนนี้ว่า quantizing error หรือ quantizing noise ซึ่งเป็นแหล่งกำเนิดที่สำคัญของความเพี้ยน ในขั้นตอนของ quantizing ถ้าทำการเพิ่มจำนวนของ quantizing level ให้มากขึ้น ก็จะทำให้ quantizing noise มีค่าน้อยลงแต่ในการเพิ่มจำนวน quantizing level ให้มากขึ้นนั้นจะทำให้รหัสฐานสองมีจำนวนบิตมากขึ้นตามไปด้วย และเป็นผลทำให้การส่งสัญญาณรหัสฐานสองต้องการแบนด์วิธ (bandwidth) กว้างมากขึ้น ดังนั้นโดยทั่วไปแล้วเราจำเป็นต้องกำหนดจำนวน quantizing level และจำนวนบิตในกลุ่มรหัสหนึ่ง โดยมีวัตถุประสงค์เพื่อให้การส่งสัญญาณ binary code ให้ได้ผลเป็นที่พึงพอใจและใช้ bandwidth ให้น้อยที่สุดเป็นหลัก

### 2.7.2.3 การเข้ารหัส

เมื่อได้ทำการสุ่มตัวอย่างสัญญาณอนาล็อกเรียบร้อยแล้ว เราก็จะได้ PAM signal ที่มีขนาดของแอมพลิจูดที่แตกต่างกันถูกส่งเข้าไปยัง quantizer โดยกำหนดให้ quantizing level ที่ตรงกันหรือใกล้เคียงกันที่สุดกับระดับของแอมพลิจูดที่สุ่มมาได้ ตัวเข้ารหัสก็จะผลิต binary code signal ตรงตาม quantizing level นั้นๆ แล้วจึงจะส่งออกไปในสายส่ง (transmission line) จากรูปที่ 2.19 สมมติว่าเราได้กำหนดค่าของรหัส (code number) ที่ใช้กับ quantizing Level ที่ระดับต่างๆ คือ -3.5 ถึง 3.5 โวลต์ เป็น 0 ถึง 7 ตามลำดับแล้ว ตัวอย่างที่สุ่มมาได้อันดับแรกคือ 1.3 โวลต์ quantizing level ที่ใกล้เคียงที่สุดของมันคือ 1.5 โวลต์และตรงกับค่าของรหัส 5 ดังนั้นรหัสที่ส่งออกไปเป็น code word ขนาด 3 บิต คือ 101

ตัวอย่างที่สุ่มมาได้อันที่สอง คือ 3.6 โวลต์ quantizing level ที่ใกล้เคียงที่สุดคือ 3.5 โวลต์ และตรงกับค่าของรหัส 7 ดังนั้นรหัสที่ส่งออกไปเป็น code word คือ 111 เช่นนี้เป็นต้น ดังแสดงตามรูปที่ 2.20



รูปที่ 2.20 การเข้ารหัส (coding)

## 2.8 บอร์ดไมโครคอนโทรลเลอร์

บอร์ดไมโครคอนโทรลเลอร์นั้นมีหน้าที่ในการแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล แล้วส่งให้ส่งสัญญาณอินพุตเข้าไปที่เครื่องคอมพิวเตอร์ โดยบอร์ดที่ใช้ในงานวิจัยนี้ก็คือ บอร์ดไมโครคอนโทรลเลอร์รุ่น ET-AVR STAMP ATmega128 เป็นบอร์ดไมโครคอนโทรลเลอร์ในตระกูล AVR ของบริษัท Atmel ซึ่งบอร์ดนี้เลือกใช้ MCU เบอร์ ATmega64 และ ATmega128 ขนาด 64 pins ในการออกแบบโครงสร้างของบอร์ดนั้นจะเน้นเรื่องการจัดวางบอร์ดให้มีขนาดเล็กเพื่อให้ง่ายต่อการนำไปประยุกต์ใช้งาน โดยได้นำ MCU มาจัดวางร่วมกับอุปกรณ์พื้นฐานที่จำเป็นและจัดขาออกมาใช้งานภายนอก ซึ่งการจัดเรียงขาสัญญาณจะทำการจัดเรียงอย่างเป็นระเบียบเพื่อให้สามารถต่อใช้งานได้โดยสะดวก โดยที่ตัวบอร์ดจะใช้ไฟเลี้ยง +5V

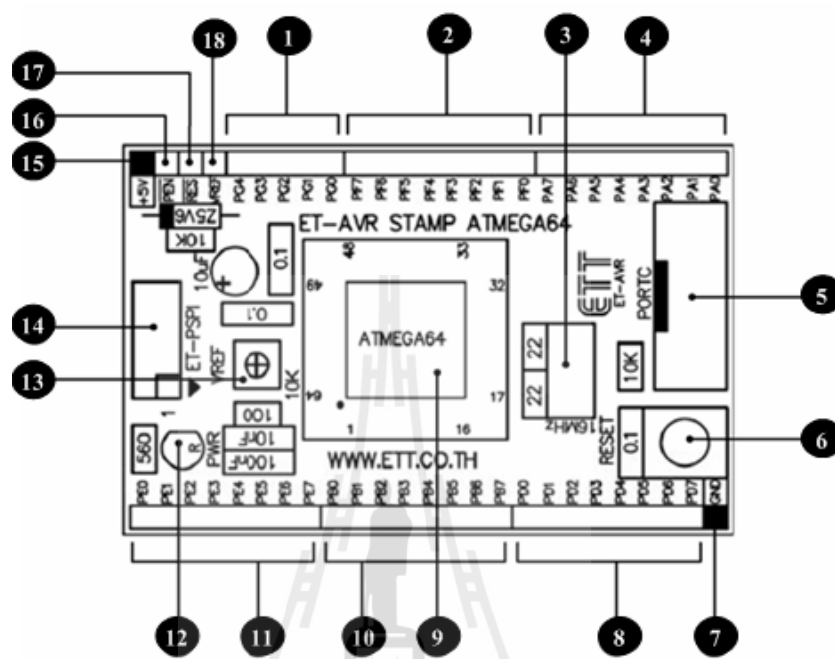
### 2.8.1 คุณสมบัติของบอร์ด ET-AVR STAMP ATmega64/128

1. ใช้ MCU ตระกูล AVR เบอร์ ATmega64 และ ATmega128 ของ Atmel ซึ่งเป็น MCU ขนาด 8 บิต
2. ความเร็วสัญญาณนาฬิกา crystal 16MHz
3. รองรับการโปรแกรมแบบ SPI และ JTAG (ต้องใช้ร่วมกับบอร์ด ET-AVRSTART KIT V1.0)
4. power supply ใช้แรงดันไฟฟ้า 4.5 - 5.5 V



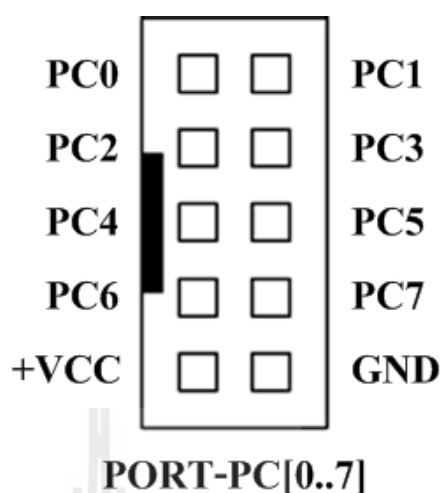
5. ภายใน MCU มีหน่วยความจำโปรแกรมแบบ flash ขนาด 64 KB (ATmega64) และ 128 KB (ATmega128), หน่วยความจำข้อมูล RAM ขนาด 4 KB, หน่วยความจำข้อมูลถาวรแบบ EEPROM ขนาด 2 KB (ATmega64) และ 4 KB (ATmega128) โดยสามารถลบและเขียนซ้ำได้กว่า 100,000 ครั้ง
6. จำนวน I/O สูงสุดถึง 53 I/O pins ซึ่งขาสัญญาณ I/O จะมีการใช้งานร่วมกันของฟังก์ชันอื่นๆ อีกดังนี้
  - SPI จำนวน 1 ช่อง, I2C จำนวน 1 ช่อง, 10-bits ADC จำนวน 8 ช่อง
  - programmable serial USARTs จำนวน 2 ช่อง
  - timers /counters 8 บิตจำนวน 2 ช่อง, timers /counters 16 บิตจำนวน 2 ช่อง, 8 บิต PWM 2 ช่อง, watchdog timer, real time counter
7. ทนอุณหภูมิใช้งานระหว่าง -40 ถึง +85°C
8. ขนาดของบอร์ด:
  - PCB size = 1653.543 mil x 2559 mil (42 x 65 mm)
  - ระยะขา = ความกว้าง 1500 mil ความยาว 2400 mil (~ 38.1 x 61 mm)
  - ระยะระหว่างขา = 2 x 25 pins I/O connector 100 mil (~ 2.54 mm)

## 2.8.2 โครงสร้างบอร์ด ET-AVR STAMP ATmega64/128



รูปที่ 2.21 โครงสร้างของบอร์ด ET-AVR STAMP ATmega64/128

- หมายเลข 1 คือ พอร์ต G ซึ่งประกอบไปด้วย PG0 – PG4 จำนวน 5 pins
- หมายเลข 2 คือ พอร์ต F ซึ่งประกอบไปด้วย PF0 – PF7 จำนวน 8 pins
- หมายเลข 3 คือ crystal 16 MHz
- หมายเลข 4 คือ พอร์ต A ซึ่งประกอบไปด้วย PA0 – PA7
- หมายเลข 5 คือ พอร์ต C ซึ่งประกอบไปด้วย PC0 – PC7 โดยพอร์ตนี้จะถูกเชื่อมต่อออกมารอไว้ยัง connector ขนาด 10 pins แบบ IDE โดยการจัดเรียงขาเป็นดังรูปที่ 2.22
- หมายเลข 6 คือ สวิตช์ reset
- หมายเลข 7 คือ จุดต่อกราวนด์ (GND)
- หมายเลข 8 คือ พอร์ต D ซึ่งประกอบไปด้วย PD0 – PD7

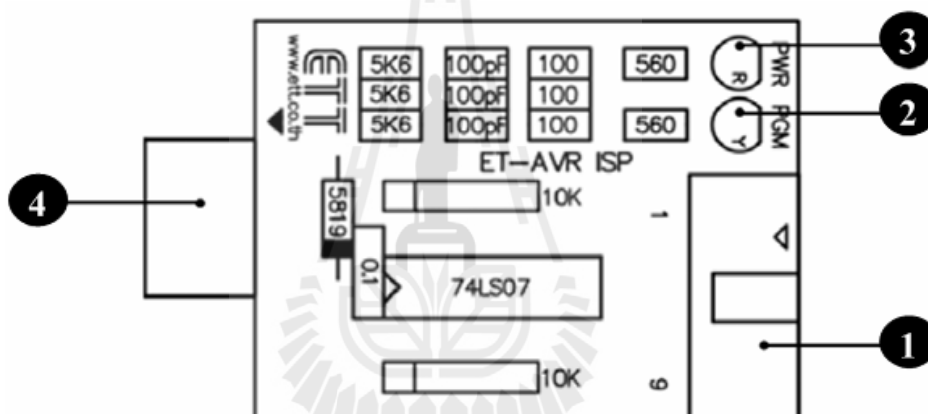


รูปที่ 2.22 การจัดเรียงขาของพอร์ต C ในหมายเลข 5

- หมายเลข 9 คือ MCU ATMEGA64/128 ของ Atmel
- หมายเลข 10 คือ พอร์ต B ซึ่งประกอบไปด้วย PB0 – PB7
- หมายเลข 11 คือ พอร์ต E ซึ่งประกอบไปด้วย PE0 – PE7
- หมายเลข 12 คือ LED PWR (สีแดง) แสดงสถานะของไฟเลี้ยงบอร์ด
- หมายเลข 13 คือ ตัวต้านทานปรับค่าได้ สำหรับปรับค่าแรงดันอ้างอิงของ ADC ที่ขา AREF
- หมายเลข 14 คือ พอร์ต ET-PSPI สำหรับเชื่อมต่อกับวงจร ISP programmer เพื่อโปรแกรม hex file ให้กับ AVR โดยการจัดเรียงขาแบบเดียวกันกับชุด STKxxx ของ Atmel
- หมายเลข 15 คือ จุดต่อ power supply +5V ของบอร์ด
- หมายเลข 16 คือ ขา PEN ใช้สำหรับเข้าโหมด SPI serial programming (ปกติไม่ได้ใช้งาน)
- หมายเลข 17 คือ จุดต่อสัญญาณ reset สำหรับใช้ reset จากอุปกรณ์ภายนอก
- หมายเลข 18 คือ จุดต่อขา AREF สำหรับใช้วัดค่าแรงดันอ้างอิงที่ขา AREF

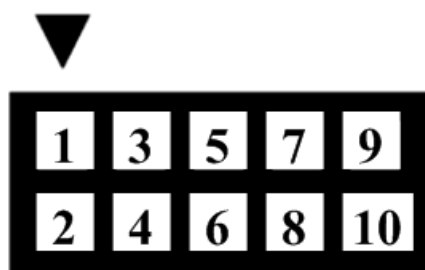
### 2.8.3 โครงสร้างบอร์ด ET-AVR ISP

ET-AVR ISP เป็นบอร์ดที่ออกแบบมาเพื่อใช้ในการดาวน์โหลด hex file ให้กับ MCU ตระกูล AVR ของ Atmel โดยใช้วิธีการแบบ serial programming การดาวน์โหลด hex file จะกระทำผ่านทางพอร์ตขนานของคอมพิวเตอร์ ซึ่งจะต้องใช้งานร่วมกับ ET-CAP10P ของอีทีที และ ซอร์ฟแวร์ที่ใช้ร่วมกับ ET-AVR ISP ก็คือ PonyProg2000 ซึ่งเป็นโปรแกรมดาวน์โหลดข้อมูลแบบ hex file ให้กับ CPU ตระกูล AVR โดยใช้วิธีการแบบ serial programming สามารถใช้งานกับบอร์ดตระกูล AVR ของอีทีทีได้เป็นอย่างดี วิธีการใช้งานโปรแกรมโดยทั่วไปนั้นสามารถศึกษาได้จาก help ของโปรแกรมได้เอง



รูปที่ 2.23 โครงสร้างของบอร์ด ET-AVR ISP

- หมายเลข 1 คือ พอร์ตสำหรับเชื่อมต่อกับ ET-CAP10P ของอีทีที เพื่อโปรแกรม hex file ให้กับ MCU
- หมายเลข 2 คือ LED PGM (สีเขียว) แสดงสถานะของการโปรแกรมหรือดาวน์โหลด hex file ลง MCU
- หมายเลข 3 คือ LED PWR (สีแดง) แสดงสถานะของไฟเลี้ยงบอร์ด
- หมายเลข 4 คือ พอร์ตสำหรับเชื่อมต่อกับบอร์ด target ซึ่งสามารถใช้โปรแกรม hex file ให้กับบอร์ด ET-AVR STAMP ATmega64/128 โดยเสียบบอร์ด ET-AVR ISP เข้าที่พอร์ต ETPSPI ซึ่งมีการจัดเรียงขาสัญญาณดังรูปที่ 2.24

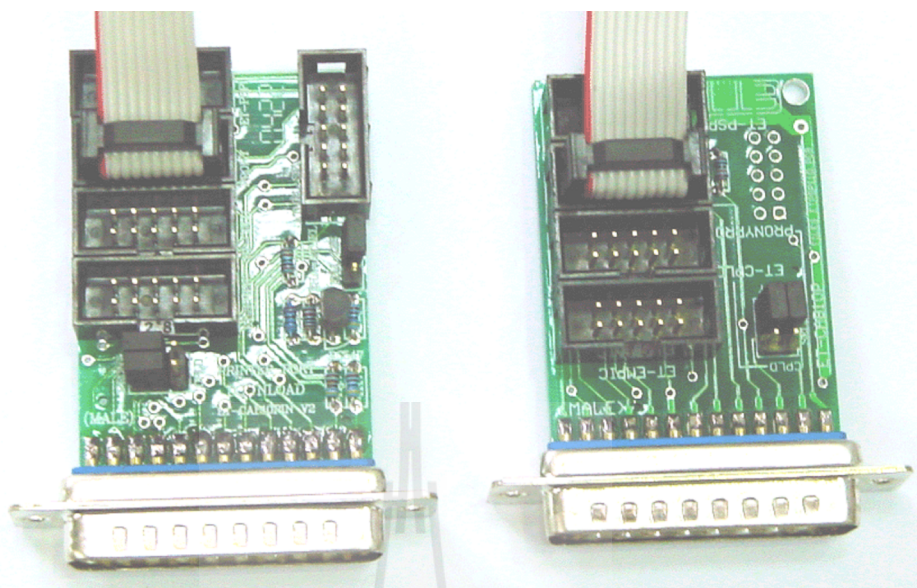


ตำแหน่งขา	ชื่อสัญญาณ
1	MOSI
2	VCC
3	ไม่ได้ใช้งาน
4,6,8,10	GND
5	RESET
7	SCK
9	MISO

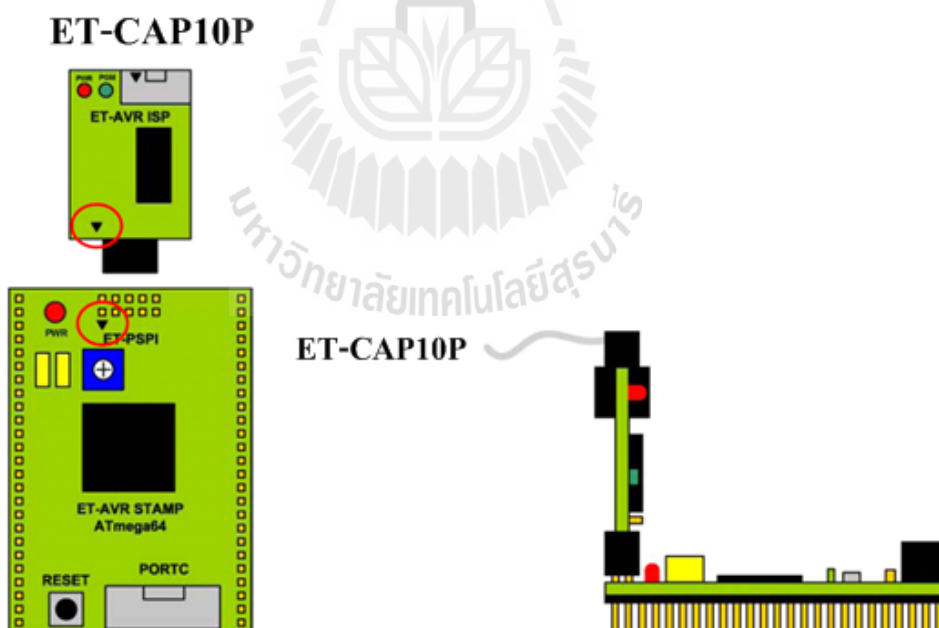
รูปที่ 2.24 การจัดเรียงขาของพอร์ต ETPSPI

#### 2.8.4 การเชื่อมต่ออุปกรณ์สำหรับโปรแกรม hex file

การโปรแกรม hex file ให้กับ AVR MCU ต้องใช้งานร่วมกับ ET-CAB10PIN และโปรแกรม PonyProg2000 โดยต่อ ET-CAP10PIN เข้ากับพอร์ต printer พร้อมทั้งเลือก jumper สำหรับใช้งานกับโปรแกรม PonyProg2000 แล้วต่อสาย download ที่ขั้วต่อ ETPSPI download ของบอร์ดพร้อมทั้งจ่ายไฟเข้าบอร์ดให้เรียบร้อย ถ้ามีการต่ออุปกรณ์ภายนอกที่พอร์ต PB ให้ปลดออกก่อนโดยการเชื่อมต่อจะมีลักษณะดังรูปที่ 2.26 โดยการเชื่อมต่อบอร์ดทั้งสองเข้าด้วยกันนั้นจะต้องสังเกตที่เครื่องหมายสามเหลี่ยมตรงกัน และในส่วน of ขึ้นตอนในการโปรแกรม hex file ให้กับบอร์ด ET-AVR STAMP ATmega64/128 นั้น จะได้กล่าวถึงในบทถัดไป



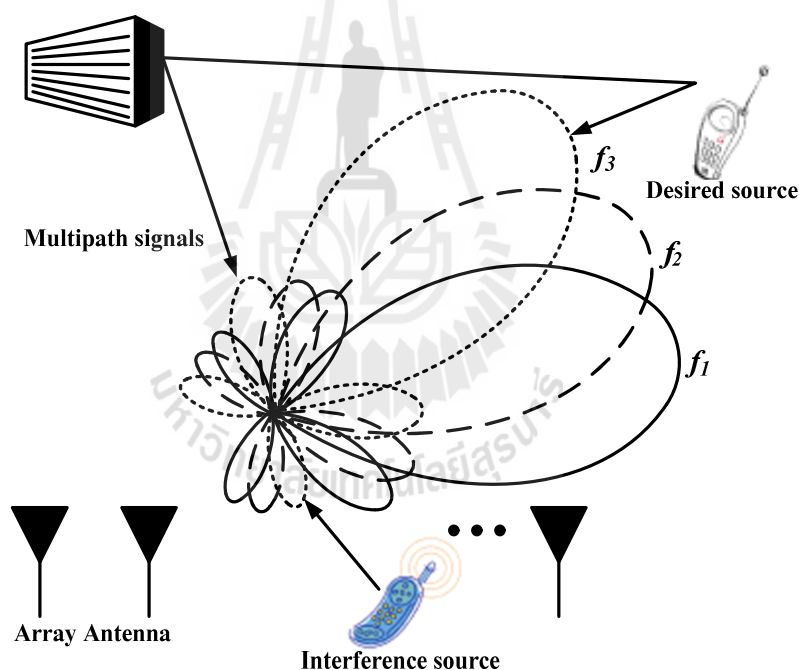
รูปที่ 2.25 บอร์ด ET-CAP10P โดย (ซ้าย) คือ ET-CAP10P V2.0 และ (ขวา) คือ ET-CAP10P V1.0



รูปที่ 2.26 การต่อ ET-AVR ISP เข้ากับ ET-AVR STAMP ATmega64/128

## 2.9 ระบบสายอากาศเก่ง

ระบบสายอากาศเก่ง (smart antenna system) ได้ถูกพัฒนาขึ้นครั้งแรกในช่วง ค.ศ.1960 โดยได้พัฒนามาจากเทคโนโลยีของสายอากาศแถวลำดับที่มีการปรับเฟส (phased array) ร่วมกับเทคโนโลยีการประมวลผลของสัญญาณ ซึ่งเริ่มแรกมีจุดประสงค์ที่จะนำมาใช้งานในระบบเรดาร์ แต่ในปัจจุบันระบบสายอากาศเก่งได้ถูกนำมาใช้ในเครือข่ายไร้สายอย่างแพร่หลาย เนื่องจากความสามารถในการหันลำคลื่นหลักไปในทิศทางที่ต้องการ ในขณะที่เดียวกันก็จะทำการหันจุดศูนย์หรือพุ่งไปในทิศทางที่เราไม่ต้องการหรือทิศทางของสัญญาณแทรกสอด กระบวนการดังกล่าวเรียกว่าการก่อรูปลำคลื่น (beamforming) ตามที่ได้แสดงในงานวิจัยของ Alexiou, A., and Haardt, M. (2004); Wintersand, J.H. (2004) ซึ่งมีข้อดีดังนี้



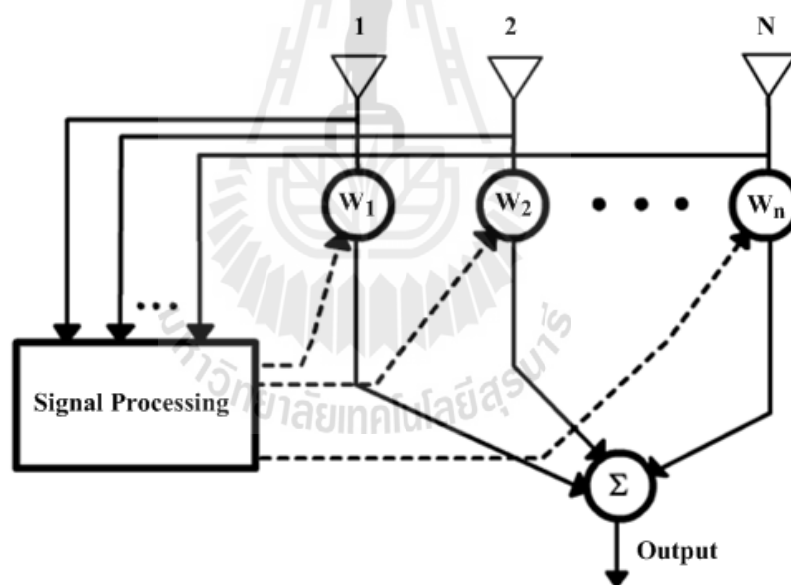
รูปที่ 2.27 ระบบสายอากาศเก่ง

- เพิ่มระยะพื้นที่ให้บริการ เนื่องจากอัตราขยายจากสายอากาศแถวลำดับหรือการก่อรูปลำคลื่นทำให้กำลังของสัญญาณ โดยเฉลี่ยเพิ่มขึ้น จึงส่งผลให้สามารถส่งสัญญาณได้ไกลขึ้น
- ประหยัดพลังงาน เนื่องจากระบบสายอากาศเก่งสามารถที่จะสร้างลำคลื่นไปยังผู้ใช้ได้โดยตรง จึงทำให้ไม่สูญเสียพลังงานโดยเปล่าประโยชน์

- ลดสัญญาณแทรกสอด (interference) และสัญญาณคลื่นหลายวิถี (multipath) เนื่องจากระบบสายอากาศเก่งสามารถหันจุดศูนย์หรือพุ่งไปยังทิศทางของสัญญาณแทรกสอดหรือสัญญาณคลื่นหลายวิถีได้

- มีอัตราส่วนระหว่างสัญญาณต่อสัญญาณรบกวนดีขึ้น
- ปรับปรุงเสถียรภาพของระบบให้ดีขึ้น

รูปที่ 2.28 แสดงส่วนประกอบของระบบสายอากาศเก่งซึ่งจะประกอบไปด้วยสองส่วนหลักๆ คือ สายอากาศแฉกลำดับ และระบบประมวลผลสัญญาณ ซึ่งในระบบประมวลผลสัญญาณนั้นจะทำหน้าที่ในการวิเคราะห์หาทิศทางการมาถึงของสัญญาณ (Direction-Of-Arrival: DOA) ซึ่งข้อมูลที่ได้มานั้นจะถูกนำมาใช้ในการคำนวณเพื่อทำการก่อรูปลำคลื่นต่อไป โดยระบบสายอากาศเก่งนั้นแบ่งออกเป็น 2 ประเภท ดังนี้



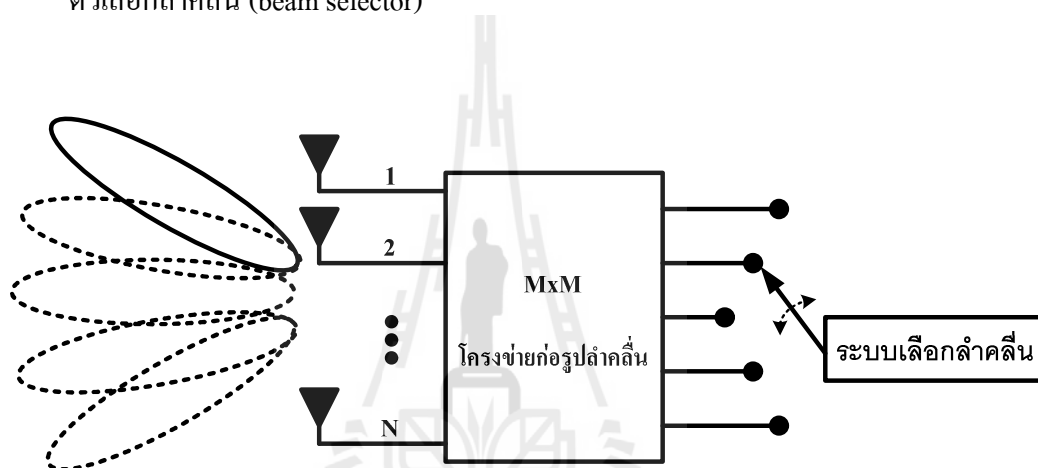
รูปที่ 2.28 ส่วนประกอบของระบบสายอากาศเก่ง

### 2.9.1 สายอากาศเก่งแบบสลับลำคลื่น

สายอากาศเก่งแบบสลับลำคลื่น (switched beam systems) นั้นประกอบไปด้วยสายอากาศชนิดระบุทิศทางแบบเจาะจงทิศทาง (directional antenna) ซึ่งมีลักษณะลำคลื่นแคบและชี้ไปในทิศทางที่ต้องการ เนื่องจากระบบสายอากาศเก่งแบบสลับลำคลื่นนั้นจะมี



ตัวถ่วงน้ำหนัก (weight) เป็นองค์ประกอบที่สำคัญ เพื่อที่จะนำหน้าที่เป็นตัวปรับเลือกลำคลื่นให้เข้าไปในทิศทางที่ต้องการ โดยตัวถ่วงน้ำหนักที่เราได้กล่าวถึงนั้น เรียกว่า โครงข่ายก่อรูปลำคลื่น (beamforming network) นั่นเอง ซึ่งสายอากาศแบบสลับลำคลื่นนั้นจะมีโครงข่ายก่อรูปลำคลื่นเป็นตัวปรับเลือกลำคลื่นเพื่อเข้าไปในทิศทางที่ต้องการแบบเจาะจงทิศทาง ดังนั้นจากรูปที่ 2.29 แสดงโครงสร้างพื้นฐานของสายอากาศแบบสลับลำคลื่น ซึ่งประกอบไปด้วยส่วนสำคัญๆ คือ สายอากาศแถวลำดับ โครงข่ายก่อรูปลำคลื่น และตัวเลือกลำคลื่น (beam selector)



รูปที่ 2.29 โครงสร้างพื้นฐานของระบบสายอากาศแบบสลับลำคลื่น

การทำงานของสายอากาศแบบสลับลำคลื่น ประกอบด้วย 4 ขั้นตอนดังนี้

1. ตรวจสอบทิศทางความแรงของสัญญาณ
2. ทำการเลือกลำคลื่นเพียงหนึ่งลำคลื่นในทิศทางที่มีความแรงของสัญญาณแรงที่สุด
3. ใช้ลำคลื่นในทิศทางที่เลือกไว้จากข้อ 2 เมื่อผู้ใช้ไม่มีการเคลื่อนที่
4. สับเปลี่ยนลำคลื่นเดิมไปยังลำคลื่นใหม่เมื่อผู้ใช้เคลื่อนที่ไปยังทิศทางอื่นๆ

นอกจากนี้การรวมสัญญาณขาออกของสายอากาศหลายๆ ตัว ทำให้สายอากาศแบบสลับลำคลื่นสามารถสร้างรูปแบบการแพร่กระจายคลื่นได้หลากหลายมากขึ้น ซึ่งทำให้ระบบมีทางเลือกของรูปแบบการแพร่กระจายแบบรูปการแผ่พลังงานที่มากกว่าการใช้

