ตัวเข้ารหัสสัญญาณภาพสีแดง เขียว น้ำเงิน เป็นสัญญาณภาพเม็ดสร็จ แบบใช้วงจรเชิงเลขทั้งหมด

AN ALL-DIGITAL BUILD-IN RGB TO COMPOSITE-VIDEO ENCODER

นิยม ชาญสูงเนิน
NIYOM CHANSUNGNOEN

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมอิเล็กทรอนิกส์
บัณฑิตวิทยาลัย
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
พ.ศ. 2549

ISBN 974-15-2467-6

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ตัวเข้ารหัสสัญญาณภาพสีแดง เขียว น้ำเงิน เป็นสัญญาณภาพเบ็ดเสร็จแบบใช้วงจรเชิงเลขทั้งหมด

AN ALL-DIGITAL BUILD-IN RGB TO COMPOSITE-VIDEO ENCODER



นิยม ชาญสูงเนิน

NIYOM CHANSUNGNOEN

เลขหมู่.....
เลขทะเบียน 63433
วัน,เดือน,ปี 28 ส.ค. 2549

.b.....
.i.....

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมอิเล็กทรอนิกส์

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2549

ISBN 974-15-2467-6

AN ALL-DIGITAL BUILD-IN RGB TO COMPOSITE-VIDEO ENCODER

NIYOM CHANSUNGNOEN

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
MASTER OF ENGINEERING IN ELECTRONICS ENGINEERING
SCHOOL OF GRADUATE STUDIES
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
2006

ISBN 974-15-2467-6

COPYRIGHT 2006

SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ใบรับรองวิทยานิพนธ์

หัวข้อวิทยานิพนธ์ ตัวเข้ารหัสสัญญาณภาพสีแดง, เขียว, น้ำเงิน เป็นสัญญาณภาพเบ็ดเสร็จแบบใช้วงจรเชิงเลขทั้งหมด

AN ALL-DIGITAL BUILD-IN RGB TO COMPOSITE-VIDEO ENCODER

นักศึกษา นายนิยม ชาญสูงเนิน

รหัสประจำตัว 44061425

ปริญญา วิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชา วิศวกรรมอิเล็กทรอนิกส์

อาจารย์ผู้ควบคุมวิทยานิพนธ์ ผศ.ประภากร สุวรรณะ

คณะกรรมการสอบวิทยานิพนธ์		ลายมือชื่อ
ผศ.พลผดุง	ผดุงกุล	
ดร.กสิน	วิเชียรชม	
รศ.ดร.มนัส	สังวรศิลป์	
ดร.กิติพล	ชิตสกุล	
ผศ.ประภากร	สุวรรณะ	

วัน / เดือน/ ปี ที่สอบ 2 มีนาคม 2549 เวลา 14.15-16.15 น.

สถานที่สอบ ณ อาคาร 12 ชั้น ชั้น 4 (ห้อง E12-404)

บัณฑิตวิทยาลัยรับรองแล้ว

(ผศ.ดร.จารุวัตร เจริญสุข)

คณบดีบัณฑิตวิทยาลัย

วันที่ 27 **เดือน** กุมภาพันธ์ **พ.ศ.** 2549

หัวข้อวิทยานิพนธ์	ตัวเข้ารหัสสัญญาณภาพสีแดง เขียว น้ำเงิน เป็นสัญญาณภาพเบ็ดเสร็จแบบใช้วงจรเชิงเลขทั้งหมด
นักศึกษา	นาย นิยม ชาญสูงเนิน
รหัสนักศึกษา	44061425
ปริญญา	วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชา	วิศวกรรมอิเล็กทรอนิกส์
พ.ศ.	2549
อาจารย์ผู้ควบคุมวิทยานิพนธ์	ผศ. ประภากร สุวรรณะ

บทคัดย่อ

วิทยานิพนธ์ฉบับนี้นำเสนอวิธีการออกแบบตัวเข้ารหัสสัญญาณภาพสีแดง เขียว น้ำเงิน เป็นสัญญาณภาพเบ็ดเสร็จระบบพาลโดยมุ่งเน้นเพื่อการประยุกต์ใช้งานกับการสร้างสัญญาณภาพทดสอบเพื่อการปรับแต่งสำหรับอุตสาหกรรมการผลิตเครื่องรับโทรทัศน์ การออกแบบใช้ภาษาวีเอชดีแอลเพื่อสร้าง แกนทรัพย์สินทางปัญญา เป็นผลทำให้สามารถใช้งานได้กับอุปกรณ์เอฟพีจีเอ(FPGAs) เอสิค (ASICs) และความสามารถในการนำไปพัฒนาต่อโดยเฉพาะกับเครื่องมือทดสอบอัตโนมัติ

Thesis Title	An All-Digital Build-In RGB to Composite-Video Encoder
Student	Niyom Chansungnoen
Student ID.	44061425
Degree	Master
Programme	Electronics Engineering
Year	2006
Thesis Advisor	Asst. Prof. Prabhakorn Suwanna

ABTRACT

This thesis proposes a design of an all-digital build-in RGB to composite-video encoder. The design is focused on test pattern generator for TV production. An intellectual property core (IP core) is generated by modeling the VHDL; therefore, it can be implemented to both of ASICs and FPGAs. Moreover, it will be capable of furtherance development, especially with integrating into the automatic test equipment (ATE).

กิตติกรรมประกาศ

การศึกษาในระดับปริญญาโทรวมไปถึงวิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงลงด้วยดี เนื่องมาจากการได้รับคำปรึกษาจาก ผศ. ประภากร สุวรรณะ ผู้ซึ่งเป็นอาจารย์ที่ปรึกษาตลอดการศึกษาในระดับปริญญาโท ข้าพเจ้าขอกราบขอบพระคุณเป็นอย่างสูง นอกจากนี้คำแนะนำจากท่านอาจารย์คณะกรรมการสอบวิทยานิพนธ์ทำให้ข้าพเจ้ามองเห็นจุดบกพร่องของการทำงานวิจัยครั้งนี้ซึ่งจะเป็นประโยชน์อย่างมากกับการศึกษาต่อและยังสามารถประยุกต์ใช้กับงานปัจจุบันที่รับผิดชอบอยู่ ข้าพเจ้าขอกราบขอบพระคุณเป็นอย่างสูงมา ณ โอกาสนี้

ขอขอบคุณบริษัท Thomson Television Thailand ที่ให้การอนุเคราะห์เครื่องมือวัดทดสอบตลอดการทำงานวิจัยชิ้นนี้ รวมไปถึงพี่ๆน้องๆที่แผนกวิศวกรรมทดสอบทุกท่านที่ได้ให้การช่วยเหลือโดยเฉพาะ คุณ ศะศิน ฉิมประภา คุณ สำเริง คำกมล และ คุณ สันติ ชวนนอก

กราบขอบพระคุณคณาจารย์สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังทุกๆท่านที่ได้ให้การประสิทธิ์ประสาทวิชาความรู้

ขอขอบคุณเจ้าหน้าที่บัณฑิตศึกษาและบัณฑิตวิทยาลัยสำหรับการช่วยเหลือในเรื่องต่างๆด้วยดีตลอดช่วงการศึกษา

สุดท้ายนี้ขอกราบขอบพระคุณ บิดา มารดา พี่ๆ และสมาชิกทุกคนในครอบครัวที่ได้ให้กำลังใจที่มีค่าและการช่วยเหลือด้วยดีเสมอมา

คุณความดีของวิทยานิพนธ์ฉบับนี้ ข้าพเจ้าขอมอบให้แด่ผู้มีพระคุณทุกท่าน

นิยม ชาญสูงเนิน

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....I

บทคัดย่อภาษาอังกฤษ.....II

กิตติกรรมประกาศ.....III

สารบัญ.....IV

สารบัญตาราง.....IX

สารบัญรูป.....X

บทที่ 1 บทนำ.....1

1.1 ความเป็นมาและความสำคัญของปัญหา.....1

1.2 ความมุ่งหมายและวัตถุประสงค์ของงานวิจัย.....2

1.3 ขอบเขตของงานวิจัย.....3

1.3.1 ขอบเขตของระบบซอร์ฟแวร์.....3

1.3.2 ขอบเขตของระบบฮาร์ดแวร์.....4

1.4 การแบ่งเนื้อหาของวิทยานิพนธ์.....4

บทที่ 2 สัญญาณและระบบโทรทัศน์.....6

2.1 บทนำ.....6

2.2 หลักการสร้างสัญญาณภาพ.....6

2.2.1 การกวาดเส้นภาพ (Scanning).....7

2.2.2 การแบ่งจำนวนเส้นของภาพ.....8

2.2.3 ความถี่ของการฉายภาพซ้ำ (Picture Repetition Frequency).....9

2.2.4 แถบความถี่ของสัญญาณภาพ (Bandwidth of Picture Signal).....11

2.3 สัญญาณภาพสีเบ็ดเสร็จ (Composite Color Video Signal: CCVS).....14

2.3.1 ส่วนประกอบของสัญญาณภาพสีเบ็ดเสร็จระบบพาล.....15

2.3.1.1 สัญญาณแบล้งกิ้ง.....16

2.3.1.2 สัญญาณซิงโครไนเซชั่น.....16

2.3.1.3 สัญญาณความส่องสว่าง.....17

สารบัญ (ต่อ)

หน้า

2.3.1.4 สัญญาณสี....18
2.3.2 ระบบที่ใช้ในการสร้างสัญญาณภาพสีเบ็ดเสร็จ....21
2.3.2.1 ภาค RGB to YUV Encoding Matrix....21
2.3.2.2 ภาค U Low Pass Filter และ V Low Pass Filter....22
2.3.2.3 ภาค Y Delay....22
2.3.2.4 ภาค U และ V Modulation....22
2.3.2.5 ภาค Sub Carrier Generator....28
2.3.2.6 ภาค Burst Generator....29
2.3.2.7 ภาค Blanking & Sync Generator....29
2.4 การส่งสัญญาณวิทยุของสัญญาณภาพและเสียง....29
2.4.1 ส่วนการประมวลผลสัญญาณภาพ (Video Signal Processing)....30
2.4.2 ส่วนการประมวลผลสัญญาณเสียง (Sound Signal Processing)....31
2.4.3 ส่วนคอนเวอเตอร์และวงจรภาคเอาท์พุท (Converter and Output Circuits)....32
2.5 การวัดทดสอบคุณสมบัติของระบบโทรทัศน์....32

บทที่ 3 การประมวลผลสัญญาณเชิงเลข....34
3.1 บทนำ....34
3.2 สัญญาณและระบบแบบไม่ต่อเนื่อง....35
3.2.1 ทฤษฎีการสุ่ม (Sampling Theorem)....35
3.2.2 การแปลงแซดและการแปลงแซดผกผัน....39
3.2.3 การวิเคราะห์ฟูริเยร์....40
3.2.4 ตัวกรองเชิงเลข....42
3.3 ระบบการประมวลผลสัญญาณเชิงเลข....44
3.3.1 ตัวกรองเพื่อป้องกันเอเลียส (Anti-alias Filter)....44
3.3.2 ตัวแปลงสัญญาณเชิงอุปมานเป็นสัญญาณเชิงเลข....45
3.3.3 ตัวประมวลผลสัญญาณเชิงเลข (Digital Signal Processor)....45

สารบัญ (ต่อ)

หน้า

3.3.4 ระบบเลขฐานสองที่ใช้ในการประมวลผล....46

3.3.4.1 เลขฐานสองแบบจำนวนโดยตรง (Fixed Point)....46

3.3.4.2 เลขฐานสองแบบจำนวนอิงดัชนี (Floating Point)....48

3.3.5 ตัวแปลงสัญญาณเชิงเลขเป็นสัญญาณเชิงอุปมาน....49

3.3.6 ตัวกรองสร้างสัญญาณคืน (Reconstruction Filter)....50

3.4 Direct Digital Synthesis (DDS)....51

บทที่ 4 ภาษาวีเอชดีแอล....56

4.1 บทนำ....56

4.2 การออกแบบจากบนลงล่าง (Top Down Design)....57

4.3 โครงสร้างและส่วนการออกแบบ....59

4.3.1 ลักษณะของรูปแบบ (Model Styles)....59

4.3.2 คำศัพท์ที่ใช้ในภาษาวีเอชดีแอล (VHDL Term)....60

4.3.3 หน่วยของแบบ (Design Units)....63

4.3.3.1 Entity Design Unit....63

4.3.3.2 Architecture Design Unit....64

4.3.3.3 Package Design Unit....67

4.3.3.4 Configuration Design Unit....68

4.4 ตัวอย่างการออกแบบระบบเชิงเลขด้วยภาษาวีเอชดีแอล....70

4.4.1 การออกแบบวงจรเชิงเลขแบบ Combination....70

4.4.1.1 ตัวถอดรหัสจาก 2 เป็น 4 (2 to 4 Decoder)....70

4.4.1.2 ตัวมัลติเพล็ก (Multiplexers)....70

4.4.1.3 ตัวบวก (Adder)....71

4.4.2 การออกแบบวงจรเชิงเลขแบบ Sequential....71

4.4.2.1 เอส-อาร์ แลทช์ (SR Latch)....72

4.4.2.2 ดี-ฟลิปฟล็อป (D-Flip-Flop)....72

สารบัญ (ต่อ)

หน้า

4.4.2.3 หน่วยความจำชนิดรอม (Read Only Memory: ROM).....74
4.4.2.4 ไฟไนท์สเตทแมชีนส์ (Finite State Machines: FSM).....74

บทที่ 5 ลำดับขั้นการดำเนินงาน.....79
5.1 บทนำ.....79
5.2 ศึกษาคุณสมบัติของสัญญาณและการทำงานของระบบ.....79
5.3 กำหนดคุณสมบัติเริ่มต้นของการออกแบบ.....82
5.3.1 คุณสมบัติของสัญญาณเชิงอุปมานที่ต้องการจากระบบ.....82
5.3.2 คุณสมบัติของระบบเชิงเลขที่ใช้ในการสร้างสัญญาณภาพสีเบ็ดเสร็จ.....83
5.4 การออกแบบระบบโดยรวม.....84
5.5 การออกแบบส่วนการทำงานย่อยของระบบ.....86
5.5.1 ภาค CLOCK_LINE_REF.....86
5.5.2 ภาค COLOR_SPACE.....90
5.5.3 ภาค CSYNC_COMBINER.....93
5.5.4 ภาค CVS_LPF3TAB.....94
5.5.5 ภาค CVS_DELAY.....100
5.5.6 ภาค NCO_MUX.....101
5.5.7 ภาค U_LPF3TAB_FAC.....103
5.5.8 ภาค V_LPF3TAB_FAC.....107
5.5.9 ภาค UVMOD.....111
5.5.10 ภาค C_LPF3TAB.....114

บทที่ 6 ผลการทดลอง.....119
6.1 การทดสอบระบบซอฟท์แวร์.....119
6.2 การทดสอบระบบฮาร์ดแวร์.....119

สารบัญ (ต่อ)

หน้า

บทที่ 7 สรุปการวิจัยและข้อเสนอแนะ.....129

เอกสารอ้างอิง.....131

ผลงานวิจัยที่ได้รับการตีพิมพ์เผยแพร่.....133

ประวัติผู้เขียน.....134

สารบัญตาราง

ตารางที่ หน้า

6.1 แสดงผลการวัดคาบเวลาและความถี่ของสัญญาณภาพสีเบ็ดเสร็จที่ได้จากระบบ.....127
6.2 แสดงผลการวัดเฟสของสัญญาณภาพสีเบ็ดเสร็จที่ได้จากระบบ.....127
6.3 แสดงผลการวัดขนาดของสัญญาณภาพสีเบ็ดเสร็จที่ได้จากระบบ.....128

สารบัญรูป

รูปที่ หน้า

1.1 แสดงมุมมองการเชื่อมต่อของโครงสร้างระดับบนสุด (Top Level) ของตัวเข้ารหัสฯที่นำเสนอ.....3

1.2 แสดงบล็อคไดอะแกรมของระบบฮาร์ดแวร์ต้นแบบ.....4

2.1 แสดงหลักการส่งภาพของระบบโทรทัศน์.....7

2.2 แสดงการกวาดเส้นภาพและรูปร่างของกระแสไฟฟ้าที่ได้จากเซนเซอร์ (Sensor).....7

2.3 แสดงความสัมพันธ์ของระยะและมุมการมองภาพของสายตา.....8

2.4 แสดงการแบ่งการกวาดเส้นภาพ.....10

2.5 แสดงการสร้างความถี่ที่ใช้ในการกวาดเส้นภาพ.....11

2.6 แสดงความละเอียดและคาบเวลาของการกวาดเส้นภาพ.....11

2.7 แสดงผลของช่วงฟลายแบ็คทำให้พื้นที่ของราสเตอร์ลดลง.....12

2.8 แสดงการลดความคมของสัญญาณภาพเนื่องจากเส้นผ่าศูนย์กลางที่จำกัดของลำอิเล็กตรอน.....14

2.9 แสดงสัญญาณภาพสีเบ็ดเสร็จของสัญญาณภาพแท่งสี 75 %.....16

2.10 แสดงค่าความไวต่อความส่องสว่างของตามนุษย์ในช่วงของแสงที่มีความยาวคลื่นค่าต่างๆ.....17

2.11 แสดงรูปแบบการส่งสัญญาณภาพเพื่อความเข้ากันได้ของระบบโทรทัศน์สีและขาวดำ.....19

2.12 แสดงควอดราเจอร์มอดูเลเตอร์.....20

2.13 แสดงบล็อคไดอะแกรมของตัวเข้ารหัสสัญญาณภาพสีแดง, เขียวและน้ำเงินเป็นสัญญาณภาพสีเบ็ดเสร็จ.....21

2.14 แสดงบล็อกไดอแกรมของตัวดีมอดูเลตสัญญาณความแตกต่างสีของระบบพาล.....26

2.15 แสดงเวคเตอร์ของสัญญาณในกรณีที่สัญญาณไม่มีความเพี้ยนทางเฟส.....27

2.16 แสดงเวคเตอร์ของสัญญาณในกรณีที่สัญญาณมีความเพี้ยนทางเฟส.....28

2.17 แสดงบล็อคไดอะแกรมมอดูเลเตอร์ของบริษัทฟิลิปส์รุ่น PM 5671.....30

3.1 แสดงระบบพื้นฐานของการประมวลผลสัญญาณเชิงเลข.....35

3.2 แสดงสัญญาณในโดเมนเวลาที่ถูกสุ่มด้วยความถี่ค่าต่างๆและสเปกตรัมในโดเมนความถี่.....38

3.3 แสดงสมการที่ใช้ในการแปลงฟูริเยร์.....41

3.4 แสดงสมการที่ใช้ในการคำนวณอนุกรมฟูริเยร์.....41

3.5 แสดงสมการที่ใช้ในการคำนวณดีสครีตไทม์ฟูริเยร์ทรานสฟอร์ม.....42

3.6 แสดงสมการที่ใช้ในการคำนวณดีสครีตฟูริเยร์ทรานสฟอร์ม.....42

สารบัญรูป (ต่อ)

รูปที่ หน้า

3.7 แสดงการจำแนกชนิดของตัวกรองเชิงเลขตามกลุ่มของการใช้งานและวิธีการสร้าง....43

3.8 แสดงบล็อคไดอะแกรมของระบบการประมวลผลสัญญาณเชิงเลข....44

3.9 แสดงข้อมูลคุณสมบัติของตัวกรองที่นิยมใช้ในการป้องกันเอเลียส....45

3.10 แสดงตัวเลขฐานสองแบบจำนวนโดยตรงชนิดต่างๆ....47

3.11 แสดงการเปรียบค่าระหว่างตัวเลขแบบ Two's Complement และ Offset Two's Complement....48

3.12 แสดงวิธีการแทนค่าด้วยตัวเลขแบบจำนวนอิงดัชนี....49

3.13 แสดงสัญญาณในโดเมนเวลาและความถี่ของสัญญาณก่อนและหลังการคงค่า....50

3.14 แสดงผลตอบสนองความถี่ของตัวกรองสร้างสัญญาณคืนในอุดมคติ....51

3.15 แสดงสัญญาณในโดเมนเวลาและความถี่ของสัญญาณที่ได้จากตัวกรองสร้างสัญญาณคืนในอุดมคติ....51

3.16 แสดงระบบดีดีเอสอย่างง่าย....52

3.17 แสดงระบบดีดีเอสที่สามารถปรับความถี่ได้....53

3.18 แสดงวงล้อเฟสเชิงเลข (Digital Phase Wheel)....54

3.19 แสดงสัญญาณที่จุดต่างๆของระบบดีดีเอส....54

4.1 แสดงตัวดำเนินการและผลที่ได้จากการดำเนินการในภาษาวีเอชดีแอล....62

4.2 (ซ้าย) แสดงโครงสร้างของหน่วย Entity (ขวา) แสดงมุมมองการเชื่อมต่อ....64

4.3 แสดงโครงสร้างของหน่วย Architecture....64

4.4 แสดงวงจรมัลติเพล็กขนาด 2:1....65

4.5 แสดงหน่วย Architecture ที่ใช้ Model Style แบบ Register Transfer Level (RTL)....65

4.6 แสดงหน่วย Architecture ที่ใช้ Model Style แบบ Structural Description....66

4.7 แสดงหน่วย Architecture ที่ใช้ Model Style แบบ Behavioral Description....67

4.8 แสดงหน่วย Package ซึ่งภายในประกอบด้วยโปรแกรมย่อยชื่อ Mean....68

4.9 แสดงรูปแบบ (Model) ที่มีหน่วย Architecture สองหน่วย....69

4.10 แสดงรูปแบบของตัวถอดรหัสจาก 2 เป็น 4....71

4.11 แสดงรูปแบบของตัวมัลติเพล็กขนาด 4:1....72

4.12 แสดงรูปแบบของตัวบวกขนาด 1 บิต....73

สารบัญรูป (ต่อ)

รูปที่ หน้า

4.13 แสดงรูปแบบของเอสอาร์แลชท์....73
4.14 แสดงรูปแบบของดีฟลิปฟล็อป....74
4.15 แสดงรูปแบบของหน่วยความจำรอม....75
4.16 แสดงสเตทไดอะแกรมของไฟไนท์สเตทแมชีนส์....76
4.17ก แสดงรูปแบบของไฟไนท์สเตทแมชีนส์....76
4.17ข แสดงรูปแบบของไฟไนท์สเตทแมชีนส์ (ต่อ)....77
4.17ค แสดงรูปแบบของไฟไนท์สเตทแมชีนส์ (ต่อ)....78
5.1 แสดงส่วนประกอบของสัญญาณภาพสีเบ็ดเสร็จระบบพาลซึ่งมีคาบซ้ำทุกๆ 8 ฟิลด์....80
5.2 แสดงส่วนประกอบของสัญญาณภาพสีเบ็ดเสร็จในช่วงเวลาการกวาดเส้นภาพทางแนวนอน....81
5.3 แสดงเวคเตอร์สัญญาณความแตกต่างสีของสัญญาณแท่งสี 75 %....82
5.4 แสดงการแบ่งกลุ่มสัญญาณที่ใช้ในการสร้างสัญญาณภาพสีเบ็ดเสร็จ....82
5.5 แสดงตัวเข้ารหัสสัญญาณภาพสีแดง เขียว น้ำเงิน เป็นสัญญาณภาพสีเบ็ดเสร็จที่ได้จาก [5]....85
5.6 แสดงตัวเข้ารหัสสัญญาณภาพสีแดง เขียว น้ำเงิน เป็นสัญญาณภาพสีเบ็ดเสร็จที่ได้จาก [19]....85
5.7 แสดงโครงสร้างระดับบนสุด (Top Level) ของตัวเข้ารหัสฯที่นำเสนอ....86
5.8 แสดงโครงสร้างในระดับ RTL ของภาค CLOCK_LINE_REF....87
5.9 แสดงมุมมองการเชื่อมต่อของภาค CLOCK_LINE_REF....87
5.10 แสดงผลการจำลองการทำงานในระดับไทม์มิ่งของภาค CLOCK_LINE_REF....89
5.11 แสดงผลการจำลองการทำงานในระดับไทม์มิ่งของภาค CLOCK_LINE_REF (ต่อ)....89
5.12 แสดงผลการจำลองการทำงานในระดับไทม์มิ่งของภาค CLOCK_LINE_REF (ต่อ)....90
5.13 แสดงโครงสร้างในระดับ RTL ของภาค COLOR_SPACE....91
5.14 แสดงมุมมองการเชื่อมต่อของภาค COLOR_SPACE....92
5.15 แสดงผลการจำลองการทำงานในระดับไทม์มิ่งของภาค COLOR_SPACE....92
5.16 แสดงโครงสร้างในระดับ RTL ของภาค CSYNC_COMBINER....93
5.17 แสดงมุมมองการเชื่อมต่อของภาค CSYNC_COMBINER....93
5.18 แสดงผลการจำลองการทำงานในระดับไทม์มิ่งของภาค CSYNC_COMBINER....94

สารบัญรูป (ต่อ)

รูปที่ หน้า

5.19 แสดงค่าสัมประสิทธิ์ของตัวกรอง, ซ้าย (QuantizedCoefficient) เป็นค่าสัมประสิทธิ์ที่แบ่งระดับให้เหลือขนาดเป็น 8 บิต, ขวา (Reference Coefficient) เป็นค่าสัมประสิทธิ์ที่ยังไม่ได้แบ่งระดับ.....96
5.20 แสดงผลตอบสนองความถี่ของตัวกรองภาค CVS_LPF3TAB.....96
5.21 แสดงผลตอบสนองเฟสของตัวกรองภาค CVS_LPF3TAB.....97
5.22 แสดงผลตอบสนองความถี่เทียบกับผลตอบสนองเฟสของตัวกรองภาค CVS_LPF3TAB.....97
5.23 แสดงผลตอบสนองอิมพัลส์ของตัวกรองภาค CVS_LPF3TAB.....98
5.24 แสดงโครงสร้างแบบโดยตรงของตัวกรองไม่ป้อนกลับเชิงเลขที่ใช้ในงานวิจัยนี้.....98
5.25 แสดงโครงสร้างในระดับ RTL ของตัวกรองภาค CVS_LPF3TAB.....99
5.26 แสดงมุมมองการเชื่อมต่อของภาค CVS_LPF3TAB.....99
5.27 แสดงผลการจำลองการทำงานในระดับไทม์มิ่งของภาค CVS_LPF3TAB.....99
5.28 แสดงโครงสร้างในระดับ RTL ของภาค CVS_DELAY.....100
5.29 แสดงมุมมองการเชื่อมต่อของภาค CVS_DELAY.....100
5.30 แสดงผลการจำลองการทำงานในระดับไทม์มิ่งของภาค CVS_DELAY.....101
5.31 แสดงโครงสร้างในระดับ RTL ของภาค NCO_MUX.....101
5.32 แสดงมุมมองการเชื่อมต่อของภาค NCO_MUX.....101
5.33 แสดงผลการจำลองการทำงานในระดับไทม์มิ่งของภาค NCO_MUX เมื่อ PAL SW = 0.....102
5.34 แสดงผลการจำลองการทำงานในระดับไทม์มิ่งของภาค NCO_MUX เมื่อ PAL SW = 1.....103
5.35 แสดงค่าสัมประสิทธิ์ของตัวกรอง, ซ้าย (QuantizedCoefficient) เป็นค่าสัมประสิทธิ์ที่แบ่งระดับให้เหลือขนาดเป็น 8 บิต, ขวา (Reference Coefficient) เป็นค่าสัมประสิทธิ์ที่ยังไม่ได้แบ่งระดับ....104
5.36 แสดงผลตอบสนองความถี่ของตัวกรองภาค U_LPF3TAB_FAC.....104
5.37 แสดงผลตอบสนองเฟสของตัวกรองภาค U_LPF3TAB_FAC.....105
5.38 แสดงผลตอบสนองความถี่เทียบกับผลตอบสนองเฟสของตัวกรองภาค U_LPF3TAB_FAC.....105
5.39 แสดงผลตอบสนองอิมพัลส์ของตัวกรองภาค U_LPF3TAB_FAC.....106
5.40 แสดงโครงสร้างในระดับ RTL ของตัวกรองภาค U_LPF3TAB_FAC.....106
5.41 แสดงมุมมองการเชื่อมต่อของภาค U_LPF3TAB_FAC.....106
5.42 แสดงผลการจำลองการทำงานในระดับไทม์มิ่งของภาค U_LPF3TAB_FAC.....107

สารบัญรูป (ต่อ)

รูปที่ หน้า

5.43 แสดงค่าสัมประสิทธิ์ของตัวกรอง, ซ้าย (QuantizedCoefficient) เป็นค่าสัมประสิทธิ์ที่แบ่งระดับให้เหลือขนาดเป็น 8 บิต, ขวา (Reference Coefficient) เป็นค่าสัมประสิทธิ์ที่ยังไม่ได้แบ่งระดับ....108
5.44 แสดงผลตอบสนองความถี่ของตัวกรองภาค V_LPF3TAB_FAC....108
5.45 แสดงผลตอบสนองเฟสของตัวกรองภาค V_LPF3TAB_FAC....109
5.46 แสดงผลตอบสนองความถี่เทียบกับผลตอบสนองเฟสของตัวกรองภาค V_LPF3TAB_FAC....109
5.47 แสดงผลตอบสนองอิมพัลส์ของตัวกรองภาค V_LPF3TAB_FAC....110
5.48 แสดงโครงสร้างในระดับ RTL ของตัวกรองภาค V_LPF3TAB_FAC....110
5.49 แสดงมุมมองการเชื่อมต่อของภาค V_LPF3TAB_FAC....111
5.50 แสดงผลการจำลองการทำงานในระดับไทม์มิ่งของภาค U_LPF3TAB_FAC....111
5.51 แสดงโครงสร้างในระดับ RTL ของภาค UVMOD....112
5.52 แสดงมุมมองการเชื่อมต่อของภาค UVMOD....112
5.53 แสดงค่าสัมประสิทธิ์ของตัวกรอง ซ้าย (Quantized Coefficient) เป็นค่าสัมประสิทธิ์ที่แบ่งระดับให้เหลือขนาดเป็น 8 บิต ขวา (Reference Coefficient) เป็นค่าสัมประสิทธิ์ที่ยังไม่ได้แบ่งระดับ....115
5.54 แสดงผลตอบสนองความถี่ของตัวกรองภาค C_LPF3TAB....115
5.55 แสดงผลตอบสนองเฟสของตัวกรองภาค C_LPF3TAB....116
5.56 แสดงผลตอบสนองความถี่เทียบกับผลตอบสนองเฟสของตัวกรองภาค C_LPF3TAB....116
5.57 แสดงผลตอบสนองอิมพัลส์ของตัวกรองภาค C_LPF3TAB....117
5.58 แสดงโครงสร้างในระดับ RTL ของตัวกรองภาค C_LPF3TAB....117
5.59 แสดงมุมมองการเชื่อมต่อของภาค C_LPF3TAB....118
5.60 แสดงผลการจำลองการทำงานในระดับไทม์มิ่งของภาค C_LPF3TAB....118
6.1 แสดงรูปแบบการเชื่อมต่อเครื่องมือวัดที่ใช้ในการทดสอบคุณสมบัติของระบบฮาร์ดแวร์....120
6.2 แสดงผลสรุปการใช้ทรัพยากรภายในเอฟพีจีเอหลังการสังเคราะห์และเพลซแอนเร้าท์(Place & Route)....120
6.3 แสดงสัญญาณในช่วงของการกวาดเส้นภาพกลับทางแนวตั้ง....121
6.4 แสดง Broad Pulse Separator ในช่วงของการกวาดเส้นภาพกลับทางแนวตั้ง....121
6.5 แสดง Equalizer Pulse ในช่วงของการกวาดเส้นภาพกลับทางแนวตั้ง....122

สารบัญรูป (ต่อ)

รูปที่ หน้า

6.6 แสดงสัญญาณภาพสีเบ็ดเสร็จของการกวาดเส้นภาพทางแนวนอน 1 เส้น....122

6.7 แสดงสัญญาณภาพสีเบ็ดเสร็จของการกวาดเส้นภาพทางแนวนอนหลังจากผ่านวงจรกรองความถี่ต่ำผ่าน....123

6.8 แสดงสัญญาณภาพสีเบ็ดเสร็จของการกวาดเส้นภาพทางแนวนอนหลังจากผ่านวงจรกรองความถี่สูงผ่าน....123

6.9 แสดงสัญญาณซิ้งค์ทางแนวนอนและสัญญาณเบิรสต์....124

6.10 แสดงระดับของสัญญาณความส่องสว่าง ระดับและเฟสของสัญญาณสี....124

6.11 แสดงเวคเตอร์สัญญาณความแตกต่างสีและสัญญาณเบิรสต์ของสัญญาณแท่งสี100%....125

6.12 แสดงเวคเตอร์ของสัญญาณเบิรสต์....125

6.13 แสดงเครื่องมือวัดขณะทำการทดสอบเครื่องต้นแบบ....126

6.14 แสดงเครื่องต้นแบบตัวเข้ารหัสฯที่นำเสนอ....126

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

การผลิตเครื่องรับโทรทัศน์ในอุตสาหกรรมการผลิตเครื่องรับโทรทัศน์โดยทั่วไปประกอบไปด้วยหลายขบวนการด้วยกันเช่น การใส่อุปกรณ์ลงแผ่นพีซีบี (Printed Circuit Board: PCB) การบัดกรี ฯลฯ เป็นต้น การปรับแต่งภาพคุณภาพของภาพ (Picture Quality Alignment) เป็นขั้นตอนที่สำคัญมากขั้นตอนหนึ่งในขบวนการผลิต เหตุผลที่ทำให้ต้องมีการปรับแต่งก็เนื่องมาจากเครื่องรับโทรทัศน์ถูกสร้างจากอุปกรณ์อิเล็กทรอนิกส์ที่มีคุณสมบัติไม่เป็นไปตามอุดมคติหลายชนิด ทั้งอุปกรณ์ที่ทำงานเป็นเชิงเส้นและไม่เป็นเชิงเส้นโดยที่อุปกรณ์เหล่านี้จะมีค่าความคลาดเคลื่อนที่เกิดจากการผลิตเป็นคุณสมบัติประจำตัวที่หลีกเลี่ยงไม่ได้เช่น ค่าเปอร์เซ็นต์ความคลาดเคลื่อนของตัวต้านทาน ตัวเก็บประจุหรือ อัตราการขยายที่ไม่เท่ากันของชุดยิงลำอิเล็กตรอน (Electron Gun) สีแดง เขียวและน้ำเงินของจอภาพ (Cathode Ray Tube: CRT) ฯลฯ เป็นต้น ดังนั้นหลังจากการนำอุปกรณ์ที่มีคุณสมบัติไม่เป็นไปตามอุดมคติดังกล่าวมาใช้ในการสร้างเป็นระบบที่ทำงานอย่างใดอย่างหนึ่ง จึงจำเป็นต้องมีการปรับแต่งบางพารามิเตอร์ (Parameter) เพื่อให้ระบบทำงานได้ถูกต้องและได้สัญญาณขาออก (Output) ที่ดีที่สุดซึ่งสำหรับเครื่องรับโทรทัศน์ก็คือสัญญาณภาพที่หน้าจอนั่นเอง

การปรับแต่งคุณภาพของภาพแบ่งออกเป็น2กลุ่มใหญ่ๆได้แก่ การปรับแต่งทางกลและการปรับแต่งทางไฟฟ้า การปรับแต่งทั้งสองประเภทดังกล่าวจำเป็นต้องใช้สัญญาณภาพ (Video Signal) ซึ่งมีคุณสมบัติเฉพาะที่เหมาะสมกับการปรับแต่งนั้นๆ เช่น ภาพพื้นสีใช้ในการปรับความบริสุทธิ์ของสี (Purity) ภาพครอสแฮช (Crosshatch) ใช้ในการปรับคอนเวอร์เจนซ์ ภาพโมโนสโคป (Mono Scope) ใช้ในการปรับความคมชัด (Focus) ภาพฟิลิปส์ (Philips) ใช้ในการปรับความสมมาตรเชิงเรขาคณิตของภาพ (Geometry) ฯลฯ แต่ในปัจจุบันเครื่องสร้างสัญญาณภาพโทรทัศน์มาตรฐาน (Color TV Pattern Generator) ถูกออกแบบให้สามารถผลิตสัญญาณภาพได้เพียงบางรูปแบบเท่านั้น อันเนื่องมาจากเหตุผลทางธุรกิจ เช่น ภาพโมโนสโคป (Mono Scope) หรือ ภาพฟิลิปส์ (Philips) ฯลฯ ในขณะเดียวกันการพัฒนาขบวนการของการผลิตทำให้เกิดความต้องการสัญญาณภาพหลายรูปแบบ ฉะนั้นถ้าต้องการสัญญาณภาพตามที่ได้กล่าวในข้างต้นจะต้องใช้เครื่องสร้างสัญญาณภาพมากกว่าหนึ่งเครื่องขึ้นไปทำให้

ต้นทุนการผลิตสูง ในขณะที่เครื่องสร้างสัญญาณภาพโทรทัศน์ชนิดโปรแกรมได้ก็มีราคาสูงมากเกินความจำเป็นที่จะนำมาใช้กับขบวนการผลิต จากปัญหาดังกล่าวผู้เขียนจึงได้เสนอวิธีการออกแบบเครื่องสร้างสัญญาณรูปแบบชนิดโปรแกรมได้ (Programmable Pattern Generator) [1] เพื่อลดต้นทุนการผลิตและที่สำคัญก็คือสร้างความยืดหยุ่นให้กับขบวนการผลิต

โดยทั่วไประบบที่ใช้ในการสร้างสัญญาณภาพรวมไปถึงระบบควบคุมซึ่งประกอบด้วยตัวประมวลผล (Processor) หน่วยความจำ (Memory) อุปกรณ์ต่อพ่วง (Peripheral) ฯลฯ ถูกสร้างจากวงจรเชิงเลขเสียเป็นส่วนใหญ่เนื่องจากการออกแบบโดยใช้วงจรเชิงเลขมีข้อได้เปรียบการออกแบบโดยใช้วงจรเชิงอุปมานอยู่หลายประการ [2]-[4] แต่อย่างไรก็ตามยังคงมีข้อจำกัดอยู่หลายประการเช่นกัน ดังเช่นการออกแบบที่ได้กล่าวในข้างต้น [1] ซึ่งใช้อุปกรณ์ประเภทซีพีแอลดี (CPLDs) ที่มีจำนวนลอจิกเกตน้อย ทำให้บางส่วนการทำงานของวงจรที่เป็นวงจรเชิงอุปมานไม่สามารถทำได้ เช่น ตัวเข้ารหัสสัญญาณภาพสีแดง เขียว น้ำเงิน เป็นสัญญาณภาพสีเบ็ดเสร็จเป็นต้น แต่เนื่องจากในปัจจุบันเทคโนโลยีการผลิตทำให้จำนวนลอจิกเกตของอุปกรณ์ประเภทเอฟพีจีเอ (FPGAs) เพิ่มขึ้นอย่างมาก ในขณะที่ราคาต่อจำนวนลอจิกเกตก็ได้ลดลงอย่างมากเช่นกัน การใช้วงจรเชิงเลขทั้งหมดจึงสามารถทำได้

ตัวเข้ารหัสสัญญาณภาพสีแดง เขียว น้ำเงิน เป็นสัญญาณภาพสีเบ็ดเสร็จแบบใช้วงจรเชิงเลขทั้งหมดที่นำเสนอ มุ่งเน้นไปที่การแก้ไขปัญหาในการเชื่อมต่อ (Interfacing) [2] เพื่อการนำไปประยุกต์ใช้กับระบบที่เป็นเชิงเลขทั้งหมด ซึ่งไม่เพียงแต่ระบบของการสร้างสัญญาณภาพทดสอบเท่านั้น ตัวเข้ารหัสฯที่นำเสนอยังสามารถนำไปฝังตัวรวมกับระบบอื่นๆที่ผู้ใช้มีปฏิสัมพันธ์กับภาพ เช่น ระบบควบคุมที่มีการแสดงผลที่ซับซ้อน ยกตัวอย่างเช่น ระบบดีซีเอส (Distributed Control System: DCS) ระบบฝังตัว (Embedded System) เครื่องใช้ไฟฟ้าเพื่อความบันเทิง ฯลฯ การออกแบบได้เลือกใช้ภาษาวีเอชดีแอล (Very high speed integrated circuits Hardware Description Language : VHDL) เนื่องจากมีข้อดีหลายประการ เช่น เป็นมาตรฐานของ IEEE (IEEE 1076) ไม่ยึดติดกับเทคโนโลยีใดเทคโนโลยีหนึ่งกล่าวคือหลังการออกแบบสามารถใช้ได้กับทุกบริษัทผู้ผลิตทั้งเอสิค (Application Specific Integrated Circuit: ASIC) และเอฟพีจีเอ (FPGAs) สามารถสร้างแกนทรัพย์สินทางปัญญา (Intellectual Property Core: IP Core) เพื่อการพัฒนาต่อ ฯลฯ เป็นต้น

1.2 ความมุ่งหมายและวัตถุประสงค์ของงานวิจัย

วัตถุประสงค์หลักของงานวิจัยนี้ก็คือการสร้างแกนทรัพย์สินทางปัญญาของตัวเข้ารหัสสัญญาณภาพสีแดง เขียว น้ำเงิน เป็นสัญญาณภาพสีเบ็ดเสร็จแบบใช้วงจรเชิงเลขทั้งหมดโดยใช้ภาษาวี

เอชดีแอลและมีจุดมุ่งหมายหลักคือการนำไปประยุกต์ใช้งานกับการสร้างสัญญาณภาพทดสอบเพื่อการปรับแต่งสำหรับอุตสาหกรรมการผลิตเครื่องรับโทรทัศน์และเพื่อการนำไปพัฒนาต่อ

เนื่องจากการออกแบบตัวเข้ารหัสสัญญาณภาพสีแดง เขียว น้ำเงิน เป็นสัญญาณภาพสีเบ็ดเสร็จแบบใช้วงจรเชิงเลขทั้งหมดที่นำเสนอเป็นการออกแบบจากระบบต้นแบบที่เป็นระบบเชิงอุปมานดังนั้นจึงต้องใช้ทั้งทฤษฎีของการออกแบบระบบเชิงอุปมานและระบบเชิงเลขจึงทำให้เกิดจุดประสงค์ของการเรียนรู้ดังนี้

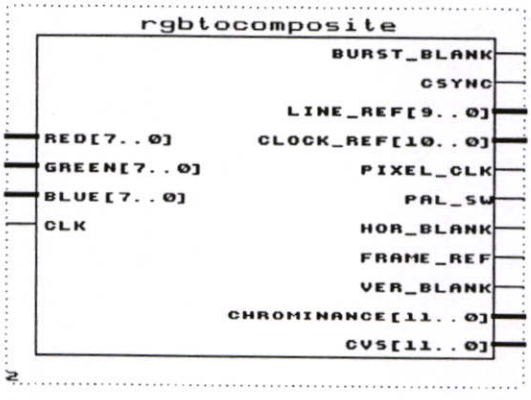
- เพื่อศึกษาการทำงานของตัวเข้ารหัสสัญญาณภาพสีแดง เขียว น้ำเงิน เป็นสัญญาณภาพสีเบ็ดเสร็จ
- เพื่อศึกษาการแปลงระบบเชิงอุปมานเป็นระบบเชิงเลขโดยใช้ทฤษฎีการประมวลผลสัญญาณร่วมกับเทคนิคการออกแบบวงจรเชิงเลข
- เพื่อศึกษาการออกแบบระบบเชิงเลขโดยใช้ภาษาวีเอชดีแอล

1.3 ขอบเขตของงานวิจัย

จากวัตถุประสงค์ทำให้สามารถกำหนดขอบเขตของงานวิจัยซึ่งแบ่งออกเป็นสองระบบใหญ่ๆได้แก่

1.3.1 ขอบเขตของระบบซอร์ฟแวร์

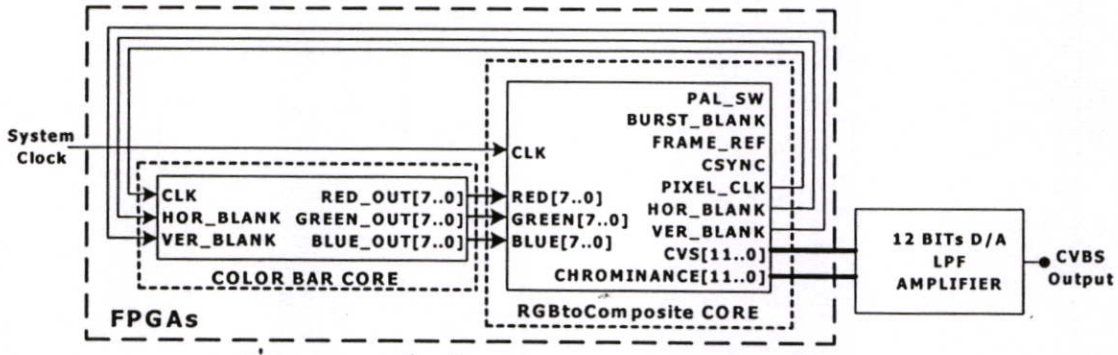
สร้างแกนทรัพย์สินทางปัญญาของตัวเข้ารหัสสัญญาณภาพสีแดง เขียว น้ำเงิน เป็นสัญญาณภาพสีเบ็ดเสร็จแบบใช้วงจรเชิงเลขทั้งหมดโดยใช้ภาษาวีเอชดีแอลซึ่งประกอบไปด้วยสัญญาณขาเข้าและสัญญาณขาออกมีรายละเอียดตามมุมมองของการเชื่อมต่อแสดงดังรูปที่ 1.1



รูปที่ 1.1 แสดงมุมมองการเชื่อมต่อของโครงสร้างระดับบนสุด (Top Level) ของตัวเข้ารหัสฯที่นำเสนอ

1.3.2 ขอบเขตของระบบฮาร์ดแวร์

สร้างฮาร์ดแวร์ต้นแบบโดยใช้อุปกรณ์เอฟพีจีเอ (FPGAs) ในการทดสอบคุณสมบัติของระบบโดยใช้เครื่องมือวัดทางอิเล็กทรอนิกส์ซึ่งมีรายละเอียดตามบล็อคไดอะแกรมแสดงดังรูปที่ 1.0



รูปที่ 1.2 แสดงบล็อคไดอะแกรมของระบบฮาร์ดแวร์ต้นแบบ

1.4 การแบ่งเนื้อหาของวิทยานิพนธ์

วิทยานิพนธ์เล่มนี้ประกอบด้วยเนื้อหาที่แบ่งออกเป็นบทได้ทั้งหมด 7 บท ตามลำดับความสำคัญของการเรียนรู้ ในบทแรกได้กล่าวถึงความเป็นมาและความสำคัญของปัญหา ความมุ่งหมายวัตถุประสงค์และ ขอบเขตของงานวิจัย ในส่วนของเนื้อหาตั้งแต่บทที่ 2 ประกอบด้วยรายละเอียดโดยสังเขปดังนี้

บทที่ 2 สัญญาณและระบบโทรทัศน์ กล่าวถึงสัญญาณและระบบเชิงอุปมานของโทรทัศน์ที่ใช้เป็นต้นแบบในการสร้างโดยจะกล่าวถึงระบบของตัวส่ง (Transmitter) เสียเป็นส่วนใหญ่ ซึ่งถือได้ว่าเป็นส่วนของเนื้อหาที่มีความสำคัญเนื่องจากเป็นจุดเริ่มต้นของการศึกษาที่เข้าไปถึงรายละเอียดของส่วนประกอบต่างๆของสัญญาณและระบบที่จะสร้างว่ามีคุณสมบัติเป็นอย่างไร

บทที่ 3 การประมวลผลสัญญาณเชิงเลข (Digital Signal Processing: DSP) เนื่องจากงานวิจัยที่กำลังนำเสนอเป็นการออกแบบระบบเชิงเลขที่ใช้ระบบเชิงอุปมานเป็นต้นแบบในการออกแบบ ดังนั้นเนื้อหาในบทนี้จึงแสดงถึงทฤษฎีที่แสดงความสัมพันธ์ของสัญญาณ ระบบเชิงอุปมานและสัญญาณระบบเชิงเลขเพื่อใช้ในการแปลงระหว่างสองระบบดังกล่าว

บทที่ 4 ภาษาวีเอชดีแอล (Very high speed integrated circuits Hardware Description Language) เนื่องจากงานวิจัยชิ้นนี้เป็นการสร้างสัญญาณจากระบบเชิงเลข ดังนั้นในบทนี้จึงกล่าวถึง

รายละเอียดของการออกแบบระบบเชิงเลขโดยใช้ภาษาบรรยายพฤติกรรมของฮาร์ดแวร์เป็นเครื่องมือในการออกแบบ

บทที่ 5 ลำดับขั้นการดำเนินงาน เนื้อหาของบทนี้จะกล่าวถึงลำดับขั้นของการดำเนินงานหลังจากการศึกษาหลักการและทฤษฎีของสัญญาณและระบบรวมทั้งเครื่องมือที่จะใช้ในการออกแบบโดยเริ่มจากการออกแบบระบบในส่วนของซอร์ฟแวร์ไปจนถึงขั้นตอนสุดท้ายคือการสร้างฮาร์ดแวร์เพื่อการทดสอบคุณสมบัติของระบบ

บทที่ 6 ผลการทดลอง เนื้อหาของบทนี้จะกล่าวถึงผลของการทดลองที่ได้จากระบบฮาร์ดแวร์ต้นแบบโดยใช้เครื่องมือวัดทดสอบทางอิเล็กทรอนิกส์และนำผลที่ได้ไปเปรียบเทียบกับค่าจริงตามคุณสมบัติที่ควรจะเป็นเพื่อการคำนวณหาค่าความคลาดเคลื่อนของสัญญาณที่ได้จากระบบ

บทที่ 7 สรุปการวิจัยและข้อเสนอแนะ หลังจากได้ศึกษาข้อมูล ออกแบบและทำการทดลองจนได้ข้อมูลที่เป็นประโยชน์และบรรลุวัตถุประสงค์ที่ตั้งไว้ เนื้อหาในบทนี้จะกล่าวถึงบทสรุปของการวิจัยและข้อเสนอแนะจากข้อมูลที่ได้จากการทำงานวิจัยชิ้นนี้เพื่อเป็นแนวทางสำหรับผู้ที่กำลังศึกษาวิจัยในเรื่องที่มีเนื้อหาคล้ายหรือใกล้เคียงกัน

วิทยานิพนธ์ฉบับนี้นำเสนอการออกแบบและสร้าง “ตัวเข้ารหัสสัญญาณภาพสีแดง เขียว น้ำเงิน เป็นสัญญาณภาพสีเบ็ดเสร็จแบบใช้วงจรเชิงเลขทั้งหมด” ซึ่งคำว่า “สัญญาณภาพสีเบ็ดเสร็จ” (Composite Color Video Signal: CCVS) เป็นชื่อที่ถูกตั้งขึ้นในช่วงของการพัฒนาระบบโทรทัศน์สี แต่ในปัจจุบันสัญญาณภาพสีเบ็ดเสร็จถูกเรียกสั้นๆว่า “สัญญาณภาพเบ็ดเสร็จ” (Composite Video with Blanking Sync: CVBS) ซึ่งในอดีตหมายถึงสัญญาณภาพขาวดำที่ประกอบด้วยสัญญาณความส่องสว่าง สัญญาณแบล้งกิ้งและสัญญาณซิงโครไนเซชั่น และปัจจุบันเป็นที่เข้าใจกันดีว่าหมายถึงสัญญาณภาพสีเบ็ดเสร็จนั่นเอง เนื่องจากในปัจจุบันไม่มีการส่งกระจายสัญญาณภาพในระบบขาวดำแล้ว แต่อย่างไรก็ตามในเนื้อความของวิทยานิพนธ์ยังคงใช้ชื่อ “สัญญาณภาพสีเบ็ดเสร็จ” ในการอธิบายเพื่อให้ง่ายต่อการทำความเข้าใจในเรื่องของการจัดกลุ่มสัญญาณและจะแตกต่างจากชื่อของวิทยานิพนธ์ซึ่งใช้คำว่า “ตัวเข้ารหัสสัญญาณภาพสีแดง เขียว น้ำเงิน เป็นสัญญาณภาพเบ็ดเสร็จแบบใช้วงจรเชิงเลขทั้งหมด” ดังนั้นขอให้ทำความเข้าใจตรงส่วนนี้เพื่อไม่ให้เกิดความสับสนของการเรียกชื่อสัญญาณดังกล่าว

บทที่ 2

สัญญาณและระบบโทรทัศน์

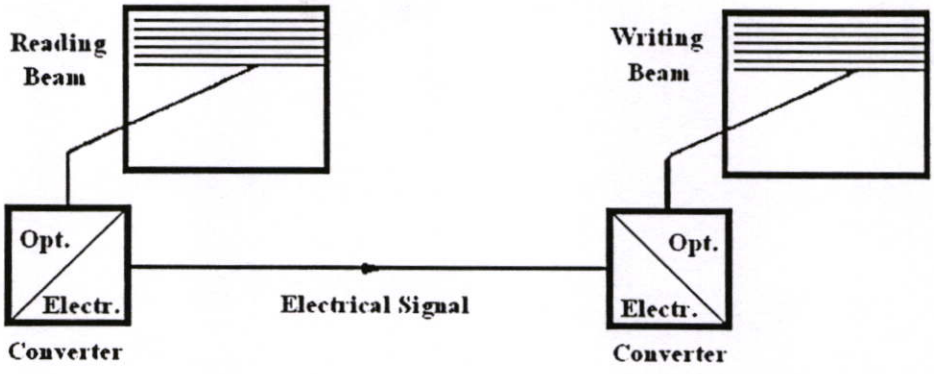
2.1 บทนำ

งานวิจัยชิ้นนี้เป็นการนำเสนอวิธีการออกแบบระบบการสร้างสัญญาณภาพสีเบ็ดเสร็จซึ่งถูกสร้างขึ้นเพื่อใช้กับระบบของการส่งภาพผ่านสื่อกลางอย่างใดอย่างหนึ่งไปยังเครื่องรับโทรทัศน์เช่น จากสถานีส่ง, จากเครื่องใช้ไฟฟ้าเพื่อความบันเทิงชนิดต่างๆ ฯลฯ ดังนั้นจึงมีความจำเป็นอย่างยิ่งที่จะต้องกล่าวถึงรายละเอียดในส่วนประกอบของสัญญาณและระบบที่ใช้ในการสร้างซึ่งงานวิจัยนี้ระบบที่ใช้เป็นต้นแบบก็คือระบบเชิงอุปมาน (Analog)