สายอากาศต้นเดี่ยว ดังนั้นสามารถสรุปข้อดีและข้อเสียของสายอากาศแบบปรับเลือกลำคลื่น ได้ดังนี้

### ข้อดี

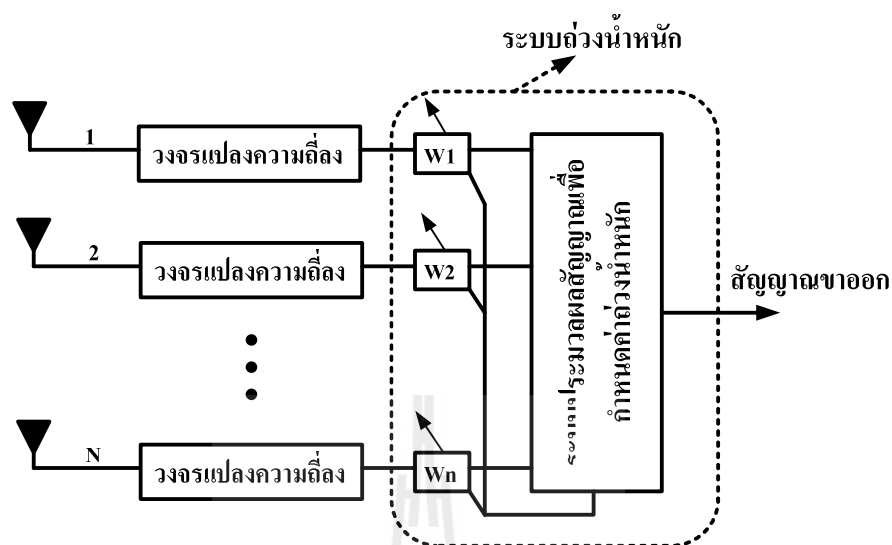
1. มีความซับซ้อนน้อย
2. สามารถติดตามสัญญาณได้รวดเร็วตามอัตราการปรับเปลี่ยนลำคลื่น
3. ระบบสายอากาศแ่งแบบสลับลำคลื่นจะมีประสิทธิภาพใกล้เคียงกับระบบสายอากาศแ่งแบบปรับลำคลื่น ในกรณีที่ระบบใช้จำนวนสายอากาศไม่มากนัก ตามที่ได้แสดงในงานวิจัยของ Peng, M., and Wang, W. (2005); Seungwon, C., Shim, D., and Sarkar, T. K. (1999)

### ข้อเสีย

1. อัตราการขยายสัญญาณต่ำในทิศทางที่อยู่ระหว่างลำคลื่น
2. การลดจำนวนสัญญาณแทรกมีข้อจำกัด
3. ในกรณีที่สัญญาณไม่ชัดเจน มีการบดบังสัญญาณ มีการแทรกสอดของสัญญาณ หรือมีสัญญาณมาถึงในมุมกว้างหลายๆ มุม อาจเกิดความผิดพลาดในการเลือกสัญญาณได้

## 2.9.2 สายอากาศแ่งแบบปรับลำคลื่น

สายอากาศแ่งแบบปรับลำคลื่น (adaptive array antenna) เป็นสายอากาศชนิดที่สามารถปรับเปลี่ยนลำคลื่นให้ชี้ไปในทิศทางใดๆ ได้อย่างอิสระ เนื่องจากจะมีตัวถ่วงน้ำหนักเพื่อทำการปรับลำคลื่น และมีส่วนที่เรียกว่า อัลกอริทึมแบบปรับตัว (adaptive algorithm) เป็นตัวคำนวณหาค่าสัมประสิทธิ์การถ่วงน้ำหนักของสัญญาณ โดยหลักการทำงานของระบบคือเมื่อสัญญาณตกกระทบสายอากาศแต่ละต้นสัญญาณจะถูกส่งมาที่ส่วนอัลกอริทึมแบบปรับตัวเพื่อคำนวณหาค่าสัมประสิทธิ์การถ่วงน้ำหนักแล้วส่งค่ากลับไปให้ตัวถ่วงน้ำหนักเพื่อคูณเข้ากับสัญญาณที่ตกกระทบสายอากาศ ซึ่งเป็นผลให้เกิดการปรับเปลี่ยนสัญญาณที่สายอากาศแต่ละต้นให้มีรูปแบบการแพร่กระจายของลำคลื่นในทิศทางที่ต้องการให้มีกำลังแรงขึ้น และในขณะเดียวกันก็จะลดทอนความแรงของสัญญาณในทิศทางที่ไม่ต้องการด้วย ตามที่ได้แสดงในงานวิจัยของ Agee, B. (1989); Frost, L. III (1972) ซึ่งจะมีโครงสร้างดังรูปที่ 2.30 และสามารถสรุปข้อดีและข้อเสียของสายอากาศแบบปรับตัวได้ดังนี้



รูปที่ 2.30 โครงสร้างพื้นฐานของสายอากาศเก่งแบบปรับตัวได้

#### ข้อดี

1. มีอัตราส่วนสัญญาณต่อสัญญาณรบกวน (Signal to Noise Ratio: SNR) สูง
2. ไม่ต้องมีการปรับเทียบสายอากาศ (calibration)
3. มีประสิทธิภาพดี แม้ในกรณีที่มีจำนวนสัญญาณรบกวนมากกว่าจำนวนสายอากาศ

#### ข้อเสีย

1. มีความซับซ้อนสูงมากกว่าสายอากาศเก่งแบบสลับลำคลื่น
2. ต้องการสัญญาณอ้างอิงที่ดีเพื่อให้ได้ประสิทธิภาพสูงสุด
3. ต้องการหน่วยประมวลผลความเร็วสูง

จากที่กล่าวมาข้างต้นนั้นได้อธิบายถึงโครงสร้างของระบบสายอากาศเก่ง ซึ่งประกอบด้วย 2 ส่วนหลักๆ คือ สายอากาศแถวลำดับ และระบบการประมวลผลสัญญาณ ซึ่งในส่วนของสายอากาศแถวลำดับนั้นจะทำหน้าที่รับและส่งสัญญาณที่แตกต่างกันไป และในส่วน of ระบบการประมวลผลสัญญาณนั้นจะทำหน้าที่ในการวิเคราะห์หาทิศทางการมาถึงของสัญญาณ (Direction-Of-Arrival: DOA) ข้อมูลที่ได้มานั้นจะถูกนำมาใช้ในการคำนวณเพื่อทำการก่อรูปลำคลื่นต่อไป โดย

ในส่วนของระบบประมวลผลสัญญาณนั้นถือว่าเป็นส่วนสำคัญของระบบสายอากาศเก่ง ซึ่งจะทำหน้าที่ในการวิเคราะห์หาทิศทางการมาถึงของสัญญาณที่ต้องการและสัญญาณที่ไม่ต้องการ ดังนั้นระบบยิ่งวิเคราะห์หาทิศทางการมาถึงของสัญญาณได้แม่นยำมากเท่าไร ก็จะยิ่งส่งผลต่อประสิทธิภาพของระบบมากขึ้นเท่านั้น ดังนั้นจึงได้มีการพัฒนาเทคนิคขึ้นมาหลายในการวิเคราะห์หาทิศทางการมาถึงของสัญญาณ ยกตัวอย่างเช่น Delay and Sum Method , Capon's Minimum Variance Method, Multiple Signal Classification (MUSIC) algorithm และ Estimation of Signal Parameters via Rotational Invariance Technique (ESPRIT) เป็นต้น ตามที่ได้แสดงในหนังสือของ Liberti, Jr. J. C., and Rappaport, T. S. (1999) แต่วิธีที่ได้รับความนิยมและถูกนำมาใช้ในการวิเคราะห์หาทิศทางการมาถึงของสัญญาณมากที่สุดได้แก่ วิธี MUSIC เนื่องจากเป็นวิธีที่ไม่ซับซ้อน และสามารถวิเคราะห์หาทิศทางการมาถึงของสัญญาณได้อย่างถูกต้องและแม่นยำ ดังนั้นในงานวิจัยฉบับนี้จึงจะนำเสนอการวิเคราะห์หาทิศทางการมาถึงของสัญญาณแบบวิธี MUSIC เพียงเทคนิคเดียว ซึ่งจะได้นำเสนอในหัวข้อถัดไป

## 2.10 วิธี Multiple Signal Classification (MUSIC) algorithm

วิธี MUSIC ได้ถูกนำเสนอโดย Schmidt ในปี 1979 เพื่อแก้ปัญหาเทคนิคพื้นฐานในการแบ่งประเภทของหลายๆ สัญญาณด้วยรูปแบบของไอเกน (eigen) โดยวิธี MUSIC นั้นจะเป็นขั้นตอนของการวิเคราะห์หาค่าพารามิเตอร์ต่างๆ ของสัญญาณ เช่น จำนวนของสัญญาณที่เกิดขึ้น, ทิศทางการมาถึงของแต่ละสัญญาณ, ความแรงของสัญญาณและความสัมพันธ์กันระหว่างสัญญาณที่เกิดขึ้น รวมไปถึงความแรงของสัญญาณรบกวนอีกด้วย ซึ่งวิธี MUSIC จะมีขั้นตอนและแนวทางการหาทิศทางอย่างละเอียด และให้ความถูกต้องแม่นยำที่สูงมาก ตามที่ได้แสดงในงานวิจัยของ Bo, W. (2006); Hong, Z. (2005) ซึ่งวิธี MUSIC นั้นจะมีขั้นตอนในการวิเคราะห์การมาถึงของสัญญาณแบ่งออกเป็น 5 ขั้นตอนด้วยกันดังนี้

1. ทำการรวมสัญญาณอินพุต  $u_k$ , โดยที่  $k = 0, \dots, T-1$  และวิเคราะห์หาค่าเมทริกซ์ความแปรปรวนร่วมของสัญญาณอินพุต (input covariance matrix) ได้ดังนี้

$$\hat{R}_{uu} = \frac{1}{T} \sum_{k=0}^{T-1} u_k u_k^H \quad (2.33)$$

โดยกำหนดให้  $T$  คือ จำนวนการสุ่มตัวอย่าง (snapshot) และ  $H$  คือ การสลับเปลี่ยนสังยุค (conjugate transpose)

2. หาค่าไอเกนของค่าเมทริกซ์ความแปรปรวนร่วมของสัญญาณอินพุตหรือค่า  $\hat{R}_{uu}$  ที่หาได้จากขั้นตอนที่ 1 ได้ดังนี้

$$\hat{R}_{uu} V = V \Lambda \quad (2.34)$$

จากสมการที่ (2.34) จะทำให้เราได้ค่า eigenvalues และค่า eigenvectors ออกมาก็คือ  $\Lambda = \text{diag}\{\lambda_0, \lambda_1, \dots, \lambda_{M-1}\}$ ,  $\lambda_0 \geq \lambda_1 \geq \dots \geq \lambda_{M-1}$  และ  $V = [q_0 q_1 \dots q_{M-1}]$  ตามลำดับ โดยค่า eigenvalues และค่า eigenvectors ที่ได้นั้นจะต้องทำการเรียงลำดับจากค่ามากไปหาน้อย ในที่นี้ก็คือ ค่ามากจะหมายถึงสัญญาณจริงที่ต้องการ และค่าน้อยจะหมายถึงสัญญาณรบกวนในระบบนั่นเอง

3. ทำการวิเคราะห์หาจำนวนของสัญญาณ ( $\hat{D}$ )

$$\hat{D} = M - K \quad (2.35)$$

โดยที่  $M$  คือ จำนวนของสายอากาศ และ  $K$  คือ จำนวนของสัญญาณที่มาถึง ซึ่งสามารถหาได้จากค่า eigenvalues และค่า eigenvectors ที่มีค่ามากจากขั้นตอนที่ 2

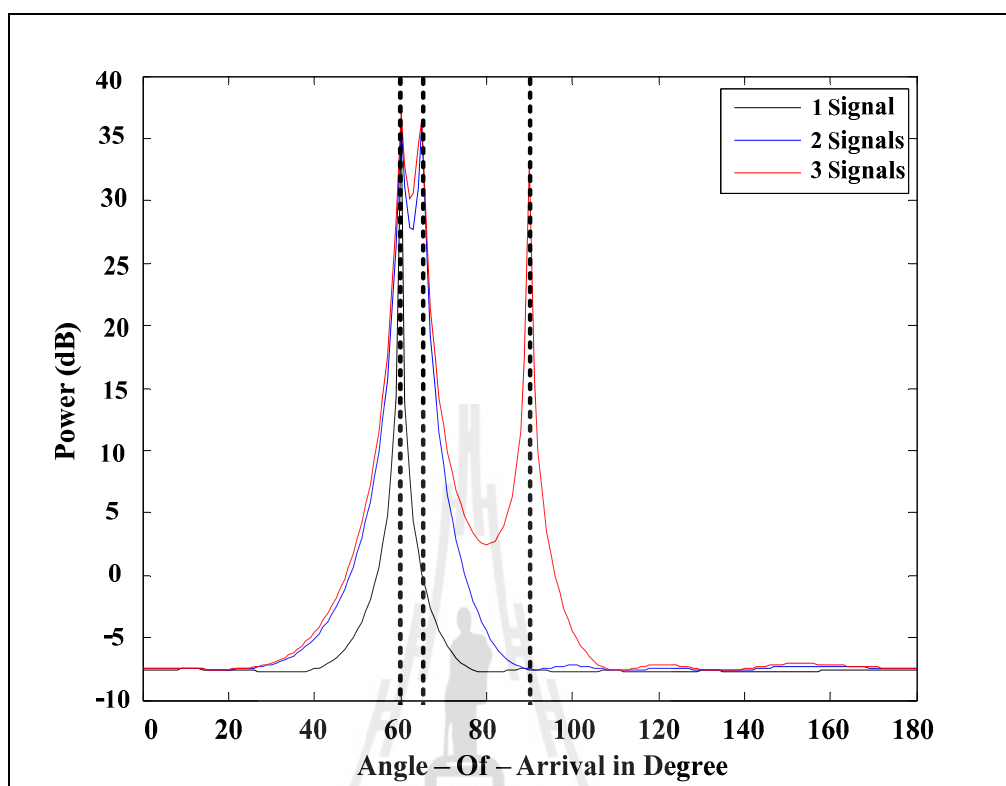
4. ทำการหาค่าสเปกตรัมแบบวิธี MUSIC ได้ดังสมการนี้

$$\hat{P}_{MUSIC}(\phi) = \frac{1}{a^H(\phi) V_n V_n^H a(\phi)} \quad (2.36)$$

โดยที่  $V_n$  คือ ค่า eigenvectors ซึ่งจากเดิมในขั้นตอนที่ 2 นั้นมีค่า  $V = [q_0 q_1 \dots q_{M-1}]$  แต่ในขั้นตอนนี้จะต้องทำการตัดค่า eigenvectors ที่มีค่ามากออกไปให้เหลือเท่ากับจำนวน  $\hat{D}$  ที่หาได้จากขั้นตอนที่ 3 ก็คือ  $V_n = [q_D q_{D+1} \dots q_{M-1}]$  และ  $a(\phi)$  คือ ค่า steering vector ซึ่งหาได้จาก  $a(\phi) = A e^{jkd(M-1)\cos(\phi)}$  โดยค่า steering vector จะทำการกวาดมุมค้นหาสัญญาณจริงในเมทริกซ์  $V_n$  ไปทุกๆ มุมตั้งแต่ 0-180 องศา ถ้ามุมจากการกวาดไปตรงกับมุมใดของสัญญาณจริงในเมทริกซ์  $V_n$  ค่าที่ได้นั้นก็จะเป็นค่าสเปกตรัมสูงสุด

5. ทำการหาค่ากำลังของสัญญาณที่มีค่าสเปกตรัมสูงสุดจากขั้นตอนที่ 4 แล้วนำมาพล็อตกราฟก็จะได้กราฟออกมาดังรูปที่ 2.31 ดังนั้นกราฟที่ได้จึงแสดงผลออกมาเป็นกำลังของสัญญาณที่มีหน่วยเป็น dB

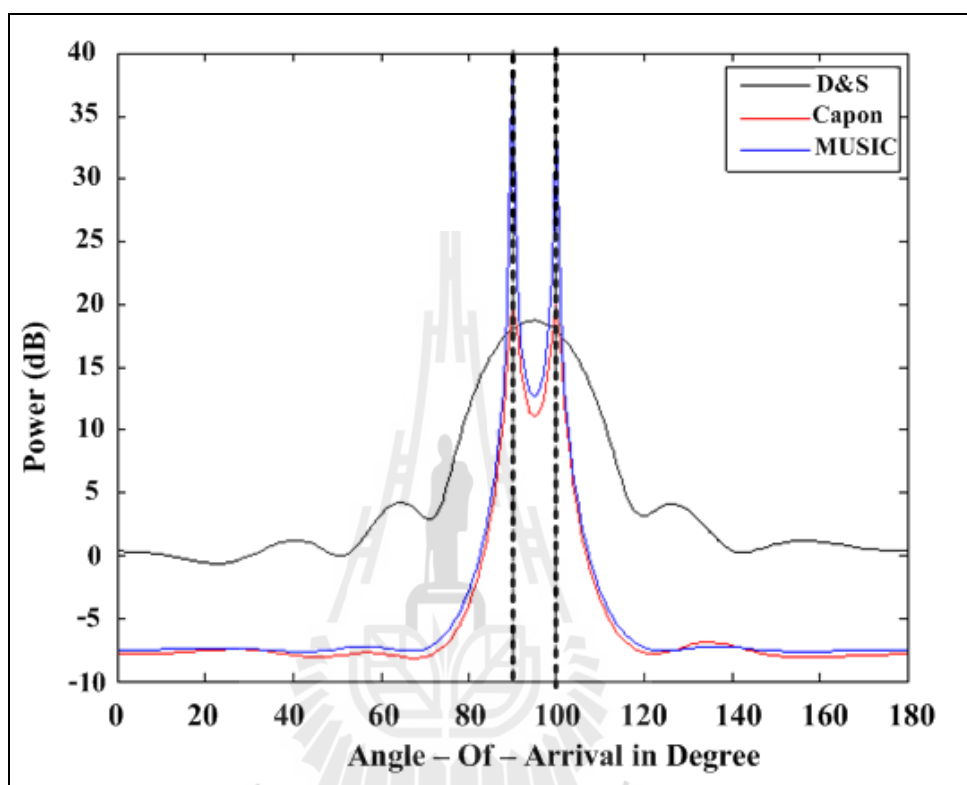
รูปที่ 2.31 แสดงการวิเคราะห์หาทิศทางการมาถึงของสัญญาณแบบวิธี MUSIC เมื่อกำหนดให้ระบบใช้สายอากาศแถวลำดับแบบเชิงเส้นจำนวน 6 ต้นและมีค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวนในระบบเท่ากับ 20dB โดยกราฟเส้นสีดำแสดงการวิเคราะห์หาทิศทางการมาถึงของสัญญาณเมื่อมี 1 สัญญาณเข้ามาในทิศทาง 60 องศา ในขณะที่เดียวกันเมื่อมี 2 สัญญาณเข้ามาในทิศทางที่ 60 และ 65 องศาจะแสดงการวิเคราะห์หาทิศทางการมาถึงของสัญญาณด้วยกราฟเส้นสีน้ำเงิน และสุดท้ายกราฟเส้นสีแดงจะแสดงการวิเคราะห์หาทิศทางการมาถึงของสัญญาณเมื่อมี 3 สัญญาณเข้ามาในทิศทางที่ 60 65 และ 90 องศา จะเห็นได้ว่าวิธี MUSIC นั้นสามารถวิเคราะห์หาทิศทางการมาถึงของสัญญาณได้อย่างถูกต้องและแม่นยำไม่ว่าจะมีสัญญาณเข้ามาในระบบเพียง 1 ทิศทางหรือในหลายๆ ทิศทางพร้อมกันก็ตาม



รูปที่ 2.31 การวิเคราะห์หาทิศทางการมาถึงของสัญญาณแบบวิธี MUSIC

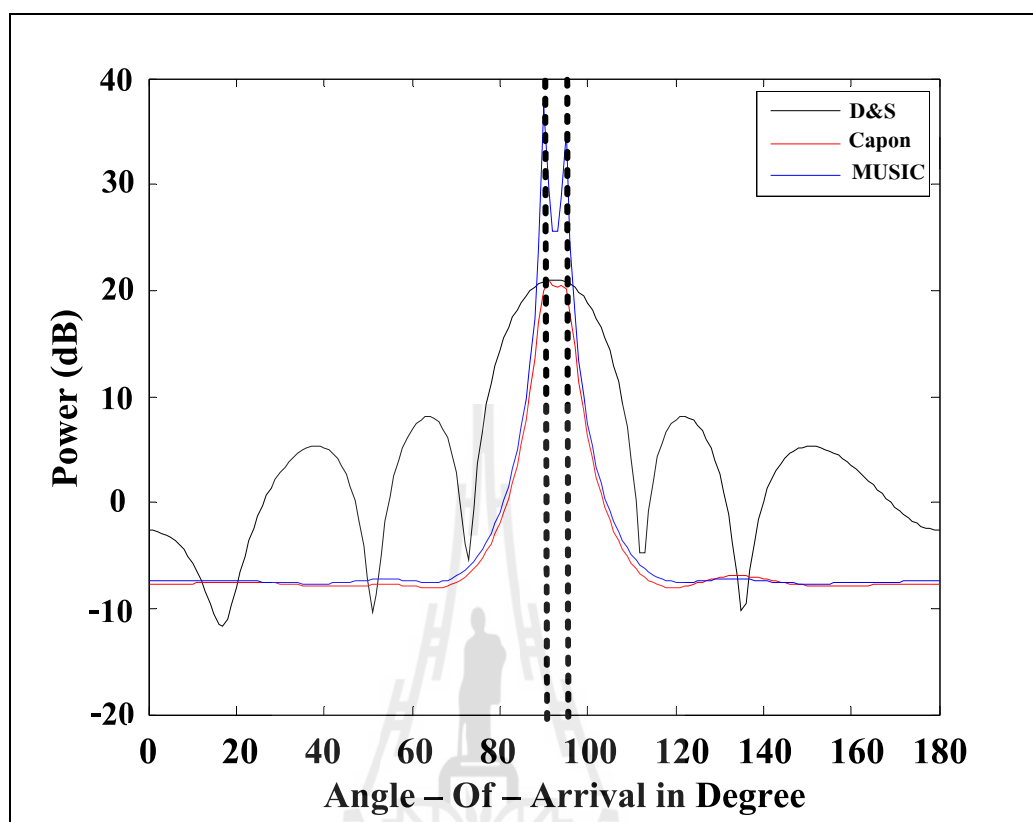
และรูปที่ 2.32 แสดงการวิเคราะห์หาทิศทางการมาถึงของสัญญาณแบบวิธี MUSIC เมื่อนำไปเปรียบเทียบกับวิธี Delay and Sum และวิธี Capon's ซึ่งแสดงแทนด้วยกราฟเส้นสีน้ำเงิน สีดำ และสีแดงตามลำดับ โดยกำหนดให้ระบบใช้สายอากาศแถวลำดับแบบเชิงเส้นจำนวน 6 ต้นและมีค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวนในระบบเท่ากับ 20 dB จากรูปที่ 2.32(ก) เมื่อกำหนดให้มีสัญญาณเข้ามาในทิศทาง 90 และ 100 องศา จะเห็นได้ว่า เมื่อระบบมีสัญญาณเข้ามามากกว่า 1 ทิศทาง วิธี Delay and Sum นั้นจะไม่สามารถวิเคราะห์หาทิศทางการมาถึงของสัญญาณได้ แต่ในขณะเดียวกันวิธี Capon's และวิธี MUSIC นั้นสามารถที่จะวิเคราะห์หาทิศทางได้อย่างถูกต้องและแม่นยำ และจากรูปที่ 2.32(ข) เมื่อกำหนดให้มีสัญญาณเข้ามาในทิศทาง 90 และ 95 องศา จะเห็นได้ว่า เมื่อระบบมีสัญญาณเข้ามาในทิศทางที่ใกล้เคียงกันมากๆ วิธี Capon's นั้นจะไม่สามารถวิเคราะห์หาทิศทางการมาถึงของสัญญาณได้ แต่ในขณะเดียวกันวิธี MUSIC นั้นยังคงสามารถวิเคราะห์หาทิศทางได้อย่างถูกต้องและแม่นยำอีกเช่นเคย ดังนั้นสรุปได้ว่าวิธี MUSIC นั้นสามารถที่จะวิเคราะห์หาทิศทางได้อย่างถูกต้องและแม่นยำ ไม่ว่าจะในสถานการณ์ที่มีสัญญาณเข้ามาในระบบมากกว่า 1

ทิศทางหรือเมื่อระบบมีสัญญาณเข้ามาในทิศทางที่ใกล้เคียงกันมากๆ ดังนั้นในงานวิจัยฉบับนี้จึงได้เลือกที่จะนำเสนอวิธี MUSIC มาใช้งานร่วมกับแนวคิดที่จะได้นำเสนอในส่วนถัดไป



(ก)





(ข)

รูปที่ 2.32 เปรียบเทียบการวิเคราะห์หาทิศทางการมาถึงของสัญญาณระหว่างวิธี Delay and Sum วิธี Capon's และวิธี MUSIC (ก) เมื่อมีสัญญาณเข้ามาที่ 90 และ 100 องศา (ข) เมื่อมีสัญญาณเข้ามาที่ 90 และ 95 องศา

## 2.11 เทคนิคการปรับจำนวนการสุ่มตัวอย่าง

การวิเคราะห์หาทิศทางการมาถึงของสัญญาณแบบวิธี MUSIC ได้รับความนิยมนเป็นอย่างมากและถูกนำไปพัฒนาอย่างแพร่หลายในการหาทิศทางของระบบสายอากาศเก่ง เนื่องจากเป็นวิธีที่ทำความเข้าใจและนำไปใช้งานได้ง่าย เพราะมีกระบวนการทำงานเป็นขั้นตอนและไม่ซับซ้อน อีกทั้งยังสามารถวิเคราะห์หาทิศทางได้อย่างถูกต้องและแม่นยำอีกด้วย แต่อย่างไรก็ตามวิธี MUSIC นั้นก็ยังคงมีปัญหาอยู่หากระบบมีค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวนที่ต่ำ ส่งผลให้ประสิทธิภาพการวิเคราะห์หาทิศทางการมาถึงของสัญญาณก็จะลดต่ำลงไปด้วย ดังนั้นในการแก้ปัญหาจึงจำเป็นต้องเพิ่มจำนวนการสุ่มตัวอย่างให้เหมาะสมกับค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวนที่มี

ในระบบ โดยในการเพิ่มจำนวนการสุ่มตัวอย่างนั้น ถ้าหากเราเพิ่มจำนวนการสุ่มตัวอย่างน้อยเกินไป ระบบก็จะไม่สามารถวิเคราะห์หาทิศทางการมาถึงของสัญญาณได้อย่างถูกต้องและแม่นยำ แต่ในขณะเดียวกันถ้าเราเพิ่มจำนวนการสุ่มตัวอย่างมากเกินไป หน่วยประมวลผลกลางหรือซีพียูของตัวประมวลผลก็จะทำงานหนักขึ้น เนื่องจากยิ่งระบบมีจำนวนการสุ่มตัวอย่างเพิ่มมากขึ้นเท่าไร เวลาที่ใช้ในการประมวลผลในระบบก็จะนานมากขึ้นเท่านั้น ซึ่งส่งผลให้ซีพียูของตัวประมวลผลทำงานหนักขึ้นตามไปด้วย

รูปที่ 2.33 แสดงตัวอย่างกราฟค่าความผิดพลาดเฉลี่ยของการวิเคราะห์หาทิศทาง (RMSE) เมื่อเรากำหนดให้มีสัญญาณเข้ามาที่สายอากาศแถวลำดับแบบเชิงเส้นขนาด  $4 \times 1$  ในทิศทาง 30 องศา โดยโปรแกรมจะทำการวิเคราะห์หาทิศทางที่มุม 30 องศาโดยการทำซ้ำทั้งหมด 100 ครั้งในแต่ละจำนวนการสุ่มตัวอย่าง (ซึ่งในที่นี้ก็คือ 10 30 50 100 และ 200) จากนั้นจึงนำค่าของทิศทางที่โปรแกรมหาได้ไปหาผลต่างจากทิศทางจริงของสัญญาณ ซึ่งค่าของผลต่างที่หาได้ทั้งหมด 100 ครั้งนั้นจะถูกนำไปหาค่าความผิดพลาดเฉลี่ยของการวิเคราะห์หาทิศทางก็จะได้กราฟออกมาดังรูปที่ 2.33 ที่แปรเปลี่ยนไปตามค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวนที่เพิ่มขึ้น (ซึ่งในที่นี้ก็คือ 0 ถึง 50 dB) จะเห็นได้ว่าเมื่อระบบมีค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวนที่น้อยลงเรื่อยๆ กราฟที่ได้ของแต่ละจำนวนการสุ่มตัวอย่างก็จะมีค่าความผิดพลาดเฉลี่ยของการวิเคราะห์หาทิศทางเพิ่มสูงขึ้นตามไปด้วย และสังเกตได้ว่ากราฟของจำนวนการสุ่มตัวอย่างที่มากที่สุด ในที่นี้ก็คือ 200 จะอยู่ต่ำสุด หรือมีค่าความผิดพลาดเฉลี่ยที่น้อยที่สุด ดังนั้นยิ่งระบบมีจำนวนการสุ่มตัวอย่างที่น้อยลงเท่าไร กราฟที่ได้ก็จะมีค่าความผิดพลาดเฉลี่ยของการวิเคราะห์หาทิศทางเพิ่มมากขึ้นตามไปด้วยอีก เช่นเดียวกัน ซึ่งก็เป็นไปตามที่เราได้ทำการวิเคราะห์ไว้ในข้างต้น

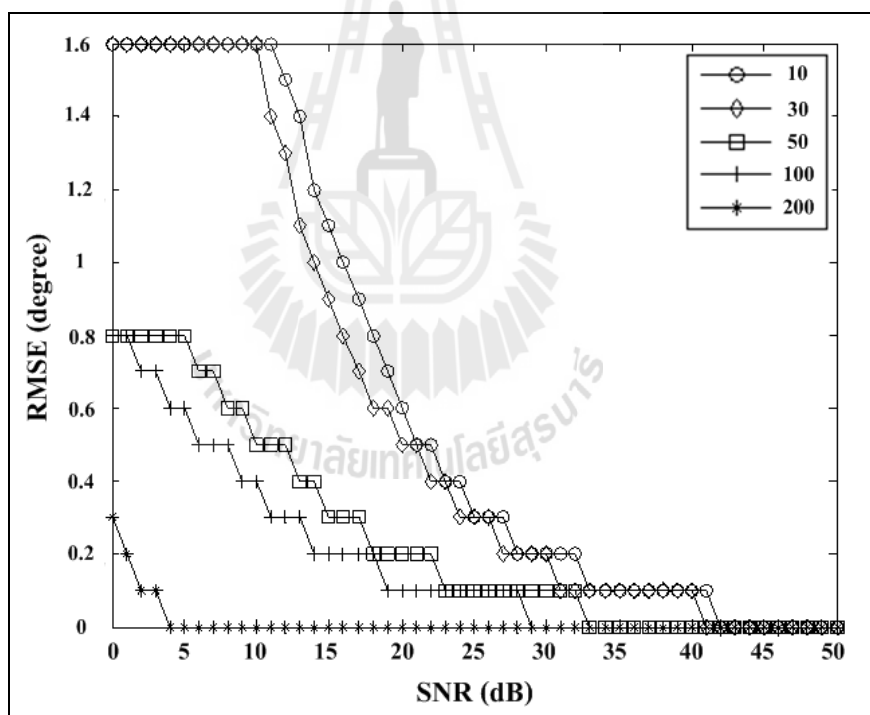
ค่าความผิดพลาดเฉลี่ยนั้นสามารถหาได้จากผลต่างของทิศทางที่โปรแกรมหาได้กับทิศทางจริงของสัญญาณ โดยทิศทางที่โปรแกรมหาได้และทิศทางจริงของสัญญาณนั้นมีหน่วยเป็นองศา (degree) ดังนั้นค่าความผิดพลาดเฉลี่ยที่หาได้นั้นจึงมีหน่วยเป็นองศาเช่นเดียวกัน ซึ่งสามารถหาได้จากสมการดังต่อไปนี้

$$(MSE)_{RMU} = \left[ \frac{\lambda}{2\pi \cos(\theta)} \right]^2 \times \left[ \frac{12\sigma_\phi^2}{M(M^2 - 1)} \left( 1 + \frac{1}{N} \right) + \frac{6\sigma_n^2}{M(M^2 - 1)PN} \left( 1 + \frac{2\sigma_{sum}^2}{M} \right) \right] \quad (2.37)$$

ดังนั้น ค่าความผิดพลาดเฉลี่ยของการวิเคราะห์หาทิศทางสามารถหาได้จาก

$$RMSE = \sqrt{(MSE)_{RMU}} \quad (2.38)$$

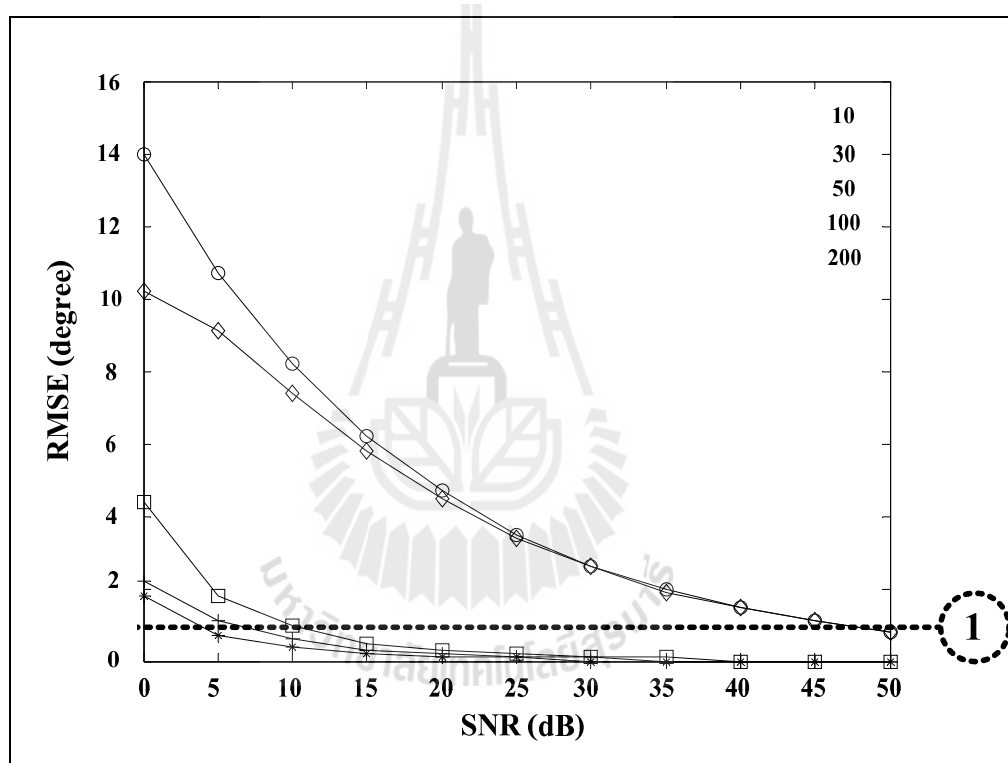
จากสมการ (2.37) เป็นการพิจารณาในกรณีที่มีสัญญาณเข้ามาที่สายอากาศแถวลำดับแบบเชิงเส้นเพียง 1 ทิศทางเท่านั้น และจะไม่คำนึงถึงค่าการสูญเสียต่างๆ ( $\sigma_g^2 = 0$  และ  $\sigma_\phi^2 = 0$ ) ซึ่งในที่นี้ก็คือ  $\sigma_{sum}^2 = \sigma_g^2 + \sigma_\phi^2 = 0$  โดยที่  $P$  คือ กำลังของสัญญาณ,  $\sigma_n^2$  คือ กำลังของสัญญาณรบกวนในระบบ,  $M$  คือ จำนวนของสายอากาศ,  $N$  คือ จำนวนของการสุ่มตัวอย่าง และ  $\theta$  คือ ผลต่างของทิศทางที่หาได้กับทิศทางจริงของสัญญาณ



รูปที่ 2.33 กราฟค่าความผิดพลาดเฉลี่ยของการวิเคราะห์หาทิศทาง เมื่อมีสัญญาณเข้ามาในทิศทาง 30 องศา

จากปัญหาดังกล่าว งานวิจัยนี้ได้นำเสนอเทคนิคการปรับจำนวนการสุ่มตัวอย่างมาประยุกต์ใช้ร่วมกับวิธี MUSIC ในการวิเคราะห์หาทิศทางการมาถึงของสัญญาณในระบบสายอากาศ

เก่ง โดยจะทำการปรับเลือกจำนวนการสุ่มตัวอย่างให้เหมาะสมกับค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวนในระบบ เพื่อทำการวิเคราะห์หาทิศทางมาถึงของสัญญาณให้ได้อย่างถูกต้องและแม่นยำ นอกจากนี้ระบบยังไม่จำเป็นต้องใช้จำนวนการสุ่มตัวอย่างที่มากเกินไปจนความเหมาะสม ดังนั้นตัวประมวลผลในระบบจึงสามารถที่จะประมวลผลได้อย่างรวดเร็ว ซึ่งถือเป็นการถนอมซีพียูของตัวประมวลผลไม่ทำให้ทำงานหนักได้อีกด้วย โดยเทคนิคการปรับจำนวนการสุ่มตัวอย่างนั้นจะมีแนวคิดดังต่อไปนี้



รูปที่ 2.34 กราฟค่าความผิดพลาดเฉลี่ยในการวิเคราะห์หาทิศทาง กรณีที่แย่ที่สุดของแต่ละการสุ่มตัวอย่าง

รูปที่ 2.34 เราสามารถหาได้จากการจำลองแบบในคอมพิวเตอร์เพื่อทำการหาค่าความผิดพลาดเฉลี่ยของการวิเคราะห์หาทิศทางอย่างในกรณีของรูปที่ 2.33 ไปทุกๆ ทิศทางตั้งแต่ 0-180 องศาในแต่ละจำนวนการสุ่มตัวอย่าง โดยจะทำการเลือกค่าความผิดพลาดเฉลี่ยของทิศทางที่แย่ที่สุดในแต่ละจำนวนการสุ่มตัวอย่างมาพล็อตดังกราฟรูปที่ 2.34 จากนั้นเราจะทำการลากเส้นกำหนดค่าความผิดพลาดเฉลี่ย ซึ่งในกรณีนี้เราได้กำหนดให้ระบบมีค่าความผิดพลาดเฉลี่ยให้ไม่เกิน 1 องศา

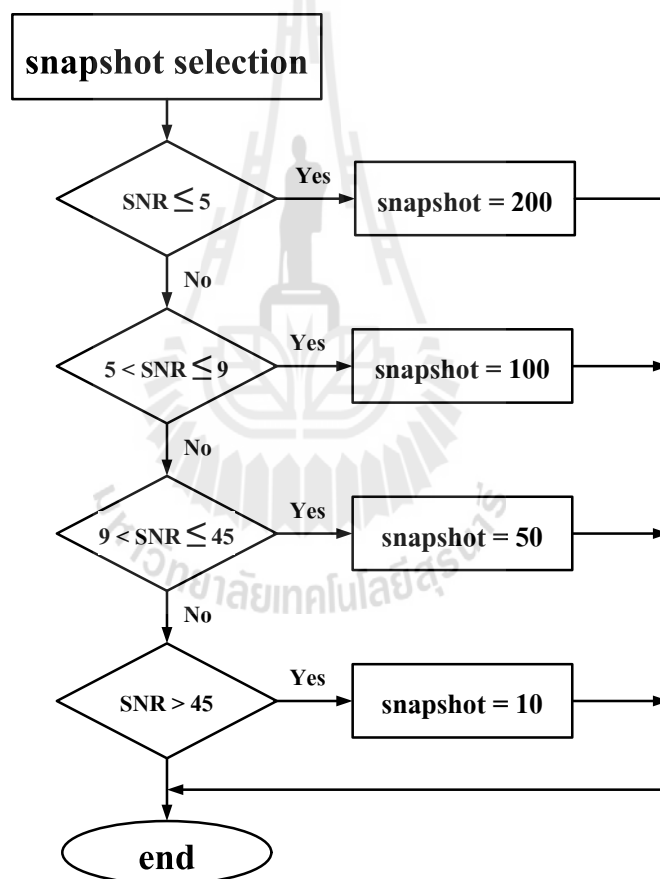
จะได้เงื่อนไขออกมาดังตารางที่ 2.1 ก็คือ เมื่อระบบมีค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวนในระบบ 0-5 dB เทคนิคนี้ก็จะทำการปรับเลือกจำนวนการสุ่มตัวอย่างเท่ากับ 200 แต่ถ้าเทคนิคนี้ทำการปรับเลือกจำนวนการสุ่มตัวอย่างเท่ากับ 50 ก็แสดงว่าระบบมีค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวนอยู่ระหว่าง 10-45 dB เป็นต้น ดังนั้นเมื่อนำเงื่อนไขที่ได้ขึ้นไปเขียน โปรแกรมก็จะได้แผนภาพการทำงานออกมาดังรูปที่ 2.35 ซึ่งจะมีลักษณะของการทำงานดังนี้ ก็คือ ระบบจะทำการหาค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวนออกมาก่อน จากนั้นก็จะนำค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวนที่หาได้ไปทำการเปรียบเทียบกับเงื่อนไขต่างๆ ที่เราได้กำหนดในตารางที่ 2.1 ถ้าค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวนไปตรงกับเงื่อนไขใดก็จะทำการปรับเลือกจำนวนการสุ่มตัวอย่างของเงื่อนไขนั้นมาใช้ในการวิเคราะห์หาทิศทางมาถึงของสัญญาณแบบวิธี MUSIC ต่อไป

เมื่อนำเงื่อนไขที่ได้จากตารางที่ 2.1 มาใช้ในการทดสอบวิเคราะห์หาทิศทางมาถึงของสัญญาณก็จะได้กราฟค่าความผิดพลาดเฉลี่ยของการวิเคราะห์หาทิศทางได้ดังรูปที่ 2.36 และ 2.37 ซึ่งจะแสดงการปรับเลือกจำนวนการสุ่มตัวอย่าง เมื่อมีสัญญาณเข้ามาในทิศทาง 10 และ 40 องศาตามลำดับ จะเห็นได้ว่า กราฟเส้นสีแดงจะมีการปรับเลือกจำนวนการสุ่มตัวอย่างให้เหมาะสมกับค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวนในระบบที่มีค่าเพิ่มขึ้นเรื่อยๆ นั่นก็คือ เมื่อระบบมีค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวนที่มากขึ้น เทคนิคนี้ก็จะทำการปรับเลือกจำนวนการสุ่มตัวอย่างที่มีจำนวนน้อยมาใช้ แต่ในขณะเดียวกันถ้าหากระบบมีค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวนที่น้อยลง ก็จะทำให้ทำการปรับเลือกจำนวนการสุ่มตัวอย่างที่มีจำนวนมากขึ้นมาใช้ เป็นต้น และกราฟที่ได้ของทั้ง 2 กรณี นั้นก็จะมีค่าความผิดพลาดเฉลี่ยของการวิเคราะห์หาทิศทางไม่เกิน 1 องศาตามเงื่อนไขที่เราได้ทำการกำหนดไว้ได้อีกด้วย ซึ่งจากรูปที่ 2.36 และ 2.37 นั้นจะเห็นได้ว่า เมื่อเราทำการเปลี่ยนมุมในการทดลอง กราฟค่าความผิดพลาดเฉลี่ยของการวิเคราะห์หาทิศทางในแต่ละจำนวนการสุ่มตัวอย่าง (นั่นก็คือกราฟเส้นสีค่าทั้ง 5 เส้น) ก็จะแตกต่างกันออกไป แต่การปรับเลือกจำนวนการสุ่มตัวอย่างให้เหมาะสมกับค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวนในระบบก็จะยังคงเป็นไปตามเงื่อนไขเช่นเดิม

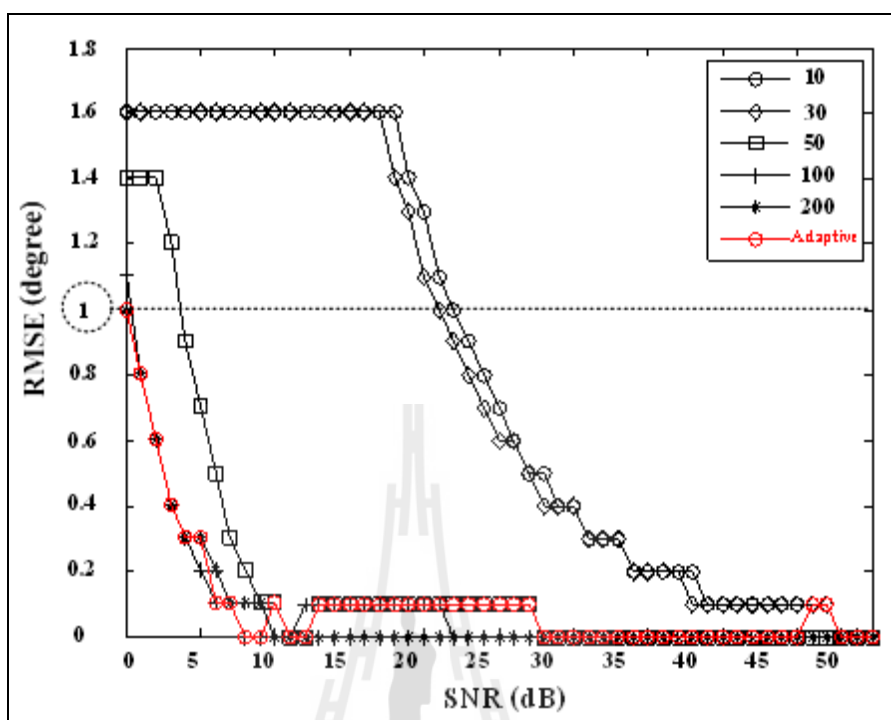
ดังนั้นจากผลที่ได้จากการจำลองแบบเราจึงสรุปได้ว่า เทคนิคการปรับจำนวนการสุ่มตัวอย่างสามารถทำให้ระบบมีการปรับเลือกจำนวนการสุ่มตัวอย่างให้เหมาะสมกับค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวนที่แปรเปลี่ยนไปได้ และผลที่ได้ยังมีค่าความผิดพลาดเฉลี่ยของการวิเคราะห์หาทิศทางไม่เกินเงื่อนไขที่เรากำหนดไว้ได้อีกด้วย โดยแนวคิดที่ได้นี้จะถูกนำไปทำการทดสอบวัดจริงและจะแสดงผลทั้งหมดในบทที่ 4 ต่อไป

ตารางที่ 2.1 เงื่อนไขการปรับจำนวนการสุ่มตัวอย่างที่ให้ความผิดพลาดไม่เกิน 1 องศา

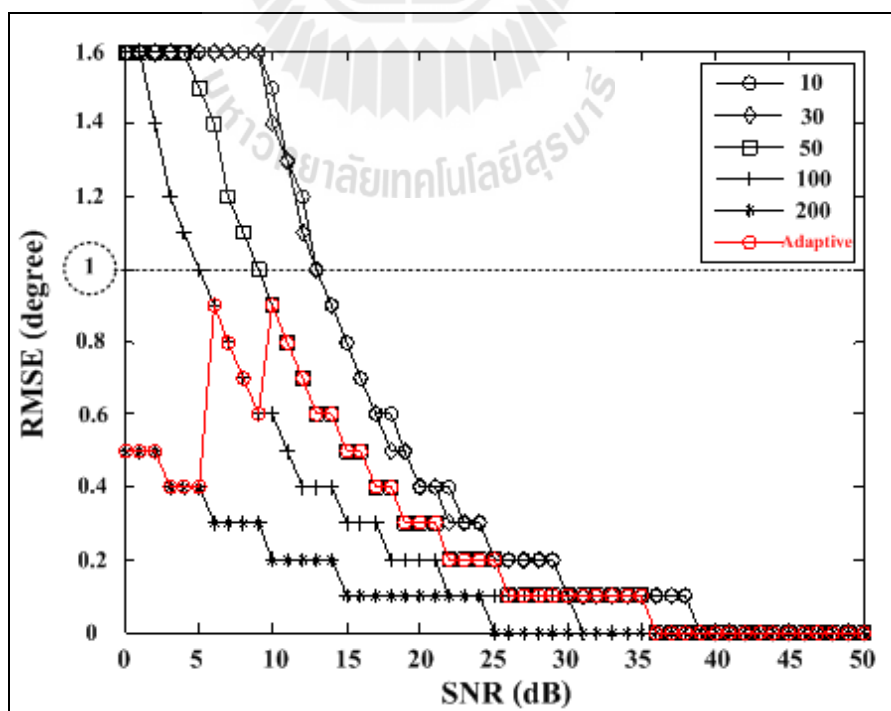
ค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวน	จำนวนการสุ่มตัวอย่าง
0 - 5	200
6 - 9	100
10 - 45	50
46 - 50	10



รูปที่ 2.35 แผนภาพการทำงานของโปรแกรมตามเงื่อนไขดังตารางที่ 2.1



รูปที่ 2.36 การปรับเลือกจำนวนการสุ่มตัวอย่าง เมื่อมีสัญญาณเข้ามาที่ 10 องศา



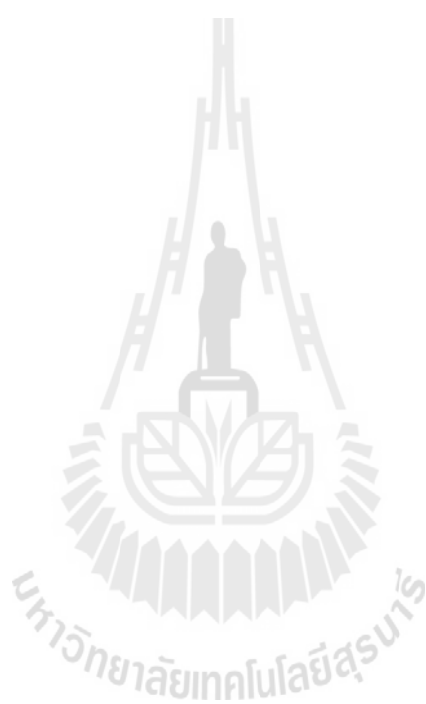
รูปที่ 2.37 การปรับเลือกจำนวนการสุ่มตัวอย่าง เมื่อมีสัญญาณเข้ามาที่ 40 องศา

## 2.12 สรุป

จากเนื้อหาข้างต้นได้กล่าวถึงหลักการและทฤษฎีต่างๆ ของชุดอุปกรณ์ภาครับ ได้แก่ วงจรแปลงสัญญาณ ความถี่ต่ำ วงจรขยายสัญญาณ และวงจรกรองสัญญาณ รวมไปถึงบอร์ดไมโครคอนโทรลเลอร์ที่ใช้เป็นตัวประมวลผลของระบบ ซึ่งทำให้ทราบถึงวิธีการทำงานและผลกระทบต่างๆ ในการออกแบบและสร้างขึ้นจริงของแต่ละอุปกรณ์ ก็เพื่อไม่ให้เกิดความเสียหายกับชุดอุปกรณ์ต่างๆ ในขณะที่ทำการทดสอบ นอกจากนั้นยังทำให้ทราบถึงข้อเสียของการวิเคราะห์หาทิศทางการมาถึงของสัญญาณแบบวิธี MUSIC ในระบบสายอากาศเก่ง ทำให้เราได้แนวคิดเกี่ยวกับการแก้ปัญหาดังกล่าว โดยการนำเสนอเทคนิคการปรับจำนวนการสุ่มตัวอย่างมาประยุกต์ใช้ร่วมกับวิธี MUSIC ในการวิเคราะห์หาทิศทาง ซึ่งส่งผลดีต่อระบบและตัวประมวลผลนั่นก็คือ ระบบสามารถประมวลผลได้รวดเร็วขึ้น ซึ่งถือเป็นการถนอมชีพียูของตัวประมวลผลไม่ให้งานหนักได้อีกด้วย





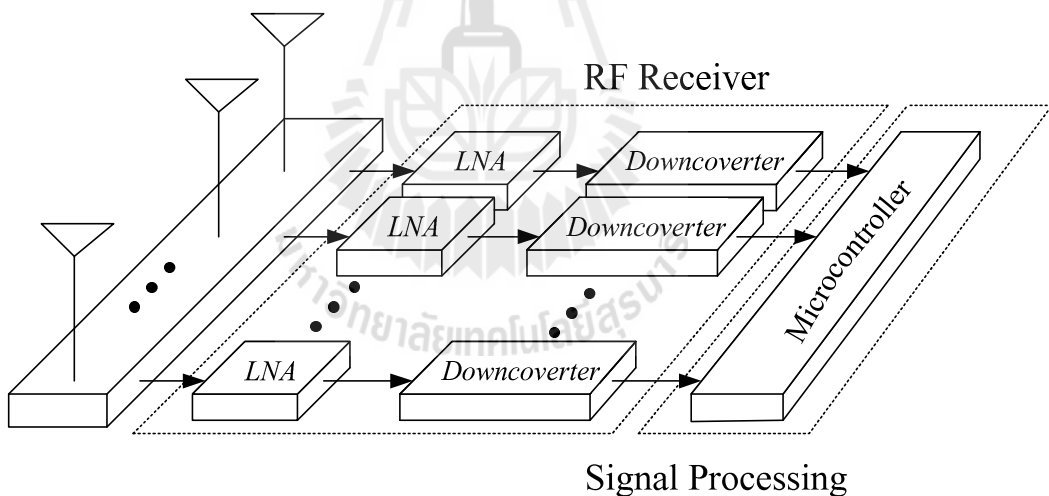


## บทที่ 3

### การออกแบบชุดอุปกรณ์ต้นแบบหาทิศทางการมาถึงของสัญญาณ

#### 3.1 กล่าวนำ

เนื้อหาในบทนี้กล่าวถึงการออกแบบชุดอุปกรณ์ต้นแบบหาทิศทางการมาถึงของสัญญาณ โดยใช้เทคนิคการปรับจำนวนการสุ่มตัวอย่างดังที่ได้นำเสนอมาแล้วในข้างต้น รูปที่ 3.1 แสดงส่วนประกอบของระบบของงานวิจัยนี้ ซึ่งสามารถแบ่งเป็น 2 ส่วนหลักๆ คือ ชุดอุปกรณ์ภาครับสัญญาณ ซึ่งจะประกอบไปด้วยสายอากาศแถวลำดับเชิงเส้น วงจรขยายสัญญาณรบกวนต่ำ วงจรแปลงสัญญาณความถี่ต่ำ และในส่วนของตัวประมวลผลสัญญาณคือ บอร์ดไมโครคอนโทรลเลอร์ทำหน้าที่แปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล และส่งสัญญาณไปยังคอมพิวเตอร์ส่วนบุคคลเพื่อทำการหาทิศทางด้วยโปรแกรมแมทแล็บ



รูปที่ 3.1 ส่วนประกอบของชุดอุปกรณ์ต้นแบบหาทิศทางการมาถึงของสัญญาณ

### 3.2 สายอากาศแถวลำดับ

สายอากาศที่ใช้ในอุปกรณ์ต้นแบบหาทิศทางการมาถึงของสัญญาณนั้นจะเป็นสายอากาศประเภทแถวลำดับเชิงเส้นขนาด  $4 \times 1$  นั่นก็คือ ในระบบจะใช้สายอากาศจำนวน 4 ต้น เนื่องจากเราจำเป็นต้องเลือกใช้สายอากาศที่มีคุณสมบัติในการแผ่กระจายคลื่นได้รอบทิศทางในมุมแนวระนาบ เราจึงเลือกใช้สายอากาศชนิดโมโนโพล (monopole antenna) ที่สามารถรองรับความถี่ได้ที่ 2.45GHz ซึ่งเป็นย่านความถี่ที่เรานำไปใช้งาน ดังรูปที่ 3.2 แสดงลักษณะการวางตัวของสายอากาศที่ใช้ในระบบการวิเคราะห์หาทิศทางมีการวางตัวเป็นเส้นตรง ซึ่งมีระยะห่างระหว่างสายอากาศ  $d = \lambda / 2$  เนื่องจากเราได้จำลองแบบด้วยโปรแกรมแมทแลบเพื่อดูแบบรูปการแผ่พลังงานของสายอากาศ พบว่าสายอากาศชุดนี้สามารถก่อรูปลำคลื่นหลักตามที่เรต้องการได้ ซึ่งแสดงดังรูปที่ 3.3 และสามารถคำนวณหาระยะห่างระหว่างสายอากาศแต่ละต้นได้จากสมการที่ (3.2)

$$v = f\lambda \quad (3.1)$$

ความยาวคลื่น

$$\lambda = \frac{v}{f} = \frac{3 \times 10^8}{2.45 \times 10^9}$$

$\therefore \lambda = 12.24$  เซนติเมตร

โดยระยะห่างระหว่างสายอากาศในสมการ (3.2)

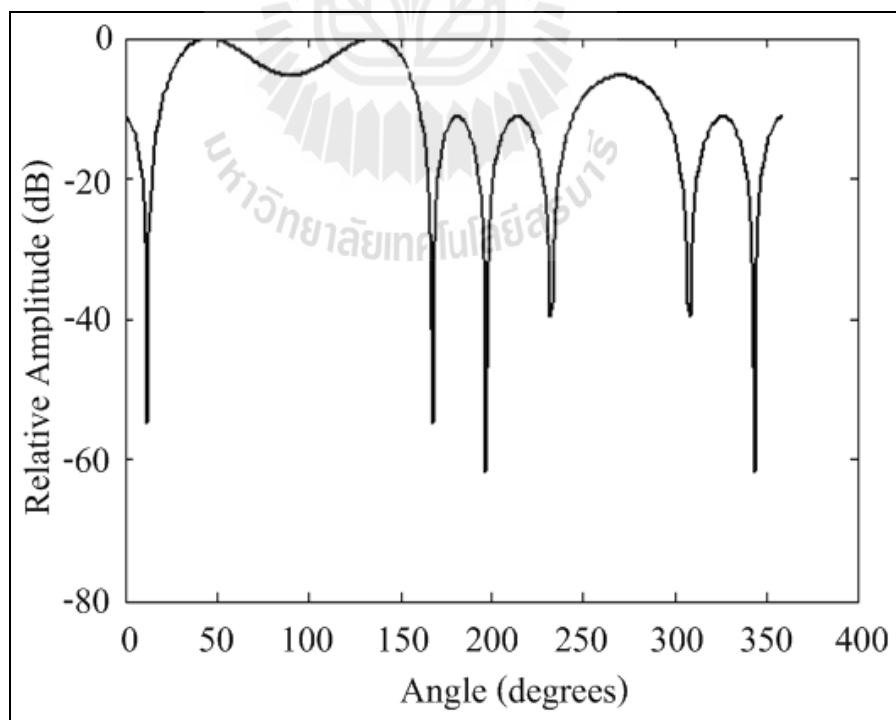
$$d = \frac{\lambda}{2} = \frac{12.24}{2} \quad (3.2)$$

$\therefore d = 6.12$  เซนติเมตร

ดังนั้นระยะห่างของสายอากาศเท่ากับ 6.12 เซนติเมตร



รูปที่ 3.2 สายอากาศแกลวลำดับเชิงเส้นขนาด  $4 \times 1$  ที่ใช้ในงานวิจัยนี้



รูปที่ 3.3 แบบรูปการแผ่พลังงานของสายอากาศแกลวลำดับเชิงระนาบขนาด  $4 \times 1$  มีระยะห่างระหว่างสายอากาศเท่ากับ  $\lambda/2$

### 3.3 วงจรขยายสัญญาณรบกวนต่ำ

สัญญาณที่รับมาได้จากสายอากาศนั้นอาจจะมีกำลังที่ต่ำ และอาจจะมีสัญญาณรบกวนปะปนมากับสัญญาณ ดังนั้นระบบจึงจำเป็นต้องมีวงจรขยายสัญญาณรบกวนต่ำเพื่อทำหน้าที่ในการขยายกำลังของสัญญาณและกำจัดสัญญาณรบกวน ในงานวิจัยนี้ได้ใช้วงจรสำเร็จรูปดังรูปที่ 3.4 ซึ่งมีรูปแบบโครงสร้างและขนาดของโครงสร้างดังรูปที่ 3.5 โดยรูปที่ 3.6 และ 3.7 จะแสดงค่าเจาะจงทางไฟฟ้าและข้อมูลประสิทธิภาพโดยทั่วไปของวงจรสัญญาณรบกวนต่ำตามลำดับ ซึ่งจะมีการแสดงกราฟเปรียบเทียบถึงประสิทธิภาพต่างๆ ได้ดังรูปที่ 3.8 ถึงรูปที่ 4.2