สัญญาณและระบบโทรทัศน์ที่ถูกคิดค้นขึ้นในยุคเริ่มต้นเป็นระบบขาวดำ หลังจากนั้นได้มีการพัฒนาระบบจนทำให้เกิดระบบโทรทัศน์สีแต่หลักการส่วนใหญ่ก็ยังคงใช้หลักการของระบบขาวดำต่างกันที่ระบบโทรทัศน์สีได้เพิ่มในส่วนของสัญญาณสี (Chrominance: C) บวกไปกับสัญญาณภาพเบ็ดเสร็จ (Composite Video Signal: CVS) ที่ใช้รูปแบบเดิมของระบบขาวดำ ในช่วงของการพัฒนาระบบโทรทัศน์จากขาวดำเป็นระบบสีแต่ละประเทศก็ได้พัฒนาระบบเพื่อใช้เป็นมาตรฐานของตนเอง เช่น สหรัฐอเมริกาและญี่ปุ่นใช้ระบบเอ็นทีเอสซี (National Television System Committee: NTSC) เป็นระบบมาตรฐาน เยอรมันใช้ระบบพาล (Phase Alternation Line: PAL) เป็นระบบมาตรฐาน ฝรั่งเศสใช้ระบบซีแคม (Sequentiel Couleur Avec Memoire: SECAM) เป็นระบบมาตรฐาน ส่วนประเทศไทยของเราก็ได้เลือกใช้ระบบพาลเป็นระบบมาตรฐานที่ใช้มาจนถึงทุกวันนี้ (พ.ศ. 2547) ดังนั้นงานวิจัยที่กำลังนำเสนอจะกล่าวถึงในรายละเอียดเฉพาะระบบพาลเท่านั้น

2.2 หลักการสร้างสัญญาณภาพ

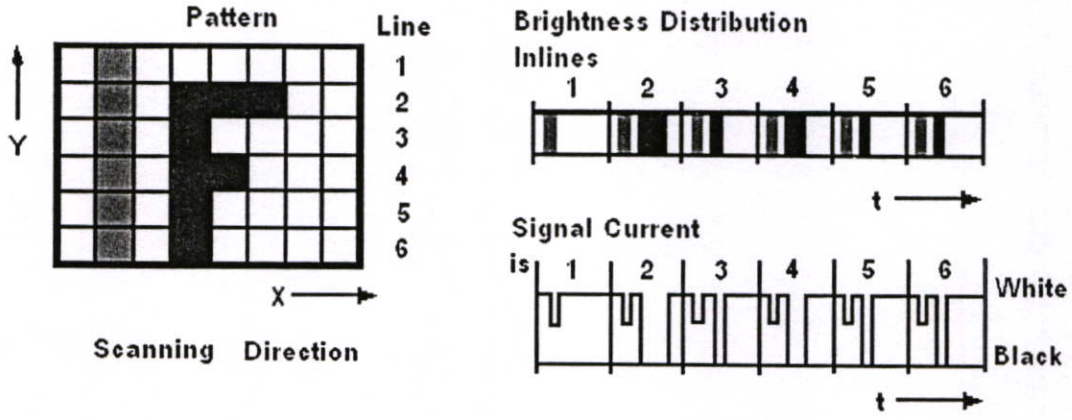
การส่งภาพจากที่ใดที่หนึ่งไปยังอีกที่หนึ่งสามารถทำได้โดยทางด้านเครื่องส่ง (Transmitter) ทำการแปลงข้อมูลความส่องสว่างของภาพให้อยู่ในรูปของสัญญาณทางไฟฟ้าแล้วส่งผ่านสื่อกลางชนิดใดชนิดหนึ่งเช่น อากาศ, แผ่นซีดีรอม (CD Read Only Memory: CD-ROM), ฯลฯ ไปยังเครื่องรับ (Receiver) ซึ่งมีตัวแสดงผลทำงานในทางกลับกันคือ ทำการแปลงสัญญาณภาพที่เป็นสัญญาณทางไฟฟ้าให้เป็นแสงเพื่อสร้างภาพจากการเรียงจุด ระบบของการรับและส่งภาพอย่างง่ายแสดงดังรูปที่ 2.1



รูปที่ 2.1 แสดงหลักการส่งภาพของระบบโทรทัศน์

2.2.1 การกวาดเส้นภาพ (Scanning)

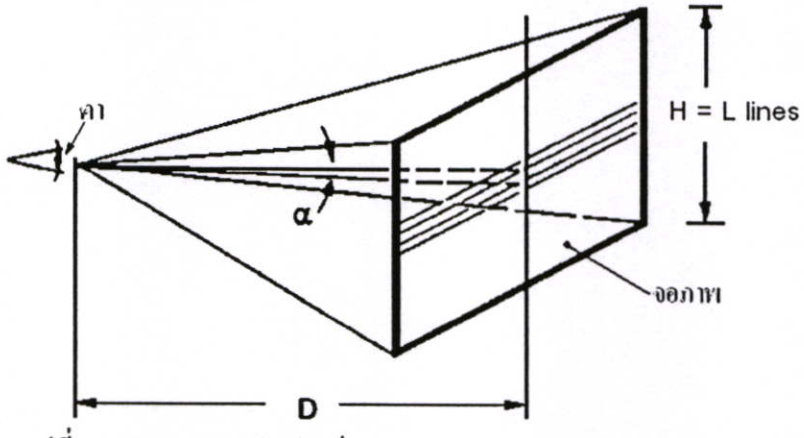
หลักการส่งภาพของระบบโทรทัศน์จะเริ่มจากการแปลงภาพให้เป็นสัญญาณทางไฟฟ้าด้วยตัวเซนเซอร์ (Sensor) โดยการแบ่งภาพออกเป็นเส้นทางแนวนอนและกวาดจากมุมบนซ้ายลงไปมุมล่างขวาตามแนวตรงของการมองภาพจะทำให้ได้สัญญาณที่เปลี่ยนแปลงค่าตามความส่องสว่างเป็นข้อมูลอนุกรมที่สัมพันธ์กับคาบเวลาของการกวาด สัญญาณภาพจะต้องถูกส่งไปคู่กับสัญญาณซิงโครไนเซชั่น (Synchronization) ซึ่งจะใช้ในการควบคุมการกวาดเส้นภาพกลับทางฝั่งเครื่องรับเพื่อสร้างภาพให้ถูกต้องตรงตามตำแหน่งเหมือนกับทางฝั่งภาคส่ง จากรูปที่ 2.2 จะเห็นว่ารูปตัวอักษรทางด้านซ้ายมือถูกแบ่งออกเป็น 6 เส้นและตัวเซนเซอร์ทำการกวาดเส้นภาพทำให้ได้สัญญาณกระแสไฟฟ้า (Is) ซึ่งเปลี่ยนแปลงค่าตามความส่องสว่างของภาพ



รูปที่ 2.2 แสดงการกวาดเส้นภาพและรูปร่างของกระแสไฟฟ้าที่ได้จากเซนเซอร์ (Sensor)

ทางด้านเครื่องส่งตัวแปลงภาพให้เป็นสัญญาณทางไฟฟ้าอย่างหนึ่งที่เป็นที่รู้จักกันเป็นอย่างดีก็คือกล้องวิดีโอ (Video Camera) ส่วนทางด้านเครื่องรับจอภาพแสดงผลเป็นอุปกรณ์ที่ใช้ในการสร้างภาพซึ่งในปัจจุบันมีหลายชนิด โดยทั่วไปจะใช้หลักการเดียวกันแต่จะแตกต่างกันตรงที่วิธีการสร้างแสงให้ปรากฏที่หน้าจอยกตัวอย่างเช่น จอภาพชนิดซีอาร์ที (Cathode Ray Tube: CRT) ใช้การสร้างแสงจากการยิงลำอิเล็กตรอนจากด้านหลังจอผ่านชุดเบี่ยงเบนลำอิเล็กตรอน (Deflection Yoke) และตกกระทบกับสารเรืองแสง (Phosphor) ที่ฉาบไว้ที่หน้าจอทำให้เกิดเป็นแสงจุดเล็กๆหลายแสนจุดรวมกันจนสามารถทำให้ตามนุษย์มองเห็นเป็นภาพที่หน้าจอได้

2.2.2 การแบ่งจำนวนเส้นของภาพ



รูปที่ 2.3 แสดงความสัมพันธ์ของระยะและมุมการมองภาพของสายตา

คุณภาพของการสร้างภาพกลับสามารถวัดได้โดยความละเอียดซึ่งก็คือจำนวนเส้นของการกวาดนั่นเอง จำนวนเส้นที่มากกว่าจะให้รายละเอียดของภาพที่สูงกว่าดังนั้นจึงจำเป็นจะต้องมีการคำนวณหาจำนวนของเส้นที่เหมาะสม โดยมีสองตัวแปรที่จะนำมาพิจารณาคือ ความไว (Acuity) และระยะห่างจากระหว่างจอภาพแสดงผลกับตาของมนุษย์ จากรูปที่ 2.3 ค่าที่ดีที่สุดของระยะห่างก็คือ 5 เท่าของความสูงจอ [5] ที่จุดนี้ตาของมนุษย์จะมองภาพต่อเนื่องไม่มีรอยต่อที่เกิดจากโครงสร้างของเส้นภาพ ภายใต้เงื่อนไขปกติมุมของ α ประมาณ $\alpha 0 = 1.5$ องศา ($\alpha 0$ วัดโดยอ้างอิงจากจุดกึ่งกลางของจอภาพซึ่งเป็นจุดที่ใกล้ที่สุดระหว่างตากับจอภาพ) จากสมการที่ 2.1

$$\tan \alpha = \frac{H / L}{D} \approx \hat{\alpha} \qquad (2.1)$$

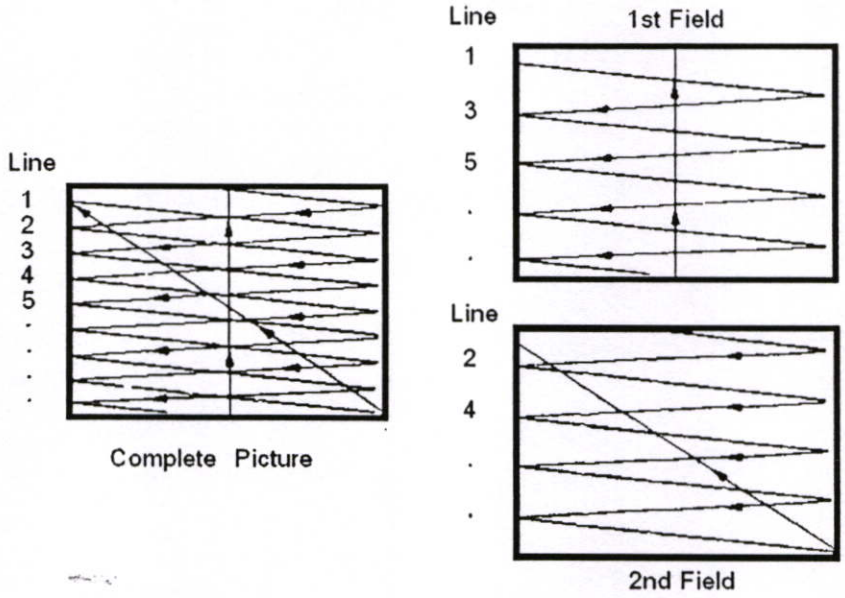
ด้วยค่า $\alpha = \alpha 0 = 1.5$ และ $\tan \alpha 0 \approx 4 \times 10^{-4}$ การคำนวณหาค่าจำนวนของเส้นภาพน้อยที่สุดที่เป็นไปได้สามารถทำได้โดยใช้สูตรการประมาณตามสมการที่ (2.2)

$$L = \frac{2500}{D / H} \tag{2.2}$$

สำหรับค่า D/H = 5 จะได้จำนวนเส้นภาพที่สามารถมองเห็นได้ทั้งหมด 500 เส้น แต่ระบบพาลตามมาตรฐานของ ซีซีไออาร์ (Consultative Committee on International Radio: CCIR) ได้แบ่งพื้นที่ของราสเตอร์ (Raster) ออกเป็นทั้งหมด 625 เส้นและเส้นภาพที่สามารถมองเห็นได้ทั้งหมดคือ 576 เส้น ส่วนที่หายไปก็เนื่องจากช่วงของการกวาดเส้นภาพกลับทางแนวตั้ง

2.2.3 ความถี่ของการฉายภาพซ้ำ (Picture Repetition Frequency)

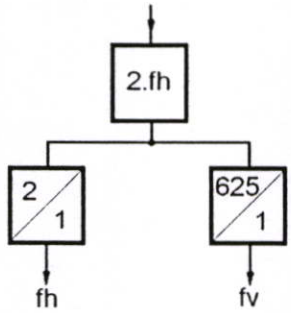
ภาพเคลื่อนไหวที่มองเห็นจากหน้าจอภาพแสดงผลเกิดจากการฉายภาพนิ่งหลายเฟรม (Frame) ต่อเนื่องกันด้วยความเร็วค่าหนึ่งจนทำให้ตามองเห็นเป็นภาพเคลื่อนไหว ดังนั้นการสร้างภาพเคลื่อนไหวต่อเนื่องสิ่งที่ต้องคำนึงถึงอย่างมากคือจำนวนน้อยที่สุดของการฉายภาพซ้ำเพื่อที่จะทำให้ตามนุษย์มองไม่เห็นความไม่ต่อเนื่อง การหาค่าความถี่ของการฉายภาพซ้ำจะต้องใช้คุณลักษณะทางสรีระของตาในการพิจารณา จากการทดลองพบว่าค่าต่ำสุดของการฉายภาพซ้ำอยู่ที่ 16-18 เฟรมต่อวินาที (Frame Per Second: FPS) [5] สำหรับเครื่องฉายภาพยนตร์ใช้อยู่ที่ประมาณ 24 เฟรมต่อวินาที ระบบของโทรทัศน์จึงได้นำค่านี้ไปพิจารณาเทียบความสัมพันธ์กับความถี่ไฟบ้านที่ใช้อยู่ในขณะนั้น (50 Hz. ในยุโรป) ดังนั้นระบบพาลที่ถูกพัฒนาขึ้นโดยชาวเยอรมนีจึงเลือกใช้ค่าความถี่ของการฉายภาพซ้ำเท่ากับ 25 เฟรมต่อวินาที แต่อย่างไรก็ตามการฉายภาพซ้ำจำนวน 25 เฟรมต่อวินาทีนั้นไม่เพียงพอ เนื่องจากตามนุษย์สามารถมองเห็นการกระพริบ (Flicker) สำหรับเครื่องฉายภาพยนตร์ได้ใช้การขัดจังหวะของ ฟลิคเกอร์ชัตเตอร์ (Flicker Shutter) เพิ่มความถี่ของการฉายภาพซ้ำเป็นสองเท่าในการแก้ปัญหานี้ แต่กับระบบโทรทัศน์ไม่สามารถใช้วิธีการนี้ได้ ดังนั้นระบบโทรทัศน์ได้ใช้อีกวิธีหนึ่งในการแก้ปัญหาคือ ใช้การกวาดเส้นภาพแบบสอดแทรกความหมายก็คือภาพหนึ่งภาพหรือหนึ่งเฟรมจะถูกแบ่งออกเป็นสองฟิลด์ (Field) ประกอบด้วยฟิลด์เส้นคี่ 1..3..5..7... และฟิลด์เส้นคู่ 2..4..6..8... การฉายภาพหนึ่งเฟรมต้องใช้ภาพทั้งสองฟิลด์ดังกล่าว ดังนั้นการฉายภาพด้วยความถี่ 25 เฟรมต่อวินาทีจะทำให้ได้ความถี่ของการกระพริบเท่ากับ 50 ครั้งต่อวินาทีหรือ 50 ฟิลด์ต่อวินาทีทำให้สามารถแก้ปัญหาการมองเห็นภาพกระพริบของตามนุษย์ได้ รูปที่ 2.4 แสดงการแบ่งการกวาดเส้นภาพ



รูปที่ 2.4 แสดงการแบ่งการกวาดเส้นภาพ

การสร้างภาพกลับจากภาพที่ถูกแบ่งออกเป็นสองฟิลด์ของจอภาพแสดงผล การกวาดเส้นภาพแบบสอดแทรกจะต้องมีความแม่นยำเพื่อหลีกเลี่ยงการเกิดราสเตอร์ ในระบบที่ใช้จำนวนเส้นภาพเป็นจำนวนคี่เช่น โทรทัศน์ระบบพาลที่ใช้จำนวนเส้นภาพเท่ากับ 625 เส้น ช่วงต่อระหว่างฟิลด์แรกกับฟิลด์ที่สอง การกวาดเส้นภาพกลับเพื่อการเริ่มต้นกวาดเส้นภาพฟิลด์ที่สองจะทำที่ครึ่งหลังของการกวาดเส้นสุดท้ายของฟิลด์แรก ด้วยวิธีการนี้จึงไม่จำเป็นต้องใช้สัญญาณพิเศษในการแก้ปัญหาความคลาดเคลื่อนคาบเวลาของสองฟิลด์ดังกล่าว ด้วยเหตุนี้เส้นภาพจำนวน 312.5 เส้น 50 ฟิลด์ถูกฉายในหนึ่งวินาทีเรียกว่า ความถี่การฉายภาพซ้ำของฟิลด์ (Field Repetition) หรือความถี่ของการกวาดเส้นภาพทางแนวตั้ง (Vertical Frequency: FV) จากค่าดังกล่าว FV = 50 Hz. เป็นผลให้สามารถคำนวณหาค่าความถี่ของการกวาดเส้นภาพทางแนวนอน (Horizontal Frequency: FH) ได้ด้วยความสัมพันธ์ตามสมการที่ (2.3) คาบเวลาของการกวาดเส้นภาพทางแนวนอนจึงมีค่าเท่ากับ 64 uS. และคาบเวลาของการกวาดเส้นภาพทางแนวตั้งมีค่าเท่ากับ 20 mS. แต่การกวาดเส้นภาพทางแนวนอนและแนวตั้งดังกล่าวจะต้องซิงโครไนซ์กัน (Synchronize) วิธีที่สามารถทำได้ง่ายก็คือการหารจากความถี่สองเท่าของความถี่การกวาดเส้นภาพทางแนวนอนแสดงดังรูปที่ 2.5

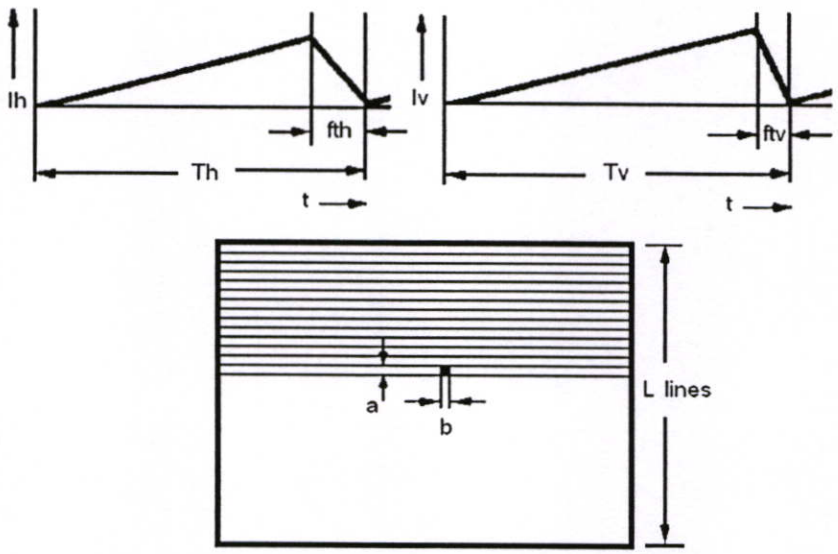
$$FH = 25 \times 625 = 50 \times 312.5 = 15625\,Hz. \qquad (2.3)$$



รูปที่ 2.5 แสดงการสร้างความถี่ที่ใช้ในการกวาดเส้นภาพ

2.2.4 แถบความถี่ของสัญญาณภาพ (Bandwidth of Picture Signal)

ความละเอียดของภาพที่ส่งออกไปสามารถวัดได้โดยจำนวนของเส้นกวาด ด้วยค่าความละเอียดที่เท่ากันของสองทิศทางทั้งแนวนอนและแนวตั้งดังนั้นความกว้างของจุดภาพ (b ของรูปที่ 2.6) จะต้องมีขนาดเท่ากับขนาดของเส้นภาพ (a ของรูปที่ 2.6) เส้นภาพถูกกวาดจากทางด้านซ้ายไปทางด้านขวาของจอภาพหลังจากนั้นเส้นภาพจะกวาดกลับ(สะบัดกลับ)เพื่อเริ่มการกวาดเส้นภาพต่อไปซึ่งในช่วงนี้เส้นกวาดจะต้องถูกทำให้มืดเนื่องจากไม่มีข้อมูลภาพเรียกว่าช่วงแบล้งกิ้งทางแนวนอน (Horizontal Blanking) ทางด้านแนวตั้งก็มีหลักการคล้ายกันคือหลังจากการกวาดภาพจนถึงด้านล่างของจอภาพเส้นภาพจะถูกกวาดกลับไปที่ด้านบนของจอและถูกทำให้มืดเรียกว่า ช่วงแบล้งกิ้งทางแนวตั้ง (Vertical Blanking) ช่วงแบล้งกิ้งทั้งทางแนวนอนและแนวตั้งอีกชื่อหนึ่งเรียกว่าช่วงฟลายแบ็ค (Fly Back) ตามมาตรฐานของซีซีไออาร์กำหนดคาบเวลาตามสมการที่ (2.4)-(2.5)



รูปที่ 2.6 แสดงความละเอียดและคาบเวลาของการกวาดเส้นภาพ

$$tfh = 0.18 \times Th = 11.52 \mu s . \quad (2.4)$$

$$tfv = 0.08 \times Tv = 1.6 ms . \quad (2.5)$$

เมื่อ

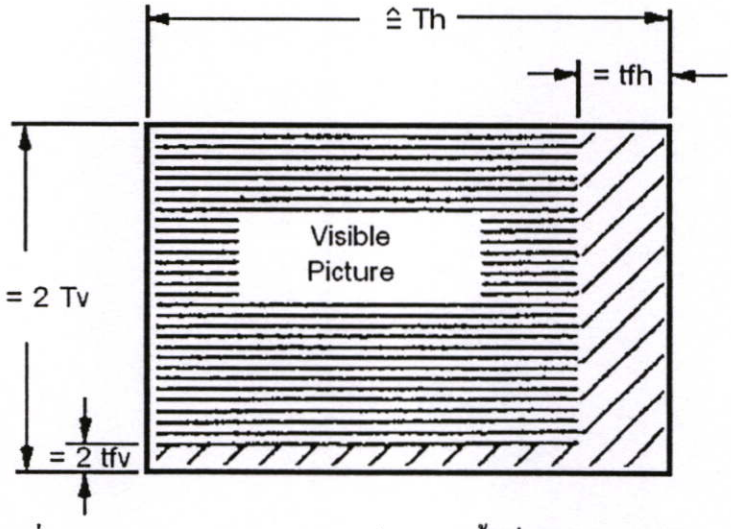
tfh คือ เวลาของช่วงแบล้งกิ้งทางแนวนอน

tfv คือ เวลาของช่วงแบล้งกิ้งทางแนวตั้ง

Th คือ เวลาของการกวาดเส้นภาพทางแนวนอน

Tv คือ เวลาของการกวาดเส้นภาพทางแนวตั้ง

ด้วยเหตุนี้ข้อมูลของสัญญาณภาพจะเท่ากับ Th(1-0.18) = 52.48 uS.ของเวลาที่ใช้ในการกวาดเส้นภาพทั้งหมดทางแนวนอน (Th) และ 625(1-0.08) = 575 เส้น สำหรับการกวาดเส้นภาพทางแนวตั้ง ทำให้พื้นที่ในการมองเห็นภาพลดลงอันเนื่องมาจากเวลาของการกวาดเส้นภาพกลับดังกล่าว รูปที่ 2.7 แสดงผลของช่วงฟลายแบ็คที่ทำให้พื้นที่ของราสเตอร์ลดลง



รูปที่ 2.7 แสดงผลของช่วงฟลายแบ็คทำให้พื้นที่ของราสเตอร์ลดลง

สำหรับจอภาพโทรทัศน์อัตราส่วนของความยาวต่อความสูงเท่ากับ 4:3 ด้วยขนาดที่เท่ากันของความกว้างทางแนวนอนและขนาดของเส้นเพื่อสร้างจุดของแสงที่มีรูปทรงสี่เหลี่ยมจัตุรัส ดังนั้นถ้าพิจารณาอัตราส่วนของจอและจำนวนทั้งหมดของเส้นภาพทำให้สามารถคำนวณหาค่าความละเอียดทางแนวนอนได้ตามสมการที่ (2.6)

$$\frac{4}{3} \times 625(1 - 0.08) = 767 \qquad (2.6)$$

ดังนั้นอาจกล่าวได้ว่าจำนวนเส้นภาพทั้งหมดเป็นความละเอียดทางด้านแนวตั้งส่วนความละเอียดทางด้านแนวนอนหาได้จากสมการ (2.6) ทำให้สามารถคำนวณหาจำนวนของจุดภาพ (Picture Element: Pixel) ทั้งหมดบนหน้าจอได้ตามสมการที่ (2.7)

$$\frac{4}{3} \times 625(1 - 0.08) \times 625(1 - 0.08) = 440833 \text{ Pixels.} \qquad (2.7)$$

จำนวนของจุดภาพที่ได้จากสมการ (2.7) สามารถคำนวณหาเวลาที่ใช้ในการส่งทั้งหมดได้ตามสมการที่ (2.8)

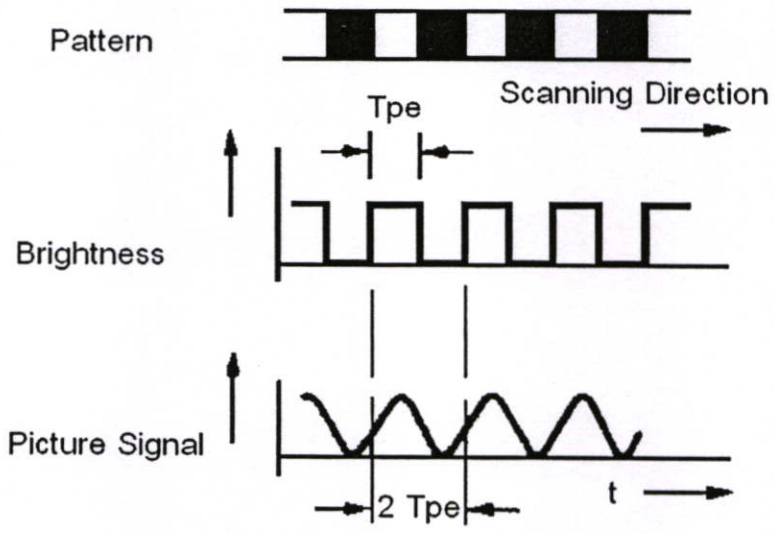
$$64\,\mu s\,(1 - 0.18) \times 625(1 - 0.08) = 30.176\,ms \qquad (2.8)$$

จากสมการที่ (2.7) และ (2.8) เวลาที่ใช้ในการกวาดเส้นภาพให้ได้หนึ่งจุดภาพ (Pixel) หรือ Tpe สามารถคำนวนหาได้จากสมการที่ (2.9)

$$Tpe = \frac{30.176\,ms\ .}{440833} = 0.0684\,\mu s \qquad (2.9)$$

ภาพที่ทำให้เกิดความถี่ของสัญญาณภาพสูงสุดคือภาพที่มีสีดำสลับกับสีขาวจุดภาพเว้นจุดภาพ (Pixel) ดังนั้นคาบเวลาของสัญญาณภาพนี้คือ Tp = 2 Tpe โดยปกติภาพที่มีสีดำสลับกับสีขาวดังกล่าว จะต้องใช้คลื่นสแควเวฟ (Square Wave) ในการสร้าง แต่เนื่องจากลำอิเล็กตรอนที่ใช้ในการสร้างแสงมีเส้นผ่าศูนย์กลางที่จำกัดการใช้คลื่นสแควเวฟจะทำให้ช่วงการเปลี่ยนแปลงจากขาวเป็นดำถูกลบความคมแสดงดังรูปที่ 2.8 หรืออาจกล่าวได้ว่าการใช้คลื่นสแควเวฟเกินความจำเป็นเนื่องจากข้อจำกัดของลำอิเล็คตรอนดังกล่าว ดังนั้นการส่งเฉพาะความถี่มูลฐาน (Fundamental Frequency) ของคลื่นสแควเวฟจึงเพียงพอสำหรับการสร้างภาพ ดังนั้นจึงสามารถคำนวณหาค่าความถี่สูงสุดที่เป็นไปได้สำหรับสัญญาณภาพได้จากสมการที่ (2.10)

$$Fp_{\max} = \frac{1}{Tp} = 7.3 \text{ MHz.} \qquad (2.10)$$



รูปที่ 2.8 แสดงการลดความคมของสัญญาณภาพเนื่องจากเส้นผ่าศูนย์กลางที่จำกัดของลำอิเล็กตรอน

เมื่อพิจารณาเส้นผ่าศูนย์กลางที่จำกัดของลำอิเล็กตรอนจะพบว่าความละเอียดทางแนวตั้งจะลดลงเปรียบเทียบกับสมการที่ 2.10 ซึ่งสามารถอธิบายได้โดยใช้ค่าคงที่เคล (KELL Factor: K) [5] ด้วยค่าของ $K \approx 2/3$ ซีซีไออาร์จึงได้กำหนดแถบความถี่ของสัญญาณภาพเท่ากับ 5 MHz.

จากทั้งหมดที่ได้กล่าวในข้างต้นเป็นหลักการสร้างสัญญาณภาพที่เป็นหลักการโดยรวมทั้งระบบขาวดำและระบบสี อย่างไรก็ตามทั้งสองระบบดังกล่าวยังมีส่วนที่แตกต่างกันได้แก่จุดภาพ (Pixel) สำหรับระบบขาวดำจุดภาพที่ใช้ในการสร้างภาพจะไม่มีสีแต่จะมีเฉพาะความส่องสว่าง (Lumiance: Y) ที่มีระดับต่างกัน แต่สำหรับระบบสีจุดภาพสามารถให้แสงที่มีสีได้เนื่องจากใช้การสร้างจุดภาพจากแสงที่เป็นแม่สีได้แก่ สีแดง, สีเขียวและสีน้ำเงินซึ่งจะกล่าวถึงรายละเอียดในหัวข้อต่อไป

2.3 สัญญาณภาพสีเบ็ดเสร็จ (Composite Color Video Signal: CCVS)

ในปัจจุบันเครื่องรับโทรทัศน์ถูกออกแบบมาให้สามารถรับสัญญาณภาพได้หลายรูปแบบด้วยกันเช่น สัญญาณอาร์เอฟ (Radio Frequency: RF), สัญญาณซูปเปอร์วิดีโอ (Super Video: S-Video), สัญญาณคอมโพเน้นท์ (Component YPbPr), สัญญาณอาร์จีบีจากสคาร์ท (Scart) แต่อย่างไรก็ตามสำหรับระบบการส่งกระจาย (Broadcast) ภาพในระบบเชิงอุปมาน สัญญาณภาพที่จำเป็นต้องใช้ในการ

มอดูเลตก็คือ สัญญาณภาพสีเบ็ดเสร็จ (Composite Color Video Signal: CCVS) นอกจากนั้นสัญญาณขาออกที่ได้จากเครื่องใช้ไฟฟ้าเพื่อความบันเทิงบางชนิดเช่น เครื่องเล่นดีวีดี (DVD), วีซีดี (VCD) ก็ยังเป็นสัญญาณภาพชนิดนี้อยู่เช่นกัน

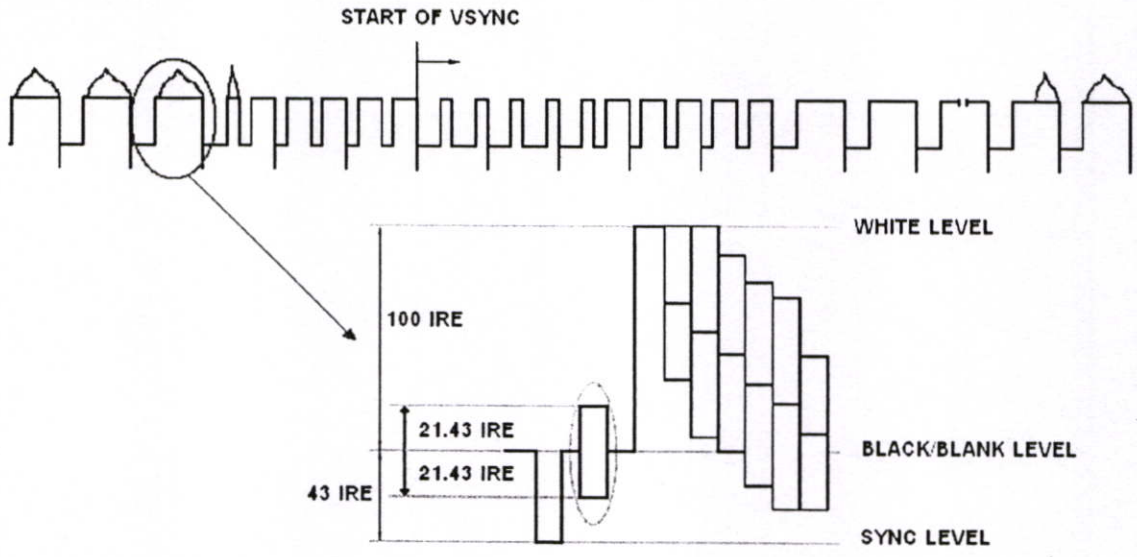
สัญญาณภาพสีเบ็ดเสร็จถูกสร้างขึ้นระหว่างช่วงรอยต่อของการเปลี่ยนระบบการส่งกระจายภาพโทรทัศน์จากระบบขาวดำไปเป็นระบบสี วัตถุประสงค์หลักของการสร้างก็คือเพื่อการส่งสัญญาณภาพสีในการรับชมกับโทรทัศน์ระบบสีในขณะที่โทรทัศน์ระบบเก่าที่เป็นระบบขาวดำก็ยังคงต้องสามารถรับชมสัญญาณภาพจากสถานีส่งได้เหมือนเดิม หลักการคือส่งสัญญาณภาพเบ็ดเสร็จ (Composite Video) ซึ่งเป็นสัญญาณเดิมที่ใช้อยู่ของระบบขาวดำในการส่งความส่องสว่างของภาพกับสัญญาณสี (Chrominance) ไปในเวลาเดียวกันด้วยการมัลติเพล็กทางความถี่ [5] เพื่อส่งข้อมูลสีของภาพโดยใช้ย่านความถี่ของช่องสัญญาณระบบโทรทัศน์ขาวดำเดิมที่ใช้อยู่ซึ่งถือได้ว่าเป็นการบีบอัดข้อมูลภาพวิธีหนึ่ง [6]

2.3.1 ส่วนประกอบของสัญญาณภาพสีเบ็ดเสร็จระบบพาล

จากหลักการสร้างภาพที่ได้กล่าวถึงไปแล้วนั้นพบว่าสัญญาณภาพจะต้องประกอบด้วยสัญญาณที่จำเป็นสำหรับการสร้างภาพกลับทางด้านเครื่องรับได้แก่

- สัญญาณแบล้งกิ้ง (Blanking) เป็นสัญญาณที่ใช้ในการควบคุมความสว่างในช่วงของการสบัดกลับของเส้นภาพเพื่อเริ่มการกวาดเส้นหรือฟิลด์ต่อไป
- สัญญาณซิงโครไนเซชั่น (Synchronization) เป็นสัญญาณที่ใช้ในการควบคุมการกวาดเส้นภาพระหว่างเครื่องส่งและเครื่องรับให้มีคาบเวลาที่ตรงกัน
- สัญญาณความส่องสว่าง (Luminance: Y) หรือสัญญาณภาพขาวดำ
- สัญญาณสี (Chrominance: C) เป็นสัญญาณข้อมูลสีที่ผ่านการมอดูเลตประกอบด้วยสัญญาณความแตกต่างสี (Color Difference) R-Y, B-Y และสัญญาณเบิรสต์

สำหรับระบบโทรทัศน์ขาวดำสัญญาณภาพจะประกอบด้วยสัญญาณซิงโครไนเซชั่น, สัญญาณแบล้งกิ้งและสัญญาณความส่องสว่างเรียกสัญญาณภาพที่เกิดจากการบวกกันของสามสัญญาณดังกล่าวว่า “สัญญาณภาพเบ็ดเสร็จ” (Composite Video Signal: CVS) [5] ส่วนของระบบโทรทัศน์สีมีความแตกต่างก็คือได้บวกสัญญาณสีเข้ากับสัญญาณภาพเบ็ดเสร็จของระบบขาวดำและเรียกสัญญาณนี้ว่า “สัญญาณภาพสีเบ็ดเสร็จ” (Composite Color Video Signal: CCVS) [5] ตำราบางเล่มเรียกสัญญาณนี้ว่า “สัญญาณซีวีบีเอส” (Composite Video Blanking Sync: CVBS)



รูปที่ 2.9 แสดงสัญญาณภาพสีเบ็ดเสร็จของสัญญาณภาพแท่งสี 75 %

2.3.1.1 สัญญาณแบล้งกิ้ง

การกวาดเส้นภาพจะเริ่มจากมุมบนซ้ายและสิ้นสุดที่มุมล่างขวาของจอภาพ หลังจากทำการกวาดเส้นภาพสิ้นสุดในแต่ละเส้นหรือแต่ละฟิลด์ เส้นกวาด (หรือลำอิเล็กตรอน) จะถูกลดความเข้มเพื่อทำให้แสงที่จอภาพมืดลงชั่วขณะเพื่อเริ่มการกวาดเส้นภาพในเส้นหรือฟิลด์ต่อไป เหตุผลที่ต้องดับเส้นกวาดก็เนื่องจากว่าไม่มีข้อมูลสัญญาณภาพในช่วงของการสบัดกลับ ระดับแรงดันของสัญญาณในช่วงนี้จึงถูกเรียกว่า "ระดับแบล้งกิ้ง" ซึ่งโดยทั่วไปจะมีขนาดเท่ากับระดับดำของสัญญาณภาพแต่จะมีเพียงบางระบบเท่านั้นที่ระดับแบล้งกิ้งและระดับดำมีขนาดแตกต่างกันเช่น ระบบเอ็นทีเอสซีเอ็ม (NTSC-M)หรือระบบพาลเอ็ม (PAL-M) [7] เป็นต้น เนื่องจากการกวาดเส้นภาพเป็นการกวาดแบบสองมิติดังนั้นสัญญาณแบล้งกิ้งจึงประกอบด้วย สัญญาณแบล้งกิ้งทางแนวนอน (Horizontal Blanking)และสัญญาณแบล้งกิ้งทางแนวตั้ง (Vertical Blanking) คาบเวลาของทั้งสองสัญญาณแสดงโดยสมการที่ (2.4) – (2.5)

2.3.1.2 สัญญาณซิงโครไนเซชั่น

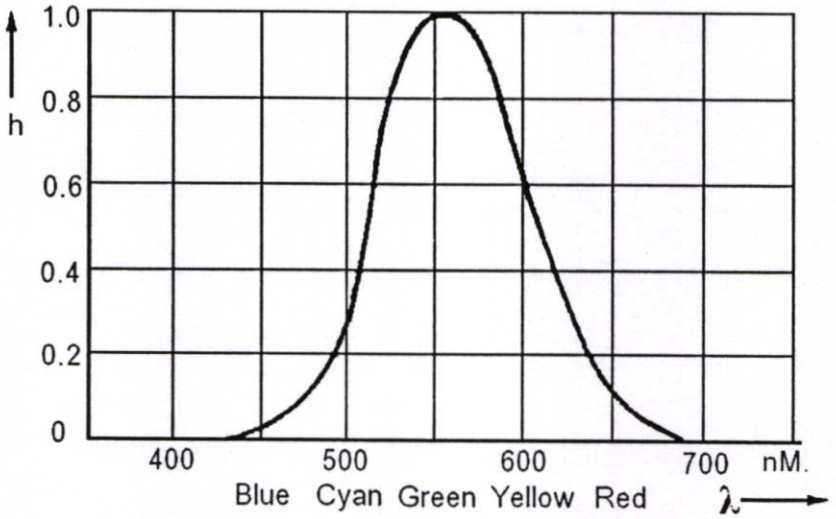
การส่งสัญญาณภาพของระบบโทรทัศน์เป็นการส่งเส้นภาพกวาดแบบสอดแทรก ลำดับของการเรียงเส้นระหว่างเครื่องส่งกับเครื่องรับจะต้องตรงกันเพื่อให้ได้ภาพที่ถูกต้อง สัญญาณซิงโครไนเซชั่นเป็นสัญญาณถูกส่งมาจากเครื่องส่งหรือต้นกำเนิดของสัญญาณภาพใช้เพื่อควบคุมการกวาดเส้นภาพทางฝั่งเครื่องรับและเนื่องจากจอภาพเป็นการแสดงผลแบบสองมิติคือมีด้านกว้างกับด้านยาวดังนั้นจึงต้องใช้สองสัญญาณในการควบคุมซึ่งประกอบด้วย สัญญาณซิงโครไนเซชั่นทางแนวนอน

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

(Horizontal Synchronization) และสัญญาณซิงโครไนเซชั่นทางแนวตั้ง (Vertical Synchronization) สำหรับสัญญาณภาพสีเบ็ดเสร็จสัญญาณซิงโครไนเซชั่นเป็นสัญญาณที่มีระดับแรงดันต่ำที่สุด

2.3.1.3 สัญญาณความส่องสว่าง

แสงที่ตามนุษย์มองเห็นเกิดจากการเรดิเอท (Radiate) ของคลื่นสนามแม่เหล็กไฟฟ้า แต่ละช่วงของความถี่ของแสงทำให้ตามนุษย์มองเห็นเป็นสีที่มีความสว่างไม่เท่ากันเช่น ความยาวคลื่นประมาณ 700 nM. ทำให้ตามองเห็นเป็นแสงสีแดง หรือแสงจากดวงอาทิตย์เป็นแสงที่มีองค์ประกอบของความถี่กว้างทำให้ตามนุษย์มองเห็นเป็นสีขาว ความไวของตามนุษย์เปลี่ยนแปลงตามความยาวคลื่นของแสงโดยแสงที่มีพลังเท่ากันแต่มีความยาวคลื่นต่างกันจะทำให้ตามองเห็นความส่องสว่างมีค่าต่างกันแสดงดังรูปที่ 2.10 จากรูปจะเห็นว่าแสงในช่วงความยาวคลื่นประมาณ 435 nM. ตามนุษย์จะมองเห็นเป็นสีน้ำเงินและมีความส่องสว่างน้อยกว่าแสงสีเขียวซึ่งมีความยาวคลื่นประมาณ 546 nM.



รูปที่ 2.10 แสดงค่าความไว (h) ต่อความส่องสว่างของตามนุษย์ต่อแสงที่มีความยาวคลื่นค่าต่างๆ

ในยุคแรกๆของการพัฒนาระบบโทรทัศน์เริ่มจากระบบขาวดำ แต่ในความเป็นจริงภาพที่เรามองเห็นเป็นภาพสี ดังนั้นสำหรับระบบขาวดำจึงใช้การส่งค่าความส่องสว่างของภาพตามความไวของตามนุษย์แทนการส่งสีของภาพซึ่งในขณะนั้นยังไม่สามารถทำได้ ตัวเซ็นเซอร์ที่มีคุณสมบัติในการแปลงภาพสีให้เป็นความส่องสว่างตามความไวของตามนุษย์ก็คือกล้องวิดีโอขาวดำ (Black & White Video Camera) แต่เนื่องจากความต้องการส่งสัญญาณภาพสีจึงจำเป็นต้องใช้แหล่งกำเนิดหรือตัวเซ็นเซอร์ที่ให้สัญญาณขาออกเป็นสัญญาณสีเช่น กล้องวิดีโอสี (Color Video Camera) ดังนั้นจึงต้องมีการแปลงกลับสัญญาณสีให้เป็นสัญญาณความส่องสว่าง (Luminance: Y) ตามมาตรฐานซีไออี (CIE

63433

illuminate C) สามารถคำนวณได้จากสมการที่ (2.11) ความหมายก็คือความส่องสว่างสามารถสร้างได้จากผลรวมของแสงที่เป็นแม่สีทางแสงได้แก่สีแดง, เขียวและน้ำเงินที่มีค่าสัมประสิทธิ์แตกต่างกันอันเนื่องมาจากความไวต่อความส่องสว่างของตากับแม่สีของแสงทั้งสามดังกล่าวมีค่าไม่เท่ากัน

$$Y = 0.299R + 0.587G + 0.114B \tag{2.11}$$

2.3.1.4 สัญญาณสี

หลังจากได้พัฒนาและส่งกระจายภาพในระบบขาวดำมาระยะหนึ่งแต่ละประเทศก็ได้พัฒนาระบบโทรทัศน์สีเพื่อใช้เป็นระบบมาตรฐานของตนเอง แต่ทุกระบบของการเข้ารหัสสีพบกับปัญหาเดียวกันคือ จะต้องส่งสัญญาณภาพสีที่มีข้อมูลความส่องสว่างของภาพเพื่อทำให้โทรทัศน์ที่เป็นระบบขาวดำรับชมภาพได้เหมือนเดิม จากการศึกษาพบว่าสีที่ปรากฏในธรรมชาติสามารถสร้างได้จากการรวมกันของแสงที่เป็นแม่สีหลักได้แก่ สีแดง, เขียวและน้ำเงิน ฉะนั้นนั้นการส่งสัญญาณสีแดง,เขียวและน้ำเงินของระบบโทรทัศน์ จึงเพียงพอกับการสร้างภาพที่มีสีตามธรรมชาติทางด้านเครื่องรับ [5]

ถึงแม้ว่าภาพสีสามารถสร้างได้จากการรวมกันของแสงที่เป็นแม่สีหลักทั้งสาม แต่ด้วยข้อจำกัดของความเข้ากันได้ (Compatibility) ของระบบการส่งกระจายภาพระหว่างระบบสีและขาวดำ การส่งกระจายภาพโทรทัศน์จึงต้องใช้การส่งสัญญาณความแตกต่างสีไปกับสัญญาณภาพเบ็ดเสร็จแทนการส่งสัญญาณของแม่สีทั้งสามโดยตรงซึ่งสามารถทำให้แก้ปัญหาเรื่องข้อจำกัดของความเข้ากันได้ เนื่องจากทั้งสองสัญญาณนี้เพียงพอสำหรับการสร้างสัญญาณที่จำเป็นในการสร้างภาพกลับทางฝั่งเครื่องรับได้ทั้งระบบขาวดำและสี วิธีการนี้จึงถูกนำมาใช้แก้ปัญหาดังกล่าวกับทุกระบบของการเข้ารหัสสี สามารถแสดงความสัมพันธ์ได้ดังรูปที่ 2.11

สำหรับระบบโทรทัศน์ขาวดำสามารถใช้สัญญาณความส่องสว่างที่ส่งมาได้โดยตรงในขณะที่ระบบโทรทัศน์สีต้องใช้สัญญาณความแตกต่างสีในการสร้างกลับสัญญาณแม่สีทางแสงดังนี้

$$R = (R - Y) + Y \tag{2.12}$$

$$G = (G - Y) + Y \tag{2.13}$$

$$B = (B - Y) + Y \tag{2.14}$$

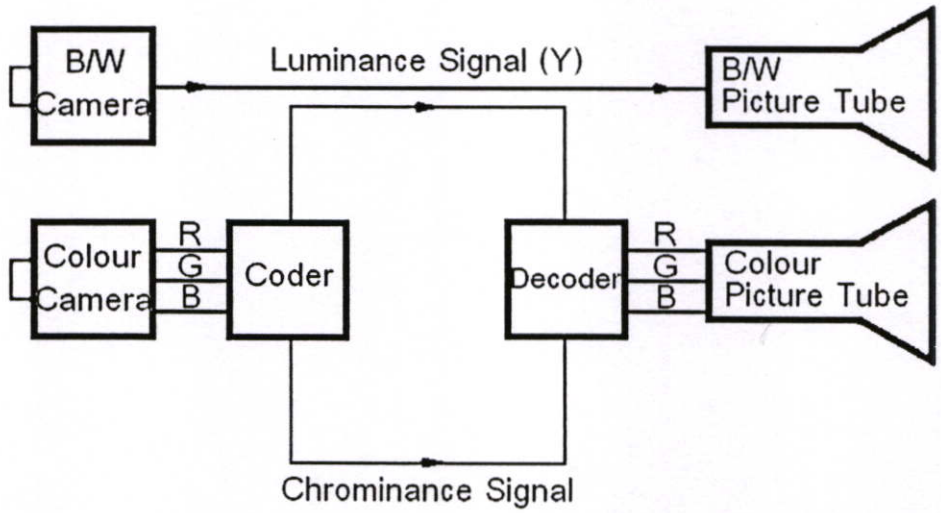
สัญญาณความแตกต่างสีจะมีเฉพาะสัญญาณ R-Y และ B-Y ทางฝั่งเครื่องรับสามารถสร้างสัญญาณ G-Y ได้ดังนี้คือ จาก (2.11) ทำให้ทราบว่า

$$Y = 0.299Y + 0.587Y + 0.114Y \tag{2.15}$$

(2.11)-(2.15) จะได้

$$0 = 0.299(R - Y) + 0.587(G - Y) + 0.114(B - Y)$$

$$G - Y = -\left(\frac{0.299}{0.587}\right)(R - Y) - \left(\frac{0.114}{0.587}\right)(B - Y) \tag{2.16}$$

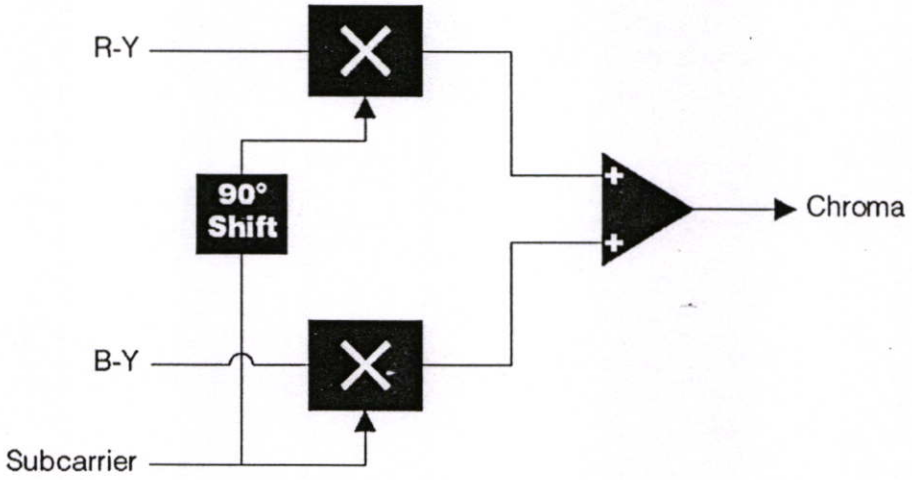


รูปที่ 2.11 แสดงรูปแบบการส่งสัญญาณภาพเพื่อความเข้ากันได้ของระบบโทรทัศน์สีและขาวดำ

สัญญาณแบนด์มูลฐาน (Base Band) ของสัญญาณความแตกต่างสีไม่สามารถส่งไปพร้อมกับสัญญาณภาพเบ็ดเสร็จได้โดยตรงด้วยการบวก เนื่องจากมีสเปกตรัม (Spectrum) อยู่ในช่วงเดียวกัน ดังนั้นการเข้ารหัสสีจึงถูกนำมาใช้ในการแก้ปัญหาดังกล่าว สัญญาณความแตกต่างสีของระบบพาลและเอ็นทีเอสซีได้ใช้การมอดูเลตเชิงแอมปลิจูดแบบแถบข้างคู่ขจัดคลื่นพาห์ (Amplitude Modulation Double Side Band Suppress Carrier: AMDSBSC) กับคลื่นพาหะที่มีค่าความถี่ที่เหมาะสมค่าหนึ่ง [5] ส่วนระบบซีแคมได้ใช้การมอดูเลตเชิงความถี่ (Frequency Modulation: FM)

สัญญาณความแตกต่างสี B-Y และ R-Y หลังจากการมอดูเลตเชิงแอมปลิจูดแบบแถบข้างคู่ขจัดคลื่นพาห์ทำให้สเปกตรัม (Spectrum) ถูกย้ายไปอยู่ที่ความถี่พาหะรองของสัญญาณสีเพื่อทำการมัลติเพล็กทางความถี่ (Frequency Division Multiplex: FDM) กับสัญญาณภาพเบ็ดเสร็จ [5] วัตถุประสงค์ก็คือ เพื่อทำให้สัญญาณสีที่เพิ่มเข้ามาใช้ย่านความถี่ช่องสัญญาณเดิมของระบบขาวดำ ในขณะที่สัญญาณพาหะรองของสัญญาณ B-Y และ R-Y มีเฟสต่างกัน 90° เพื่อทำการมัลติเพล็กทาง

เวลา (Time Division Multiplex: TDM) ของทั้งสองสัญญาณดังกล่าวและเรียกการมอดูเลตที่ใช้สัญญาณพาหะรองที่มีเฟสต่างกัน 90° นี้ว่า ควอดราเจอร์ (Quadrature Modulation) ซึ่งมีบล็อคไดอะแกรมแสดงดังรูปที่ 2.12