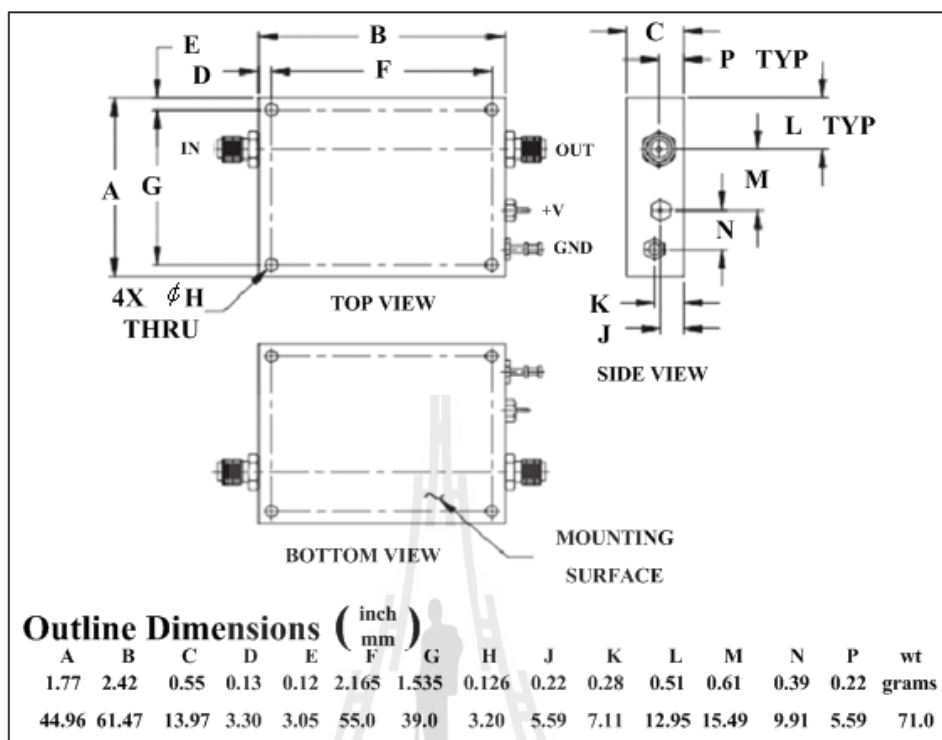
รูปที่ 3.4 วงจรขยายสัญญาณรบกวนต่ำแบบสำเร็จรูป

#### คุณลักษณะ

- แบนด์วิธกว้าง 2,200 ถึง 2,700MHz
- มีสัญญาณรบกวนต่ำมาก สูงสุด 1.5dB
- มีกำลังขยาย +38dBm
- ใช้ไฟเลี้ยงสูงสุด +17V
- ทนอุณหภูมิขณะทำงาน -40°C ถึง 70°C
- อุณหภูมิในการเก็บรักษา -55°C ถึง 100°C
- มีกล่องป้องกันที่แข็งแรง

#### การประยุกต์ใช้งาน

- MMDS
- ระบบสื่อสาร



รูปที่ 3.5 รูปแบบและขนาดของโครงสร้างวงจรถยายสัญญาณรบกวนต่ำ

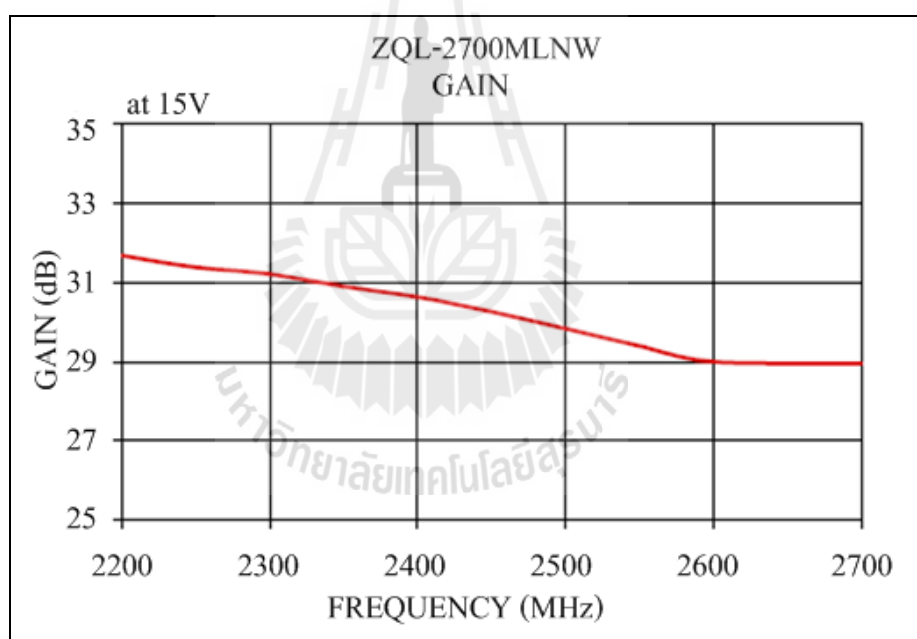
MODEL NO.	FREQUENCY (MHz)		NOISE FIGURE (dB) Max.	GAIN (dB)		MAXIMUM POWER (dBm)	
	$f_L$	$f_U$		Min.	Flatness Max.	Output (1 dB Compr.) Typ.	Input (no damage)
ZQL-2700MLNW(+)	2200	2400	1.3	25	$\pm 1.0$	+25	+3
	2200	2700	1.5	25	$\pm 2.3$	+25	+3

INTERCEPT POINT (dBm)	VSWR (:1) Typ.		DC POWER		
	IP3 Typ.	In	Out	Volt (V) Nom.	Current (mA) Max.
+38		1.25	1.15	15	325
+38		1.25	1.15	15	325

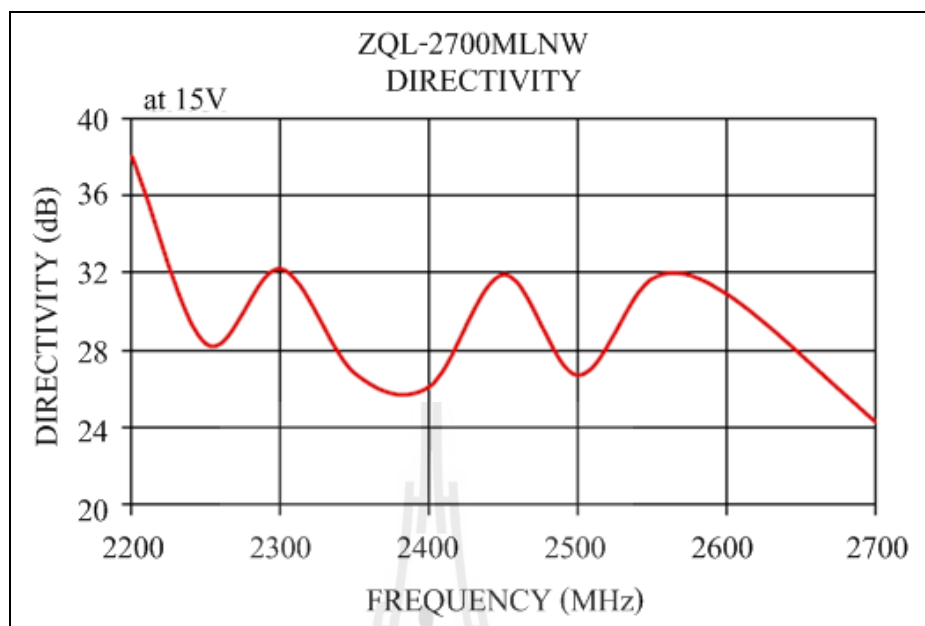
รูปที่ 3.6 ค่าเจาะจงทางไฟฟ้า

FREQUENCY (MHz)	GAIN (dB)	DIRECTIVITY (dB)	VSWR (:1)		NOISE FIGURE (dB)	POUT at 1dB COMPR. (dBm)
	15V		IN	OUT		
2200.00	31.68	38.00	1.23	1.20	0.77	25.38
2250.00	31.38	28.30	1.25	1.16	0.78	25.40
2300.00	31.21	32.20	1.27	1.10	0.81	25.49
2350.00	30.90	26.80	1.27	1.06	0.84	25.71
2400.00	30.63	26.10	1.27	1.05	0.90	26.02
2450.00	30.26	31.90	1.24	1.09	0.98	26.45
2500.00	29.83	26.70	1.21	1.14	1.06	26.33
2550.00	29.39	31.70	1.15	1.19	1.10	25.95
2600.00	29.00	30.90	1.09	1.23	1.14	25.60
2700.00	28.95	24.30	1.15	1.17	1.28	25.27

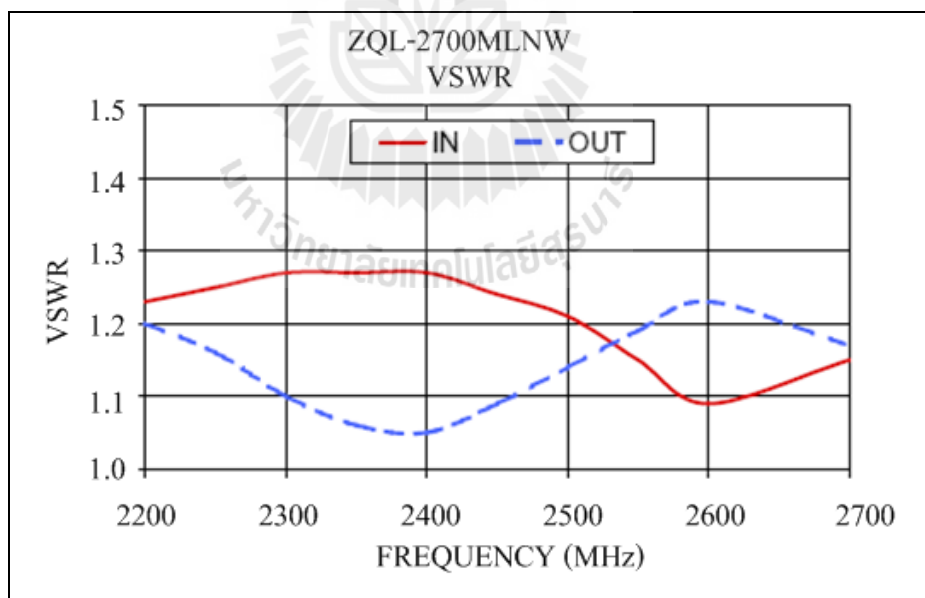
รูปที่ 3.7 ข้อมูลประสิทธิภาพของวงจรขยายสัญญาณรบกวนต่ำ



รูปที่ 3.8 ความสัมพันธ์ระหว่างความถี่และอัตราขยายสัญญาณ

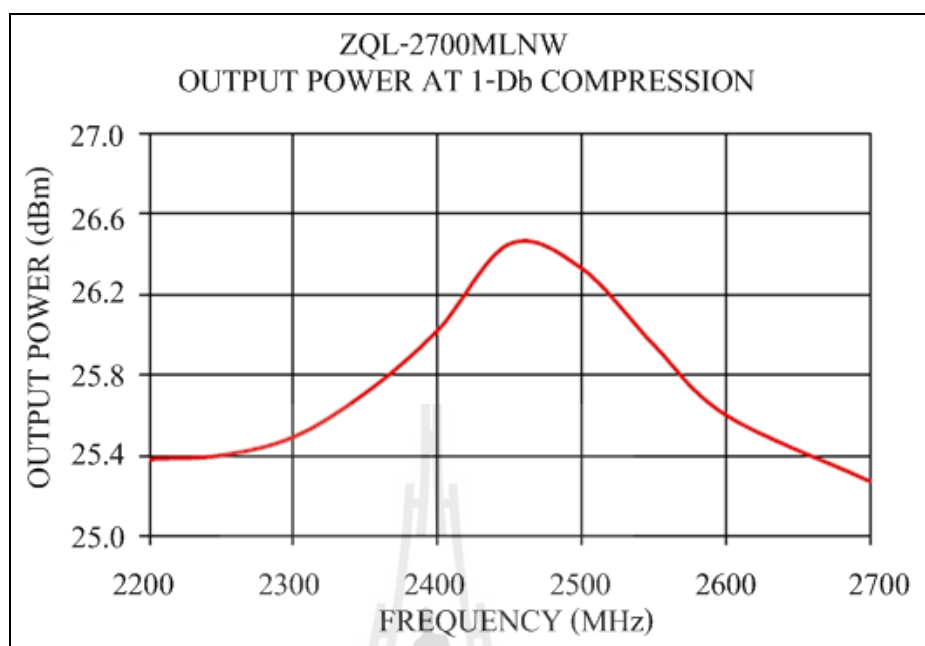


รูปที่ 3.9 ความสัมพันธ์ระหว่างความถี่และค่า directivity

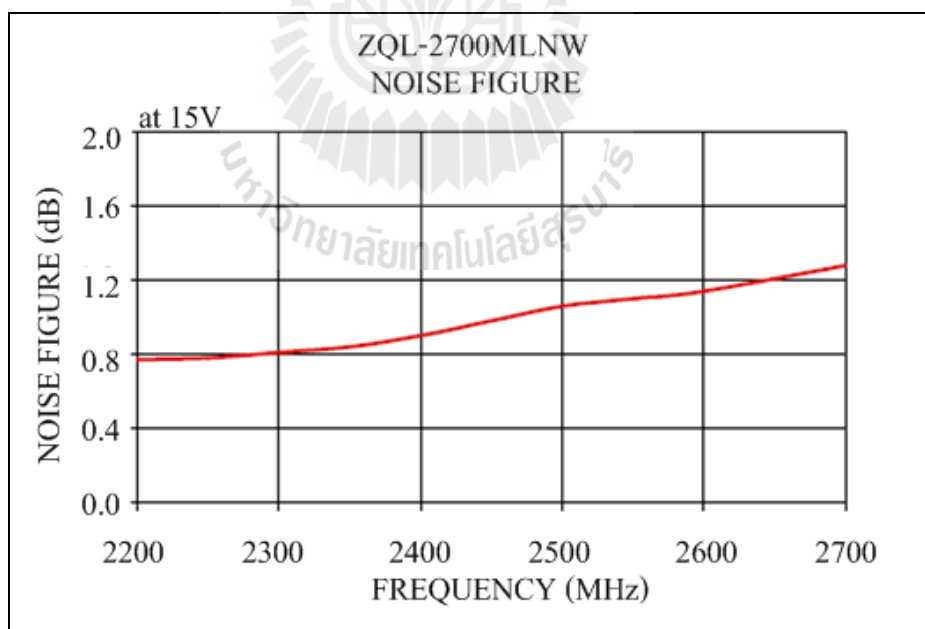


รูปที่ 3.10 ความสัมพันธ์ระหว่างความถี่และค่า VSWR





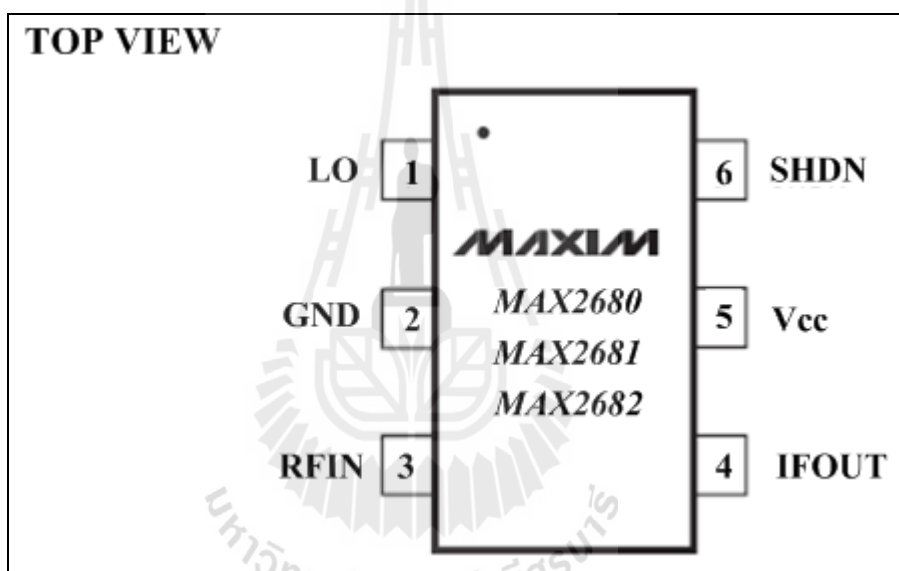
รูปที่ 3.11 ความสัมพันธ์ระหว่างความถี่และค่ากำลังของสัญญาณ



รูปที่ 3.12 ความสัมพันธ์ระหว่างความถี่และสัญญาณรบกวน

### 3.4 วงจรแปลงสัญญาณความถี่ต่ำ

ชุดอุปกรณ์ต้นแบบหาทิศทางนั้นจำเป็นต้องมีวงจรแปลงสัญญาณความถี่ต่ำเพื่อแปลงความถี่ของสัญญาณที่รับได้ (RF signal) ให้มีค่าที่ต่ำลงเพียงพอที่จะส่งไปตามสายนำสัญญาณ โดยการนำสัญญาณ LO ที่สร้างขึ้นเข้าไปผสมกับสัญญาณ RF จะได้สัญญาณผลบวกและผลต่างออกมา โดยเราจำเป็นต้องออกแบบและสร้างวงจรกรองความถี่เพื่อทำการกรองให้เหลือเพียงสัญญาณผลต่างมาใช้งานเท่านั้น ซึ่งในที่นี้ก็คือ สัญญาณ IF นั่นเอง แต่วงจรแปลงสัญญาณความถี่ต่ำที่ใช้ในงานวิจัยนี้จะมีวงจรกรองความถี่ต่ำผ่านอยู่ภายในตัววงจรอยู่แล้ว ซึ่งมีรูปแบบของตัวไอซีดังนี้



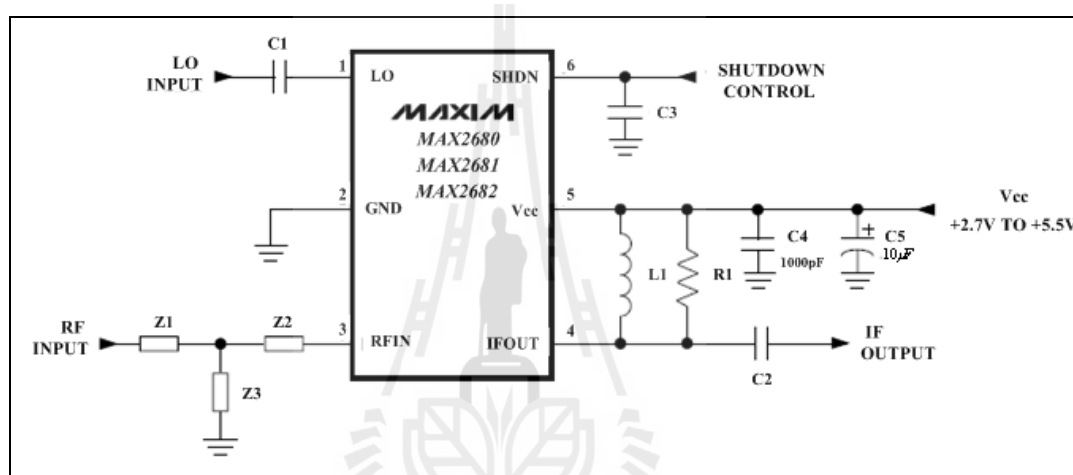
รูปที่ 3.13 ไอซีของวงจรแปลงสัญญาณความถี่ต่ำ

#### คุณลักษณะ

- แบนด์วิธกว้าง 400 ถึง 2,500MHz
- ความถี่เอาต์พุตสูงสุด 500MHz
- กำลังสูงสุด 11.6dB
- ทนอุณหภูมิได้สูงสุด 85°C
- ใช้ไฟเลี้ยงตั้งแต่ +2.7V. ถึง +5.5V.
- มี 6 ขาใช้ติดตั้งบนพื้นผิว และมีขนาดเล็ก

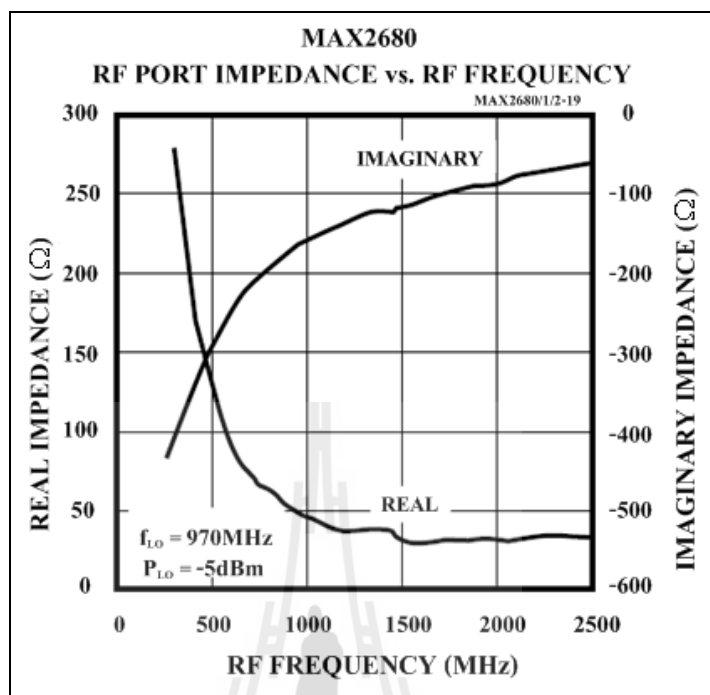
### การประยุกต์ใช้งาน

- 400MHz/900MHz/2.4GHz ISM-band radios
- Personal Communications Systems (PCS)
- cellular and cordless phones
- wireless local loop
- IEEE-802.11 and wireless data

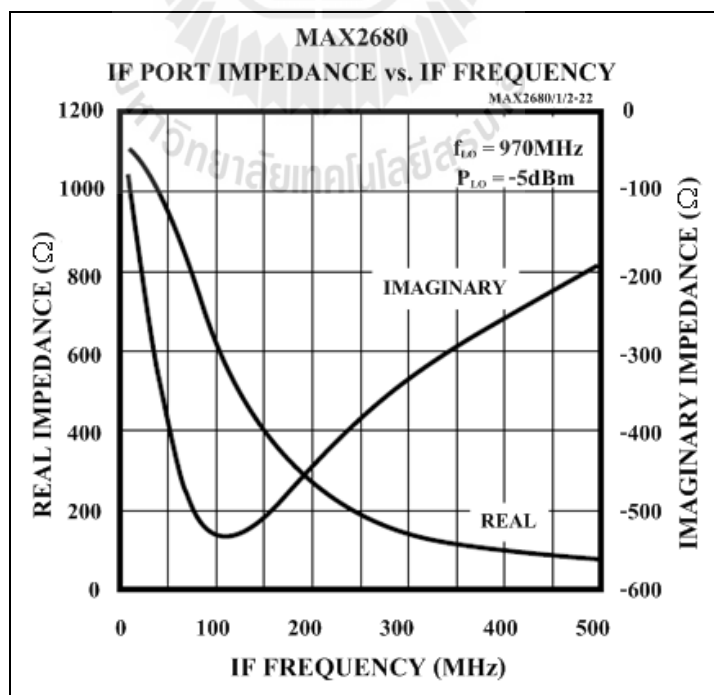


รูปที่ 3.14 วงจรแปลงสัญญาณความถี่ต่ำ

รูปที่ 3.14 แสดงวงจรแปลงสัญญาณความถี่ต่ำ ซึ่งสามารถหาค่า R, L และ C ให้เสถียรกับ ความถี่ที่ต้องการใส่เข้าไปในพอร์ต RF และ IF โดยการคำนวณเมทริกซ์จากค่าจริงและจินตภาพใน รูปที่ 3.15 และ 3.16 ตามลำดับ ดังนั้นจะได้ค่าพารามิเตอร์ต่างๆ ดังรูปที่ 3.17 และ 3.18 โดยสามารถ คู่อค่าสัมประสิทธิ์การสะท้อนที่พอร์ต LO ได้จากรูปที่ 3.19



รูปที่ 3.15 ค่าจริงและจินตภาพเพื่อนำไปคำนวณแมทซิ่งที่พอร์ต RF



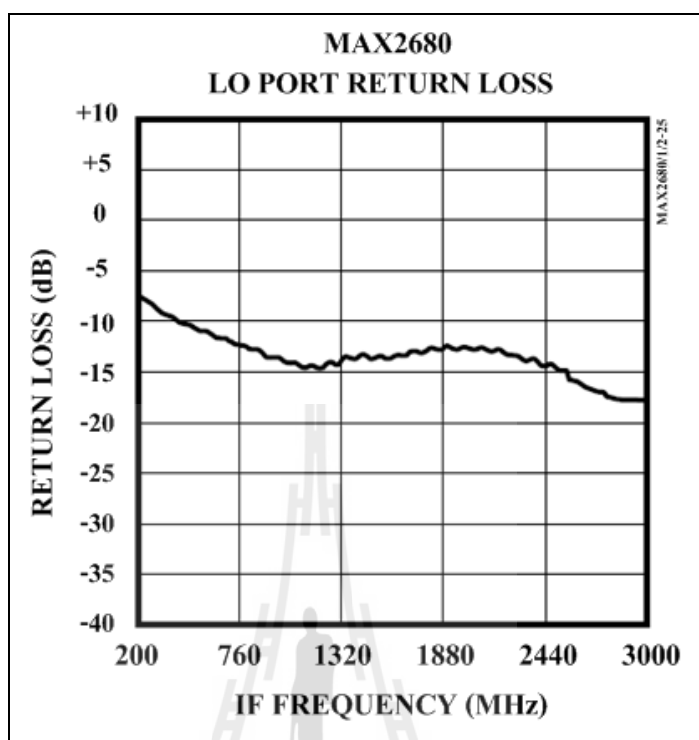
รูปที่ 3.16 ค่าจริงและจินตภาพเพื่อนำไปคำนวณแมทซิ่งที่พอร์ต IF

MATCHING COMPONENTS	FREQUENCY			
	MAX2680			
	400 MHz	900 MHz	1950 MHz	2450 MHz
Z <sub>1</sub>	86nH	270pF	1.5pF	Short
Z <sub>2</sub>	270pF	22nH	270pF	270pF
Z <sub>3</sub>	Open	Open	1.8nH	1.8nH

รูปที่ 3.17 ค่าพารามิเตอร์ต่างๆ ที่พอร์ต RF

MATCHING COMPONENTS	FREQUENCY		
	45MHz	70MHz	240MHz
L <sub>1</sub>	390nH	330nH	82nH
C <sub>2</sub>	39pF	15pF	3pF
R <sub>1</sub>	250Ω	Open	Open

รูปที่ 3.18 ค่าพารามิเตอร์ต่างๆ ที่พอร์ต IF



รูปที่ 3.19 ค่าสัมประสิทธิ์การสะท้อนที่พอร์ต LO

### 3.5 บอร์ดไมโครคอนโทรลเลอร์

บอร์ดไมโครคอนโทรลเลอร์ที่ใช้ในงานวิจัยนี้ เป็นบอร์ดไมโครคอนโทรลเลอร์รุ่น ET-AVR STAMP ATmega128 ซึ่งจะทำหน้าที่อ่านข้อมูลจากสัญญาณอนาล็อกเปลี่ยนเป็นสัญญาณดิจิทัล ดังนั้นเราจึงได้ทำการออกแบบและเขียนโปรแกรมด้วยภาษาซีดังแสดงในภาคผนวก ข.1 โดยโปรแกรมนั้นจะถูกเขียนและคอมไพล์อยู่บนโปรแกรมที่ชื่อว่า CodeVisionAVR ซึ่งจะทำการตรวจสอบว่าโปรแกรมที่เขียนไปนั้นมีข้อผิดพลาดตรงไหนหรือไม่ ก่อนที่จะทำการโปรแกรมลงในบอร์ด ET-AVR STAMP ATmega128 ด้วยโปรแกรม PonyProg2000 ซึ่งมีขั้นตอนต่างๆ ดังแสดงในภาคผนวก ข.2 และภาคผนวก ข.3 ตามลำดับ

### 3.6 โปรแกรมแมทแลบ

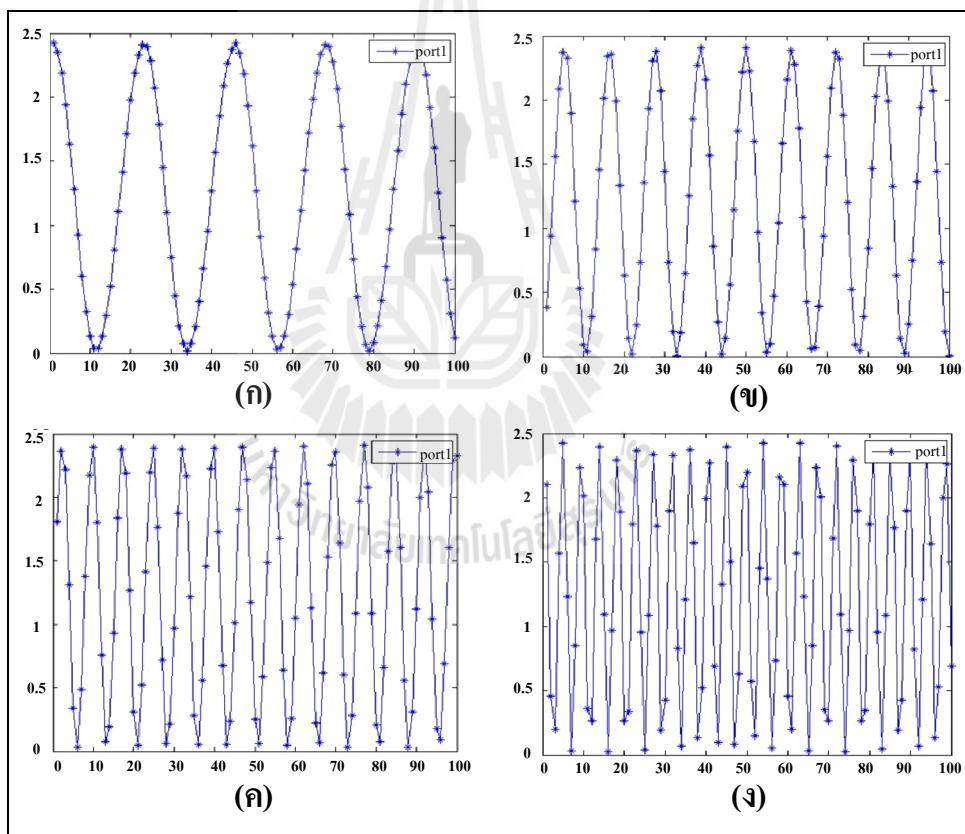
โปรแกรมแมทแลบที่ได้ทำการออกแบบจะมีการนำข้อมูลดิจิทัลที่ได้จากการอ่านค่าจากบอร์ดไมโครคอนโทรลเลอร์มาแปลงค่าให้เป็นข้อมูลอนาล็อกเหมือนเดิม ซึ่งข้อมูลอนาล็อกที่ได้จะ