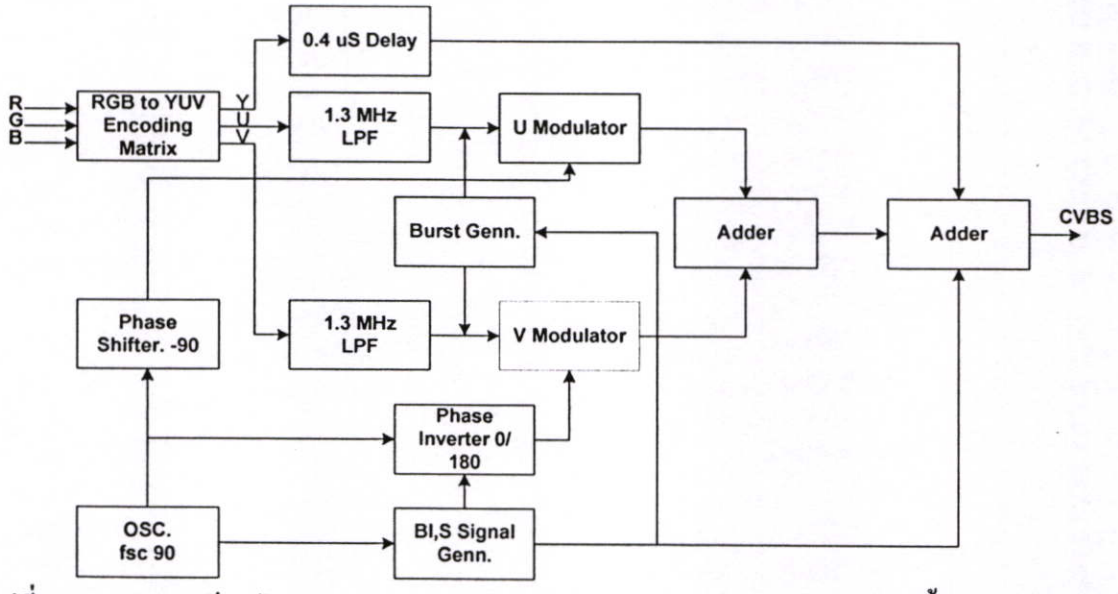


รูปที่ 2.12 แสดงควอดราเจอร์มอดูเลเตอร์

สำหรับระบบการสร้างสัญญาณสีที่ใช้งานจริงนั้นสัญญาณความแตกต่างสี B-Y, R-Y จะถูกลดทอนลงและเรียกสัญญาณที่ถูกลดทอนนี้ว่าสัญญาณ U และ V เพื่อแก้ปัญหาการมอดูเลตเกิน (Over Modulation) ของภาคมอดูเลตสัญญาณภาพ [5] กล่าวคือ เนื่องจากระบบพาลบีและจี ใช้การมอดูเลตสัญญาณภาพทางลบ (Negative) สัญญาณซิงโครไนเซชั่นเป็นช่วงสัญญาณที่ทำให้สัญญาณพาหะมีขนาดสูงสุดอยู่ที่ 100% และสัญญาณภาพที่มีความส่องสว่างมากจะทำให้สัญญาณพาหะมีขนาดลดลง ในกรณีที่ไม่มีการลดทอนสัญญาณความแตกต่างสี สัญญาณแท่งสี 100% ทำให้เกิดการมอดูเลตเกินเมื่อสัญญาณสีมีการอิ่มตัวสูงสุดเท่ากับ 78% แต่หลังจากการลดทอนสัญญาณความแตกต่างสีจะทำให้เกิดการมอดูเลตเกินเท่ากับ 33% ซึ่งหมายถึงว่าแม้มีการลดทอนสัญญาณความแตกต่างสีลงก็ยังคงมีการมอดูเลตเกินอยู่ถ้าสัญญาณสีมีการอิ่มตัวสูงสุด (Fully Saturated Colors) แต่ในทางปฏิบัติเกิดขึ้นน้อยมาก [5] ด้วยเหตุนี้จึงได้มีการกำหนดและสร้างสัญญาณภาพแท่งสี 75% ขึ้นมาใช้ในการทดสอบเพื่อแก้ปัญหาการมอดูเลตเกินดังกล่าว นอกจากนี้เทคนิคการเข้ารหัสสีของระบบพาลยังสามารถแก้ความเพี้ยนทางเฟสของสัญญาณความแตกต่างสีที่เกิดจากระบบการส่ง ซึ่งจะกล่าวโดยละเอียดในหัวข้อระบบที่ใช้ในการสร้างสัญญาณภาพสีเบ็ดเสร็จ

2.3.2 ระบบที่ใช้ในการสร้างสัญญาณภาพสีเบ็ดเสร็จ

ระบบที่ใช้ในการสร้างสัญญาณภาพสีเบ็ดเสร็จได้มีการทำวิจัยและพัฒนามาอย่างต่อเนื่อง โดยเฉพาะบริษัทผู้ผลิตไอซี การพัฒนายังคงใช้โครงสร้างของระบบที่เป็นมาตรฐานเดียวกัน ส่วนที่แตกต่างกันจะเป็นเทคนิคการออกแบบระบบและวงจรเพื่อแก้ไขข้อบกพร่องเพื่อให้ได้สัญญาณที่ใกล้เคียงกับอุดมคติมากที่สุด รูปที่ 2.13 แสดงบล็อคไดอะแกรมของตัวเข้ารหัสสัญญาณภาพสีแดง, เขียวและน้ำเงินเป็นสัญญาณภาพสีเบ็ดเสร็จที่เป็นมาตรฐานของระบบพาล [5]



รูปที่ 2.13 แสดงบล็อคไดอะแกรมของตัวเข้ารหัสสัญญาณภาพสีแดง เขียวและน้ำเงินเป็นสัญญาณภาพสีเบ็ดเสร็จ

2.3.2.1 ภาค RGB to YUV Encoding Matrix

ทำหน้าที่แปลงสัญญาณภาพสีแดง, เขียวและน้ำเงิน ให้อยู่ในรูปของสัญญาณความส่องสว่างซึ่งมีรูปแบบการแปลงตามสมการที่ (2.11) และสัญญาณความแตกต่างสี B-Yและ R-Y โดยที่สัญญาณ B-Yและ R-Y จะไม่ถูกนำมาใช้โดยตรงแต่มันจะถูกลดทอนขนาดลง ก่อนที่จะส่งให้วงจรภาค U และ V มอดูเลชั่น เพื่อป้องกันการมอดูเลตเกินที่ภาคมอดูเลตสัญญาณภาพ (Vision Modulator) [5] เรียกสัญญาณความแตกต่างสีที่ถูกลดทอนขนาดลงว่าสัญญาณ U และ V แสดงดังสมการที่ (2.17) – (2.18) สัญญาณแม่สีทั้งสามดังกล่าวอาจได้มาจากแหล่งกำเนิดชนิดใดชนิดหนึ่ง เช่น กล้องวิดีโอ (Video Camera) หรือ เครื่องเล่นดีวีดี-วีซีดี ฯลฯ เป็นต้น

$$U = 0.493 * (B - Y) \quad (2.17)$$

$$V = 0.877 * (R - Y) \quad (2.18)$$

2.3.2.2 ภาค U Low Pass Filter และ V Low Pass Filter

ทำหน้าที่กรองสัญญาณความถี่ต่ำผ่านเพื่อจำกัดย่านความถี่ของสัญญาณความแตกต่างสี U, V ที่ได้จากภาค RGB to YUV Encoding Matrix ก่อนป้อนให้ภาค U, V Modulation เพื่อจำกัดไซด์แบนด์ (Sidebands) ของสัญญาณความแตกต่างสีหลังการมอดูเลต สำหรับระบบพาลบีและพาลจีตัวกรองมีคุณสมบัติคือ อัตราการลดทอนน้อยกว่า 2 dB.ที่ความถี่ 1.3 MHz. และที่ความถี่ 3.6 MHz. มีอัตราการลดทอนมากกว่า 20 dB.

2.3.2.3 ภาค Y Delay

จากบล็อกไดอะแกรมแสดงดังรูปที่ 2.13 จะเห็นว่าสัญญาณถูกแบ่งการประมวลผลออกเป็น 2 ส่วนได้แก่ ส่วนแรกการประมวลผลสัญญาณภาพเบ็ดเสร็จซึ่งได้จากการรวมกันของสัญญาณความส่องสว่าง, สัญญาณแบล้งกิ้งและสัญญาณซิงโครไนเซชั่น ส่วนที่สองการประมวลผลสัญญาณสีซึ่งได้จากการรวมกันของสัญญาณเบิรสต์และสัญญาณความแตกต่างสีที่ผ่านการมอดูแลตแล้ว จากลำดับของการประมวลผลจะสังเกตเห็นได้ว่าการประมวลผลสัญญาณสีมีการประมวลที่มีขั้นตอนมากกว่าดังนั้นจะใช้เวลาในการประมวลผลมากกว่าส่วนของสัญญาณภาพเบ็ดเสร็จ สัญญาณความส่องสว่างจึงต้องมีการหน่วงเวลาประมาณ 0.4 uS. เพื่อชดเชยเวลาการประมวลที่ช้ากว่าของสัญญาณสี

2.3.2.4 ภาค U และ V Modulation

ทำหน้าที่มอดูเลตสัญญาณความแตกต่างสีกับสัญญาณพาหะรองที่มีความถี่เท่ากับ 4.43361875 MHz. สำหรับระบบพาล [7] ถือได้ว่าเป็นส่วนที่สำคัญมากส่วนหนึ่ง การทำงานมีรายละเอียดดังนี้

สัญญาณข่าวสาร (Message) ทั้งสองได้แก่สัญญาณ U และ V ถูกป้อนเข้าตัวมอดูเลตเชิงแอมปลิจูดแบบแถบข้างคู่ขจัดคลื่นพาห์สองตัวด้วยสัญญาณพาหะรองที่มีเฟสต่างกัน 90° ซึ่งตัวมอดูเลตสามารถสร้างได้จากตัวคูณแบบสี่ควอแดรนท์ (Four Quadrant Multiplier) เนื่องจากตัวมอดูเลเตอร์ทั้งสองตัวดังกล่าวมีการทำงานเหมือนกัน แตกต่างกันเฉพาะเฟสของคลื่นพาห์ดังนั้นจะอธิบายเฉพาะส่วนของสัญญาณ U

$$\phi_{DSB-SC}(t) = k.u(t).Cos(\omega_{sc}t + \theta_c) \quad (2.19)$$

เพื่อความสะดวกในการวิเคราะห์สัญญาณ เราสามารถที่จะสมมุติให้ k = 1 และ $\theta_c = 0$ ได้ โดยไม่เสียความหมายของการวิเคราะห์สัญญาณโดยทั่วไปแต่อย่างใด เพียงแต่จะได้รูปของสมการที่กระชับขึ้นเป็น

$$\phi_{DSB-SC}(t) = u(t).Cos(\omega_{sc}t) \tag{2.20}$$

เพื่อที่จะทำความเข้าใจถึงการเปลี่ยนแปลงที่เกิดขึ้นในโดเมนความถี่ จะสมมุติให้ $U(\omega)$ คือฟังก์ชั่นสเปกตรัมของ u(t) กล่าวคือ

$$u(t) \leftrightarrow U(\omega) \tag{2.21}$$

โดยอาศัยคุณสมบัติของการแปลงฟูริเยร์ (Fourier) จะได้

$$u(t).Cos(\omega_{sc}t) \leftrightarrow \frac{1}{2}[U(\omega+\omega_{sc}) + U(\omega-\omega_{sc})] \tag{2.22}$$

นั่นคือ สัญญาณในโดเมนความถี่จะมีค่าเป็น

$$\phi_{DSB-SC}(\omega) = \frac{1}{2}[U(\omega+\omega_{sc}) + U(\omega-\omega_{sc})] \tag{2.23}$$

ทางฝั่งเครื่องรับสามารถดีมอดูเลตได้ดังนี้คือ

$$\begin{aligned} V_d(t) &= \phi_{DSB-SC}(t)Cos(\omega_{sc}t) \\ &= \{u(t)Cos(\omega_{sc}t)\}Cos(\omega_{sc}t) \\ &= u(t)Cos^2(\omega_{sc}t) \\ &= \frac{1}{2}u(t) + \frac{1}{2}u(t)Cos(2\omega_{sc})t \end{aligned} \tag{2.24}$$

และโดยคุณสมบัติของการแปลงฟูริเยร์จะได้

$$\phi_{DSB-SC}(t)Cos(\omega_{sc}t) \leftrightarrow \frac{1}{2}U(\omega)+\frac{1}{4}[U(\omega+2\omega_{sc})+U(\omega-2\omega_{sc})] \quad (2.25)$$

เมื่อนำสัญญาณไปผ่านวงจรกรองความถี่ต่ำผ่านทำให้สัญญาณความถี่สูงถูกตัดออกจะทำให้ได้สัญญาณขาออกของการดีมอดูเลตคือ

$$V_o(t)=\frac{1}{2}u(t) \quad (2.26)$$

จากสมการที่ (2.24) ทางฝั่งเครื่องรับการดีมอดูเลตสัญญาณที่ผ่านการมอดูเลตเชิงแอมปลิจูดแบบแถบข้างคู่ขจัดคลื่นพาห์จำเป็นต้องใช้สัญญาณคลื่นพาห์ที่มีความถี่และเฟสที่ตรงกับทางฝั่งเครื่องส่งจึงจะได้สัญญาณที่มีความถูกต้อง ด้วยเหตุนี้เองการดีมอดูเลตวิธีนี้จึงถูกเรียกว่า การดีมอดูเลต หรือการดีเทกต์แบบสัมพัทธ์ (Synchronous Detection) หรือ การดีเทกต์แบบร่วมนัย (Coherent Detection) [8] แต่ด้วยสาเหตุที่ใช้การมอดูเลตแบบนี้สัญญาณพาหะจะถูกขจัด ทำให้ต้องส่งสัญญาณพาหะรองของสัญญาณสีซึ่งถูกเรียกว่าสัญญาณ "เบิรสต์" (Burst) บวกไปเป็นช่วงๆกับสัญญาณความแตกต่างสีที่ถูกมอดูเลต ทำให้ได้รูปแบบของสัญญาณตามสมการที่ (2.27)

$$\phi_{DSB-SC}(t)=u(t).Sin(\omega_{sc}t)\pm v(t).Cos(\omega_{sc}t)+Burst \quad (2.27)$$

แต่ถ้าสัญญาณคลื่นพาห์มีความถี่หรือเฟสผิดเพี้ยนไปจากทางฝั่งเครื่องส่งจะทำให้ได้สัญญาณดีมอดูเลต Vd(t) ทางฝั่งเครื่องรับที่มีความเพี้ยนดังนี้

$$\begin{aligned} V_d(t) &= \phi_{DSB-SC}(t)Cos\{(\omega_{sc}+\Delta\omega)t+\delta\} \\ &= u(t)Cos(\omega_{sc}t)Cos\{(\omega_{sc}+\Delta\omega)t+\delta\} \\ &= \frac{1}{2}u(t)Cos(\Delta\omega t+\delta)+\frac{1}{2}u(t)Cos\{(2\omega_{sc}+\Delta\omega)t+\delta\} \end{aligned} \quad (2.28)$$

เมื่อ

$v(d)$ คือสัญญาณขาออกจากตัวดีมอดูเลต (ก่อนวงจรกรองสัญญาณความถี่ต่ำผ่าน)

$\phi_{DSB-SC}(t)$ คือสัญญาณความแตกต่างสี (U) ที่รับได้ทางฝั่งเครื่องรับ (ผ่านการมอดูเลตแบบ AM-DSBSC)

$Cos\{(\omega_{sc} + \Delta\omega)t + \delta\}$ คือสัญญาณคลื่นพาห์ที่ถูกสังเคราะห์ขึ้นทางฝั่งเครื่องรับซึ่งมีความถี่และเฟสผิดเพี้ยนไปจากทางฝั่งเครื่องส่ง ($\Delta\omega$ และ δ)

สัญญาณ Vd(t) จะถูกป้อนเข้าวงจรกรองความถี่ต่ำผ่านทำให้สัญญาณความถี่สูงถูกตัดออก ซึ่งก็คือพจน์หลังของ (2.28) เพราะฉะนั้นสัญญาณขาออกของวงจรดีมอดูเลเตอร์ Vo(t) จะได้

$$V_o(t) = \frac{1}{2}u(t)Cos(\Delta\omega t + \delta) \tag{2.29}$$

ถ้าเครื่องรับผลิตคลื่นพาห์โดยไม่มีความคลาดเคลื่อน กล่าวคือ $\Delta\omega$ และ δ เท่ากับศูนย์ จะทำให้ได้สัญญาณขาออกที่ถูกต้องคือ Vo(t) = ½ u(t) แต่ถ้าเครื่องรับผลิตคลื่นพาห์ที่มีความถี่ถูกต้อง ($\Delta\omega = 0$) แต่มีความคลาดเคลื่อนทางเฟส ($\delta \neq 0$) จะได้สัญญาณขาออกที่มีความคลาดเคลื่อนดังนี้

$$V_o(t) = \frac{1}{2}u(t)Cos(\delta) \tag{2.30}$$

เนื่องจาก $|Cos(\delta)| \leq 1$ ดังนั้นผลที่เกิดขึ้นก็คือสัญญาณขาออกจะมีระดับลดลงและถ้า $\delta = \pm\pi/2$ ระดับของสัญญาณขาออกก็จะเป็นศูนย์ เนื่องจาก $Cos(\pm\pi/2) = 0$

ถ้าสมมุติว่าเครื่องรับผลิตคลื่นพาห์ที่มีเฟสเริ่มต้นถูกต้อง ($\delta = 0$) แต่มีความถี่คลาดเคลื่อน ($\Delta\omega \neq 0$) จะทำให้ได้สัญญาณขาออกที่มีความคลาดเคลื่อนดังนี้

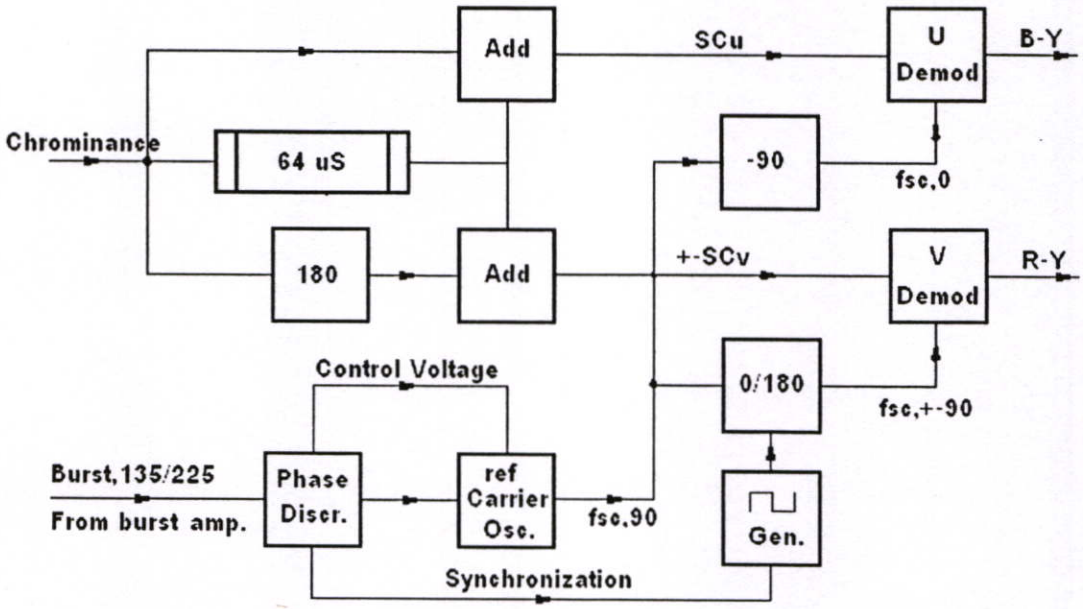
$$V_0(t) = \frac{1}{2}u(t)Cos(\Delta\omega t) \tag{2.31}$$

สมการที่ (2.31) แสดงให้เห็นว่าสัญญาณขาออกมีลักษณะผิดเพี้ยนไปจากความเป็นจริงคือสัญญาณจะถูกบังคับขนาดหรือถูกคูณด้วยคลื่นรูปไซน์ที่มีความถี่ต่ำ $\Delta\omega$ ผลที่เกิดขึ้นนี้เรียกว่า มีการบีท (Beating) จากความถี่ต่ำ

การส่งสัญญาณภาพสีเบ็ดเสร็จผ่านระบบส่งที่ประกอบด้วยวงจรขยายทรานซีสเตอร์จะทำให้เกิดความเพี้ยนสองประการคือ ความเพี้ยนทางขนาดและความเพี้ยนทางเฟส กล่าวคือ ประการแรกความเพี้ยนทางขนาดเกิดจากการทำงานที่ไม่เป็นเชิงเส้น (Nonlinear) ของทรานซีสเตอร์และสามารถแก้ความเพี้ยนชนิดนี้ได้ด้วยการป้อนกลับทางลบ (Negative Feedback) ประการที่สองความเพี้ยนทางเฟสเกิดจากค่าความจุของตัวเก็บประจุ (Capacitor) ภายในทรานซีสเตอร์ที่เปลี่ยนแปลงตามค่ากระแสอีมิต

เตอร์อันเป็นผลมาจากการเปลี่ยนแปลงระดับของสัญญาณความส่องสว่าง ความเพี้ยนทางเฟสทำให้เกิดปัญหากับสัญญาณสีที่ทับซ้อน (Superimpose) อยู่บนสัญญาณความส่องสว่าง เนื่องจากสัญญาณสีใช้การมอดูเลตแบบสัมพัทธ์ ผลก็คือสัญญาณความแตกต่างสีที่ได้จากการดีมอดูเลตมีขนาดผิดจากความเป็นจริง ทำให้การสร้างสีกลับผิดไปจากความเป็นจริงเช่นเดียวกัน เช่น จากสีม่วงกลายเป็นสีม่วงแดง เป็นต้น

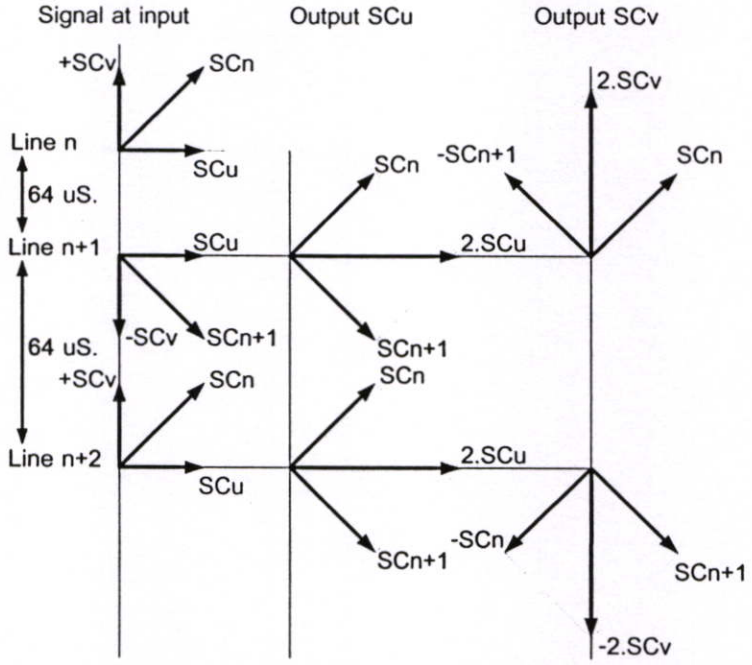
วิธีการลดความเพี้ยนทางเฟสโดยทั่วไปก็คือการลดขนาดของสัญญาณให้มีขนาดเล็กที่สุดเท่าที่จะทำได้ แต่วิธีการนี้ไม่สามารถนำมาใช้กับระบบการส่งสัญญาณภาพสีเบ็ดเสร็จได้ สำหรับระบบพาลได้ใช้วิธีการแก้ปัญหาความเพี้ยนทางเฟสโดยการกลับเฟสของสัญญาณ V เส้นเว้นเส้นสังเกตได้จากพจน์ $\pm v(t).Cos(\omega_{sc}t)$ ของสมการ (2.27) ซึ่งเป็นที่มาของชื่อระบบการเข้ารหัสแบบพาล (Phase Alternation Line: PAL) นั่นเอง การกลับเฟสของสัญญาณ V ดังกล่าวสามารถแก้ความเพี้ยนทางเฟสได้ดังนี้



รูปที่ 2.14 แสดงบล็อกไดอแกรมของตัวดีมอดูเลตสัญญาณความแตกต่างสีของระบบพาล

จากรูปที่ 2.14 การดีมอดูเลตสัญญาณของเครื่องรับจะใช้ข้อมูลสัญญาณสีเส้นก่อนหน้าซึ่งได้จากตัวหน่วงเวลา 1 เส้นกวาดที่เรียกว่า ดีเลย์ไลน์ (Delay Line) มาทำการบวกกับสัญญาณสีของเส้นกวาดปัจจุบันเพื่อแยกสัญญาณ SCu และ SCv ออกจากกัน (จากรูป SCu และ SCv คือสัญญาณ $u(t).Sin(\omega_{sc}t)$ และ $\pm v(t).Cos(\omega_{sc}t)$) สำหรับสัญญาณ SCu ใช้การบวกโดยตรงส่วนสัญญาณ SCv

มีการกลับเฟสก่อนการบวกหรืออาจมองได้ว่าเป็นการลบนั่นเอง หลังจากที่ได้สัญญาณ SCu และ SCv ทั้งสองสัญญาณจะถูกแยกไปดีมอดูเลตเพื่อนำสัญญาณแบนด์มูลฐาน (Base Band) ของสัญญาณ B-Y และ R-Y ไปใช้ในภาคต่อไป

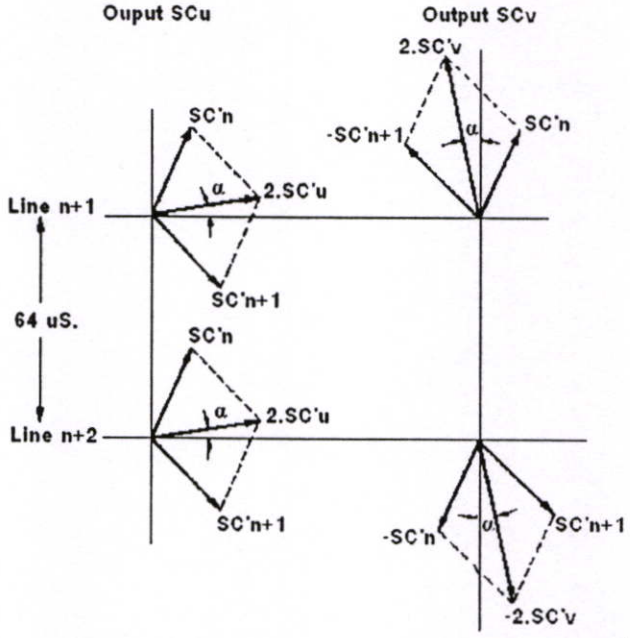


รูปที่ 2.15 แสดงเวคเตอร์ของสัญญาณในกรณีที่สัญญาณไม่มีความเพี้ยนทางเฟส

รูปที่ 2.15 แสดงในกรณีที่สัญญาณไม่มีความเพี้ยนทางเฟส จะเห็นว่าเส้นที่ n สัญญาณ SCv เป็นเฟสบวกและเส้นที่ n+1 เป็นเฟสลบซึ่งเป็นผลมาจากการกลับเฟสของสัญญาณพาหะรอง สัญญาณสีของทั้งสองเส้น (SCn เป็นสัญญาณที่ได้จากดีเลย์ไลน์และ SCn+1 เป็นสัญญาณเส้นต่อมา) จะถูกนำมาบวกกันเพื่อแยกสัญญาณ SCu จากผลการบวกจะทำให้ได้สัญญาณ SCu ที่มีขนาดเป็นสองเท่า ส่วนการแยกสัญญาณ SCv ใช้การลบ ผลจากการลบจะทำให้ได้สัญญาณ SCv ที่มีขนาดเป็นสองเท่าและกลับเฟสสลับเส้นเว้นเส้นตามการกลับเฟสของสัญญาณพาหะรอง ดังนั้นตัวดีมอดูเลตจะใช้สัญญาณไอเดนต์ซึ่งสร้างจากการดีเทกต์เฟสของสัญญาณเบิรสต์และมีความถี่เท่ากับครึ่งหนึ่งของความถี่การกวาดเส้นภาพทางแนวนอนในการควบคุมการกลับเฟสสัญญาณ SCv เพื่อให้ได้เฟสที่ถูกต้อง จากบล็อคไดอะแกรมดังรูปที่ 2.14 ใช้การกลับเฟสสัญญาณพาหะรองโดยภาค 0/180° ในการกลับเฟสสัญญาณ SCv ดังกล่าว