มีค่า 000 – 3FF มีค่าเท่ากับ 24 บิต แล้วจึงนำข้อมูลอนาล็อกที่ได้ไปเปรียบเทียบกับระดับโวลต์ตั้งแต่ 0 – 5V และการแปลงข้อมูลอนาล็อกเป็นระดับโวลต์จะสามารถหาได้จาก

$$Y = X * 5 / 1024 \quad (3.1)$$

โดยที่ Y คือ ข้อมูลระดับโวลต์ และ X คือ ข้อมูลอนาล็อก

ดังนั้น จะได้โปรแกรมที่ได้ทำการออกแบบมาดังแสดงในภาคผนวก ค



รูปที่ 3.20 สัญญาณที่รับได้จากบอร์ดไมโครคอนโทรลเลอร์ที่ความถี่

(ก) 100Hz (ข) 200Hz (ค) 300Hz และ (ง) 500Hz

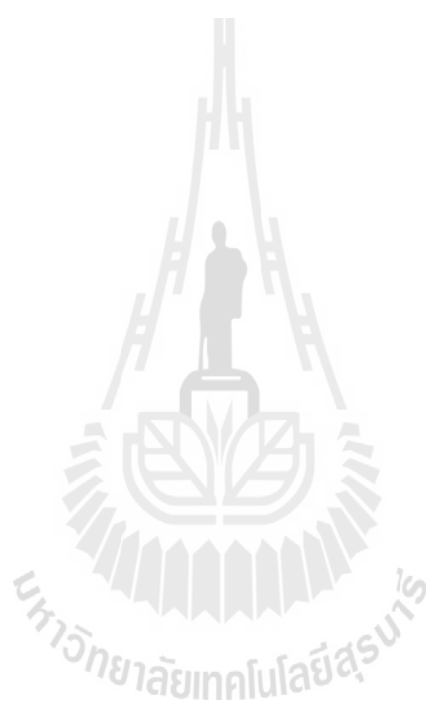
จากรูปที่ 3.20 แสดงตัวอย่างสัญญาณที่โปรแกรมแมทแลบสามารถรับได้จากบอร์ดไมโครคอนโทรลเลอร์ เมื่อเราทำการจ่ายสัญญาณความถี่ 100Hz 200Hz 300Hz และ 500Hz ซึ่งแสดงได้ดังรูปที่ 3.20(ก) ถึง 3.20(ง) ตามลำดับ ซึ่งจากกราฟของสัญญาณที่รับได้ แสดงให้เห็นว่าระหว่างบอร์ดไมโครคอนโทรลเลอร์และคอมพิวเตอร์สามารถส่งข้อมูลเชื่อมต่อถึงกันได้

### 3.7 สรุป

ในบทนี้ได้กล่าวถึงการออกแบบอุปกรณ์ภาครับ ซึ่งในงานวิจัยนี้ประกอบไปด้วยวงจรถ่ายสัญญาณรบกวนต่ำ และวงจรแปลงสัญญาณความถี่ต่ำ รวมไปถึงการสร้างโปรแกรมเพื่อควบคุมบอร์ดไมโครคอนโทรลเลอร์ในการรับข้อมูลอนาล็อกและทำการเปลี่ยนเป็นข้อมูลดิจิทัล ซึ่งจะใช้โปรแกรมแมทแลบรับข้อมูลดิจิทัลดังกล่าวเพื่อทำการหาทิศทางต่อไป จากผลที่ได้พบว่าเราสามารถนำอุปกรณ์เหล่านี้ไปทำการสร้างและวัดผลจริงได้ดังที่เราจะนำเสนอในบทที่ 4 ต่อไป







## บทที่ 4

### การทดสอบและวิเคราะห์ผล

#### 4.1 กล่าวนำ

เนื้อหาในบทนี้จะเป็นการนำเสนอผลที่ได้จากการทดสอบอุปกรณ์ในการวิเคราะห์หาทิศทางการมาถึงของสัญญาณแบบวิธี MUSIC โดยใช้เทคนิคการปรับจำนวนการสุ่มตัวอย่าง ซึ่งเราจะได้รูปแบบของอุปกรณ์ต่างๆ ที่ได้มาจากการออกแบบในบทที่ 3 และในส่วนสุดท้ายจะเป็นการทดสอบประสิทธิภาพของเทคนิคการปรับจำนวนการสุ่มตัวอย่างในพื้นที่ที่มีการจำลองขึ้น โดยนำผลที่ได้นั้นไปเปรียบเทียบกับผลที่ได้จากการจำลองแบบในหัวข้อที่ 2.11

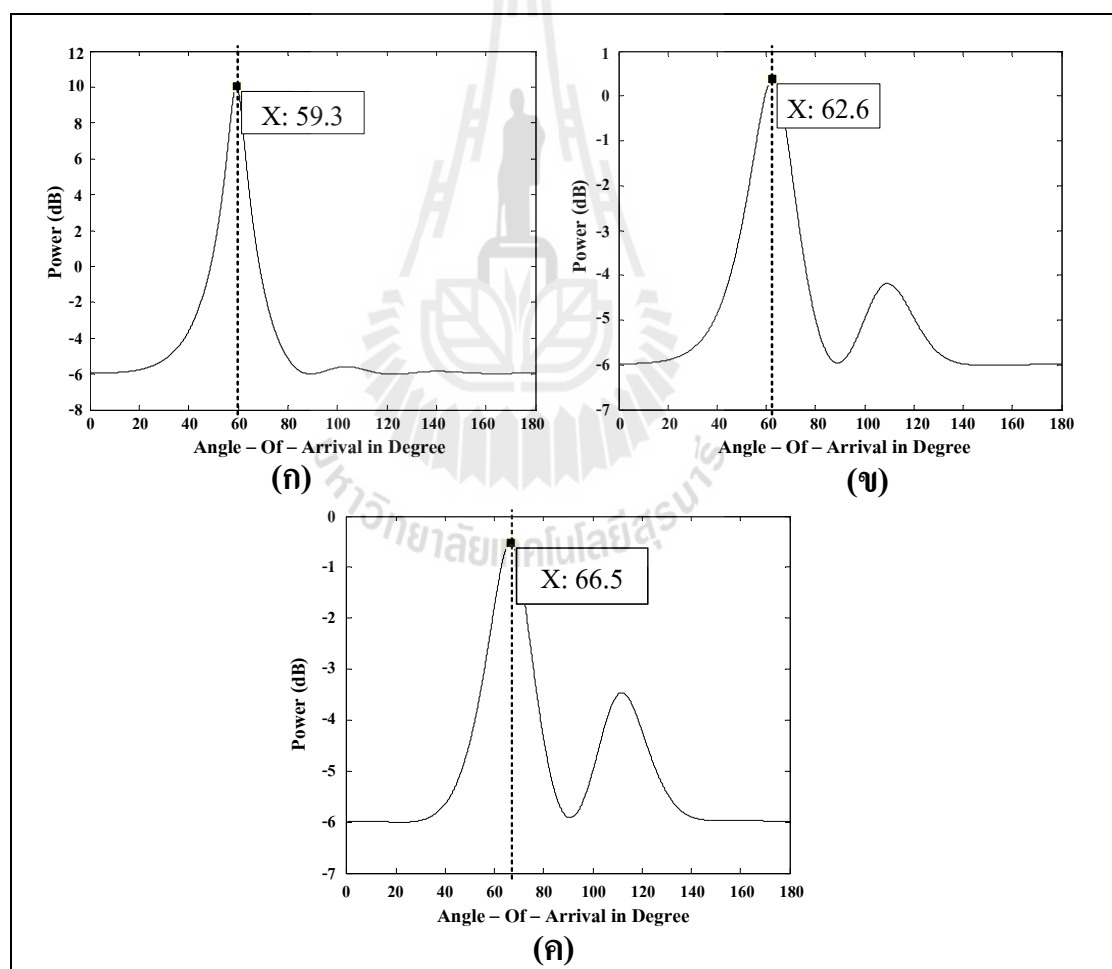
#### 4.2 การทดสอบวงจรแปลงสัญญาณความถี่ต่ำ

เนื่องจากบอร์ดไมโครคอนโทรลเลอร์ที่ใช้ในงานวิจัยนี้มีความเร็วในการสุ่มตัวอย่าง (sampling) ไม่เพียงพอกับสัญญาณความถี่ต่ำที่ทางคู่มือได้กำหนดมาให้ดังรูปที่ 3.18 เราจึงไม่สามารถนำค่าพารามิเตอร์ต่างๆ มาใช้ในการสร้างจริงได้ ดังนั้นในงานวิจัยนี้จึงจำเป็นต้องทำการแปลงสัญญาณความถี่ให้ต่ำลงไปอีก ซึ่งสัญญาณความถี่ต่ำที่เราสามารถทำการวิเคราะห์ได้ดังต่อไปนี้

รูปที่ 3.20 แสดงสัญญาณที่โปรแกรมแมทแลบรับได้จากบอร์ดไมโครคอนโทรลเลอร์ที่สัญญาณความถี่ต่างๆ นั่นก็คือ 100, 200, 300 และ 500Hz ซึ่งแสดงได้ดังรูป 3.20(ก) ถึง 3.20(ง) ตามลำดับ จะเห็นได้ว่าที่สัญญาณความถี่ 300 และ 500Hz นั้นสัญญาณที่รับได้เริ่มมีความผิดเพี้ยนไป ดังนั้นในงานวิจัยนี้จึงควรเลือกใช้สัญญาณความถี่ต่ำที่ไม่เกิน 200Hz และในการวิเคราะห์ถัดมาดังรูปที่ 4.1 แสดงการวิเคราะห์หาทิศทางการมาถึงของสัญญาณ เมื่อเราเลือกใช้สัญญาณความถี่ที่ 50, 100 และ 200Hz ดังรูป 4.1(ก) ถึง 4.1(ค) ตามลำดับ โดยเราจะทำการจำลองใส่วงจรเลื่อนเฟสที่มีเฟสต่างกัน 90 องศาแทนสัญญาณที่รับได้จากสายอากาศ จะเห็นได้ว่าที่สัญญาณความถี่ 50Hz สามารถหาทิศทางได้ที่ 59.3 องศา ซึ่งมีทิศทางที่ใกล้เคียงกับทิศทางที่ถูกต้อง นั่นก็คือ 60 องศา (เมื่อระบบมีวงจรเลื่อนเฟสที่มีเฟสต่างกันอยู่ 90 องศาจะต้องหาทิศทางได้จริงที่ 60 องศา) แต่ในขณะที่เดียวกันที่สัญญาณความถี่ 100Hz และ 200Hz จะเห็นได้ว่าทิศทางที่หาได้นั้นคลาดเคลื่อนไปจากทิศทางที่ถูกต้องมาก นั่นก็คือ 62.6 และ 66.5 องศา ดังรูปที่ 4.1(ข) และ 4.1(ค) ตามลำดับ สรุปได้

ว่างานวิจัยนี้จะเลือกใช้สัญญาณความถี่ต่ำ 50Hz มาใช้ในการออกแบบวงจรแปลงสัญญาณความถี่ต่ำ ซึ่งก็จะได้ค่าพารามิเตอร์ที่พอร์ท IF ที่ได้ทำการออกแบบดังตารางที่ 4.1

รูปที่ 4.2 แสดงวงจรแปลงสัญญาณความถี่ต่ำที่ได้ทำการสร้างขึ้นจริง โดยเราจะทำการป้อนสัญญาณความถี่ 2,450MHz เข้าที่พอร์ท RF และในขณะเดียวกันก็จะทำการป้อนสัญญาณความถี่ 2,449.999950MHz เข้าที่พอร์ท LO ดังนั้นก็จะได้สัญญาณเอาต์พุตที่พอร์ท IF ออกมาที่ความถี่ 50Hz จากนั้นเมื่อนำวงจรที่สร้างขึ้นไปทำการทดสอบวัดจริงก็จะได้ผลออกมาดังรูปที่ 4.3 ก็คือสัญญาณความถี่ต่ำที่ 49.75Hz ดังนั้นสรุปได้ว่า วงจรแปลงสัญญาณความถี่ต่ำที่เราได้ทำการสร้างขึ้นนั้นสามารถให้ผลออกมาตามที่เราได้ทำการออกแบบไว้

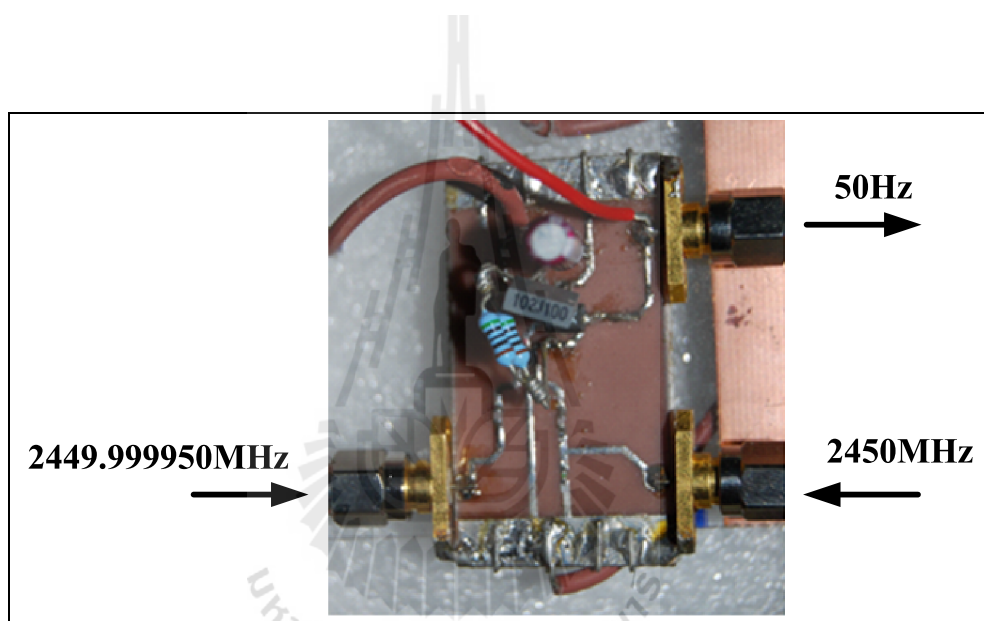


รูปที่ 4.1 การวิเคราะห์หาทิศทางการมาถึงของสัญญาณเมื่อเลือกใช้ความถี่

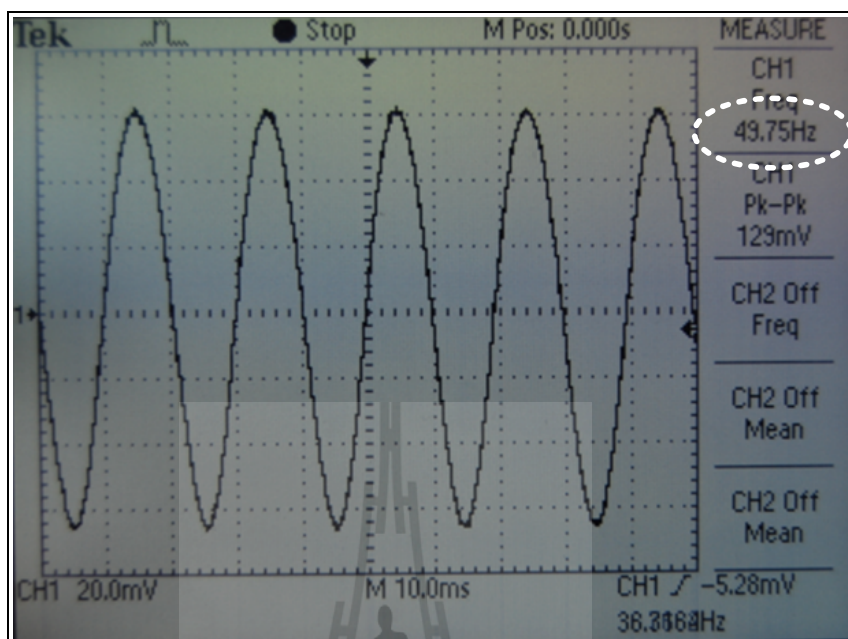
(ก) 50Hz (ข) 100Hz และ (ค) 200Hz

ตารางที่ 4.1 ค่าพารามิเตอร์ต่างๆ ที่พอร์ต IF ที่ได้ทำการออกแบบ

ส่วนประกอบของวงจร ที่พอร์ต IF	ความถี่ 50Hz
L1	680 $\mu$ H
C2	68nF
R1	250 $\Omega$



รูปที่ 4.2 วงจรแปลงสัญญาณความถี่ต่ำที่สร้างขึ้นจริง



รูปที่ 4.3 ผลที่ได้จากวงจรแปลงสัญญาณความถี่ต่ำที่พอร์ต IF

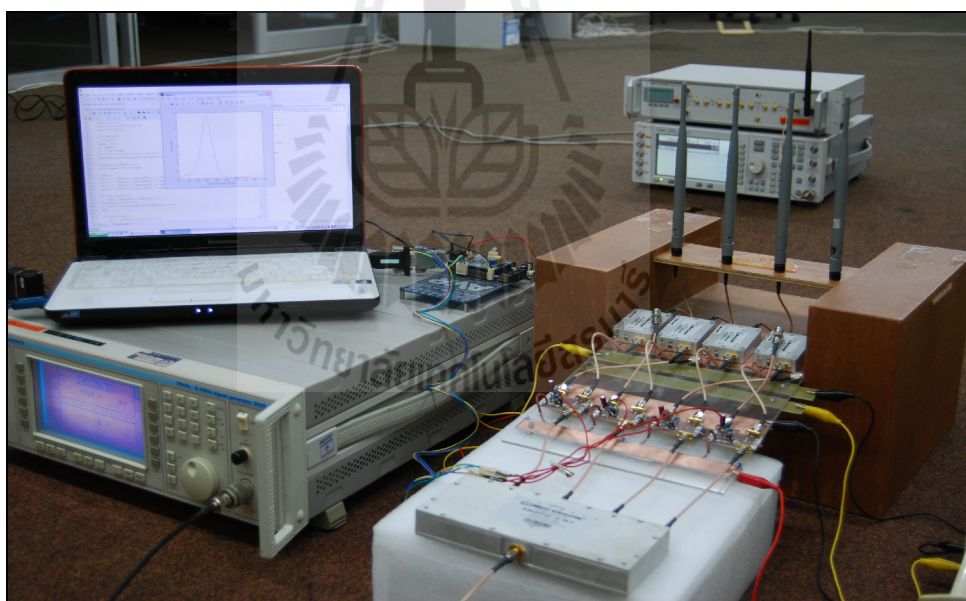
#### 4.3 การทดสอบหาทิศทางด้วยเทคนิคการปรับจำนวนการสุ่มตัวอย่าง

จากตัวอย่างการวิเคราะห์หาทิศทางการมาถึงของสัญญาณที่ผ่านมาในรูปที่ 4.1 เป็นการหาทิศทางโดยการจำลองใส่วงจรเลื่อนเฟสเพื่อเปรียบเทียบกับสัญญาณที่เข้ามาที่สายอากาศแต่ละต้น แต่ในหัวข้อนี้จะแสดงการวิเคราะห์หาทิศทางจากสัญญาณที่รับได้จริงจากสายอากาศ ดังรูปที่ 4.4 ซึ่งจะประกอบไปด้วยอุปกรณ์ต่างๆ ดังนี้ ก็จะมีภาคส่ง 1 ชุด และในส่วนของภาครับก็จะประกอบไปด้วย สายอากาศแถวลำดับแบบเชิงเส้นขนาด  $4 \times 1$  วงจรขยายสัญญาณรบกวนต่ำ วงจรแปลงสัญญาณความถี่ต่ำ และบอร์ดไมโครคอนโทรลเลอร์ จากนั้นก็จะใช้คอมพิวเตอร์มารับข้อมูลเพื่อนำไปประมวลผลต่อไป โดยในการทดสอบนั้นเราจะทำการตั้งชุดอุปกรณ์ภาครับสัญญาณไว้กับที่ และทำการเคลื่อนภาคส่งสัญญาณไปยังทิศทางต่างๆ

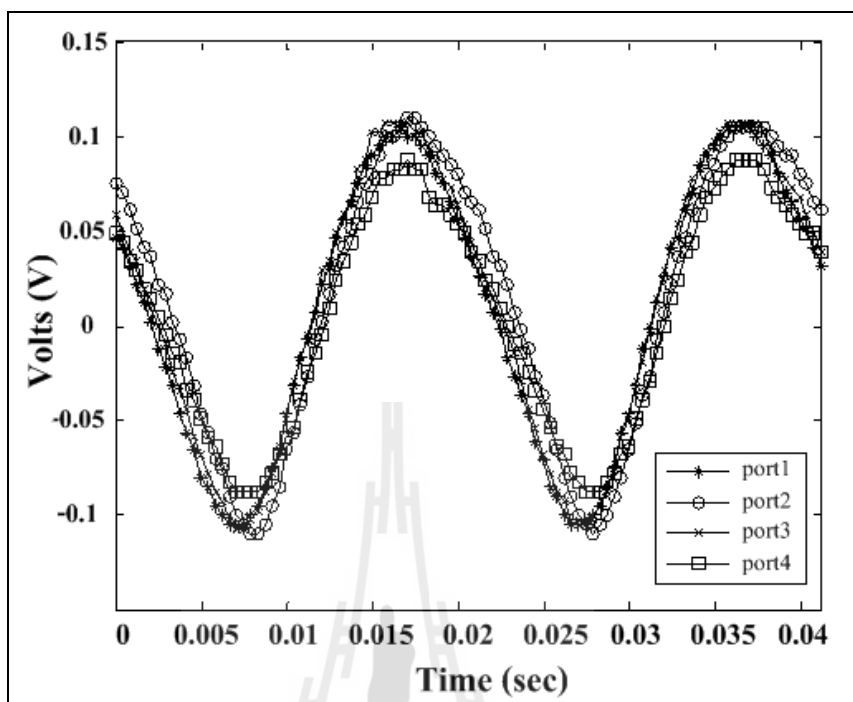
รูปที่ 4.5 และ 4.7 แสดงสัญญาณที่โปรแกรมแมทแลบรับได้จากบอร์ดไมโครคอนโทรลเลอร์ เมื่อเราใช้จำนวนของสุ่มตัวอย่างเท่ากับ 100 และทำการตั้งภาคส่งไว้ที่ตำแหน่ง 90 และ 30 องศาตามลำดับ สัญญาณที่แสดงของพอร์ต 1 2 3 และ 4 ก็คือ สัญญาณที่รับได้จากสายอากาศต้นที่ 1 2 3 และ 4 ตามลำดับ เนื่องจากสายอากาศนั้นรับสัญญาณได้ในเวลาที่แตกต่างกัน ดังนั้นจะเห็นได้ว่าสัญญาณทั้ง 4 พอร์ตจะมีการการเลื่อนเฟสกันอยู่ จากนั้นนำสัญญาณที่ได้ไปประมวลผลวิเคราะห์หาทิศทางการมาถึงของสัญญาณแบบวิธี MUSIC ดังรูปที่ 4.6 และ 4.8 จะเห็น

ได้ว่าระบบสามารถวิเคราะห์หาทิศทางได้จริงที่ประมาณ 90 และ 30 องศาตามลำดับ ดังนั้นจึงสรุปได้ว่า ชุดอุปกรณ์ต้นแบบที่เราได้ทำการออกแบบนั้นสามารถที่จะวิเคราะห์หาทิศทางการมาถึงของสัญญาณได้อย่างถูกต้องและแม่นยำ อีกทั้งชุดอุปกรณ์ต้นแบบยังสามารถนำไปใช้ทดสอบในสภาพแวดล้อมจริงได้อีกด้วย

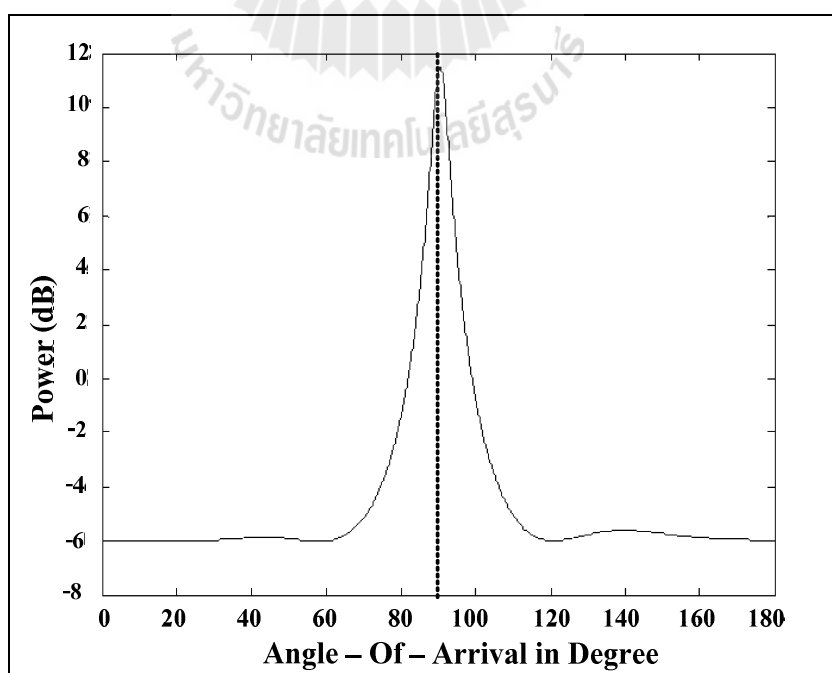
เนื่องจากอุปกรณ์ภาคส่งในห้องปฏิบัติการวิศวกรรมโทรคมนาคมมีจำนวนจำกัด ดังนั้นการทดสอบวัดจริงในงานวิจัยนี้จึงทำการทดสอบวิเคราะห์หาทิศทางได้เพียงทิศทางเดียวเท่านั้น แต่อย่างไรก็ตามถ้ามีภาคส่งเพิ่มเข้ามาอีก 1 ชุดหรือต้องการทดสอบวิเคราะห์หาทิศทางมากกว่า 1 ทิศทาง ก็เพียงแค่แก้ไขในส่วนของโปรแกรมเมทแลบแบบวิธี MUSIC ในขั้นตอนที่ 4 ซึ่งจากเดิมเมื่อเราพิจารณาเพียง 1 ทิศทางก็จะทำการตัดค่า eigenvectors ที่มีค่ามากในตัวแปร  $V_n$  ออกเพียง 1 คอลัมน์ แต่ถ้าต้องการพิจารณา 2 ทิศทางก็จะทำการตัดค่า eigenvectors ที่มีค่ามากออกไป 2 คอลัมน์ เป็นต้น



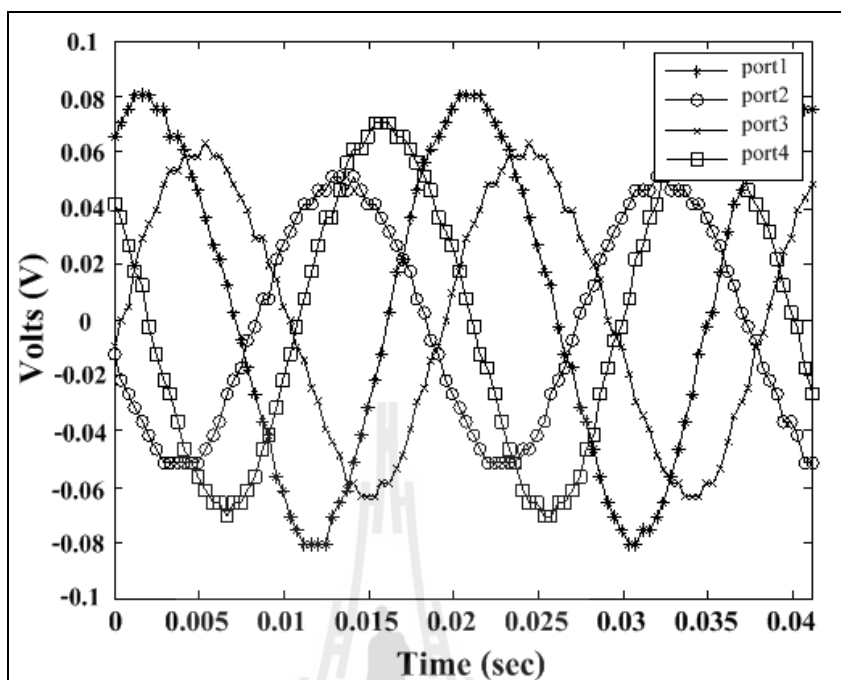
รูปที่ 4.4 ชุดอุปกรณ์ต้นแบบวิเคราะห์หาทิศทางการมาถึงของสัญญาณในการทดสอบจริง



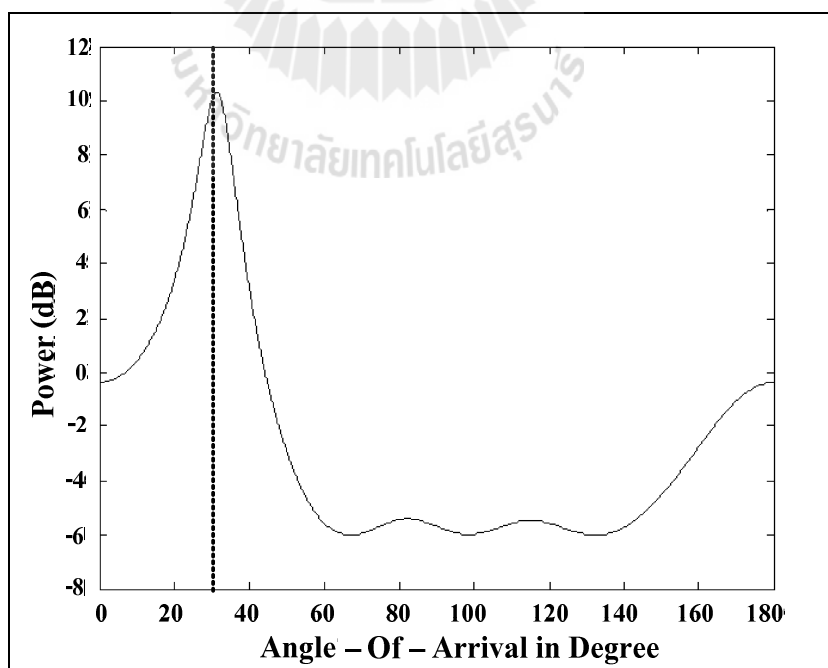
รูปที่ 4.5 สัญญาณที่รับได้จากบอร์ดไมโครคอนโทรลเลอร์ เมื่อมีสัญญาณเข้ามาในทิศทาง 90 องศา



รูปที่ 4.6 ระบบสามารถหาทิศทางได้จริงที่ 90 องศา



รูปที่ 4.7 สัญญาณที่รับได้จากบอร์ดไมโครคอนโทรลเลอร์ เมื่อมีสัญญาณเข้ามาในทิศทาง 30 องศา

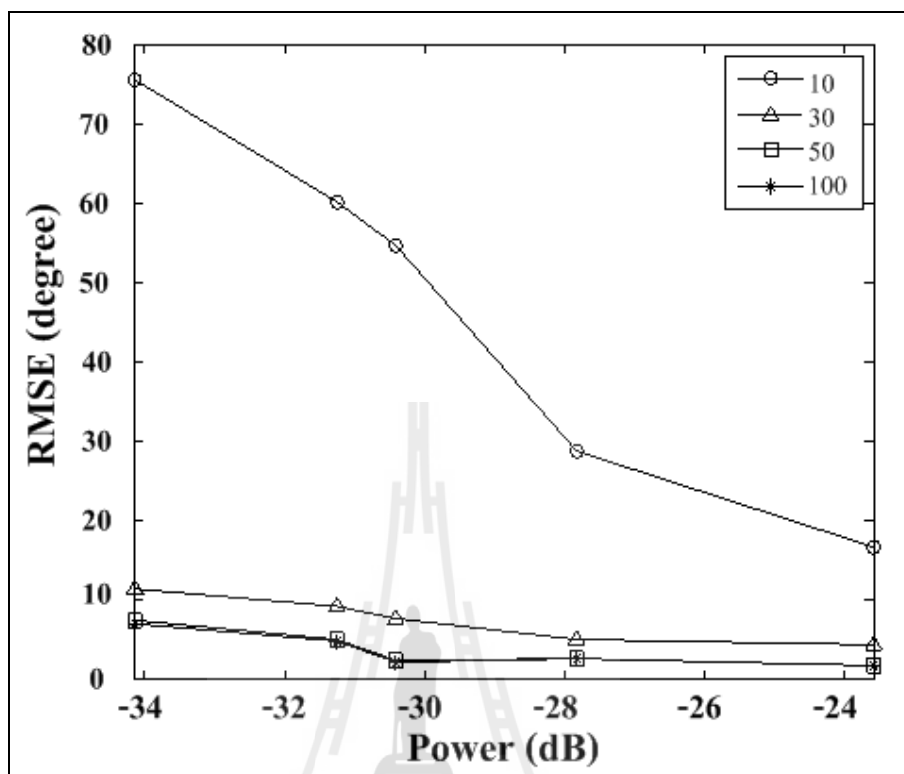


รูปที่ 4.8 ระบบสามารถหาทิศทางได้จริงที่ 30 องศา

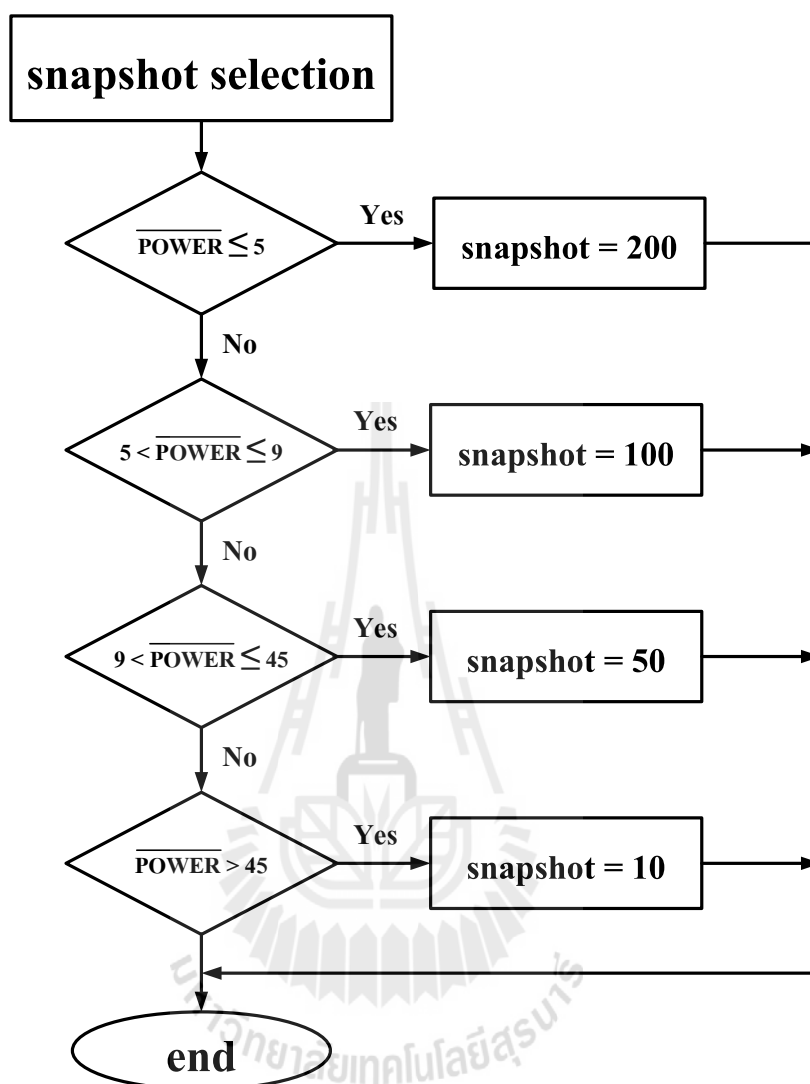


จากนั้นเมื่อเราทำการทดสอบวิเคราะห์หาทิศทางที่มุม 30 องศา ดังในกรณีของรูปที่ 4.7 และ 4.8 โดยการซ้ำทั้งหมด 50 ครั้งของแต่ละจำนวนการสุ่มตัวอย่าง (ซึ่งในที่นี้ก็คือ 10, 30, 50 และ 100) และนำไปหาค่าความผิดพลาดเฉลี่ยของการวิเคราะห์หาทิศทาง ก็จะได้กราฟออกมาดังรูปที่ 4.9 ซึ่งเป็นกราฟของค่าความผิดพลาดเฉลี่ยที่ได้จากผลการทดสอบวัดจริง เมื่อตั้งภาคส่งสัญญาณไว้ที่มุม 30 องศา แต่กราฟที่ได้จากผลการทดสอบวัดจริงนี้ จะมีความแตกต่างจากกราฟที่ได้จากการจำลองแบบดังรูปที่ 2.33 อยู่ นั่นก็คือ กราฟที่ได้จากการจำลองแบบนี้ จะแสดงการเปรียบเทียบระหว่างค่าความผิดพลาดเฉลี่ยของการวิเคราะห์หาทิศทางกับค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวน แต่เนื่องจากสถานที่ที่เราได้ทำการทดสอบนั้นคือ อาคารเครื่องมือ 4 ที่ห้องปฏิบัติการวิศวกรรมโทรคมนาคมซึ่งไม่น่ามีสัญญาณรบกวนใดๆ เข้ามารบกวนชุดอุปกรณ์ต้นแบบของเรามากนัก ดังนั้นในการทดสอบวัดจริงเราจึงได้ทำการกำหนดให้ระบบมีค่าของสัญญาณรบกวนที่เท่ากันตลอดทั้งระบบ ซึ่งในกรณีนี้กราฟที่ได้จึงแสดงได้ดังรูปที่ 4.9 ซึ่งเป็นการเปรียบเทียบระหว่างค่าความผิดพลาดเฉลี่ยของการวิเคราะห์หาทิศทางกับค่ากำลังเฉลี่ยของสัญญาณที่รับได้ แต่ถ้าเรานำชุดอุปกรณ์หาทิศทางของเราไปติดตั้งในสถานที่อื่นๆ ที่มีสัญญาณรบกวนมาก กราฟที่ได้ในรูปที่ 4.9 ก็ จะเปลี่ยนไปก็คือ กราฟที่ได้ก็จะมีค่าความผิดพลาดเฉลี่ยที่สูงขึ้นกว่าในกราฟรูปที่ 4.9 เดิม

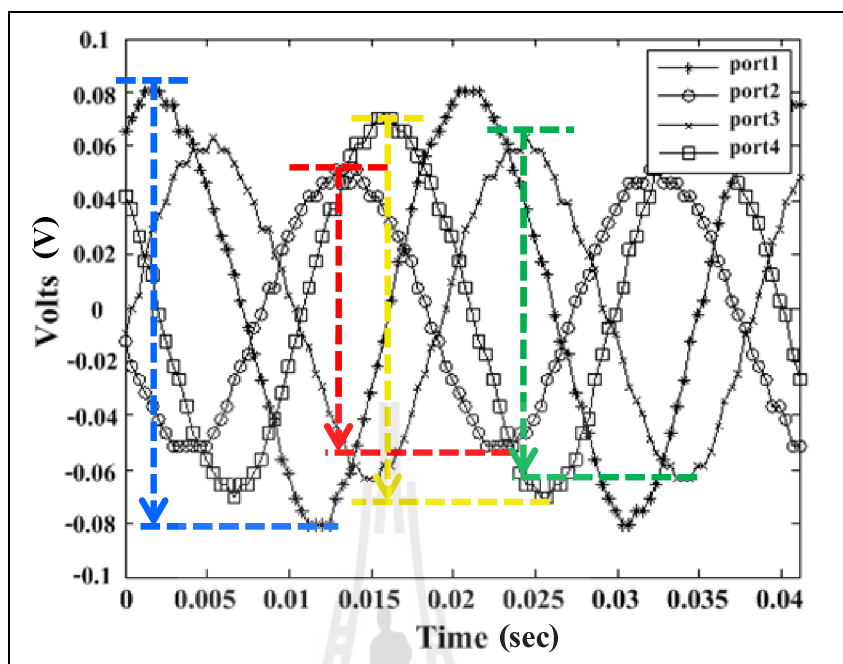
ดังนั้นเมื่อระบบของเราได้ทำการทดสอบหาค่าความผิดพลาดเฉลี่ยมาเปรียบเทียบกับค่ากำลังเฉลี่ยของสัญญาณที่รับได้ แผนภาพการทำงานของโปรแกรมในรูปที่ 2.35 จึงต้องเปลี่ยนแปลงไปซึ่งแสดงได้ดังรูปที่ 4.10 ก็คือ ระบบจะทำการหาค่ากำลังเฉลี่ยของสัญญาณออกมาก่อน จากนั้นก็จะนำค่ากำลังเฉลี่ยของสัญญาณที่หาได้ไปทำการเปรียบเทียบกับเงื่อนไขต่างๆ ที่เราได้กำหนดไว้ ถ้าค่ากำลังเฉลี่ยของสัญญาณ ไปตรงกับเงื่อนไขใดก็จะทำการปรับเลือกจำนวนการสุ่มตัวอย่างของเงื่อนไขนั้นมาใช้ในการวิเคราะห์หาทิศทางการมาถึงของสัญญาณแบบวิธี MUSIC ต่อไปนั่นเอง



รูปที่ 4.9 กราฟค่าความผิดพลาดเฉลี่ยที่ได้จากผลทดสอบวัดจริง เมื่อมีสัญญาณ  
เข้ามาในทิศทาง 30 องศา



รูปที่ 4.10 แผนภาพการทำงานของโปรแกรมในกรณีนำค่ากำลังเฉลี่ยของสัญญาณมาเปรียบเทียบเงื่อนไข



รูปที่ 4.11 การหาค่า  $V_{P_k - P_k}$  ของสัญญาณที่บอร์ดไมโครคอนโทรลเลอร์รับได้จากรูปที่ 4.7

$$dB = 10 \log \left( V_{P_k - P_k} \right)^2 \quad (4.1)$$

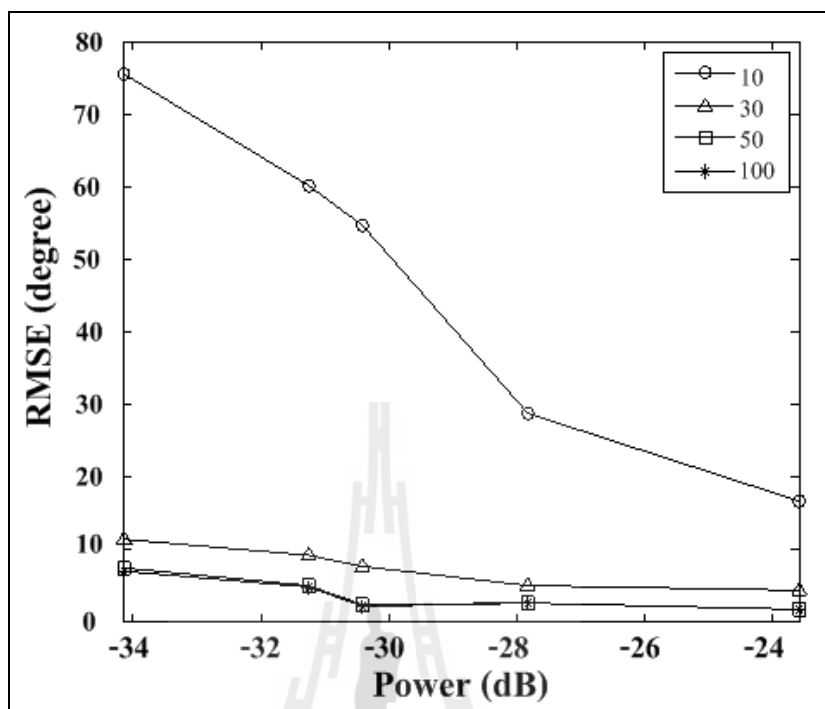
เมื่อ  $V_{P_k - P_k}$  คือ  $V_{max} - V_{min}$

โดยค่ากำลังเฉลี่ยของสัญญาณนั้นเราสามารถหาได้จากสัญญาณที่รับได้จากบอร์ดไมโครคอนโทรลเลอร์ดังรูปที่ 4.7 ซึ่งเราจะทำการหาค่ากำลังของสัญญาณในแต่ละพอร์ตออกมา ก่อน โดยหาจากค่า  $V_{P_k - P_k}$  ของทั้ง 4 พอร์ต คือ พอร์ต1 พอร์ต2 พอร์ต3 และพอร์ต4 แทนด้วยลูกศรสีน้ำเงิน แดง เขียว และเหลืองตามลำดับ แสดงได้ดังรูปที่ 4.11 จากนั้นนำค่า  $V_{P_k - P_k}$  ของทั้ง 4 พอร์ตที่ได้ไปแทนในสมการที่ (4.1) แล้วจึงนำไปหาค่าเฉลี่ยต่อไป และในการทดสอบนั้นเราจะทำการทดสอบทั้งหมด 5 ระยะด้วยกัน คือ 1 2 3 4 และ 5 เมตร ดังนั้นค่ากำลังเฉลี่ยของสัญญาณที่รับได้ในแต่ละระยะก็จะได้ออกมาที่แตกต่างกันออกมามีดังนี้

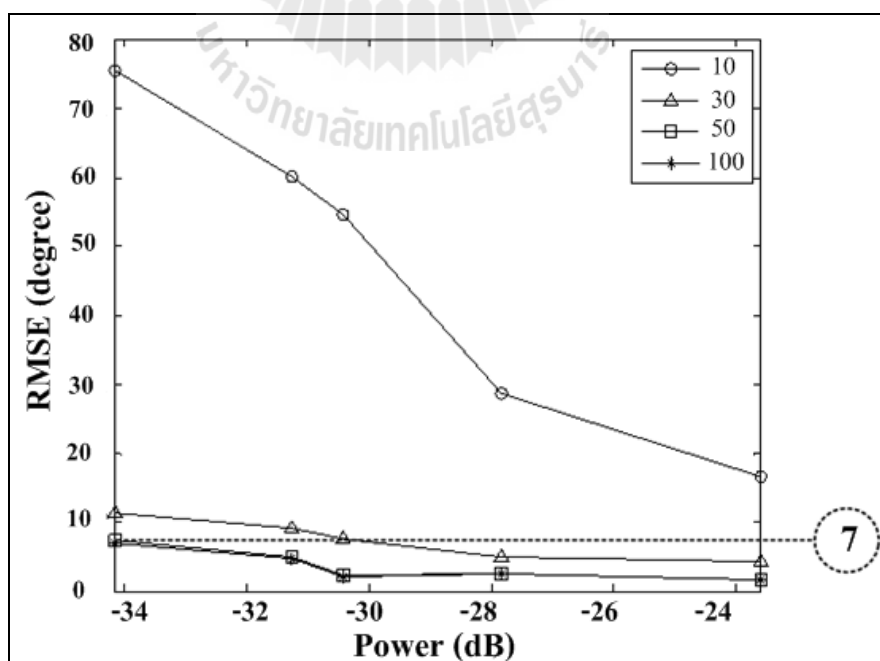
- ที่ระยะ 1 เมตร ได้ค่ากำลังเฉลี่ยของสัญญาณเท่ากับ -23.66 dB
- ที่ระยะ 2 เมตร ได้ค่ากำลังเฉลี่ยของสัญญาณเท่ากับ -27.84 dB
- ที่ระยะ 3 เมตร ได้ค่ากำลังเฉลี่ยของสัญญาณเท่ากับ -30.53 dB
- ที่ระยะ 4 เมตร ได้ค่ากำลังเฉลี่ยของสัญญาณเท่ากับ -31.25 dB
- ที่ระยะ 5 เมตร ได้ค่ากำลังเฉลี่ยของสัญญาณเท่ากับ -34.88 dB

ดังนั้นกราฟที่ได้ดังรูปที่ 4.9 จึงเป็นการเปรียบเทียบระหว่างค่าความผิดพลาดเฉลี่ยของการวิเคราะห์หาทิศทางกับค่ากำลังเฉลี่ยของสัญญาณที่รับได้ในระยะที่ใกล้ขึ้นเรื่อยๆ ซึ่งในอีกความหมายก็คือ ค่ากำลังเฉลี่ยของสัญญาณที่เพิ่มขึ้นเรื่อยๆ นั่นเอง จากนั้นเราก็จะทำการทดสอบวัดจริงเพื่อหาค่าความผิดพลาดเฉลี่ยของการวิเคราะห์หาทิศทางอย่างไรในกรณีของรูปที่ 4.9 ไปทุกๆ ทิศทางตั้งแต่ 0-180 องศาของแต่ละจำนวนการสุ่มตัวอย่าง และทำการเลือกค่าความผิดพลาดเฉลี่ยของทิศทางที่แย่ที่สุดในแต่ละจำนวนการสุ่มตัวอย่างมาพล็อตก็จะได้กราฟออกมาดังรูปที่ 4.12 จากนั้นเราจะทำการลากเส้นกำหนดค่าความผิดพลาดเฉลี่ยดังรูปที่ 4.13 ซึ่งในกรณีนี้เราได้กำหนดให้ระบบมีค่าความผิดพลาดเฉลี่ยให้ไม่เกิน 7 องศา ก็จะได้อะไรออกมาดังตารางที่ 4.2 ดังนั้นเมื่อนำเงื่อนไขที่ได้ไปทำการทดสอบวิเคราะห์หาทิศทางก็จะได้กราฟออกมาดังรูปที่ 4.14 ซึ่งก็คือการปรับเลือกจำนวนการสุ่มตัวอย่างที่ได้จากผลการทดสอบวัดจริง เมื่อมีสัญญาณเข้ามาที่ 30 องศา จะเห็นได้ว่ากราฟเส้นประจะทำการปรับเลือกจำนวนการสุ่มตัวอย่างให้เหมาะสมกับค่ากำลังเฉลี่ยของสัญญาณในระบบที่มีค่าเพิ่มมากขึ้นเรื่อยๆ และจากรูปที่ 4.15 แสดงภาพขยายของกราฟรูปที่ 4.14 ให้เห็นได้ชัดเจนขึ้น ซึ่งกราฟเส้นประที่ได้นั้นก็จะมีค่าความผิดพลาดเฉลี่ยไม่เกิน 7 องศาตามเงื่อนไขที่เราได้ทำการกำหนดไว้

จากในกรณีที่ เราได้ทำการกำหนดเงื่อนไขค่าความผิดพลาดเฉลี่ยจะเห็นได้ว่า จากผลที่ได้จากการจำลองแบบในหัวข้อที่ 2.11 เราได้ทำการกำหนดกราฟกรณีที่แย่ที่สุดดังรูปที่ 2.34 ให้มีค่าความผิดพลาดเฉลี่ยไม่เกิน 1 องศา แต่ในกราฟของกรณีที่แย่ที่สุดที่ได้จากผลการทดสอบวัดจริงนั้น เราได้ทำการกำหนดให้มีค่าความผิดพลาดเฉลี่ยไม่เกิน 7 องศา ซึ่งความแตกต่างของการกำหนดเงื่อนไขค่าความผิดพลาดเฉลี่ยที่เห็นนั้น ในความเป็นจริงเราจะกำหนดเท่าไรก็ได้ซึ่งก็ขึ้นอยู่กับผู้ที่ทำการทดสอบ ถ้าต้องการให้ระบบใช้จำนวนการสุ่มตัวอย่างที่น้อยลง ค่าความผิดพลาดเฉลี่ยที่ได้ก็จะมีค่าสูงขึ้น แต่ในขณะเดียวกันถ้าต้องการให้ระบบมีค่าความผิดพลาดเฉลี่ยที่น้อย จำนวนการสุ่มตัวอย่างที่ใช้ก็จะมากขึ้น ซึ่งก็แล้วแต่ผู้ทำการทดสอบว่าต้องการเลือกแบบใด



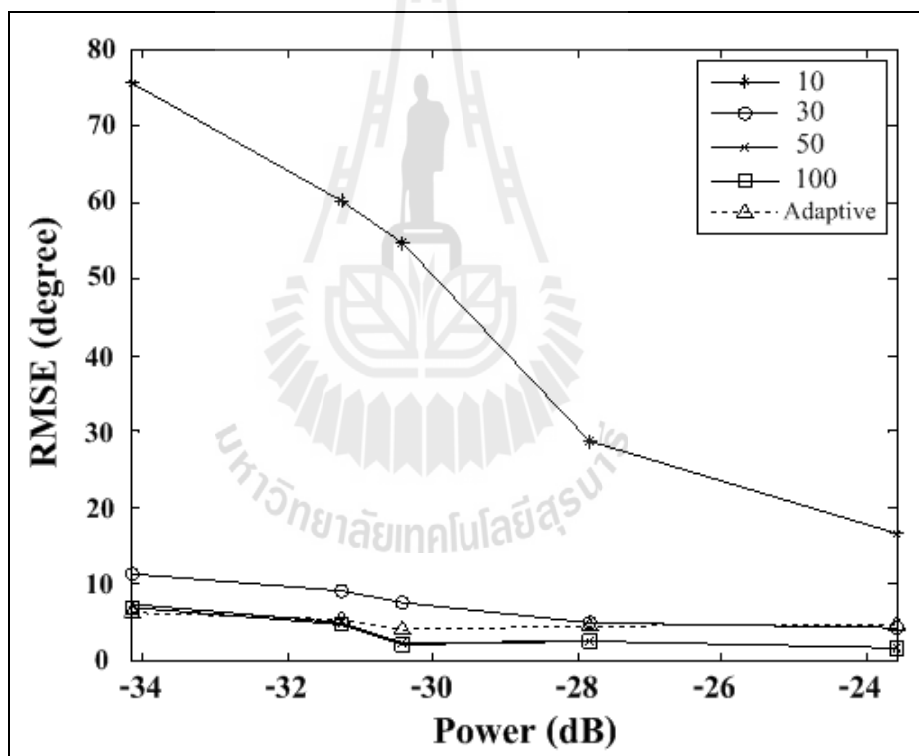
รูปที่ 4.12 กราฟค่าความผิดพลาดเฉลี่ยที่ได้จากผลทดสอบวัดจริง ในกรณีแย้สุดของแต่ละจำนวนการสุ่มตัวอย่าง



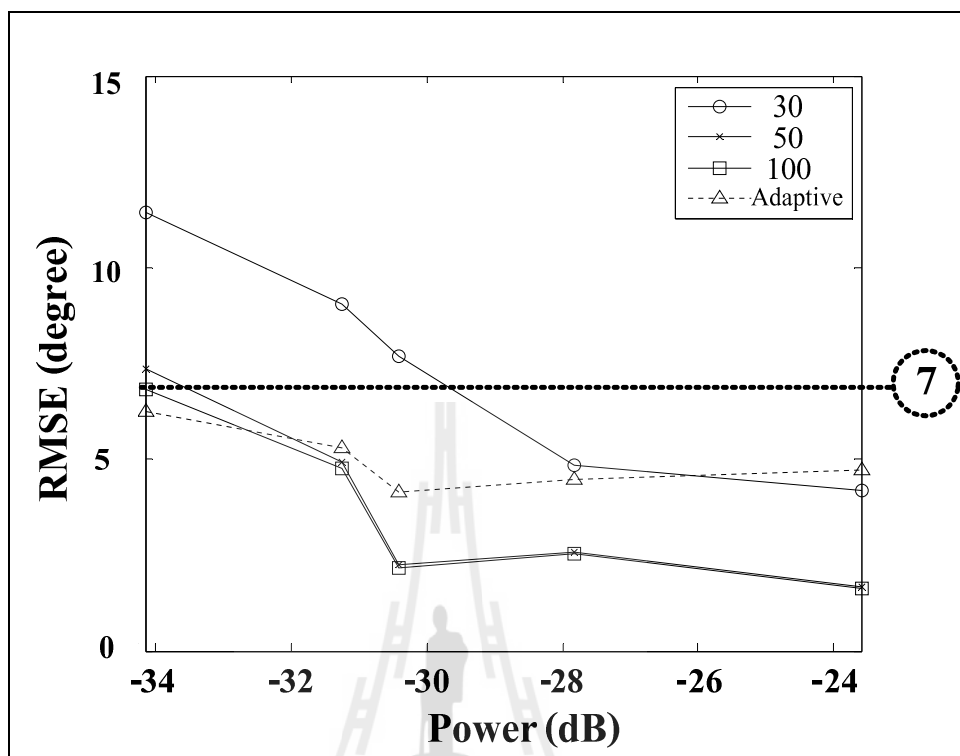
รูปที่ 4.13 การกำหนดค่าความผิดพลาดเฉลี่ยให้ไม่เกิน 7 องศา

ตารางที่ 4.2 เงื่อนไขการปรับจำนวนการสุ่มตัวอย่างที่ให้ความผิดพลาดไม่เกิน 7 องศา

กำลังเฉลี่ย (dB)	จำนวนการสุ่มตัวอย่าง
$\geq (-33.7)$	100
$(-33.7) - (-29.8)$	50
$(-29.8) - (-23.59)$	30



รูปที่ 4.14 การปรับเลือกจำนวนการสุ่มตัวอย่างที่ได้จากผลวัดจริง เมื่อมีสัญญาณเข้ามาที่ 30 องศา



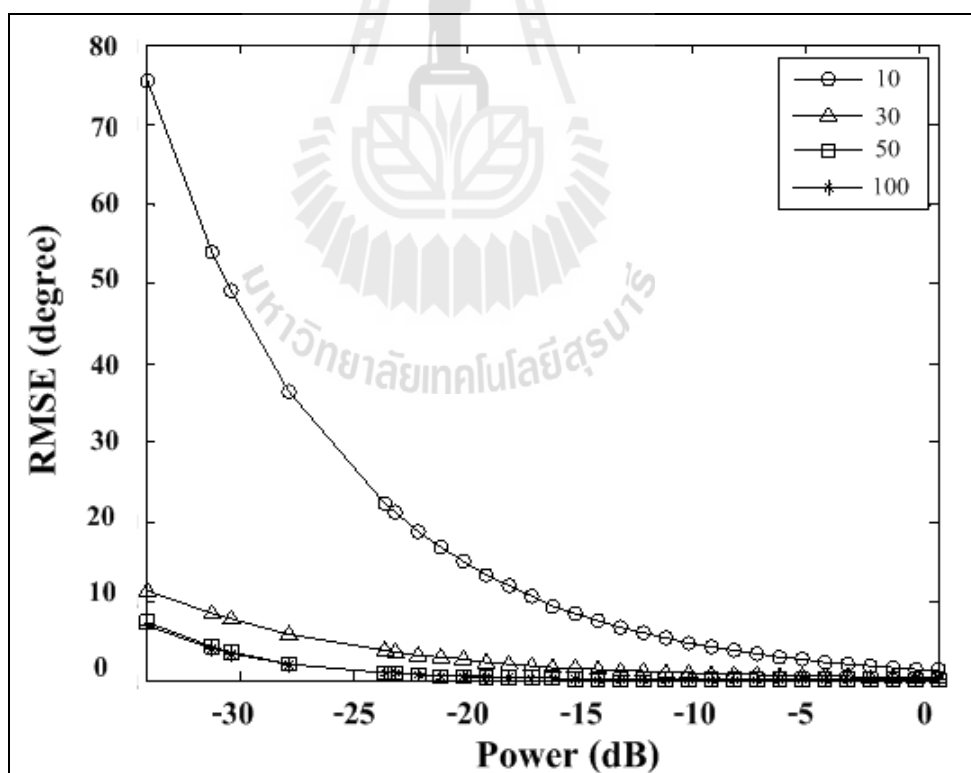
รูปที่ 4.15 ภาพขยายของรูปที่ 4.14

แต่อย่างไรก็ตามกราฟของกรณีที่แย่ที่สุดที่ได้ผลจากการทดสอบวัดจริงดังรูปที่ 4.12 นั้นก็ยังคงแสดงความสัมพันธ์ได้ไม่ชัดเจน เมื่อเราต้องการที่จะนำไปเปรียบเทียบกับกราฟที่ได้มาจากการจำลองแบบดังรูปที่ 2.34 เนื่องจากเป็นเพราะข้อจำกัดของชุดอุปกรณ์หาทิศทางก็คือ ถ้าเราทำการทดสอบไกลกว่าระยะ 5 เมตร ชุดอุปกรณ์หาทิศทางก็จะไม่สามารถรับสัญญาณได้ แต่ในขณะเดียวกันถ้าเราทำการทดสอบใกล้กว่าระยะ 1 เมตร ชุดอุปกรณ์หาทิศทางก็จะซัดกับภาคส่งจนเกินไป ดังนั้นถ้าเราต้องการเปรียบเทียบกับกราฟที่ได้มาจากการจำลองแบบ เราจึงต้องประมาณกราฟขึ้นมาดังรูปที่ 4.16 ซึ่งแสดงค่าความผิดพลาดเฉลี่ยของการวิเคราะห์หาทิศทางในกรณีที่แย่ที่สุดที่ได้จากผลทดสอบวัดจริง เมื่อเราทำการประมาณกราฟในรูปแบบของเอ็กซ์โปเนนเชียล (exponential) จะเห็นได้ว่าเมื่อระบบมีค่ากำลังเฉลี่ยของสัญญาณที่เพิ่มขึ้นเรื่อยๆ กราฟที่ได้ก็จะมีค่าความผิดพลาดเฉลี่ยของแต่ละจำนวนการสุ่มตัวอย่างที่น้อยลงตามไปด้วย ซึ่งก็คล้ายคลึงกับกรณีของกราฟที่ได้มาจากการจำลองแบบ

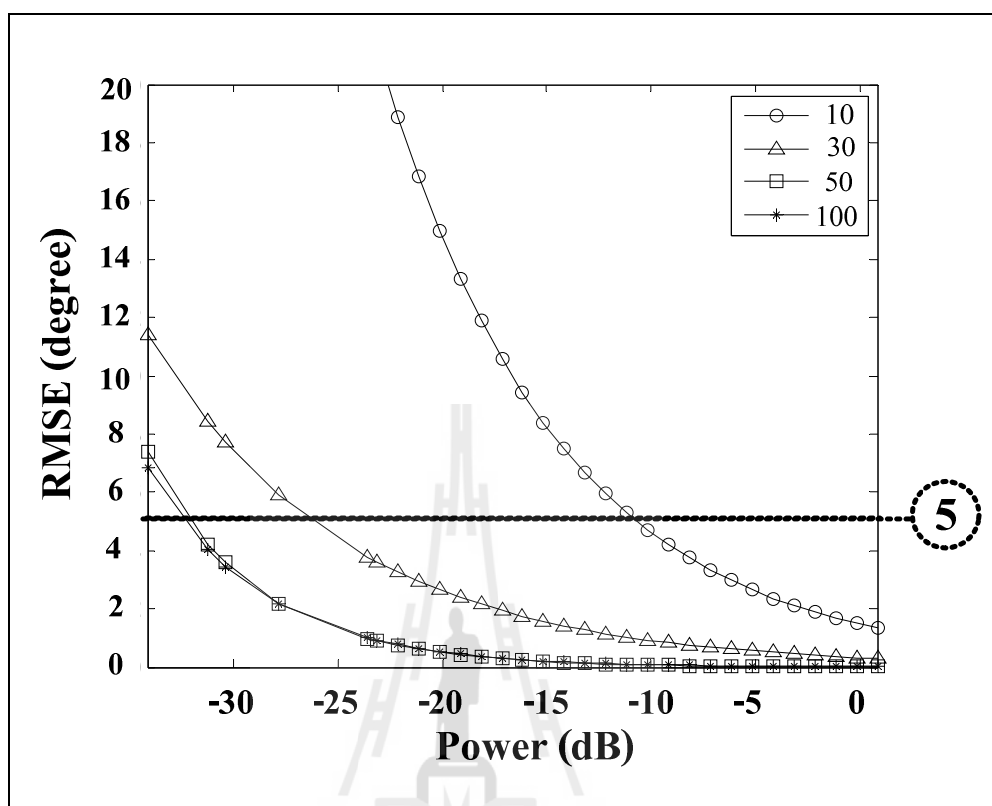
จากนั้นเราจึงได้ทำการขยายภาพของรูปที่ 4.16 เพื่อให้เห็นกราฟได้ชัดเจนขึ้นดังรูปที่ 4.17 และทำการกำหนดค่าความผิดพลาดเฉลี่ยของการวิเคราะห์หาทิศทางขึ้น ซึ่งในที่นี้เราได้ทำการ