ในกรณีที่สัญญาณมีความเพี้ยนทางเฟสแสดงดังรูปที่ 2.16 โดยสมมติให้สัญญาณมีเฟสคลาดเคลื่อนเท่ากับ α องศาซึ่งเกิดจากระบบของการส่ง ความคลาดเคลื่อนของเฟสนี้จะไม่มีผลกับ

ขนาดของสัญญาณเนื่องจากความเพี้ยนทางขนาดสามารถแก้ไขได้ด้วยการป้อนกับทางลบ (Negative Feedback: NFB) และการปรับอัตราการขยายอัตโนมัติ (Automatic Gain Control: AGC) หรืออาจกล่าวอีกนัยหนึ่งได้ว่าขนาดที่เป็นผลรวมทางเวคเตอร์ของสัญญาณ SCu และ SCv มีค่าถูกต้องแต่ขนาดของสัญญาณ U และ V หลังการดีมอดูเลตจะมีขนาดที่ผิดไปจากค่าจริงซึ่งคำนวณได้จาก $U = [SCu]Cos\alpha$ และ $V = [SCv]Cos\alpha$ ในกรณีที่ไม่มีความเพี้ยนทางเฟสมุม α จะเท่ากับศูนย์ อย่างไรก็ตามอัตราการลดทอนจากผลของความเพี้ยนทางเฟสนี้จะไม่มีผลกับสีสัน (Hue) เนื่องจากสัญญาณ U และ V ถูกลดทอนด้วยค่าเดียวกันหรืออัตราส่วนของสัญญาณ U/V มีค่าเท่าเดิมแต่จะมีผลกับค่าความอิ่มตัวของสี (Saturation) นั่นคือความเข้มของสีจะลดลงแต่จะมากหรือน้อยขึ้นอยู่กับค่าของมุม α



รูปที่ 2.16 แสดงเวคเตอร์ของสัญญาณในกรณีที่สัญญาณมีความเพี้ยนทางเฟส

2.3.2.5 ภาค Sub Carrier Generator

ทำหน้าที่สร้างสัญญาณพาหะรองความถี่ 4.43361875 MHz. สำหรับการเข้ารหัสในระบบพาลบีและจี เพื่อใช้ในการมอดูเลตกับสัญญาณความแตกต่างสี โดยทำงานร่วมกับวงจรกลับเฟสสัญญาณ 180° เพื่อป้อนให้วงจรมอดูเลตสัญญาณ V เพื่อกลับเฟสสัญญาณหลังการมอดูเลต ความถี่ในการกลับเฟสมีค่าเท่ากับครึ่งหนึ่งของความถี่การกวาดเส้นภาพทางแนวนอน สำหรับสัญญาณ U ความถี่พาหะไม่มีการกลับเฟส

2.3.2.6 ภาค Burst Generator

ทำหน้าที่สร้างสัญญาณอ้างอิงที่มีความถี่เท่ากับสัญญาณพาหะรองของสัญญาณสีและถูกเรียกว่า "สัญญาณเบิรสต์" สัญญาณเบิรสต์จะมีเฟสต่างกับสัญญาณพาหะของสัญญาณ U 180°± 45° กล่าวคือ มันจะมีเฟสนำหน้าสัญญาณพาหะของสัญญาณ U 135° สำหรับเส้นกวาดที่ไม่มีกลับเฟสสัญญาณ V และมันจะมีเฟสนำหน้าสัญญาณพาหะของสัญญาณ U 225° สำหรับเส้นกวาดที่มีการกลับเฟสสัญญาณ V ดังนั้นมันจึงมีเฟสต่างกัน 90° หรือ ±45° ดังกล่าว วัตถุประสงค์ของการสร้างมี 2 อย่าง คือ ประการแรกใช้เป็นสัญญาณอ้างอิงในการสร้างกลับสัญญาณพาหะรองของสัญญาณสีโดยทางภาครับจะใช้เฉลี่ยสัญญาณเบิรสต์ในการสร้างกลับสัญญาณพาหะรอง ประการที่สองทางภาครับจะใช้ในการผลิตสัญญาณไอเดนต์ (Identification) ซึ่งเป็นสัญญาณที่ใช้ในการควบคุมพาลสวิทช์ (PAL Switch) ทำหน้าที่กลับเฟสสัญญาณ V ให้สอดคล้องกับทางฝั่งเครื่องส่ง เพื่อให้ได้สัญญาณ V ที่มีเฟสถูกต้อง

2.3.2.7 ภาค Blanking & Sync Generator

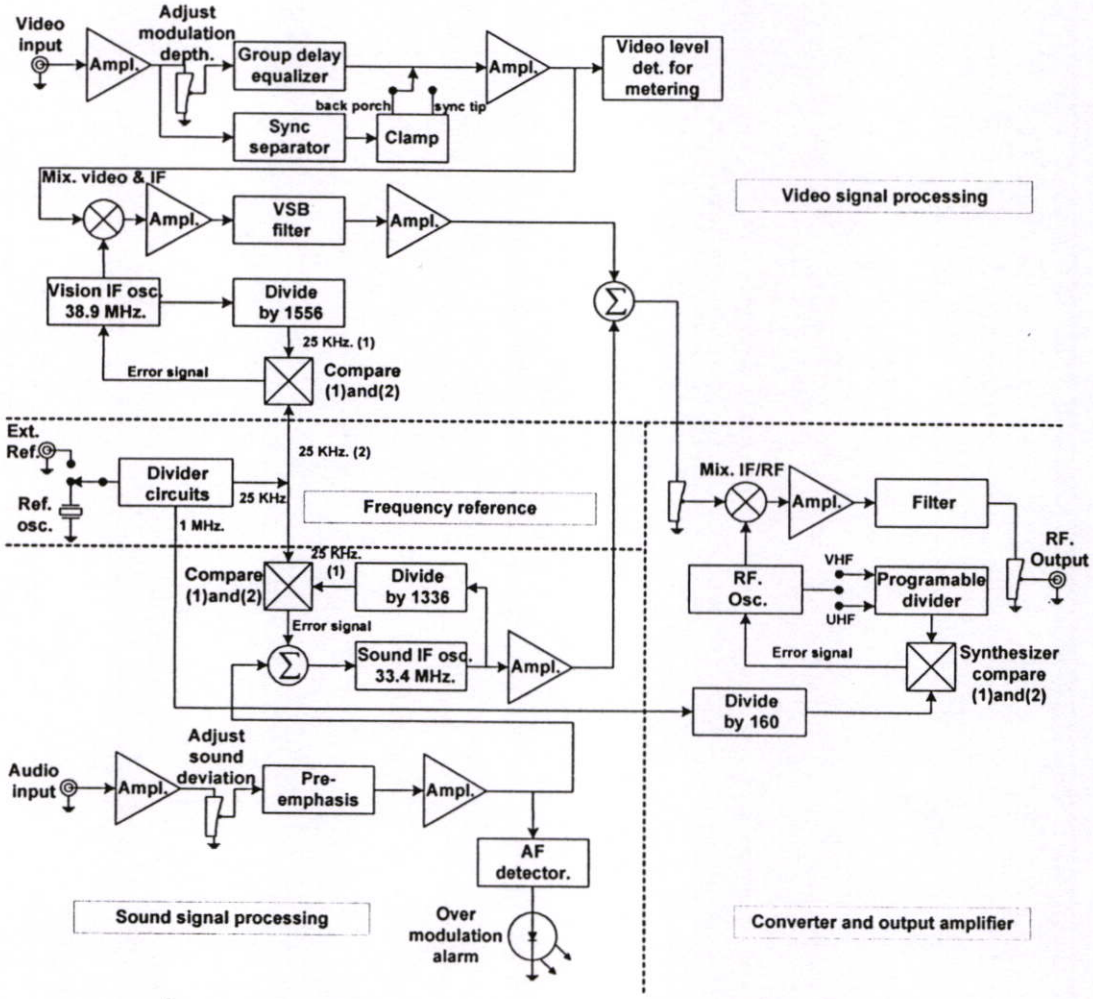
ทำหน้าที่สร้างสัญญาณแบล้งกิ้งและสัญญาณซิงโครไนเซชั่น สำหรับการกวาดเส้นภาพทั้งทางแนวตั้งและแนวนอน

2.4 การส่งสัญญาณวิทยุของสัญญาณภาพและเสียง

ตัวเข้ารหัสสัญญาณภาพสีแดง, เขียวและน้ำเงินเป็นสัญญาณภาพสีเบ็ดเสร็จที่นำเสนอสามารถประยุกต์ใช้งานได้อย่างกว้างขวางโดยเฉพาะกับระบบที่เป็นระบบเชิงเลข แต่เนื่องจากวัตถุประสงค์หลักของงานวิจัยชิ้นนี้มุ่งเน้นกับการประยุกต์ใช้งานกับระบบโทรทัศน์ไม่ว่าจะเป็นระบบส่ง (Transmitter) หรือเครื่องรับ (Receiver) ซึ่งอาจกล่าวได้ว่าเป็นระบบที่เป็นจุดกำเนิดของสัญญาณภาพสีเบ็ดเสร็จ ดังนั้นหัวข้อนี้จะเป็นการกล่าวถึงการส่งสัญญาณภาพและเสียงของระบบโทรทัศน์ที่ใช้ตัวเข้ารหัสฯเป็นส่วนการทำงานที่สำคัญส่วนหนึ่งของระบบ ตัวอย่างที่จะนำเสนอเป็นตัวมอดูเลเตอร์ของบริษัทฟิลิปส์รุ่น PM 5671 [9] ซึ่งเป็นเครื่องมือวัดทดสอบที่ใช้เป็นมาตรฐานในอุตสาหกรรมการผลิตเครื่องรับโทรทัศน์ หน้าที่ของมอดูเลเตอร์คือ ทำการมอดูเลตสัญญาณภาพและเสียงด้วยสัญญาณพาหะที่มีความถี่เท่ากับความถี่ของช่องสัญญาณมาตรฐานของระบบนั้นๆ สำหรับระบบที่กำลังนำเสนอเป็นระบบพาล บล็อคไดอะแกรมของมอดูเลเตอร์แสดงดังรูปที่ 2.17

จากรูปสามารถแบ่งระบบออกได้เป็น 3 ส่วนใหญ่ๆได้แก่ ส่วนของการประมวลผลสัญญาณภาพ, ส่วนของการประมวลผลสัญญาณเสียงและส่วนคอนเวอเตอร์และวงจรภาคเอาท์พุท ทุกส่วนการทำงานที่เกี่ยวกับการควบคุมหรือสังเคราะห์ความถี่ (Frequency Synthesis) จะใช้สัญญาณอ้างอิงซึ่ง

สร้างจากวงจรผลิตความถี่ที่มีการชดเชยอุณหภูมิ (Temperature-Compensated Crystal Oscillator: TCXO) เพื่อทำให้สัญญาณมีความถี่ที่เที่ยงตรงและเสถียร ความถี่อ้างอิงที่ใช้มี 2 ค่าคือ 1 MHz. และ 25 KHz. หน้าที่การทำงานของแต่ละส่วนโดยสังเขปมีดังนี้



รูปที่ 2.17 แสดงบล็อคไดอะแกรมมอดูเลเตอร์ของบริษัทฟิลิปส์รุ่น PM 5671

2.4.1 ส่วนการประมวลผลสัญญาณภาพ (Video Signal Processing)

สัญญาณภาพที่เป็นสัญญาณสีแดง เขียวและน้ำเงินถูกป้อนเข้าตัวเข้ารหัสสัญญาณภาพสีแดง เขียวและน้ำเงินเป็นสัญญาณภาพสีเบ็ดเสร็จ สัญญาณภาพสีเบ็ดเสร็จที่ได้จะถูกป้อนเข้ามอดูเลเตอร์ที่จุด Video Input ผ่านวงจรขยายที่มีอัตราการขยายสูงพอที่จะสามารถปรับความลึกการมอดูเลตของสัญญาณภาพ (Rest Carrier) สัญญาณภาพที่ผ่านการขยายจะถูกป้อนให้วงจรภาคชดเชยการล้าหลังของกลุ่ม (Group Delay) ตามความต้องการของแต่ละระบบซึ่งมีความแตกต่างกัน เช่น ระบบพาลบีและจีจะใช้

การชดเชยแตกต่างกับระบบพาลเอ็ม หลังจากผ่านการชดเชยการล้าหลังของกลุ่มสัญญาณจะถูกแคล้มป์ (Clamp) ค่าระดับแรงดันที่จุดหรือระดับใดระดับหนึ่งของสัญญาณเช่น ระดับแบล้งกิ้งหรือระดับของสัญญาณซิงค์โครไนเซชั่น วงจรที่ทำหน้านี้เรียกว่าวงจร "DC Restorer" จากนั้นสัญญาณจะถูกขยายอีกครั้งหนึ่งและป้อนให้ภาคมอดูเลตสัญญาณภาพ เพื่อทำการมอดูเลตเชิงแอมปลิจูดสัญญาณภาพกับสัญญาณพาหะความถี่ 38.9 MHz. ซึ่งเป็นความถี่กลางของสัญญาณภาพ (Vision Intermediate Frequency: Vision IF) สัญญาณพาหะดังกล่าวถูกสร้างจากวงจร Voltage Controlled Crystal Oscillator (VCXO) การควบคุมความถี่ใช้ระบบ Phase Locked Loop (PLL) ที่ใช้ความถี่อ้างอิงเท่ากับ 25 KHz. ในการเปรียบเทียบกับสัญญาณขาออกซึ่งถูกหารด้วย 1556 (38.9 MHz./1556 = 25 KHz.) ทำให้ได้สัญญาณที่มีความถี่และเฟสเที่ยงตรง ขั้นตอนสุดท้ายของการประมวลผล สัญญาณภาพที่ผ่านการมอดูเลตจะถูกป้อนให้ตัวกรองที่เรียกว่า ตัวกรองเวสติเจียนไซด์แบนด์ (Vestigial Side Band Filter: VSB Filter) เพื่อจำกัดช่วงความถี่ของไซด์แบนด์ที่เกิดจากการมอดูเลต [8],[5] ก่อนนำบวกกับสัญญาณความถี่กลางของสัญญาณเสียงที่ถูกมอดูเลตเชิงความถี่เพื่อส่งให้ภาค Converter & Output Amplifier ทำการประมวลผลต่อไป

2.4.2 ส่วนการประมวลผลสัญญาณเสียง (Sound Signal Processing)

สัญญาณเสียงจะถูกป้อนเข้าสู่วงจรขยายเพื่อทำให้สัญญาณแรงพอที่จะสามารถปรับระดับการเบี่ยงเบน (Frequency Deviation) ของสัญญาณหลังการมอดูเลตเชิงความถี่ จากนั้นสัญญาณจะถูกยกระดับที่ความถี่สูงโดยวงจร Pre-emphasis และป้อนให้วงจรมอดูเลเตอร์ทำการมอดูเลตเชิงความถี่สัญญาณเสียงกับสัญญาณพาหะซึ่งเป็นความถี่กลาง (Sound Intermediate Frequency: Sound IF) โดยที่แต่ละระบบใช้ค่าความถี่แตกต่างกัน เช่น ระบบพาลบีและจีใช้ความถี่ 33.4 MHz. ระบบพาลดีและเคใช้ความถี่ 32.4 MHz. ระบบพาลไอใช้ความถี่ 32.9 MHz. และ 34.4 MHz. สำหรับระบบพาลเอ็ม [9] เป็นต้น ตัวมอดูเลตเชิงความถี่สร้างจากระบบเฟสล็อคลูป (Phase Locked Loop: PLL) การทำงานของวงจรขณะไม่มีสัญญาณเสียงคือวงจรเปรียบเทียบ (Comparator) จะให้ค่าสัญญาณความคลาดเคลื่อน (Error Signal) มีค่าเป็นศูนย์ทำให้วงจรผลิตความถี่ผลิตความถี่เท่ากับความถี่กลางของสัญญาณเสียง แต่ถ้ามีสัญญาณเสียงป้อนเข้ามามันจะถูกบวกเข้ากับสัญญาณความคลาดเคลื่อนส่งผลให้วงจรผลิตความถี่ผลิตความถี่เบี่ยงเบนไปจากค่ากลางทำให้เกิดการมอดูเลตเชิงความถี่และนำสัญญาณที่ได้ไปบวกกับสัญญาณภาพที่ผ่านการมอดูเลตเชิงแอมปลิจูดที่ได้จากวงจรส่วนของการประมวลผลภาพเพื่อทำการประมวลผลในภาคต่อไป

2.4.3 ส่วนคอนเวอเตอร์และวงจรภาคเอาท์พุท (Converter and Output Circuits)

เอาท์พุทจากวงจรบวกซึ่งประกอบด้วยภาพสีเบ็ดเสร็จที่ทับ (Superimpose) อยู่บนสัญญาณความถี่ไอเอฟของสัญญาณภาพและสัญญาณความถี่กลางของสัญญาณเสียงที่ถูกมอดูเลตเชิงความถี่ สัญญาณดังกล่าวจะถูกป้อนเข้าสู่วงจร ไอเอฟ/อาร์เอฟ มิกเซอร์ (IF/RF Mixer) เพื่อทำการผสม (Mix) กับสัญญาณอาร์เอฟจากวงจรโลคอลออสซิลเลเตอร์ (Local Oscillator: LO) โดยสามารถตั้งค่าความถี่ได้ทั้งย่านวีเอชเอฟ (Very High Frequency: VHF) และยูเอชเอฟ (Ultra High Frequency: UHF) จากการเลือกตัวหาร (Divider) ที่ถูกโปรแกรมมาจากโรงงานผู้ผลิต การควบคุมความถี่ใช้ระบบเฟสล็อคลูป (Phase Locked Loop: PLL) ที่มีความถี่อ้างอิงเท่ากับ 1 MHz. วงจรภาคสุดท้ายของส่วนนี้ก็คือวงจรกรองแถบความถี่ผ่าน (Band Pass Filter: BPF) ซึ่งทำหน้าที่กรองความถี่ของสัญญาณขาออก สามารถคำนวณได้จากสมการที่ (2.32)

$$F_{out} = F_{osc} - 38.9 \quad \text{MHz.} \tag{2.32}$$

2.5 การวัดทดสอบคุณสมบัติของระบบโทรทัศน์

เนื่องจากอุปกรณ์อิเล็กทรอนิกส์ที่มีคุณสมบัติไม่เป็นไปตามอุดมคติทั้งอุปกรณ์ที่ทำงานเป็นเชิงเส้นและไม่เป็นเชิงเส้น หลังจากนำอุปกรณ์เหล่านี้มาสร้างเป็นระบบจะทำให้ได้สัญญาณที่ไม่เป็นไปตามอุดมคติเช่นเดียวกัน ดังนั้นจึงต้องมีการวัดคุณสมบัติของระบบว่าให้ค่าของความคลาดเคลื่อนอยู่ในเกณฑ์ที่ยอมรับได้หรือไม่ โดยทั่วไปในทางปฏิบัติใช้การป้อนสัญญาณทดสอบ (Test Signal) ที่ทราบคุณสมบัติผ่านเข้าไปในระบบที่ต้องการวัดจากนั้นใช้เครื่องมือวัดในการวัดสัญญาณขาออกและใช้การวิเคราะห์สัญญาณที่ได้เพื่อหาคุณสมบัติ สำหรับระบบโทรทัศน์ก็ใช้หลักดังกล่าวเช่นเดียวกัน ดังนั้นสามารถสรุปได้ว่ามี 2 ส่วนสำคัญที่ใช้ในการวัดหาคุณสมบัติของระบบโทรทัศน์ได้แก่

- สัญญาณทดสอบ (Test Signal) ได้แก่ สัญญาณที่ได้จากเครื่องสร้างสัญญาณภาพทดสอบ (Video Test Pattern Generator) ซึ่งมีคุณสมบัติเฉพาะที่เหมาะสมสำหรับการทดสอบระบบโทรทัศน์ เช่น เครื่องสร้างสัญญาณภาพทดสอบของบริษัท ROHDE & SCHWARZ
- เครื่องมือวัดทดสอบ ได้แก่ ออสซิลโลสโคป (Oscilloscope) ชนิดพิเศษที่รู้จักกันในชื่อของ เวฟฟอร์มมอนิเตอร์ (Waveform Monitor) และเว็คเตอร์สโคป (Vector Scope) เช่น เครื่องมือวิเคราะห์สัญญาณภาพของบริษัท Tektronix รุ่น VM700A

การวัดคุณสมบัติของระบบโทรทัศน์สามารถแบ่งออกได้เป็น 5 ประเภทดังนี้ [10]

- Amplitude and Timing Measurements
- Linear Distortions
- Non-linear Distortions
- Noise Measurements
- Transmitter Measurements

รายละเอียดสามารถศึกษาเพิ่มเติมได้จาก [10]

บทที่ 3

การประมวลผลสัญญาณเชิงเลข

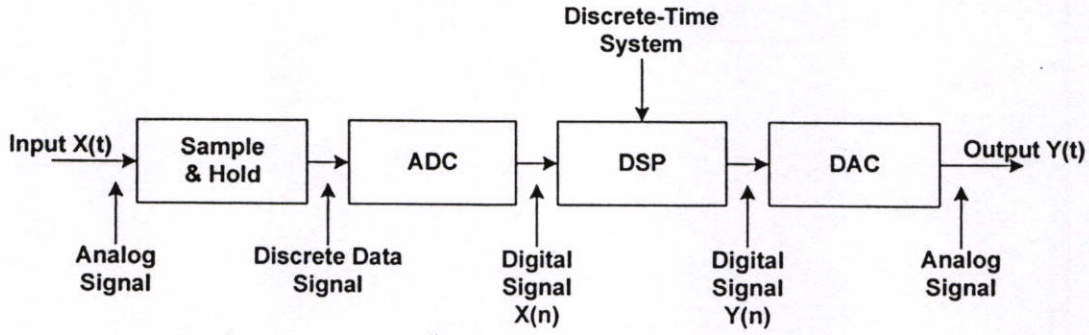
3.1 บทนำ

สัญญาณที่พบและใช้อยู่ในชีวิตประจำวันส่วนใหญ่อยู่ในรูปสัญญาณเชิงอุปมานซึ่งหมายถึงสัญญาณที่มีรูปคลื่น (Waveform) แปรไปอย่างต่อเนื่องกับพิสัยเวลาโดยที่แอมปลิจูด (Amplitude) หรือค่าขนาดของสัญญาณก็มีการแปรไปอย่างต่อเนื่องด้วย ยกตัวอย่างเช่น สัญญาณไซน์ สัญญาณเสียง ฯลฯ สัญญาณเชิงอุปมานดังกล่าวนี้ในอดีตระบบที่ใช้ในการประมวลผลส่วนใหญ่ถูกสร้างจากอุปกรณ์อิเล็กทรอนิกส์โดยอาศัยคุณสมบัติเฉพาะตัวของอุปกรณ์อิเล็กทรอนิกส์ เช่น ตัวต้านทาน ตัวเก็บประจุ ทรานซิสเตอร์ ฯลฯ ที่มีพฤติกรรมต่อปริมาณทางไฟฟ้าเช่น แรงดัน กระแส ความถี่และเฟส ผลที่ได้นี้เปรียบได้ว่ามันก็เกิดจากการคำนวณของอุปกรณ์อิเล็กทรอนิกส์นั่นเอง ยกตัวอย่างเช่น การใช้ออปแอมป์ (Operational Amplifier: Op Amp) ต่อเป็นวงจรขยายสองเท่าแบบกลับเฟส ผลที่ได้ก็คือแรงดันขาออกมีขนาดเท่ากับแรงดันขาเข้าที่ถูกคูณด้วยลบสอง ดังนั้นจึงสามารถใช้การคำนวณทางซอร์ฟแวร์แทนพฤติกรรมการคำนวณของอุปกรณ์อิเล็กทรอนิกส์ดังกล่าวได้ ด้วยเหตุนี้จึงได้มีการคิดทฤษฎีการประมวลผลสัญญาณเชิงเลขขึ้น ซึ่งในความเป็นจริงได้มีการวิจัยออกมานานแล้วแต่ไม่สามารถสร้างได้จริงในทางปฏิบัติ (Realize) เนื่องจากเทคโนโลยีการผลิตชิพไอซีไม่สามารถสร้างตัวประมวลผลที่มีความซับซ้อนสำหรับรองรับอัตราความเร็วของการคำนวณที่ต้องการได้ แต่ในปัจจุบันเทคโนโลยีวีแอลเอสไอ (VLSI: Very Large Scale Integrated Circuit) ที่พัฒนาไปอย่างรวดเร็วจึงทำให้สามารถสร้างชิพตัวประมวลผลที่มีความเร็วในการประมวลผลสูงเพียงพอกับความต้องการได้

โดยทั่วไปสัญญาณขาออกของระบบการประมวลผลสัญญาณเชิงเลขที่เราต้องการคือสัญญาณเชิงอุปมานแต่สัญญาณขาเข้าของระบบสามารถเป็นได้ทั้งสัญญาณเชิงอุปมานและสัญญาณเชิงเลข การประมวลผลสัญญาณเชิงเลขจึงสามารถแบ่งออกได้เป็นสองแบบคือ การประมวลผลสัญญาณเชิงเลขของสัญญาณใดๆที่อยู่ในรูปแบบเชิงเลขหรือการประมวลผลสัญญาณเชิงเลขของสัญญาณใดๆที่ไม่ได้อยู่ในรูปแบบเชิงเลข

3.2 สัญญาณและระบบแบบไม่ต่อเนื่อง

ในระบบการประมวลผลสัญญาณเชิงเลข สัญญาณเชิงอุปมานขาเข้าที่เป็น "สัญญาณเชิงเวลาต่อเนื่อง" (Continuous-time Signal) จะถูกสุ่มและคงค่า (Sampling and Hold) เรียกสัญญาณที่จุดนี้ว่า "สัญญาณเชิงข้อมูลเต็มหน่วย" (Discrete Data Signal) จากนั้นจะถูกแปลงเป็น "สัญญาณเชิงเลข" (Digital Signal) ทั้งสัญญาณเชิงข้อมูลเต็มหน่วยและสัญญาณเชิงเลข ถือว่าเป็น "สัญญาณเชิงเวลาเต็มหน่วย" (Discrete-time Signal) ตำราบางเล่มเรียกว่า "สัญญาณไม่ต่อเนื่อง" [3] คำว่าต่อเนื่องหรือไม่ต่อเนื่องนี้หมายถึงสัญญาณนั้นๆมีค่าต่อเนื่องทางเวลาหรือไม่ สัญญาณไม่ต่อเนื่องหมายถึงสัญญาณที่มีค่าเพียงบางจุดของเวลาเท่านั้นและใช้สัญลักษณ์ n แทนเวลาแบบไม่ต่อเนื่อง โดยที่ n เป็นตัวแปรที่มีค่าเป็นจำนวนเต็มคือ n = ..., -2, -1, 0, 1, 2,... สัญญาณไม่ต่อเนื่องจะเป็นฟังก์ชั่นของ n ดังนั้นจะเขียนแทนสัญญาณนี้ว่า x[n] ส่วนระบบไม่ต่อเนื่องก็คือ ระบบที่ทำการประมวลผลสัญญาณไม่ต่อเนื่องซึ่งก็คือโปรเซสเซอร์ (Digital Signal Processor) นั่นเอง ด้วยเหตุนี้การประมวลผลสัญญาณเชิงเลขบางครั้งถูกเรียกว่า การประมวลผลสัญญาณไม่ต่อเนื่อง (Discrete-Time Signal Processing)