กำหนดให้ระบบมีค่าความผิดพลาดเฉลี่ยให้ไม่เกิน 5 องศา ก็จะได้เงื่อนไขออกมาดังตารางที่ 4.3 ดังนั้นเมื่อนำเงื่อนไขดังกล่าวไปทำการทดสอบ โดยเราจะทำการจำลองติดตั้งภาคส่งให้อยู่ในทิศทางต่างๆ และมีระยะที่แตกต่างกันออกไปดังรูปที่ 4.18 ผลที่ได้จากการทดสอบก็จะแสดงให้เห็นดังตารางที่ 4.4 จะเห็นได้ว่าระบบมีการปรับเลือกจำนวนการสุ่มตัวอย่างให้เหมาะสมกับค่ากำลังเฉลี่ยของสัญญาณที่แปรเปลี่ยนไป เช่น ที่ระยะ 1 เมตรก็แสดงว่าระบบค่ากำลังเฉลี่ยของสัญญาณที่สูง ดังนั้นก็จะทำการปรับเลือกจำนวนการสุ่มตัวอย่างที่น้อยลงซึ่งในที่นี้ก็คือ 30 แต่ในขณะเดียวกันที่ระยะ 5 เมตรก็แสดงว่าระบบค่ากำลังเฉลี่ยของสัญญาณที่ต่ำ ดังนั้นก็จะทำการปรับเลือกจำนวนการสุ่มตัวอย่างที่มากขึ้นซึ่งในที่นี้ก็คือ 100 เป็นต้น จากนั้นเมื่อเราทำการหาค่าความผิดพลาดเฉลี่ยของการวิเคราะห์หาทิศทางของทั้ง 10 กรณีในตารางที่ 4.4 แล้วนำไปพล็อตก็จะได้กราฟออกมาดังรูปที่ 4.19 และกราฟที่ได้นั้นก็จะมีค่าความผิดพลาดเฉลี่ยไม่เกิน 5 องศาตามเงื่อนไขที่เราได้ทำการกำหนดไว้อีกด้วย



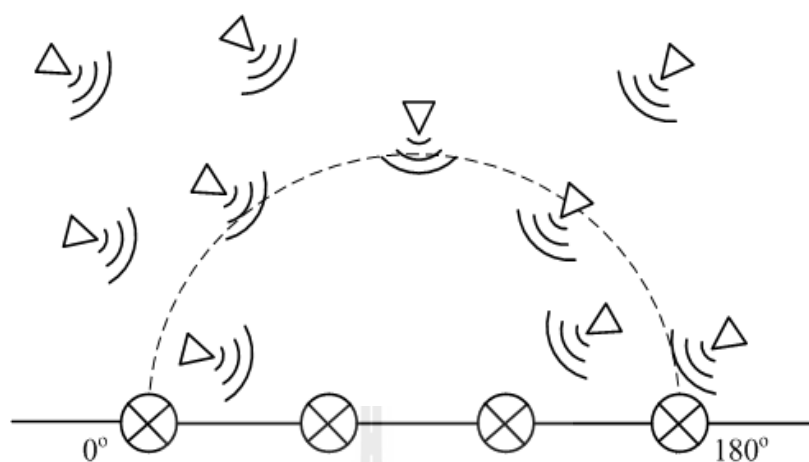
รูปที่ 4.16 กราฟค่าความผิดพลาดเฉลี่ยในกรณีแย่สุดที่ได้จากการวัดจริง เมื่อทำการประมาณกราฟในรูปแบบของเอ็กซ์โปเนนเชียล



รูปที่ 4.17 ภาพขยายกราฟค่าความผิดพลาดเฉลี่ยที่ได้จากผลวัดจริงของรูปที่ 4.14

ตารางที่ 4.3 เงื่อนไขการปรับจำนวนการสุ่มตัวอย่างที่ให้ความผิดพลาดไม่เกิน 5

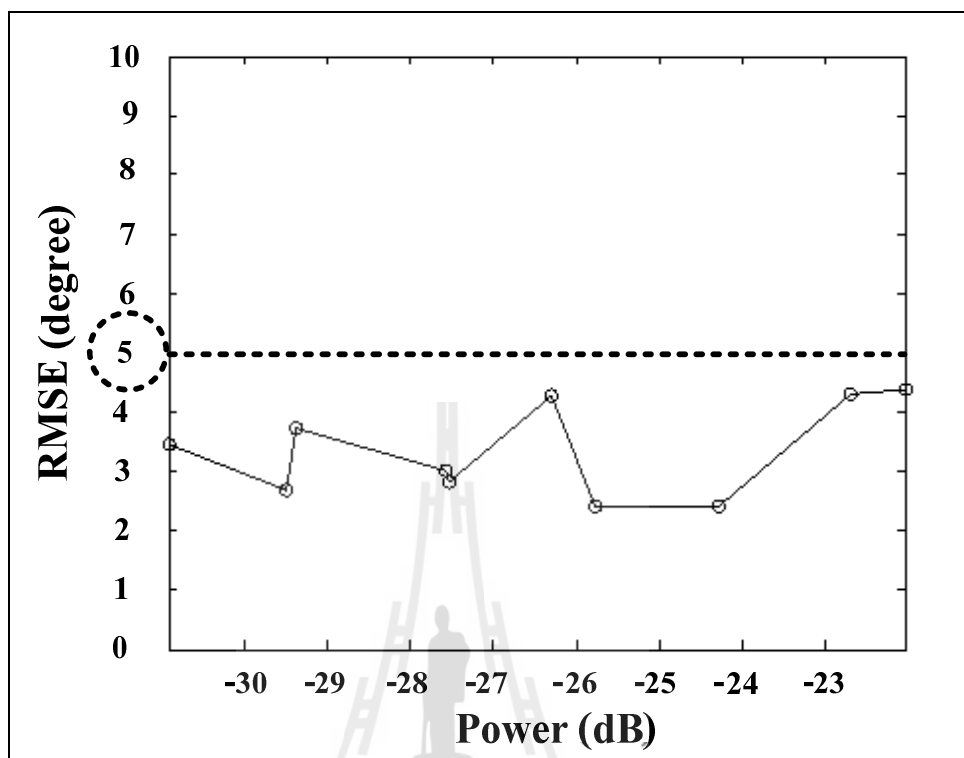
กำลังเฉลี่ย (dB)	จำนวนการสุ่มตัวอย่าง
$\geq (-31.350)$	100
$(-31.350) - (-25.715)$	50
$(-25.715) - (-10.660)$	30
$(-10.660) - (0.842)$	10



รูปที่ 4.18 แบบจำลองทิศทางของสัญญาณจากภาคส่งที่ตั้งอยู่ในระยะต่างๆ

ตารางที่ 4.4 ผลการทดสอบหาทิศทางด้วยเทคนิคการปรับจำนวนการสุ่มตัวอย่าง

ทิศทาง (°)	ระยะ (เมตร)	จำนวนการสุ่มตัวอย่าง
15	1	30
35	3	50
60	2	30
60	5	100
70	4	50
90	3	50
120	2	30
120	4	50
130	1	30
160	1	30



รูปที่ 4.19 การปรับเลือกจำนวนการสุ่มตัวอย่างที่ได้จากผลวัดจริง เมื่อมีสัญญาณเข้ามาในทิศทางต่างๆ ที่ได้จากรายที่ 4.4



## บทที่ 5

### สรุปการวิจัยและข้อเสนอแนะ

#### 5.1 สรุปเนื้อหางานวิจัย

ในปัจจุบันระบบสายอากาศเก่งได้ถูกนำมาใช้ในเครือข่ายไร้สายอย่างแพร่หลาย เนื่องจากความสามารถในการหันลำคลื่นหลักไปในทิศทางที่ต้องการ ในขณะที่เดียวกันก็จะทำการหันจุดศูนย์หรือพุ่งไปในทิศทางที่เราไม่ต้องการ กระบวนการดังกล่าวเรียกว่า การก่อรูปลำคลื่น ซึ่งจะประกอบไปด้วย 2 ส่วนหลักๆ คือ สายอากาศแถวลำดับ และตัวประมวลผลสัญญาณ ซึ่งจะทำหน้าที่ในการวิเคราะห์หาทิศทางการมาถึงของสัญญาณ โดยวิธีที่เป็นที่นิยมมากที่สุดคือ วิธี MUSIC เนื่องจากเป็นวิธีที่ไม่ซับซ้อนและสามารถหาทิศทางได้อย่างถูกต้องและแม่นยำ อย่างไรก็ตามวิธี MUSIC นั้นก็ยังคงมีปัญหาอยู่หากระบบมีค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวนที่ต่ำ ประสิทธิภาพในการวิเคราะห์หาทิศทางการมาถึงของสัญญาณก็จะลดต่ำลงไปด้วย ดังนั้นในการแก้ปัญหาจำเป็นจะต้องเพิ่มจำนวนการสุ่มตัวอย่างให้เหมาะสมกับค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวน โดยในการเพิ่มจำนวนการสุ่มตัวอย่างนั้นถ้าหากเพิ่มน้อยเกินไป ระบบก็จะไม่สามารถวิเคราะห์หาทิศทางได้ถูกต้องและแม่นยำ แต่ในทางตรงกันข้ามถ้าเพิ่มจำนวนการสุ่มตัวอย่างมากเกินไป ก็จะทำให้ตัวประมวลผลในระบบสามารถประมวลผลได้ช้าลง ส่งผลให้ซีพียูของตัวประมวลผลทำงานหนักขึ้นตามไปด้วย ดังนั้นงานวิจัยฉบับนี้จึงได้มีการนำเสนอเทคนิคการปรับจำนวนการสุ่มตัวอย่างมาประยุกต์ใช้กับวิธี MUSIC ในการปรับเลือกจำนวนการสุ่มตัวอย่างให้เหมาะสมกับค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวนในระบบ เพื่อทำการวิเคราะห์หาทิศทางการมาถึงของสัญญาณให้ได้อย่างถูกต้องและแม่นยำ นอกจากนี้ยังส่งผลให้ระบบมีการประมวลผลที่รวดเร็วขึ้น ถือเป็นภาระลดน้อยซีพียูของตัวประมวลผลไม่ทำให้ทำงานหนักอีกด้วย ดังนั้นชุดอุปกรณ์ต้นแบบในงานวิจัยฉบับนี้จึงได้นำเทคนิคการปรับจำนวนการสุ่มตัวอย่างมาประยุกต์ใช้ในการทดสอบหาทิศทาง ชุดอุปกรณ์ต้นแบบจะประกอบไปด้วย สายอากาศแถวลำดับเชิงเส้นแบบ 4×1 วงจรขยายสัญญาณรบกวนต่ำ วงจรแปลงสัญญาณความถี่ต่ำ รวมไปถึงการสร้างโปรแกรมเพื่อควบคุมบอร์ดไมโครคอนโทรลเลอร์ในการรับข้อมูลอนาล็อกและทำการเปลี่ยนเป็นข้อมูลดิจิทัล ซึ่งจะใช้โปรแกรมแมทแลบรับข้อมูลดิจิทัลดังกล่าวเพื่อทำการประมวลผลหาทิศทางแบบวิธี MUSIC ต่อไป ซึ่งจากผลได้ทำการนำเสนอสามารถสรุปได้ว่า ชุดอุปกรณ์ต้นแบบสามารถวิเคราะห์

หาทิศทางได้อย่างถูกต้องและแม่นยำ อีกทั้งเทคนิคการปรับจำนวนการสุ่มตัวอย่างยังสามารถทำการปรับเลือกจำนวนการสุ่มตัวอย่างให้เหมาะสมกับค่ากำลังเฉลี่ยของระบบ โดยมีค่าผิดพลาดของการหาทิศทางไม่เกินเงื่อนไขที่เรากำหนดไว้ได้อีกด้วย

## 5.2 ปัญหาและข้อเสนอแนะ

- เนื่องจากบอร์ดไมโครคอนโทรลเลอร์ที่ใช้ในงานวิจัยนี้มีอัตราเร็วในการสุ่มตัวอย่างที่ค่อนข้างต่ำ งานวิจัยนี้จึงจำเป็นต้องแปลงสัญญาณความถี่ที่ต่ำมาก ซึ่งมีผลต่อการออกแบบวงจรแปลงสัญญาณความถี่ต่ำทำให้สัญญาณที่ได้จะไม่นิ่งเท่าที่ควร ดังนั้นในการวิเคราะห์หาทิศทางอาจเกิดความคลาดเคลื่อนไปบ้าง

- ควรจะเพิ่มการพัฒนาเพื่อให้สามารถใช้งานได้ในระยะที่ไกลขึ้นเพื่อตอบสนองการใช้งานจริงของระบบสื่อสารไร้สาย

- ควรจะพัฒนาความสามารถของระบบให้สามารถหาทิศทางได้มากขึ้นเช่น 2 และ 3 ทิศทางขึ้นไป

- ควรเพิ่มความหลากหลายของชนิดของสายอากาศเพื่อตรวจสอบว่า ชนิดของการแผ่กระจายคลื่นมีผลต่อการทิศทางของระบบอย่างไร

## 5.3 แนวทางการพัฒนาในอนาคต

เนื่องจากชุดอุปกรณ์การหาทิศทางในงานวิจัยนี้เป็นเพียงอุปกรณ์ต้นแบบ อาจจะมีข้อจำกัดในเรื่องของระยะในการรับสัญญาณ และสามารถหาทิศทางได้เพียงทิศทางเดียว ดังนั้นเมื่อมีการพัฒนาชุดอุปกรณ์ฮาร์ดแวร์ของระบบ รวมไปถึงตัวประมวลผลสัญญาณให้มีประสิทธิภาพที่สูงขึ้น ก็อาจจะส่งผลให้ระบบสามารถหาทิศทางได้ในระยะที่ไกลขึ้นและจำนวนของทิศทางที่หาได้ก็จะเพิ่มมากขึ้น

## รายการอ้างอิง

- Wei, Z., and Xiaoli, X. (2007). **Analysis and simulation of the direction of arrival estimation algorithm of spatial signal**. ICEMI'2007.
- Ye, Z., Xiang, L., and Xu, X. (2007). **DOA estimation with circular array via spatial averaging algorithm**. IEEE Antennas and Wireless Propagation Letters, vol. 6, 2007.
- Kaveh, M., and Barabell, A.J. (1986). **The statistical performance of the MUSIC and the minimum-norm algorithms in resolving plane waves in noise**. IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. ASSP-34, No. 2, April 1986.
- Xin, J., and Sano, A. (2004). **Computationally efficient subspace-based method for direction-of-arrival estimation without eigendecomposition**. IEEE Transactions on Signal Processing, vol. 52, No. 4, April 2004.
- ลัญฉกร วุฒิสัทติกุลกิจ. (2542). **หลักการระบบโทรศัพท์เคลื่อนที่**. กรุงเทพฯ: จุฬาลงกรณ์มหาวิทยาลัย.
- สุขสันต์ เรือนแก้ว. (2548). **เทคโนโลยีไร้สายทำงานอย่างไร**. กรุงเทพฯ: ซีเอ็ดยูเคชั่น.
- Frank, B.G., Ph.D. (2005). **Smart Antennas for Wireless Communication With MATLAB**. Fairfax, Virginia, 2005.
- Allen, B., and Ghavami, M. (2005). **adaptive array system: Fundamentals and Applications**. John Wiley & Sons Ltd.
- จารุพงศ์ จีนาพันธ์. (2551). **Electronics Handbook ฉบับที่ 141 พฤษภาคม 2551**. กรุงเทพฯ: สถาบันอิเล็กทรอนิกส์กรุงเทพรังสิต.
- Lutovac, and Miroslav, D. (2001). **Filter design for signal processing using MATLAB and Mathematica**. Upper Saddle River, NJ: Prentice-Hall.
- Schanmann, and Rolf. (1941). **Design of analog filters: passive, active RC and switched capacitor**. Englewood Cliffs, NJ: Prentice-Hall.
- Graeme, and Jerald, G. (1999). **Amplifier applications of Op Amps**. New York: McGraw Hill



ปฏิบัติการ การวิเคราะห์วงจรไฟฟ้า 1. ออปแอมป์. ภาควิชาวิศวกรรมอิเล็กทรอนิกส์ มหาวิทยาลัยเทคโนโลยีมหานคร.

ธนวัฒน์ ศรีสอ้าน, และสุทธิชัย มณีรัตน์รุ่งโรจน์. (2542). การสื่อสารทางไกลและเครือข่าย. เอกสารประกอบการสอน สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี

Waggener, and Bill. (1995). **Pulse code modulation techniques with applications in communications and data recording.** New York: Van Nostrand Rein hold.

Alexiou, A., and Haardt, M. (2004). **Smart antenna technologies for future wireless systems: trends and challenges.** IEEE Communication Magazine, vol. 42, pp. 90-97.

Wintersand, J.H. (2004). **Smart antenna technologies and their application to wireless ad hoc networks.** IEEE Wireless Communications, August 2004.

Peng, M., and Wang, W. (2005). **Comparison of Capacity between Adaptive Tracking and Switched Beam Smart Antenna Techniques in TDD- CDMA Systems.** Microwave Antenna, Propagation and EMC Technologies for Wireless Communications, 2005. (MAPE 2005). IEEE International Symposium on Vol. 1, pp. 135-139.

Seungwon, C., Shim, D., and Sarkar, D. K. (1999). **A Coparison of Tracking- Beam Arrays and Swiching- Beam Arrays Operating in a CDMA Mobile Communication Channel.** IEEE Transactions on Antenna and Propagation, Mag., Vol. 41, pp. 10-56.

Agee, B. (1989). **Blind Seperation and Capture of Communications Signals Using a Multitarget Constant Modulus Beamformer.** 1989 IEEE Military Communications Conf., Boston, MA, Oct. 1989.

Frost, L. III (1972). **An Algorithm for Linearly Constrained Adaptive Array Processing.** Proc. Of the IEEE, Vol. 60, No. 8, Aug. 1972.

Liberti, Jr. J. C., and Rappaport, T. S. (1999). **Smart Antenna for Wireless Communications: IS-95 and Third Generation CDMA Applications.** NJ: Prentice-Hal.

Bo, W. (2006). **Realization and simulation of DOA estimation using MUSIC algorithm with uniform circular arrays.** CEEM'2006.

Hong, Z. (2005). **MUSIC Algorithm and Direction of Arrival Estimation**. Sichuan University, 04, 2005.



ภาคผนวก ก

บทความวิชาการที่ได้รับการตีพิมพ์เผยแพร่



**บทความทางวิชาการที่ได้รับการตีพิมพ์เผยแพร่ในวารสารระดับชาติ**

P. Chaiyaserm, M. Uthansakul and N. Seman, “Adaptive snapshot technique for DOA finder”,  
to be published in Suranaree Journal of Science and Technology, 2014.



ภาคผนวก ข

โปรแกรมของบอร์ดไมโครคอนโทรลเลอร์



## ข.1 โปรแกรมภาษาซีที่ได้ทำการออกแบบ

```

/*****/;
/* Hardware      : ET-AVR STAMP ATmega128      */;
/* CPU           : ATMEL-ATmega128            */;
/* X-TAL         : 16.00 MHz                  */;
/* Compiler      : CodeVisionAVR V1.24.7e     */;
/* Last Update   : 10-04-2006 (ETT CO.,LTD)   */;
/*              : WWW.ETT.CO.TH              */;
/* Description   : Demo ADC Channel 0        */;
/*              : Setup RS232 = 57600,N,8,1   */;
/*****/;
/* CodeVisionAVR Compiler Option Setting      */;
/* Chip type    : ATmega128                  */;
/* Program type : Application                */;
/* Clock frequency : 16.000000 MHz          */;
/* Memory model  : Small                    */;
/* External SRAM size : 0                   */;
/* Data Stack size : 1024                   */;
/*****/;
#include <mega128.h>                // ATmega128 MCU
#include <stdio.h>                  // Standard Input / Output functions
#define ADC_VREF_TYPE 0xC0         // ADC Used Internal Reference
#include <delay.h>
#include <string.h>
/* prototype section */
unsigned int read_adc(unsigned char adc_input); // Read ADC Result
void main(void)
{
    unsigned int i,j,num=300;

```

```
char command;
unsigned int i_adc1[301], i_adc2[301], i_adc3[301], i_adc4[301] ;

// USART initialization
// Communication Parameters: 8 Data, 1 Stop, No Parity
// USART Receiver: On
// USART Transmitter: On
// USART Mode: Asynchronous
// USART Baud rate: 57600
UCSR0A=0x00;
UCSR0B=0x18;
UCSR0C=0x86;
UBRR0H=0x00;
UBRR0L=0x10;

// ADC initialization
// ADC Clock frequency: 125 kHz
// ADC Voltage Reference: Int., cap. on AREF
// ADC High Speed Mode: Off
// ADC Auto Trigger Source: None

ADMUX=ADC_VREF_TYPE;
ADCSRA=0x87;
SFIO&=0xEF;

while (1) //Loop Continue
{ //delay_ms (200);
    command = getchar ( );
```

```

if (command == 0x61)
{
    command=0;
    for(i=0;i<num;i++){
        i_adc1[i]=read_adc(0);
        i_adc2[i]=read_adc(1);
        i_adc3[i]=read_adc(2);
        i_adc4[i]=read_adc(3); }

    for(j=0; j<num; j++) {
        delay_ms(10);
        if(i_adc1[j] <= 0x0F){
            printf("00%X",i_adc1[j]);
        }
        else if(i_adc1[j] <= 0xFF){
            printf("0%X",i_adc1[j]);
        }
        else {
            printf("%X",i_adc1[j]);
        }

        command = getchar( );
    }

    command = getchar();
    for(j=0; j<num; j++) {
        delay_ms(10);
        if(i_adc2[j] <= 0x0F){
            printf("00%X",i_adc2[j]);
        }
        else if(i_adc2[j] <= 0xFF){

```



```
        printf("0%X",i_adc2[j]);
    }
    else {
        printf("%X",i_adc2[j]);
    }
    command = getchar();
}

    command = getchar();
for(j=0; j<num; j++) {
    delay_ms(10);
    if(i_adc3[j] <= 0x0F){
        printf("00%X",i_adc3[j]);
    }
    else if(i_adc3[j] <= 0xFF){
        printf("0%X",i_adc3[j]);
    }
    else {
        printf("%X",i_adc3[j]);
    }
    command = getchar();
}

    command = getchar();
for(j=0; j<num; j++) {
    delay_ms(10);
    if(i_adc4[j] <= 0x0F){
        printf("00%X",i_adc4[j]);
    }
}
```

```

        else if(i_adc4[j] <= 0xFF){
            printf("0%X",i_adc4[j]);
        }
        else {
            printf("%X",i_adc4[j]);
        }
        command = getchar();
    }
}
else{ }
}
}
/*****/;
/* Read the AD conversion result */;
/*****/;
unsigned int read_adc(unsigned char adc_input) // Read Result ADC
{
    ADMUX=adc_input|ADC_VREF_TYPE;
    ADCSRA|=0x40; // Start the AD conversion
    while ((ADCSRA & 0x10)==0); // Wait for the AD conversion to complete
    ADCSRA|=0x10;
    return ADCW;
}

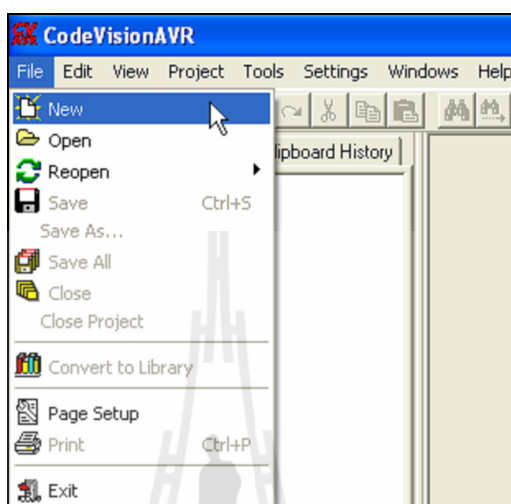
```

## ข.2 การพัฒนาโปรแกรมด้วย CodeVisionAVR

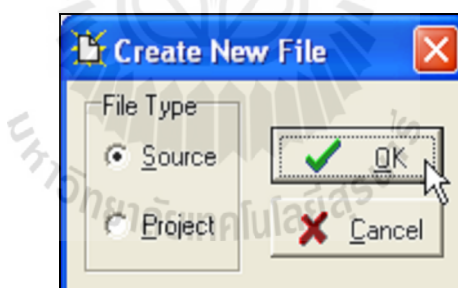
สำหรับโปรแกรมที่ใช้ในการเขียนโปรแกรมภาษาซี ก็คือโปรแกรม CodeVisionAVR เวอร์ชัน 1.24.7e ซึ่งขั้นตอนการใช้งานโปรแกรมมีดังนี้

1. เปิดโปรแกรม CodeVisionAVR C compiler และคลิกเลือกที่เมนูคำสั่ง File => New ดัง

รูป



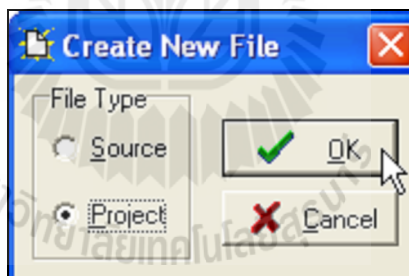
2. เลือก file type เป็น Source เพื่อสร้างไฟล์ภาษาซีใหม่และคลิกปุ่ม OK ดังรูป



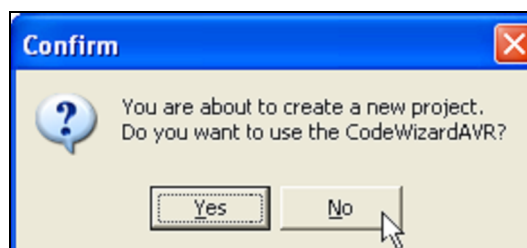
3. จากนั้นจะปรากฏหน้าต่าง Editor ให้ทำการเขียนโปรแกรมจากข้างต้น
4. ทำการบันทึกโปรแกรมภาษาซีที่เขียนโดยเลือกเมนู File => Save ทำการตั้งชื่อไฟล์ และกดปุ่ม Save ดังรูป



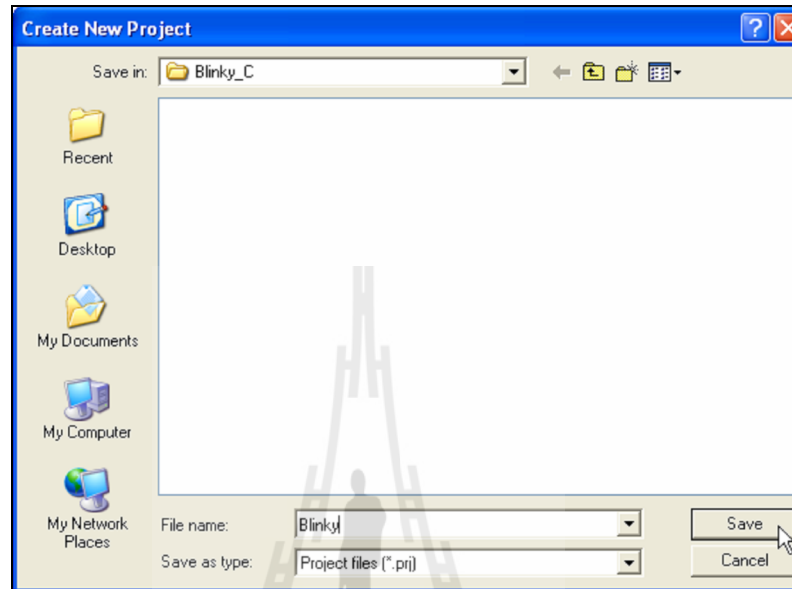
5. เลือกที่เมนู File => New และเลือก file type เป็น Project เพื่อสร้างโปรเจกใหม่และคลิกปุ่มOK ดังรูป



6. คลิกปุ่ม No เพื่อไม่ใช่ตัวช่วยในการสร้างโปรเจก (code wizard)

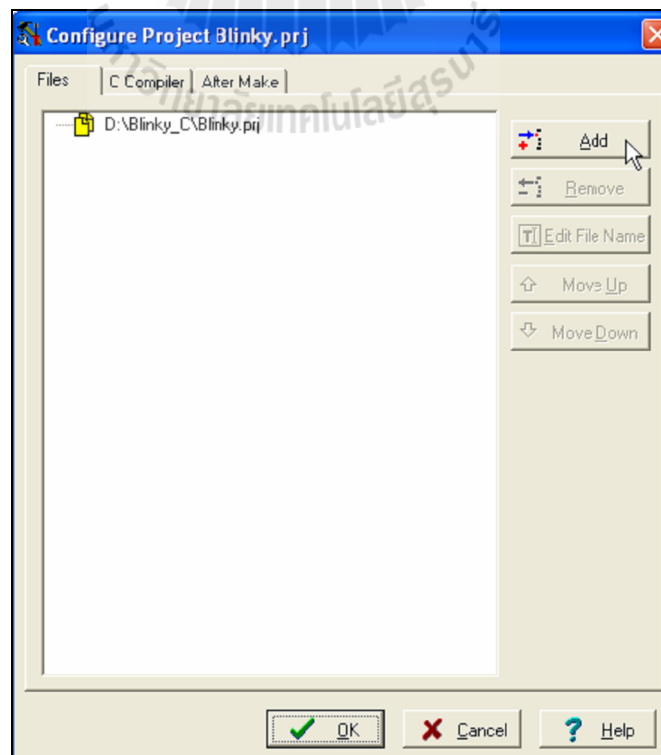


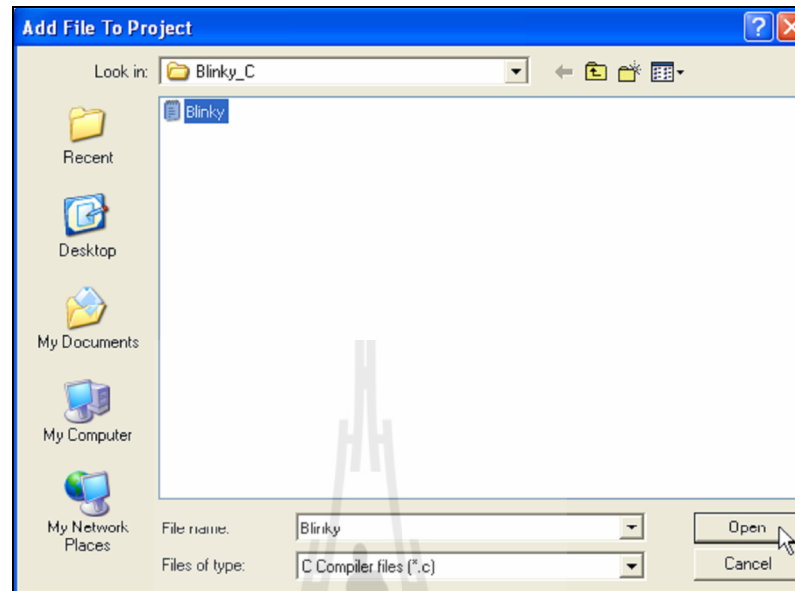
7. ทำการตั้งชื่อโปรเจกตามต้องการและคลิกปุ่ม Save ดังรูป