รูปที่ 3.1 แสดงระบบพื้นฐานของการประมวลผลสัญญาณเชิงเลข

3.2.1 ทฤษฎีการสุ่ม (Sampling Theorem)

การแปลงข้อมูลระหว่างสัญญาณเชิงอุปมานและสัญญาณเชิงเลขสิ่งที่ต้องคำนึงถึงและให้ความสำคัญเป็นอย่างมากก็คือความสมบูรณ์ครบถ้วนของข้อมูลหลังการแปลง ทฤษฎีการสุ่มกล่าวว่าสัญญาณเชิงอุปมานที่มีความหนาแน่นสเปกตรัมอยู่ในแบนด์วิดท์ที่จำกัด เราจะพบว่าในโดเมนเวลาของสัญญาณจะมีข้อมูลที่มีความซ้ำซ้อนดังนั้นถ้าเราเลือกสุ่มค่าตัวอย่างของสัญญาณด้วยคาบเวลาที่เหมาะสมและมีระยะห่างอย่างสม่ำเสมอจะทำให้เราได้ข้อมูลที่เป็นตัวแทนของสัญญาณเชิงอุปมานที่ถูกสุ่มได้อย่างสมบูรณ์ โดยที่เวลาของการสุ่มสามารถพิจารณาได้ดังนี้คือ สมมติว่าสัญญาณใดๆที่มีค่าฟังก์ชั่นมีความหนาแน่นสเปกตรัมอยู่ในช่วงความถี่ที่มีแบนด์วิดท์ไม่เกิน B เฮิรตซ์ การสุ่มค่าตัวอย่าง

ของสัญญาณด้วยคาบเวลาไม่เกิน $\frac{1}{2B}$ วินาที หรือความถี่ของการสุ่มจะต้องมากกว่าหรือเท่ากับ 2B เฮิรตซ์ ก็จะสามารถเก็บข้อมูลที่เป็นตัวแทนของสัญญาณเชิงอุปมานที่ถูกสุ่มได้อย่างสมบูรณ์และสามารถสร้างสัญญาณเชิงอุปมานกลับคืนอย่างสมบูรณ์ด้วยข้อมูลที่ได้จากการสุ่มนี้ด้วยเช่นกัน จากคำกล่าวในข้างต้นสามารถอธิบายในเชิงคณิตศาสตร์ได้ดังนี้คือ สมมติให้สัญญาณเชิงอุปมาน $f(t)$ มีค่าความหนาแน่นเชิงสเปกตรัมอยู่ไม่เกินความถี่ B เฮิรตซ์ สัญญาณ $f(t)$ นี้ถูกสุ่มด้วยเวลาคงที่ T วินาที และสมมติให้สัญญาณที่ถูกสุ่มแล้วนี้ว่า $f_S(t)$ จะได้ว่า

$$f_S(t) = f(t).\delta_T(t) \tag{3.1}$$

โดยในที่นี้ $\delta_T(t)$ คือขบวนอิมพัลส์ (Impulse Train) ที่มีการนิยามว่า

$$\delta_T(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT) \tag{3.2}$$

จากคุณสมบัติการคูณฟังก์ชั่นอิมพัลส์กับฟังก์ชั่นอื่นจะได้

$$f(t)\delta(t - t0) = f(t0)\delta(t - t0) \tag{3.3}$$

จาก (3.1) และ (3.2) จะได้

$$f_S(t) = \sum_{n=-\infty}^{\infty} f(nT)\delta(t - nT) \tag{3.4}$$

จาก (3.4) แสดงให้เห็นว่า $f_S(t)$ คือสัญญาณ $f(t)$ ที่มีค่าเพียงช่วงเวลา $t = nT(n = 1,2,..)$ เท่านั้นและผลการแปลงฟูริเยร์ของ $\delta_T(t)$ จะได้

$$\delta_T(t) \leftrightarrow \delta_{\omega_S}(\omega)\omega_S \tag{3.5}$$

โดยที่ $\omega_S = \frac{2\pi}{T}$ และจากทฤษฎีการทำคอนโวลูชัน

$$f_1(t)f_2(t) \leftrightarrow \frac{1}{2\pi}\int_{-\infty}^{\infty} F_1(\mu)F_2(\omega-\mu)d\mu \qquad (3.6)$$

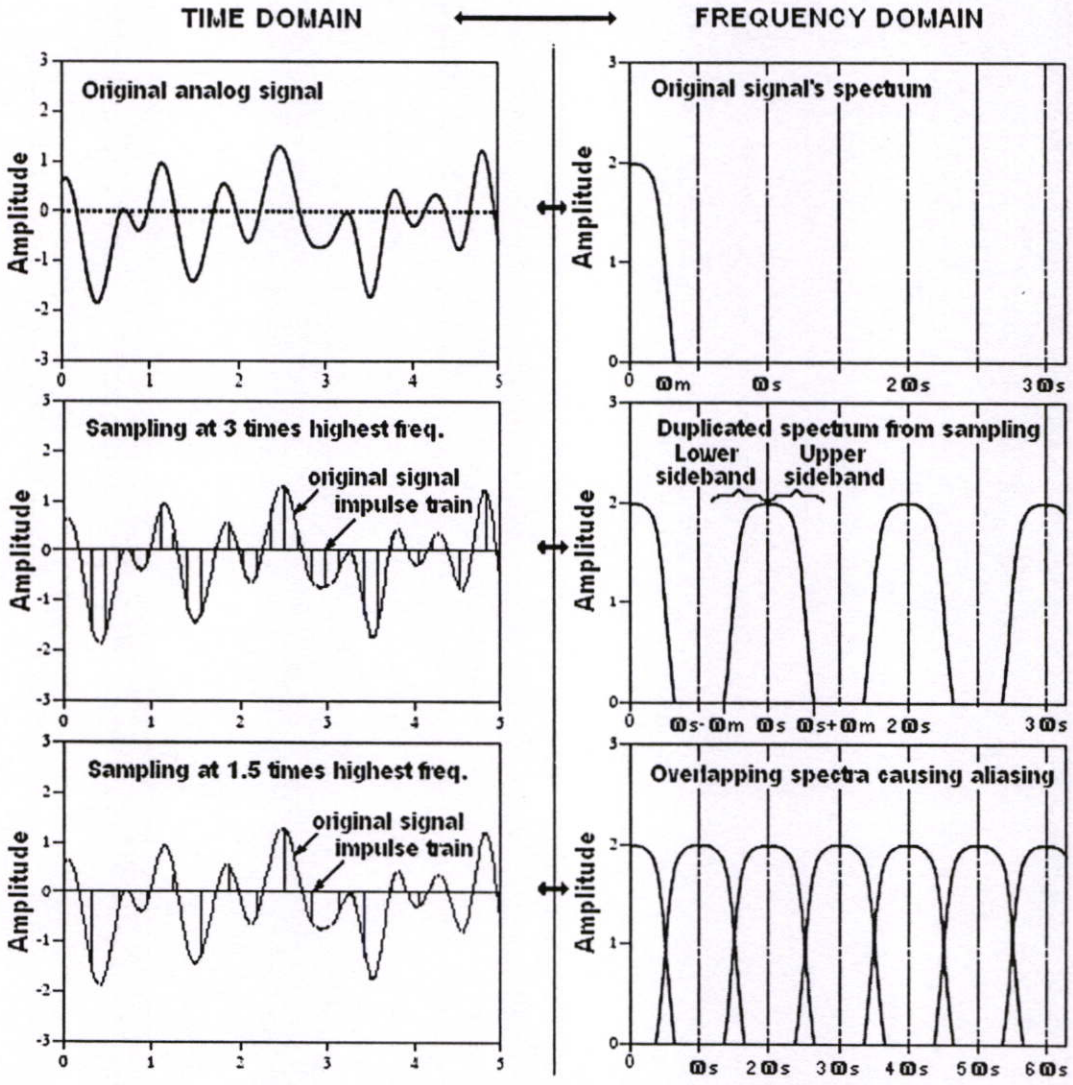
จาก (3.6) ถ้ากำหนดให้ $f_S(t) \leftrightarrow F_S(\omega)$ และ $f(t) \leftrightarrow F(\omega)$ จะได้

$$\begin{aligned} F_S(\omega) &= F(\omega)\delta_{\omega_S}(\omega)\omega_S \\ &= \frac{\omega_S}{2\pi}\int_{-\infty}^{\infty}\sum_{n=-\infty}^{\infty} F(z)\delta(\omega-n\omega_S-z)dz \\ &= \frac{1}{T}\sum_{n=-\infty}^{\infty} F(\omega-n\omega_S) \end{aligned} \qquad (3.7)$$

จาก 3.7 ทำให้ทราบว่าสัญญาณ $f_S(t)$ หรือสัญญาณที่เกิดจากการสุ่มของสัญญาณ $f(t)$ จะมีฟังก์ชั่นความหนาแน่นเชิงสเปกตรัมที่มีรูปร่างเหมือนฟังก์ชั่นความหนาแน่นเชิงสเปกตรัมของสัญญาณ $f(t)$ แต่จะเกิดอยู่ซ้ำกันทุกคาบความถี่ ω_S แสดงดังรูปที่ 3.2 เมื่อสัญญาณ $f(t)$ มีองค์ประกอบของความถี่สูงสุดคือ B เฮิรตซ์ ค่าความถี่เชิงมุมที่ตรงกันกับค่าความถี่สูงสุดนี้จะมีค่า $\omega_M = 2\pi B$ เรเดียน/วินาที จะพบว่าถ้า $\omega_S > 2\omega_M$ หรือ $T < \frac{1}{2B}$ เช่นในกรณีที่ใช้ค่าความถี่ในการสุ่มเท่ากับสามเท่าของความถี่สูงสุดของสัญญาณ $f(t)$ ($\omega_S = 3\omega_M$) จะได้ $F_S(\omega)$ มีลักษณะดังรูปที่ 3.2 (กลาง-ขวา) แต่ถ้า $\omega_S < 2\omega_M$ หรือ $T > \frac{1}{2B}$ เช่นในกรณีที่ใช้ค่าความถี่ในการสุ่มหนึ่งจุดห้าเท่าของความถี่สูงสุดของสัญญาณ $f(t)$ ($\omega_S = 1.5\omega_M$) จะได้ $F_S(\omega)$ มีลักษณะดังรูปที่ 3.2 (ล่าง-ขวา) จะสังเกตเห็นว่าเกิดการเหลื่อมทับกันของ $F(\omega)$ ทำให้เกิดผลเทียบเท่ากับการพับกลับ (Fold Over) องค์ประกอบของสัญญาณที่มีความถี่สูงเกิน $\frac{\omega_S}{2}$ ซึ่งกลับมาทับกับองค์ประกอบของสัญญาณที่มีความถี่ต่ำและรวมตัวกันจึงทำให้มีสเปกตรัมผิดเพี้ยนไปจากเดิม ปรากฏการณ์ที่มีการเหลื่อมหรือการพับกลับของส่วนประกอบของสเปกตรัมดังกล่าวนี้มีคำศัพท์เรียกเฉพาะว่าการเกิด เอเลียสซิง (Aliasing)

จากที่ได้กล่าวในข้างต้น การใช้คาบเวลาในการสุ่ม $T < \frac{1}{2B}$ วินาที จะทำให้สามารถใช้วงจรกรองความถี่ในการแยกเอาเฉพาะส่วนที่มีความถี่ต่ำของ $F_S(\omega)$ ซึ่งเหมือนกับ $F(\omega)$ ออกมาได้แต่อย่างไรก็ตามในกรณีที่ $T > \frac{1}{2B}$ แล้วมีการซ้อนทับกันของส่วนประกอบความถี่ ทำให้วงจรกรองความถี่ไม่สามารถแยกเอา $F(\omega)$ จาก $F_S(\omega)$ ได้ สำหรับในกรณีที่ใช้คาบเวลาในการสุ่ม $T = \frac{1}{2B}$ พอดีหรือที่เรียกว่าช่วงเวลาไนควิสต์ (Nyquist Interval) จะต้องใช้วงจรกรองความถี่ในอุดมคติเท่านั้น

จึงจะสามารถแยกสเปกตรัมในช่วงความถี่ที่ต้องการคือ $F(\omega)$ ออกมาได้ ดังนั้นโดยปกติในทางปฏิบัติจึงต้องใช้คาบเวลาการสุ่มให้น้อยกว่า $\frac{1}{2B}$ เพื่อแก้ปัญหาการโรลออฟ (Roll Off) หรือการค่อยๆ ลดลงของค่าฟังก์ชั่นถ่ายโอนบริเวณจุดตัดความถี่ (Cutoff) ของวงจรกรองความถี่ซึ่งไม่เป็นไปตามอุดมคติ



รูปที่ 3.2 แสดงสัญญาณในโดเมนเวลาที่ถูกสุ่มด้วยความถี่ค่าต่างๆและสเปกตรัมในโดเมนความถี่

ทฤษฎีการสุ่มนี้สามารถประยุกต์ใช้ได้ทั้งในส่วนของการแปลงสัญญาณเชิงอุปมานเป็นสัญญาณเชิงเลขและการแปลงจากสัญญาณเชิงเลขเป็นสัญญาณเชิงอุปมาน แต่จะมีเงื่อนไขอื่นๆที่จะต้องนำมาวิเคราะห์ร่วมเช่น ผลของการคงค่าข้อมูล (Hold) ฯลฯ ซึ่งจะกล่าวโดยละเอียดในหัวข้อระบบการประมวลผลสัญญาณเชิงเลข

3.2.2 การแปลงแซดและการแปลงแซดผกผัน

ในสาขาวิศวกรรมไฟฟ้าอิเล็กทรอนิกส์มีคณิตศาสตร์ที่เรียกว่าการแปลงอยู่หลายรูปแบบเพื่อใช้ในการวิเคราะห์สัญญาณ, ระบบ หรือการแก้ปัญหาโดยอ้อม ยกตัวอย่างเช่น การแปลงลาปลาซ (Laplace's Transform) ที่ช่วยให้การวิเคราะห์ระบบหรือโครงข่ายเชิงอุปมานสามารถทำได้ง่ายขึ้น ทั้งนี้โดยทั่วไปในโดเมนเวลานั้น คุณสมบัติของระบบหรือโครงข่ายสามารถเขียนอธิบายได้โดยใช้สมการเชิงอินทริกรัลอนุพันธ์ (Integro-Differential Equation) [2] ระบบที่ซับซ้อนจะมีพจน์ของสมการเชิงอินทริกรัลอนุพันธ์จำนวนมาก ซึ่งเป็นที่ทราบกันว่าการหาคำตอบจากสมการเชิงอินทริกรัล อนุพันธ์โดยตรงทำได้ค่อนข้างยาก แต่ถ้าเราทำการแก้ปัญหาโดยการประยุกต์ใช้การแปลงลาปลาซ ผลของการแปลงจะทำให้เราได้สมการพีชคณิตในโดเมนความถี่และการหาคำตอบก็สามารถใช้วิธีการของพีชคณิตซึ่งทำได้ง่ายกว่า คำตอบที่ได้จะอยู่ในรูปของตัวแปรโดเมนความถี่จะต้องใช้การแปลงผกผันเพื่อหาคำตอบที่อยู่ในโดเมนเวลาที่เราต้องการ จากหลักการดังกล่าวระบบเชิงเลขก็เช่นเดียวกัน มีการใช้การแปลงเป็นเครื่องมือที่สำคัญในการวิเคราะห์คุณสมบัติของสัญญาณและระบบแต่ต่างชนิดกับที่ใช้ในระบบเชิงอุปมาน ยกตัวอย่างเช่น การออกแบบตัวกรองความถี่เชิงอุปมานใช้การแปลงลาปลาซเป็นเครื่องมือในการวิเคราะห์ ในขณะที่การออกแบบตัวกรองเชิงเลขใช้การแปลงแซด อาจกล่าวได้ว่าการแปลงทั้งสองชนิดนี้ใช้กลยุทธเดียวกันคือการหาโพลและซีโร่ของระบบ การแปลงลาปลาซใช้กับสมการเชิงอนุพันธ์ ผลที่ได้อยู่ในโดเมน S ส่วนการแปลงแซดใช้กับสมการผลต่าง (Difference Equation) และผลที่ได้อยู่ในโดเมน Z แต่การแปลงทั้งสองนี้ยังมีความแตกต่างกันบ้างเช่นระนาบ S ใช้ระบบพิกัดฉากแต่ระนาบ Z ใช้ระบบพิกัดเชิงขั้วเป็นต้น สูตรการแปลงแซดแสดงดังสมการที่ (3.8)

$$X(z) = \sum_{n=-\infty}^{\infty} x[n] z^{-n} \tag{3.8}$$

หลังจากได้ผลเฉลยที่อยู่ในโดเมน z แล้วจะต้องแปลงกลับผลเฉลยนั้นให้กลับไปอยู่ในโดเมนเวลาโดยการแปลงแซดผกผัน (Inverse Z- Transform) แสดงดังสมการที่ (3.9) เพื่อใช้ในการออกแบบตัวประมวลผลหรือการเขียนโปรแกรม สำหรับในโดเมนเวลาจะใช้สมการผลต่าง (Difference Equation) ในการแสดงความสัมพันธ์ระหว่างสัญญาณขาออก y[n] และ สัญญาณขาเข้า x[n] ยกตัวอย่างเช่น y[n] = 0.5x[n]+2x[n-1] เป็นต้น

$$x[n] = \frac{1}{2\pi j} \oint_c X(z) z^{n-1} dz \tag{3.9}$$

เมื่อ C เป็นคอนทัวร์ในทิศทางทวนเข็มนาฬิกาที่ล้อมรอบจุดเอกฐาน (Singular Point) ของ $X(z)z^{n-1}$

3.2.3 การวิเคราะห์ฟูริเยร์

การวิเคราะห์ฟูริเยร์เป็นคณิตศาสตร์แขนงหนึ่งซึ่งใช้การแยกสัญญาณออกเป็นสัญญาณไซน์ความถี่ต่างๆ เนื่องจากการวิเคราะห์สัญญาณไซน์สามารถทำได้ง่ายกว่าสัญญาณต้นฉบับ [11] นอกจากนี้สัญญาณไซน์และโคไซน์ยังมีคุณสมบัติ Sinusoidal fidelity ในขณะที่สัญญาณต้นฉบับไม่มี สัญญาณที่พิจารณาอาจจะเป็นสัญญาณต่อเนื่องหรือสัญญาณไม่ต่อเนื่องและอาจจะเป็นสัญญาณคาบหรือไม่ใช่สัญญาณคาบ ด้วยเหตุนี้การวิเคราะห์ฟูริเยร์จึงสามารถแบ่งออกได้เป็นสี่ชนิดคือ

- สัญญาณที่สนใจไม่เป็นสัญญาณคาบและเป็นสัญญาณต่อเนื่องมีความยาวตามแกนเวลาจากลบอนันต์จนถึงบวกอนันต์ สัญญาณชนิดนี้ต้องใช้ "การแปลงฟูริเยร์" (Fourier Transform) ในการวิเคราะห์
- สัญญาณที่สนใจเป็นสัญญาณคาบและเป็นสัญญาณต่อเนื่องมีความยาวตามแกนเวลาจากลบอนันต์จนถึงบวกอนันต์ สัญญาณชนิดนี้ต้องใช้ "อนุกรมฟูริเยร์" (Fourier Series) ในการวิเคราะห์
- สัญญาณที่สนใจไม่เป็นสัญญาณคาบและเป็นสัญญาณไม่ต่อเนื่องมีความยาวจุดของข้อมูลที่ได้จากการสุ่มตามแกนเวลาจากลบอนันต์จนถึงบวกอนันต์ สัญญาณชนิดนี้ต้องใช้ "ดีสครีตไทม์ฟูริเยร์ทรานสฟอร์ม" (Discrete Time Fourier Transform: DTFT) ในการวิเคราะห์
- สัญญาณที่สนใจเป็นสัญญาณคาบและเป็นสัญญาณไม่ต่อเนื่องมีความยาวจุดของข้อมูลที่ได้จากการสุ่มตามแกนเวลาจากลบอนันต์จนถึงบวกอนันต์ สัญญาณชนิดนี้ต้องใช้ "ดีสครีตฟูริเยร์ทรานสฟอร์ม" (Discrete Fourier Transform: DFT) ในการวิเคราะห์ ซึ่งการแปลงชนิดนี้บางครั้งเรียกว่า "ดีสครีตฟูริเยร์ซีรีส์" (Discrete Fourier Series: DFS)

การวิเคราะห์ฟูริเยร์ทั้งสี่ชนิดดังกล่าวสามารถแบ่งออกเป็นชนิดย่อยได้อีกสองชนิดคือ คอมเพล็ก (Complex) และ เรียล (Real) ทั้งสองชนิดนี้แตกต่างกันที่รูปแบบของคณิตศาสตร์ที่ใช้ในการวิเคราะห์แสดงรายละเอียดดังรูปที่ 3.3-3.6 ถึงแม้ว่าการวิเคราะห์ฟูริเยร์จะมีอยู่หลายชนิดด้วยกันแต่สำหรับงานด้านการประมวลผลสัญญาณเชิงเลขจะใช้เพียงชนิดเดียวคือ DFT เนื่องจากตัวประมวลผลสามารถทำงานได้กับเฉพาะข้อมูลแบบไม่ต่อเนื่องและมีความยาวข้อมูลจำกัด

Fourier Transform

complex transform	real transform
synthesis $x(t) = \int_{-\infty}^{+\infty} X(\omega)\, e^{j\omega t}\, d\omega$	*synthesis* $x(t) = \int_{0}^{+\infty} ReX(\omega)\cos(\omega t) - ImX(\omega)\sin(\omega t)\,dt$
analysis $X(\omega) = \frac{1}{2\pi}\int_{-\infty}^{-\infty} x(t)\, e^{-j\omega t}\, dt$	*analysis* $ReX(\omega) = \frac{1}{\pi}\int_{-\infty}^{-\infty} x(t)\cos(\omega t)\,dt$ $ImX(\omega) = \frac{-1}{\pi}\int_{-\infty}^{-\infty} x(t)\sin(\omega t)\,dt$

รูปที่ 3.3 แสดงสมการที่ใช้ในการแปลงฟูริเยร์

Fourier Series

complex transform	real transform
synthesis $x(t) = \sum_{k=-\infty}^{+\infty} X[k]\, e^{j\,2\pi kt/T}$	*synthesis* $x(t) = \sum_{k=0}^{-\infty} ReX[k]\cos(2\pi kt/T) - ImX[k]\sin(2\pi kt/T)$
analysis $X[k] = \frac{1}{T}\int_{0}^{T} x(t)\, e^{-j\,2\pi kt/T}\, dt$	*analysis* $ReX[k] = \frac{2}{T}\int_{0}^{T} x(t)\cos(2\pi kt/T)\,dt$ $ImX[k] = \frac{-2}{T}\int_{0}^{T} x(t)\sin(2\pi kt/T)\,dt$

รูปที่ 3.4 แสดงสมการที่ใช้ในการคำนวณอนุกรมฟูริเยร์

Discrete Time Fourier Transform (DTFT)

complex transform	real transform
synthesis $x[n] = \int_0^{2\pi} X(\omega)\, e^{j\omega n}\, d\omega$	*synthesis* $x[n] = \int_0^{\pi} ReX(\omega)\cos(\omega n) - ImX(\omega)\sin(\omega n)\, d\omega$
analysis $X(\omega) = \frac{1}{2\pi}\sum_{n=-\infty}^{+\infty} x[n]\, e^{-j\omega n}$	*analysis* $ReX(\omega) = \frac{1}{\pi}\sum_{n=-\infty}^{+\infty} x[n]\cos(\omega n)$ $ImX(\omega) = \frac{-1}{\pi}\sum_{n=-\infty}^{+\infty} x[n]\sin(\omega n)$

รูปที่ 3.5 แสดงสมการที่ใช้ในการคำนวณดีสครีตไทม์ฟูริเยร์ทรานสฟอร์ม

Discrete Fourier Transform (DFT)

complex transform	real transform
synthesis $x[n] = \sum_{k=0}^{N-1} X[k]\, e^{j2\pi kn/N}$	*synthesis* $x[n] = \sum_{k=0}^{N/2} ReX[k]\cos(2\pi kn/N) - ImX[k]\sin(2\pi kn/N)$
analysis $X[k] = \frac{1}{N}\sum_{n=0}^{N-1} x[n]\, e^{-j2\pi kn/N}$	*analysis* $ReX[k] = \frac{2}{N}\sum_{n=0}^{N-1} x[n]\cos(2\pi kn/N)$ $ImX[k] = \frac{-2}{N}\sum_{n=0}^{N-1} x[n]\sin(2\pi kn/N)$

รูปที่ 3.6 แสดงสมการที่ใช้ในการคำนวณดีสครีตฟูริเยร์ทรานสฟอร์ม

3.2.4 ตัวกรองเชิงเลข

โดยทั่วไปคำว่าตัวกรองสัญญาณอาจหมายถึง อุปกรณ์ซึ่งอาจเป็นอุปกรณ์ทางเครื่องมือกลวงจรอิเล็กทรอนิกส์ ซอร์ฟแวร์ หรือ คอมพิวเตอร์ฮาร์ดแวร์ ที่สามารถเปลี่ยนแปลงสเปกตรัมซึ่งเป็นองค์ประกอบทางความถี่ของสัญญาณให้มีคุณสมบัติตามที่ต้องการได้ ดังนั้นในที่นี้คำว่าตัวกรองเชิงเลข

จะหมายถึงระบบการประมวลผลสัญญาณเชิงเลขที่ทำให้สเปกตรัมความถี่เปลี่ยนแปลงไป [2] วัตถุประสงค์เพื่อแยกสัญญาณที่ถูกรวมแล้วหรือสร้างคืน (Restoration) สัญญาณเมื่อมีความเพี้ยนซึ่งเกิดจากการรบกวนจากสัญญาณที่ไม่ต้องการ [11] ในการจำแนกชนิดของตัวกรองเชิงเลขสามารถแบ่งได้ตามกลุ่มของการใช้งาน และวิธีการสร้างแสดงดังรูปที่ 3.7

		FILTER IMPLEMENTED BY:	
		Convolution *Finite Impulse Response (FIR)*	Recursion *Infinite Impulse Response (IIR)*
FILTER USED FOR:	Time Domain *(smoothing, DC removal)*	Moving average	Single pole
	Frequency Domain *(separating frequencies)*	Windowed-sinc	Chebyshev
	Custom *(Deconvolution)*	FIR custom	Iterative design