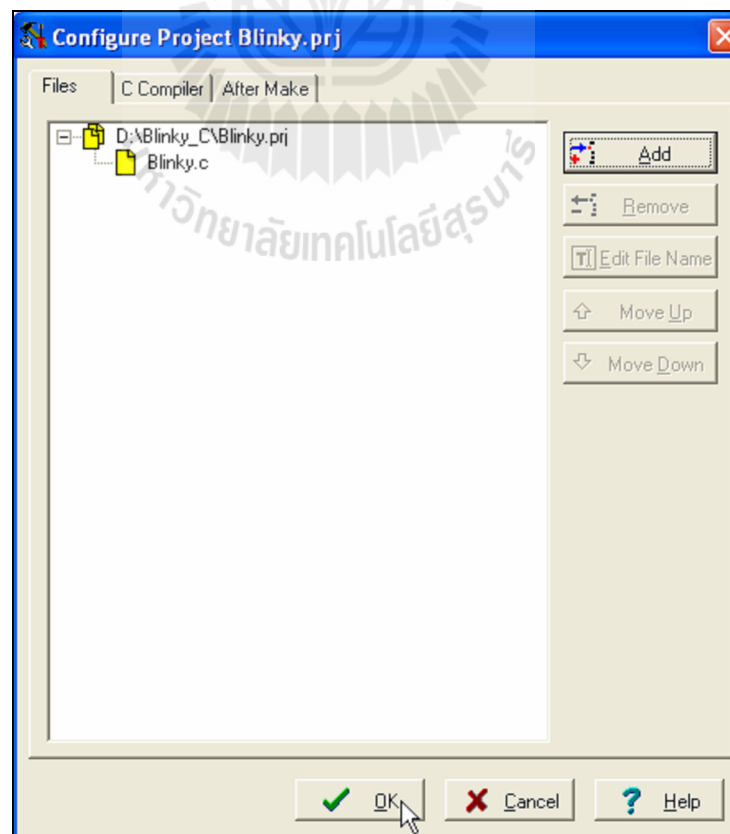
8. ทำการเพิ่มไฟล์ภาษาซีที่เขียนไว้ก่อนหน้านี้เข้ามาในโปรเจก โดยการคลิกปุ่ม Add ดัง

รูป

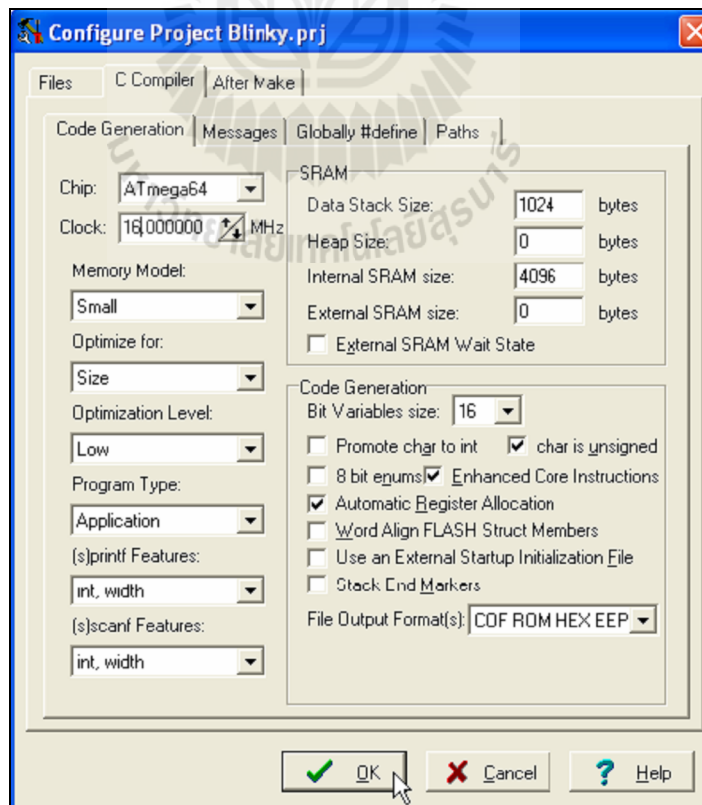
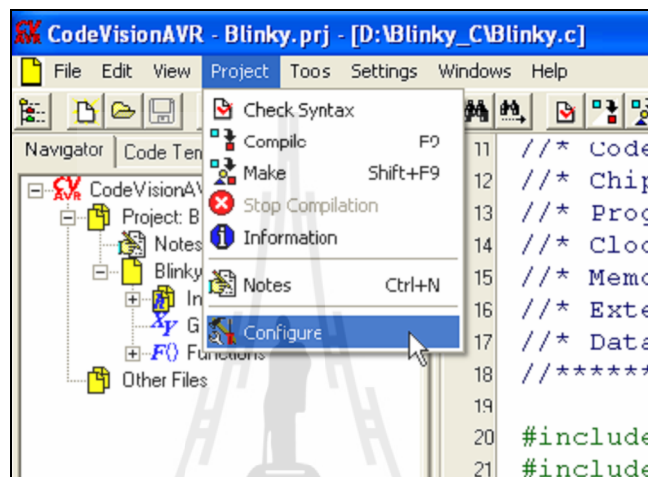




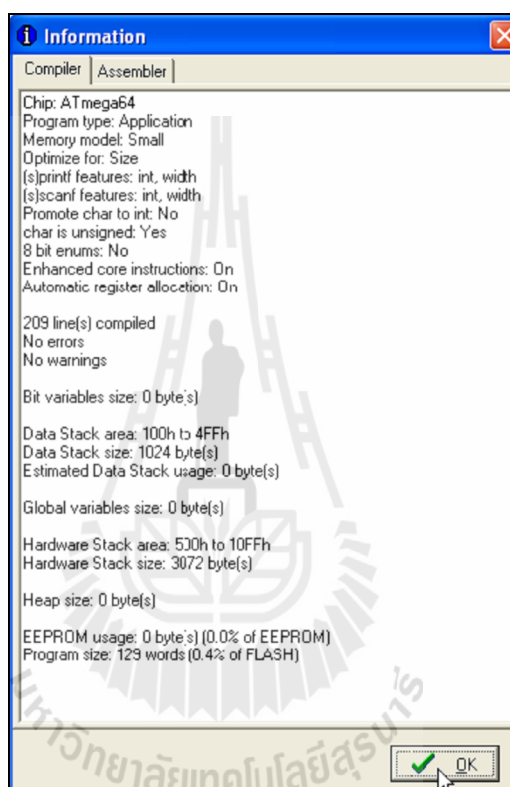
9. เมื่อทุกอย่างเรียบร้อยคลิกปุ่ม OK ดังรูป



10. ทำการตั้งค่าต่างๆ ของโปรเจกต์โดยการคลิกเมาส์ที่เมนูคำสั่ง Project => Configure จากนั้น ทำการกำหนดบอร์ด MCU เป็น ATmega64 ค่าคริสตอลเท่ากับ 16.000000 MHz และ File Output Format(s) เป็น COF ROM HEX EEP



11. ให้ทำการสั่งแปลโปรแกรมที่เราเขียนขึ้น โดยการคลิกเมาส์ที่เมนูคำสั่ง Project => Make ซึ่งหลังจากแปลโปรแกรมแล้วได้ผลถูกต้อง และไม่เกิดข้อผิดพลาดใด ๆ จะปรากฏข้อความ No errors, No warnings ต่อจากนั้นผู้ใช้ก็สามารถนำ hex file ที่ได้จากการสั่งแปลโปรแกรมนี้ไปทำการดาวน์โหลด MCU ได้ทันที



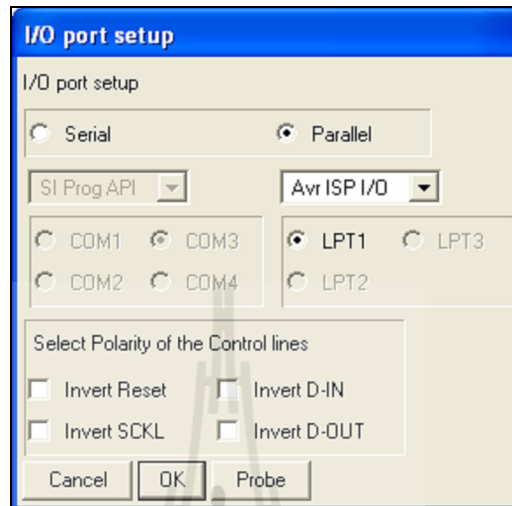
## ข.2 การโปรแกรมให้ออร์ด ET-AVR STAMP ATmega128 ด้วยโปรแกรม PonyProg2000

โปรแกรม PonyProg2000 เป็นโปรแกรมดาวน์โหลดข้อมูลแบบ hex file ให้กับ CPU ตระกูล AVR โดยใช้วิธีการแบบ serial programming ซึ่งสามารถใช้งานกับบอร์ดตระกูล AVR ของ อีทีทีได้เป็นอย่างดี ซึ่งวิธีการใช้งานโปรแกรมโดยทั่วไปนั้น สามารถศึกษาได้จาก help ของโปรแกรมได้เอง โดยในที่นี้จะขอแนะนำให้ทราบถึงวิธีการติดตั้งโปรแกรม PonyProg2000 เพื่อใช้งานกับบอร์ดตระกูล AVR ของอีทีที ซึ่งสามารถใช้งานได้กับบอร์ดตระกูล AVR ทุกรุ่นของอีทีที

โดยในการสั่งโปรแกรม CPU ตระกูล AVR ที่ใช้งานกับบอร์ดของอีทีทีนั้น ถ้าใช้การโปรแกรมด้วยโปรแกรมของ “PonyProg2000” จะต้องกำหนดตัวเลือกของโปรแกรมเพื่อให้สามารถใช้งานกับบอร์ดของอีทีที ดังนี้

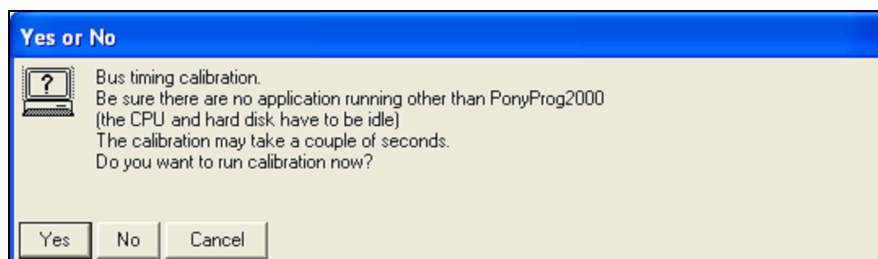


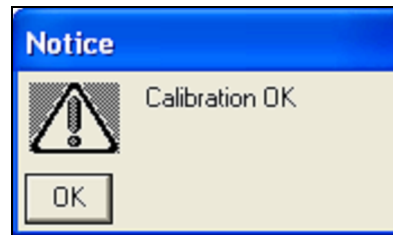
- กำหนด setup => interface setup เป็นดังนี้



- ให้เลือก I/O port setup เป็น Parallel และเลือกรูปแบบการโปรแกรมเป็น Avr ISP I/O
- ให้เลือก printer port ตามที่ต่อจริง เช่น LPT1 ในกรณีที่ใช้กับ printer port LPT1
- ส่วนของ Polarity of the Control lines ไม่ต้องเลือก
- การ setup นี้ทำเพียงครั้งเดียวตอนเริ่มใช้งาน โปรแกรมในครั้งแรกเท่านั้น

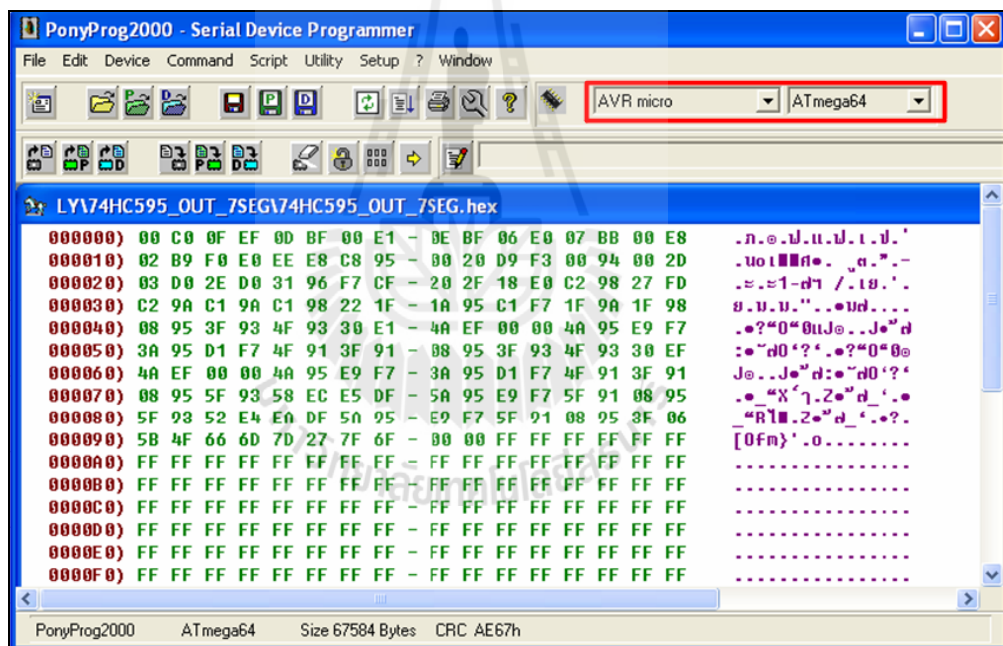
- สั่งให้โปรแกรม PonyProg2000 ทำการคำนวณหาค่าความเร็วที่เหมาะสมสำหรับใช้ในการส่งสัญญาณไปโปรแกรมให้กับ CPU โดยเลือกจาก Setup => Calibration



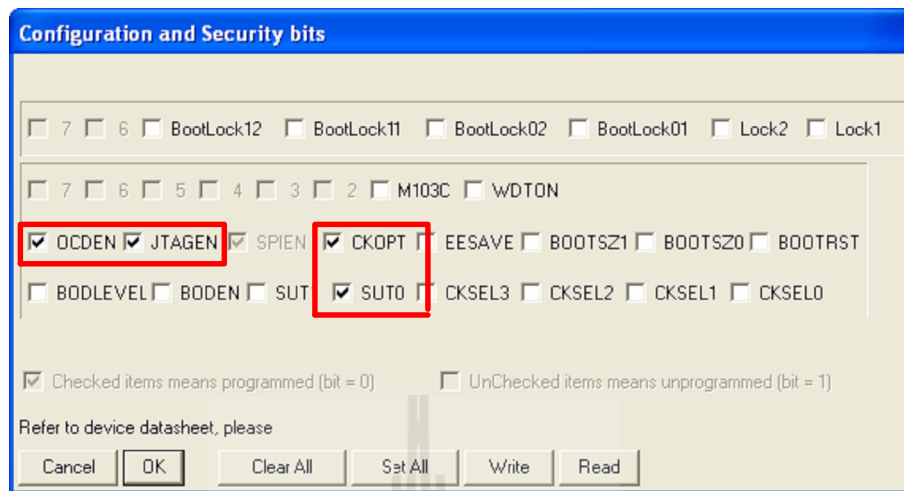


- การตั้ง calibration จะกระทำเพียงครั้งเดียวในตอนเรียกใช้งานโปรแกรมครั้งแรกเท่านั้น

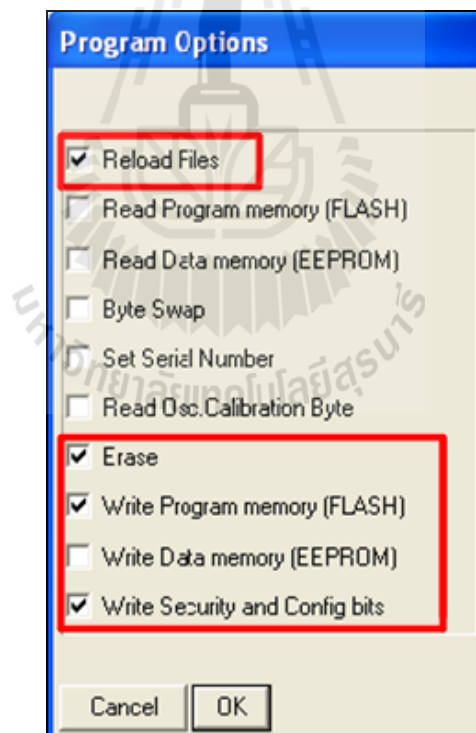
3. เลือกกำหนดเบอร์ CPU จาก Device => AVR micro เช่น ATmega64



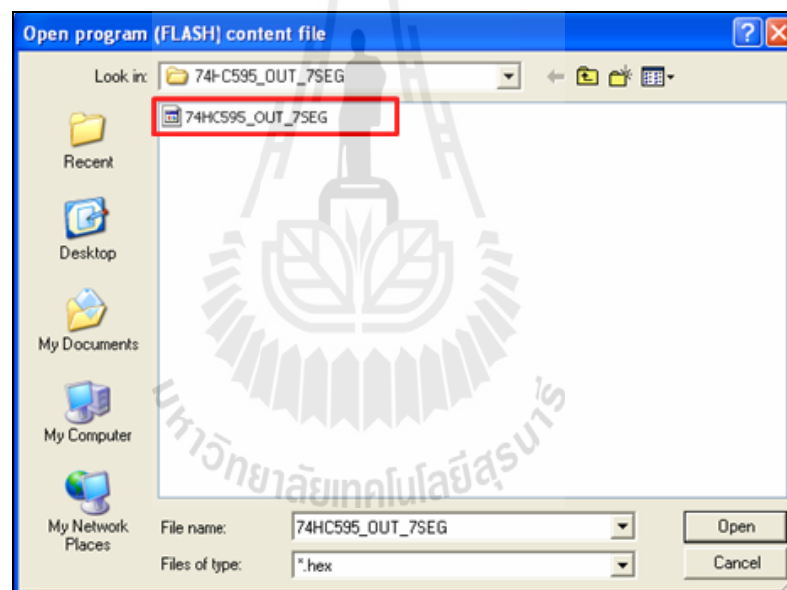
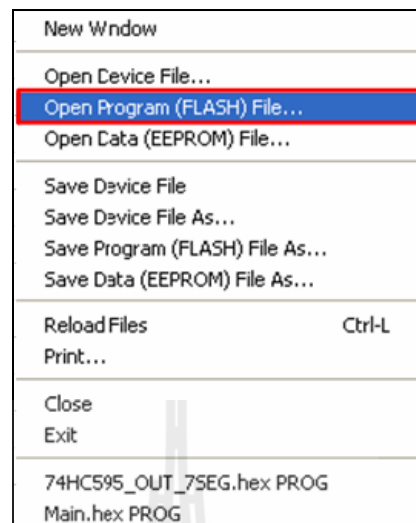
4. เลือกกำหนด Command => Configuration and Security bits โดยถ้าเป็น AVR เบอร์อื่นๆ จะสามารถกำหนดการทำงานของ fuse bit ได้ตามต้องการ โดยสามารถศึกษารายละเอียดของ fuse bit ต่างๆ ได้จาก data sheet ของ CPU ที่ใช้ได้เอง



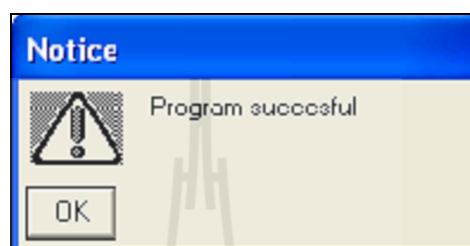
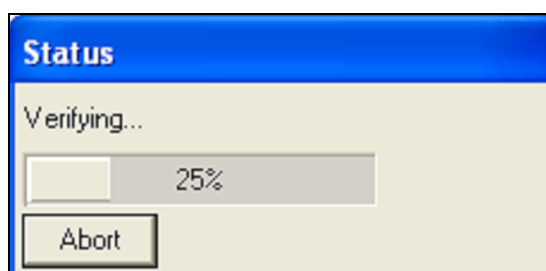
5. เลือกกำหนด Command => Program Options เป็นดังนี้



6. สั่งเปิดไฟล์สำหรับที่จะใช้โปรแกรมให้กับ CPU โดยเลือกจาก File => Open Program (FLASH) File => พร้อมทั้งระบุชื่อและที่อยู่ของ hex file ที่จะใช้โปรแกรมให้เรียบร้อย



7. สั่งเริ่มต้น โปรแกรมข้อมูลให้กับ CPU โดยเลือก Command => Program จากนั้น โปรแกรมจะเริ่มทำงานตามคำสั่งที่เราเลือกกำหนดไว้ในข้อ 5 คือ Reload Files => Erase => Write Program memory (FLASH) และ Write Security and Config bits ตามลำดับ ซึ่งให้รองกว่าการทำงาน of โปรแกรมเสร็จสมบูรณ์





### ค.1 โปรแกรมแม่แบบที่ใช้ในการรับสัญญาณจากบอร์ดไมโครคอนโทรลเลอร์

```

>> clear all; clc;
>> s = serial('com5');
>> input_data_port = 300;
>> s.BaudRate = 57600;
>> s.InputBufferSize = input_data_port*3*4;
>> fopen(s);
>> Data_port1(1) = 0;
>> Data_port2(1) = 0;
>> Data_port3(1) = 0;
>> Data_port4(1) = 0;

>> fprintf(s,'a');
>> for i = 1:1:input_data_port
>> Data_in1 = fscanf(s,'%c',3);
>> Data_1 = hex2dec(Data_in1);
>> Data_port1(i) = (Data_1*5)/1024;
>> fprintf(s,'a');
>> end

```

```

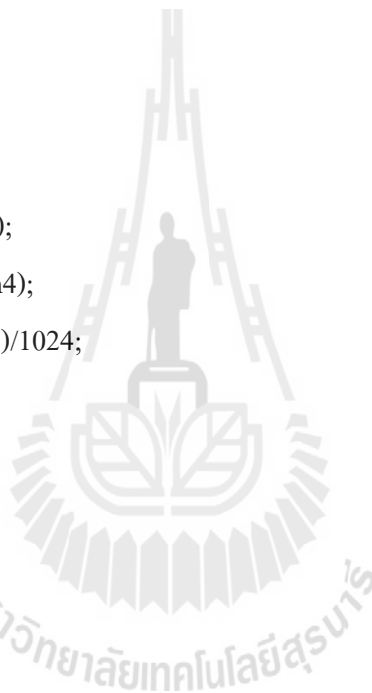
>> fprintf(s,'a');
>> for i = 1:1:input_data_port
>> Data_in2 = fscanf(s,'%c',3);
>> Data_2 = hex2dec(Data_in2);
>> Data_port2(i) = (Data_2*5)/1024;
>> fprintf(s,'a');
>> end

```

```
>> fprintf(s,'a');  
>> for i = 1:1:input_data_port  
>> Data_in3 = fscanf(s,'%c',3);  
>> Data_3 = hex2dec(Data_in3);  
>> Data_port3(i) = (Data_3*5)/1024;  
>> fprintf(s,'a');  
>> end
```

```
>> fprintf(s,'a');  
>> for i = 1:1:input_data_port  
>> Data_in4 = fscanf(s,'%c',3);  
>> Data_4 = hex2dec(Data_in4);  
>> Data_port4(i) = (Data_4*5)/1024;  
>> fprintf(s,'a');  
>> end
```

```
>> fclose(s); delete(s); clear s  
>> aa = length(Data_port1);  
>> u = 1:1:aa;  
>> plot(u,Data_port1,'b-*')  
>> hold on  
>> plot(u,Data_port2,'r-*')  
>> plot(u,Data_port3,'k-*')  
>> plot(u,Data_port4,'g-*')
```





## ค.2 โปรแกรมเมทแอมป์แบบวิธี MUSIC

```

>>clear all; clc;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%STEERING VECTOR%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

>>A = 1;

>>f = 2.45*10^9;

>>c = 3*10^8;

>>lamda = c/f;

>>d = lamda/2;

>>k = (2*pi)/lamda;

>>

>>phase = 0:0.1:180;

>>phase_rad = (phase*pi)/180;

>>

>>N = 4;

>>for n = 1:N;

>>  a(n,:) = A*exp(i*k*d*cos(phase_rad)*(n-1));

>>end

>>a_H = a';

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%DATA300%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

>>Data=[Data_port1;Data_port2;Data_port3;Data_port4];

>>ttsamp=1/((125e3/13)/4);

>>tt=0:ttsamp:(300-1)*ttsamp;

>>uu(1,:) = (Data(1,:)-(max(Data(1,:))-(max(Data(1,:))-min(Data(1,:)))/2));

>>uu(2,:) = (Data(2,:)-(max(Data(2,:))-(max(Data(2,:))-min(Data(2,:)))/2));

>>uu(3,:) = (Data(3,:)-(max(Data(3,:))-(max(Data(3,:))-min(Data(3,:)))/2));

>>uu(4,:) = (Data(4,:)-(max(Data(4,:))-(max(Data(4,:))-min(Data(4,:)))/2));

```

```

>>if uu(1,1)>uu(1,7)
>> phi1 = acos(uu(1,1)/max(uu(1,:)));
>>elseif uu(1,1)<uu(1,7)
>> phi1 = (2*pi)-(acos(uu(1,1)/max(uu(1,:))));
>>end
>>
>>if uu(2,1)>uu(2,6)
>> phi2 = acos(uu(2,1)/max(uu(2,:)));
>>elseif uu(2,1)<uu(2,6)
>> phi2 = (2*pi)-(acos(uu(2,1)/max(uu(2,:))));
>>end
>>
>>if uu(3,1)>uu(3,7)
>> phi3 = acos(uu(3,1)/max(uu(3,:)));
>>elseif uu(3,1)<uu(3,7)
>> phi3 = (2*pi)-(acos(uu(3,1)/max(uu(3,:))));
>>end
>>
>>if uu(4,1)>uu(4,9)
>> phi4 = acos(uu(4,1)/max(uu(4,:)));
>>elseif uu(4,1)<uu(4,9)
>> phi4 = (2*pi)-(acos(uu(4,1)/max(uu(4,:))));
>>end
>>
>>im11=max(uu(1,:))*sin(2*pi*50*tt+phi1);
>>im22=max(uu(2,:))*sin(2*pi*50*tt+phi2);
>>im33=max(uu(3,:))*sin(2*pi*50*tt+phi3);
>>im44=max(uu(4,:))*sin(2*pi*50*tt+phi4);

```

```

>>uu_new(1,:)=uu(1,)+1i*im11;
>>uu_new(2,:)=uu(2,)+1i*im22;
>>uu_new(3,:)=uu(3,)+1i*im33;
>>uu_new(4,:)=uu(4,)+1i*im44;

>>

>>ssamp=1:300;
>>figure(1);
>>plot(ssamp,uu_new(1,),'b-*)
>>plot(ssamp,uu_new(2,),'r-*)
>>plot(ssamp,uu_new(3,),'k-*)
>>plot(ssamp,uu_new(4,),'g-*)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%POWER%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
>>rreal=real(uu_new);
>>h=1:4;
>>for inh=1:length(h)
>>  amplitude(inh) = (max(rreal(inh,:))-min(rreal(inh,:)))/2;
>>  ppower(inh) = amplitude(inh)^2;
>>  dB(inh) = 10*log10(ppower(inh));
>>end

>>power=mean(dB)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%ADAPTIVE SNAPSHOT%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
>>  if power<-33.7
>>    snapshot=100;
>>  elseif power>=-33.7 & power<-29.8
>>    snapshot=50;
>>  elseif power>=-29.8
>>    snapshot=30;
>>  end

```

```

>>for snap=1:length(snapshot)
>>  samp=1:snapshot(:,snap);
>>  for insamp=1:length(samp)
>>    Data1(insamp)=Data_port1(:,insamp);
>>    Data2(insamp)=Data_port2(:,insamp);
>>    Data3(insamp)=Data_port3(:,insamp);
>>    Data4(insamp)=Data_port4(:,insamp);
>>  end
>>  figure(2);
>>  plot(samp,Data1,'b-*')
>>  hold on
>>  plot(samp,Data2,'r-*')
>>  plot(samp,Data3,'k-*')
>>  plot(samp,Data4,'g-*')
>>
>>tsamp=1/((125e3/13)/4);
>>t=0:tsamp:(length(samp)-1)*tsamp;
>>u(1,:) = (Data1-(max(Data1)-(max(Data1)-min(Data1))/2));
>>u(2,:) = (Data2-(max(Data2)-(max(Data2)-min(Data2))/2));
>>u(3,:) = (Data3-(max(Data3)-(max(Data3)-min(Data3))/2));
>>u(4,:) = (Data4-(max(Data4)-(max(Data4)-min(Data4))/2));
>>
>>im1=max(u(1,:))*sin(2*pi*50*t+phi1);
>>im2=max(u(2,:))*sin(2*pi*50*t+phi2);
>>im3=max(u(3,:))*sin(2*pi*50*t+phi3);
>>im4=max(u(4,:))*sin(2*pi*50*t+phi4);

```

```

>>u_new(1,:)=u(1,;)+1i*im1;
>>u_new(2,:)=u(2,;)+1i*im2;
>>u_new(3,:)=u(3,;)+1i*im3;
>>u_new(4,:)=u(4,;)+1i*im4;

>>uu=u_new;
>>u_H = uu';
>>K = length(t);
>>Ruu = uu*u_H;
>>Ruu_cov = Ruu/K;
>>
>> [V,D] = eig(Ruu_cov);
>> [DD,ind] = sort(diag(D),'descend');
>>for i2=1:length(ind)
>>  V2(:,i2)=V(:,ind(i2));
>>end
>>Vn = V2(:,2:end);
>>Vn_H = Vn';
>>V_total = Vn*Vn_H;
>>for p = 1:length(phase);
>>  Pmusic(:,p) = 1/(a_H(p,;)*V_total*a(:,p));
>>end
>>Pmusic_abs = abs(Pmusic);
>>Pmusic_dB = 10*log10(Pmusic_abs);
>>q1 = max(Pmusic_dB);
>>index = find(Pmusic_dB==q1);
>>angle(snap) = phase(index);
>>power2(snap) = max(Pmusic_dB);
>>end

```

```
>>figure(3);  
>>plot(phase,Pmusic_dB,'k')  
>>xlabel('Angle-Of-Arrival in Degree')  
>>ylabel('Power(dB)')
```