รูปที่ 3.7 แสดงการจำแนกชนิดของตัวกรองเชิงเลขตามกลุ่มของการใช้งานและวิธีการสร้าง

การใช้งานสามารถแบ่งย่อยออกเป็นสามประเภทได้แก่

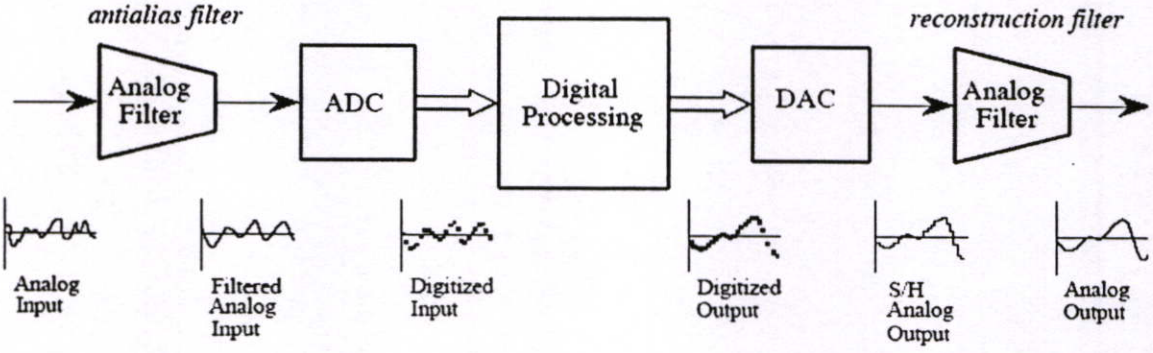
- การใช้งานในโดเมนเวลา ออกแบบเพื่อใช้กับข้อมูลที่สนใจรูปร่างของรูปคลื่นยกตัวอย่าง เช่น ตัวกรองสัญญาณไฟตรงออกจากสัญญาณที่ต้องการ หรือ ตัวกรองประเภท Waveform Shaping
- การใช้งานในโดเมนความถี่ ออกแบบเพื่อใช้กับข้อมูลที่แฝงอยู่ในขนาด, ความถี่และเฟสขององค์ประกอบสัญญาณไซน์ วัตถุประสงค์เพื่อการแยกสัญญาณช่วงความถี่ที่ต้องการออก
- การใช้งานพิเศษเฉพาะอย่าง (Custom) เป็นตัวกรองที่มีคุณสมบัติแตกต่างไปจากตัวกรองพื้นฐานทั้งสี่ชนิด (Low-pass, High-pass, Band-pass, Band reject) ซึ่งหมายถึงการตอบสนองต่อความถี่ของตัวกรองจะมีลักษณะพิเศษตามความต้องการของผู้ออกแบบ

วิธีการสร้างตัวกรองสามารถแบ่งออกได้เป็นสองชนิดได้แก่

- Convolution คือตัวกรองไม่ป้อนกลับเชิงเลขหรือที่เรียกว่าเอฟไอาร์ (Finite Impulse Response: FIR) ตัวกรองชนิดนี้มีความยาวของผลตอบสนองอิมพัลส์จำกัด
- Recursion คือตัวกรองไม่ป้อนกลับเชิงเลขหรือที่เรียกว่าเอฟไอาร์ (Infinite Impulse Response: IIR) เป็นตัวกรองที่มีความยาวของผลตอบสนองอิมพัลส์ไม่จำกัด

3.3 ระบบการประมวลผลสัญญาณเชิงเลข

ระบบพื้นฐานของการประมวลผลสัญญาณเชิงเลขประกอบด้วย ส่วนแรกคือตัวกรองเชิงอุปมานหน้าที่กรองความถี่ต่ำผ่านและสัญญาณจะถูกป้อนให้ส่วนที่สองตัวแปลงสัญญาณเชิงอุปมานเป็นสัญญาณเชิงเลข (Analog to Digital Converter: ADC) ทำหน้าที่สุ่มสัญญาณขาเข้าด้วยคาบเวลาค่าหนึ่งแล้วแบ่งระดับเพื่อแปลงเป็นสัญญาณเชิงเลขส่งต่อให้ส่วนที่สามคือตัวประมวลผลทำหน้าที่คำนวณตามขั้นตอนวิธี (Algorithm) ซึ่งอาจเป็นได้ทั้งซอร์ฟแวร์หรือฮาร์ดแวร์ หลังจากได้คำตอบจากการคำนวณแล้วขั้นตอนที่สี่คือการแปลงสัญญาณเชิงเลขกลับไปเป็นสัญญาณเชิงอุปมานโดยตัวแปลงสัญญาณเชิงเลขเป็นสัญญาณเชิงอุปมาน (Digital to Analog Converter: DAC) และขั้นตอนสุดท้ายของระบบก็คือตัวกรองเชิงอุปมานชนิดความถี่ต่ำผ่านเพื่อตัดความถี่เงา (Image) เพื่อให้ได้สัญญาณเชิงอุปมานที่สมบูรณ์



รูปที่ 3.8 แสดงบล็อคไดอะแกรมของระบบการประมวลผลสัญญาณเชิงเลข

3.3.1 ตัวกรองเพื่อป้องกันเอเลียส (Anti-alias Filter)

เป็นตัวกรองเชิงอุปมานใช้เพื่อตัดสัญญาณขาเข้าที่มีความถี่สูงกว่าครึ่งหนึ่งของความถี่ของการสุ่มเพื่อป้องกันการเกิดความเพี้ยนเอเลียส (Aliasing) จากทฤษฎีการสุ่มที่กล่าวว่าความถี่ของการสุ่มต้องเท่ากับสองเท่าหรือมากกว่าสองเท่าความถี่สูงสุดของสัญญาณขาเข้าที่เราสนใจ ดังนั้นความถี่สูงกว่าครึ่งหนึ่งของความถี่ของการสุ่มถือว่าเป็นสัญญาณรบกวนของระบบ ในการเลือกใช้งานตัวกรองมีคุณสมบัติหลายอย่างที่จะต้องพิจารณารวมไปถึงงานที่จะนำไปประยุกต์ใช้ว่าสนใจสัญญาณในโดเมนความถี่หรือเวลาเป็นหลัก ข้อมูลคุณสมบัติของตัวกรองที่นิยมใช้ในการป้องกันเอเลียสแสดงดังรูปที่ 3.9

	Voltage gain at DC	Step Response			Frequency Response		
		Overshoot	Time to settle to 1%	Time to settle to 0.1%	Ripple in passband	Frequency for x100 attenuation	Frequency for x1000 attenuation
Bessel							
2 pole	1.27	0.4%	0.60	1.12	0%	12.74	40.4
4 pole	1.91	0.9%	0.66	1.20	0%	4.74	8.45
6 pole	2.87	0.7%	0.74	1.18	0%	3.65	5.43
8 pole	4.32	0.4%	0.80	1.16	0%	3.35	4.53
Butterworth							
2 pole	1.59	4.3%	1.06	1.66	0%	10.0	31.6
4 pole	2.58	10.9%	1.68	2.74	0%	3.17	5.62
6 pole	4.21	14.3%	2.74	3.92	0%	2.16	3.17
8 pole	6.84	16.4%	3.50	5.12	0%	1.78	2.38
Chebyshev							
2 pole	1.84	10.8%	1.10	1.62	6%	12.33	38.9
4 pole	4.21	18.2%	3.04	5.42	6%	2.59	4.47
6 pole	10.71	21.3%	5.86	10.4	6%	1.63	2.26
8 pole	28.58	23.0%	8.34	16.4	6%	1.34	1.66

รูปที่ 3.9 แสดงข้อมูลคุณสมบัติของตัวกรองที่นิยมใช้ในการป้องกันเอเลียส

3.3.2 ตัวแปลงสัญญาณเชิงอุปมานเป็นสัญญาณเชิงเลข (Analog to Digital Converter: ADC)

ทำหน้าที่ในการแปลงสัญญาณเชิงอุปมานให้เป็นสัญญาณเชิงเลข ปัจจุบันได้รวมวงจรสุ่มและคงค่า (Sample and Hold) เอาไว้ภายใน ในการเลือกใช้มีสิ่งที่จะต้องพิจารณาหลายอย่างเช่น สถาปัตยกรรม, ความละเอียด (Resolution), ความถี่สุ่มสูงสุดหรือเวลาที่ใช้ในการแปลง (Conversion Time) ฯลฯ เป็นต้น ค่าต่างๆดังกล่าวมีผลทำให้คุณสมบัติทางสแตติค (Static) และไดนามิค (Dynamic) ของ ADC เกิดความแตกต่างกันด้วย [12]

3.3.3 ตัวประมวลผลสัญญาณเชิงเลข (Digital Signal Processor)

การประมวลผลสัญญาณเชิงเลขเป็นการคำนวณทางคณิตศาสตร์ซึ่งมีความซับซ้อนมากถ้าจะเปรียบเทียบกับการประมวลผลทั่วๆไปเช่น การประมวลผลคำ (Word Processing) ฯลฯ เป็นต้น นั่นหมายถึงว่าตัวประมวลผลที่ถูกออกแบบมาเพื่อใช้งานโดยทั่วไปไม่สามารถทำงานได้ประสิทธิภาพสูงสุดกับขั้นตอนวิธี (Algorithm) ของการประมวลผลสัญญาณเชิงเลขเช่น ตัวกรองเชิงเลขหรือการคำนวณดีสครีตฟูริเยร์ทรานสฟอร์ม ฯลฯ เป็นต้น ตัวประมวลผลสัญญาณเชิงเลขก็คือตัวประมวลผลที่ถูกออกแบบมาเพื่องานด้านการประมวลผลสัญญาณเชิงเลขโดยเฉพาะ ดังนั้นมันจะสามารถทำงานได้ผลเร็วกว่าตัวประมวลผลที่ถูกออกแบบมาเพื่อใช้งานทั่วๆไปเมื่อใช้ความถี่สัญญาณนาฬิกาเท่ากัน รายละเอียดของความแตกต่างทางด้านสถาปัตยกรรมสามารถศึกษาเพิ่มเติมได้จาก [11]

3.3.4 ระบบเลขฐานสองที่ใช้ในการประมวลผล

เนื่องจากตัวประมวลผลถูกสร้างจากวงจรเชิงเลขการเก็บค่าต่างๆจะใช้ระบบเลขฐานสอง แต่ในความเป็นจริงตัวเลขที่ใช้ในชีวิตประจำวันเป็นเลขจำนวนจริงที่ประกอบด้วยเลขหลังจุดทศนิยมจำนวนหลายตำแหน่ง เช่น π = 3.14159265358979323846264338327 95 ดังนั้นจึงได้มีการคิดระบบตัวเลขในการแทนเลขจำนวนจริงด้วยเลขฐานสองที่มีจำนวนบิตจำกัดเพื่อใช้ในการประมวลผล สำหรับตัวประมวลผลซึ่งมีขีดความสามารถจำกัดอันเนื่องมาจากสถาปัตยกรรมภายในที่ถูกออกแบบมา เช่น ขีดจำกัดในการเก็บข้อมูลของรีจีสเตอร์หรือขีดจำกัดในการประมวลผลของหน่วยคำนวณ (Arithmetic Logic Unit: ALU) ดังนั้นในการออกแบบระบบการประมวลผลสิ่งสำคัญที่ต้องคำนึงถึงก็คือการเลือกว่าจะใช้ระบบตัวเลขแบบไหนจึงจะเหมาะสมกับงานนั้นๆ ระบบตัวเลขสำหรับงานประมวลผลแบ่งออกเป็น 2 ประเภทใหญ่ๆได้แก่

3.3.4.1 เลขฐานสองแบบจำนวนโดยตรง (Fixed Point)

ตัวเลขฐานสองแบบจำนวนโดยตรงสามารถแบ่งย่อยได้อีกหลายชนิดแสดงดังรูปที่ 3.10 ซึ่งแต่ละชนิดมีหลักการคล้ายๆกันดังนั้นจะอธิบายเฉพาะส่วนที่เกี่ยวข้องกับงานวิจัยนี้เท่านั้น

- Two's Complement Sign Fractional เป็นระบบตัวเลขที่นิยมใช้มากในระบบการประมวลผลสัญญาณเชิงเลขเพราะสามารถแทนค่าได้ทั้งจำนวนเต็มและแบบมีเศษทั้งบวกและลบ [13] ค่าสูงสุดและต่ำสุดที่สามารถแทนได้แสดงดังสมการที่ (3.10) – (3.11)

$$MinValue = -2^{(N-1)} \tag{3.10}$$

$$MaxValue = 2^{(N-1)} - 2^{-M} \tag{3.11}$$

เมื่อ

$MinValue$ และ $MaxValue$ คือค่าต่ำสุดและสูงสุดที่สามารถแทนได้

N คือ จำนวนบิตของเลขฐานสองก่อนจุดทศนิยม

M คือ จำนวนบิตของเลขฐานสองหลังจุดทศนิยม

ยกตัวอย่างในกรณีที่ระบบมีขนาด 8 บิต ให้ $N = 1$ และ $M = 7$ ดังนั้นจะสามารถแทนค่าได้ตั้งแต่ -1 ถึง 0.9921875 โดยค่าละเอียดที่สุดที่สามารถแทนได้คือ 2^{-M} ซึ่งเท่ากับ 0.0078125 ตัวอย่างต่อไปแสดงการบวกเลขสองจำนวนระหว่าง -36.25 และ 13.90625 โดยกำหนด $N = 9$ และ $M = 5$ ดังนี้

36.25 = 000100100.01000 โดยที่เลขหลังจุดคำนวณได้จาก $Round(.25 * 2^{M})$

เมื่อ

Round คือฟังก์ชั่นการปัดเศษ (น้อยกว่า 0.5 ตัดทิ้งแต่ถ้า เท่ากับหรือมากกว่า 0.5 บวกเพิ่ม 1)

-36.25 = 111011011.11000 จากบรรทัดแรกทำเป็นจำนวนลบโดยทำ 2's Complement

13.90625 = 000001101.11101

111011011.11000 + 000001101.11101 = 111101001.10101 คำตอบบิต MSB เป็น 1 แสดงว่าค่าเป็นลบ

พิสูจน์โดยการแปลงกลับเพื่อดูขนาด

-36.25 + 13.90625 = -22.34375

111101001.10101 = 10110.01011

10110 = 22, 01011 = 11, $11/2^5 = 0.34375$

ดังนั้นคำตอบที่ได้คือ -22.34375 เป็นคำตอบที่ถูกต้อง

System	Number Range	Advantages	Disadvantages
Unsigned Integer	0 to $2^N - 1$	Universal numbering system. Easy to perform arithmetic operations such as addition and subtraction.	Cannot store negative numbers.
Two's Complement Integer	$-2^{(N-1)}$ to $2^{(N-1)} - 1$	Stores both positive and negative numbers. Easy to perform arithmetic with regular adders.	Requires one extra bit of storage space when only positive numbers are necessary.
Unsigned Fractional	0 to $2^N - 2^M$	Stores positive numbers greater than and less than 1. Operations are identical to unsigned integer operations.	Cannot store negative numbers.
Two's Complement Signed Fractional	$-2^{(N-1)}$ to $2^{(N-1)} - 2^{-M}$ in 2^{-M} steps	Stores positive and negative numbers both greater than and less than 1. Operations are identical to two's complement operations.	–
Gray Code	0 to $2^{(N-1)}$	Only one bit changes between adjacent numbers, which facilitates interfaces with physical systems.	Difficult to perform arithmetic operations without first converting to one of the systems listed above.
Signed-Magnitude	$-2^{(N-1)} - 1$ to $2^{(N-1)} - 1$	Useful for applications that require the magnitude to be distinct from the sign.	Difficult to perform arithmetic operations (although easier than with Gray code).
Offset Two's Complement	$-2^{(N-1)}$ to $2^{(N-1)} - 1$	Used by many analog-to-digital (A/D) and digital-to-analog (D/A) converters. Easy to perform arithmetic operations.	–
One's Complement	$-2^{(N-1)} - 1$ to $2^{(N-1)} - 1$	Easy to perform negations.	Difficult to perform arithmetic operations other than negations.

รูปที่ 3.10 แสดงตัวเลขฐานสองแบบจำนวนโดยตรงชนิดต่างๆ

• Offset Two's Complement เป็นระบบตัวเลขที่ใช้กับไอซีประเภท ADC หรือ DAC มีรูปแบบคล้ายกับ 2'Complement รูปที่ 3.11 แสดงการเปรียบค่าระว่าง Two's Complement และ Offset Two's Complement จะสังเกตเห็นว่าการแปลงจาก Two's Complement เป็น Offset Two's Complement สามารถทำได้โดยการกลับ MSB นั่นเอง

Decimal Value	Two's Complement	Offset Two's Complement
–4	100	000
–3	101	001
–2	110	010
–1	111	011
0	000	100
1	001	101
2	010	110
3	011	111

รูปที่ 3.11 แสดงการเปรียบค่าระหว่างตัวเลขแบบ Two's Complement และ Offset Two's Complement

3.3.4.2 เลขฐานสองแบบจำนวนอิงดัชนี (Floating Point)

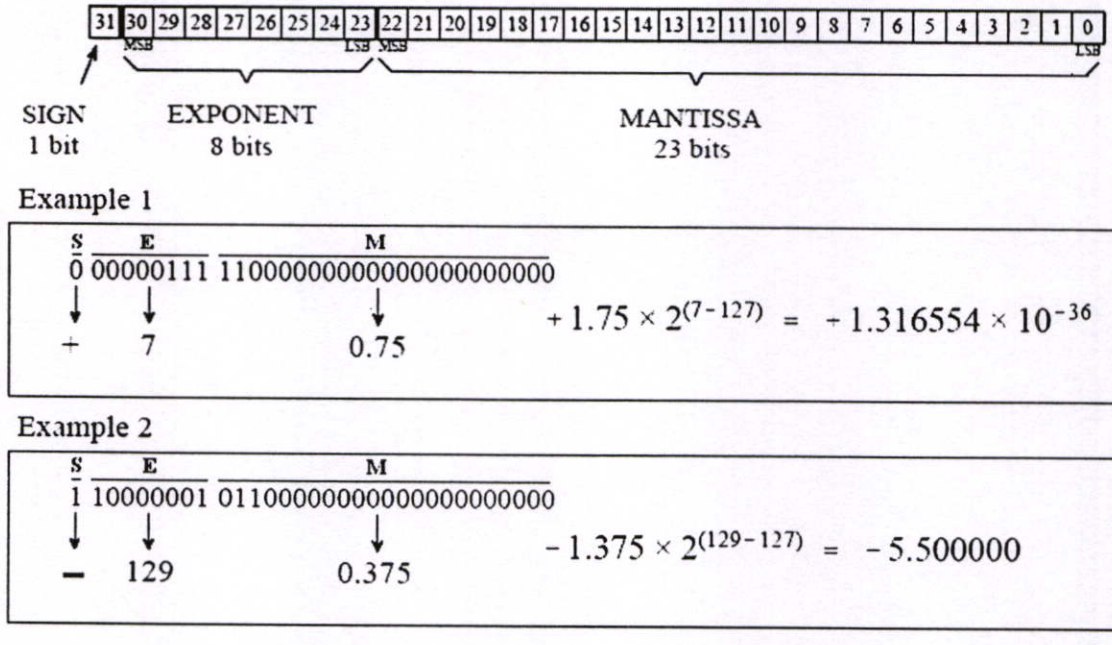
การเขียนตัวเลขแบบจำนวนอิงดัชนีตัวเลขจะถูกเขียนโดยการใช้เลขฐานสองแบบจำนวนโดยตรงสองส่วนคือ ส่วนแรกเป็นค่า แมนทีสซา (Mantissa) และส่วนที่สอง ส่วนชี้กำลัง (Exponent) รูปแบบของการเขียนตัวเลขแบบจำนวนอิงดัชนี แสดงดังสมการที่ (3.12) ตามมาตรฐาน ANSI/IEEE Std. 754-1985 จะเรียกว่า Single Precision เมื่อใช้เลขฐานสองจำนวน 32 บิตในการแสดงค่า และเรียกระบบ 64 บิตว่า Double Precision

$$Value = (-1)^S * M * 2^{E-127} \qquad (3.12)$$

สำหรับระบบ 32 บิต พจน์ $(-1)^S$ นี้ใช้เลขฐานสองจำนวน 1 บิตเป็นตัวแสดงเครื่องหมายนั่นคือ ถ้า S = 0 เป็นค่าบวกแต่ถ้า S = 1 เป็นค่าลบ ตัวแปร E มีค่าอยู่ในช่วง 0 ถึง 255 ใช้เลขฐานสองจำนวน 8 บิต จากสมการจะเห็นว่ามีการลบค่านี้ด้วย 127 เพื่อให้ส่วนชี้กำลังแสดงค่าได้ในช่วง 2^{-127} ถึง 2^{128} และค่าแมนทีสซา M ใช้เลขฐานสองจำนวน 23 บิต ด้วยวิธีการนี้ค่าสูงสุดที่สามารถแทนได้เท่ากับ

$\pm(2-2^{-23})*2^{128} = \pm 6.8*10^{38}$ และค่าต่ำสุดที่สามารถแทนได้เท่ากับ $\pm 1.0*2^{-127} = \pm 5.9*10^{-39}$ ตัวอย่างการแปลงแสดงดังรูปที่ 3.12

จากสมการที่ (3.10) จะพบว่าถ้านำตัวเลขแบบจำนวนอิงดัชนีไปเขียนโปรแกรมหรือสร้างฮาร์ดแวร์ในการประมวลผลจะทำได้ยุ่งยากมาก ยกตัวอย่างเช่น การบวก ตัวเลขที่จะบวกกันได้ส่วนชี้กำลังต้องเท่ากัน ถ้าส่วนชี้กำลังไม่เท่ากันจะต้องทำการเลื่อนจุดทวินิยม (Binary Point) ของตัวเลขที่มีค่าส่วนชี้กำลังน้อยกว่าไปทางขวาโดยเลื่อนไปเท่ากับจำนวนผลต่างของส่วนชี้กำลัง ส่วนการคูณกันของตัวเลขแบบจำนวนอิงดัชนีทำได้โดยนำเอาส่วนแมนทีสซามาคูณกันและนำส่วนชี้กำลังมาบวกกัน ข้อดีของการใช้ตัวเลขแบบจำนวนอิงดัชนีคือ สามารถเขียนแทนค่าของสัญญาณได้ละเอียดและแม่นยำกว่าแบบจำนวนโดยตรง แต่อย่างไรก็ตามฮาร์ดแวร์ที่ใช้ในการคำนวณยุ่งยากซับซ้อนและแพงกว่าแบบจำนวนโดยตรง นอกจากนั้นมันยังต้องการเวลาในการประมวลผลที่มากกว่า ดังนั้นสำหรับการประมวลผลแบบระบบเวลาจริงจึงนิยมใช้ระบบตัวเลขแบบจำนวนโดยตรงมากกว่า รวมไปถึงงานวิจัยที่กำลังนำเสนอก็ได้เลือกใช้ระบบตัวเลขแบบจำนวนโดยตรงเช่นกัน



รูปที่ 3.12 แสดงวิธีการแทนค่าด้วยตัวเลขแบบจำนวนอิงดัชนี

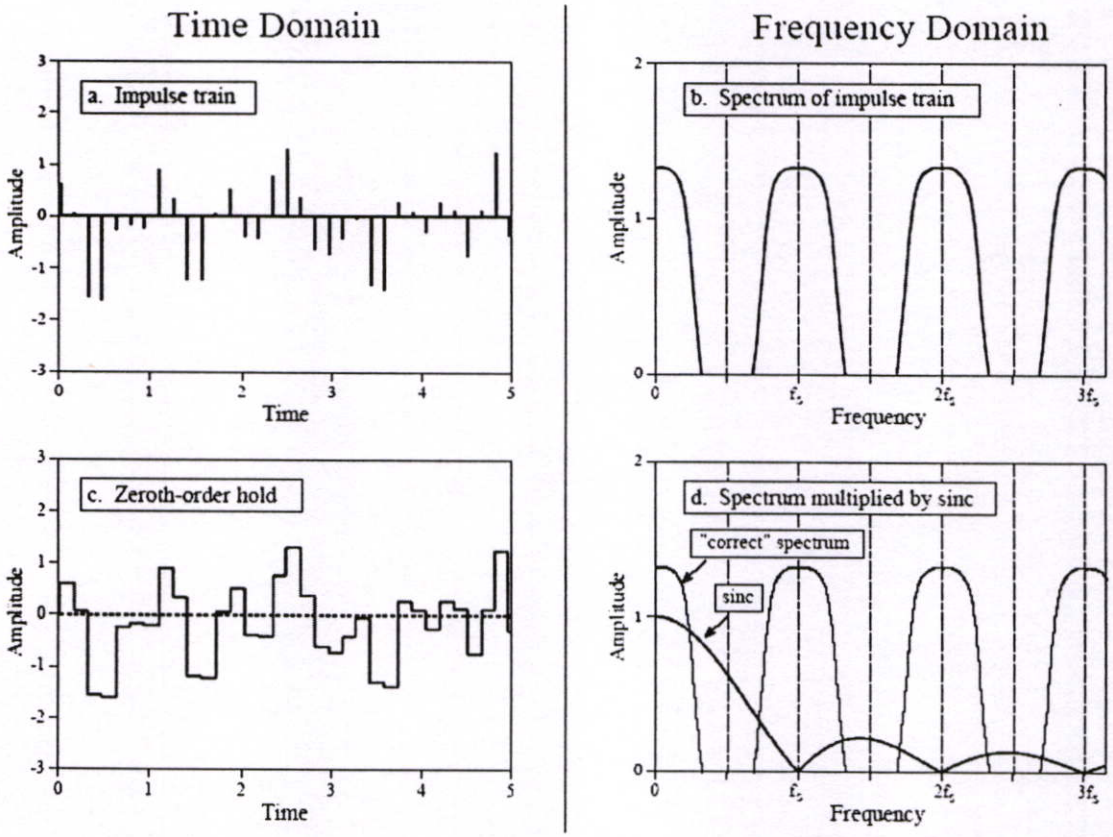
3.3.5 ตัวแปลงสัญญาณเชิงเลขเป็นสัญญาณเชิงอุปมาน (Digital to Analog Converter: DAC)

หลังการประมวลผลสัญญาณผลลัพธ์ที่ได้จะอยู่ในรูปสัญญาณเชิงเลขเป็นข้อมูลจุดซึ่งเป็นสัญญาณชั่วขณะของสัญญาณ แต่สัญญาณขาออกที่ต้องการจากระบบคือสัญญาณเชิงอุปมาน ดังนั้นจึง

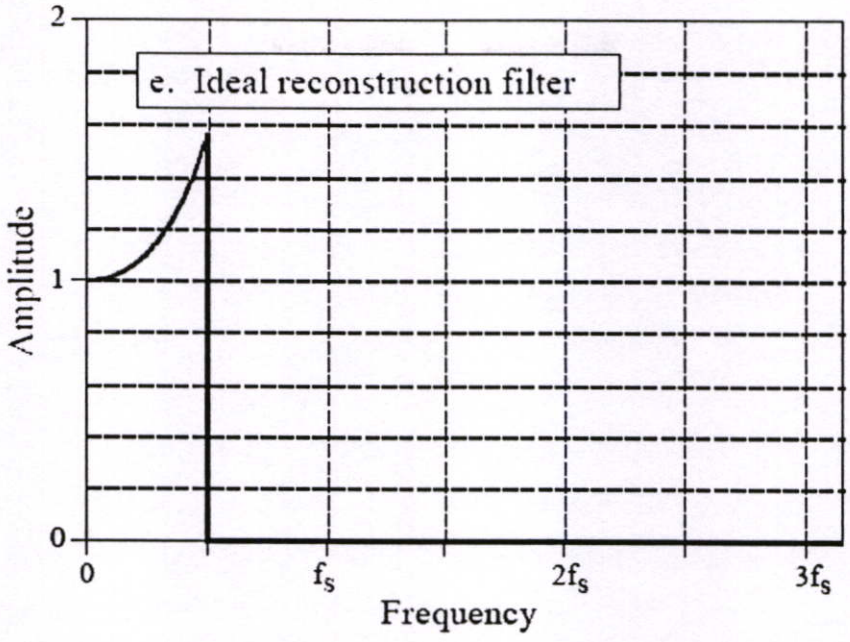
ต้องมีการแปลงสัญญาณจากเชิงเลขไปเป็นเชิงอุปมาน โดย ตัวแปลงสัญญาณเชิงเลขเป็นสัญญาณเชิงอุปมาน

3.3.6 ตัวกรองสร้างสัญญาณคืน (Reconstruction Filter)

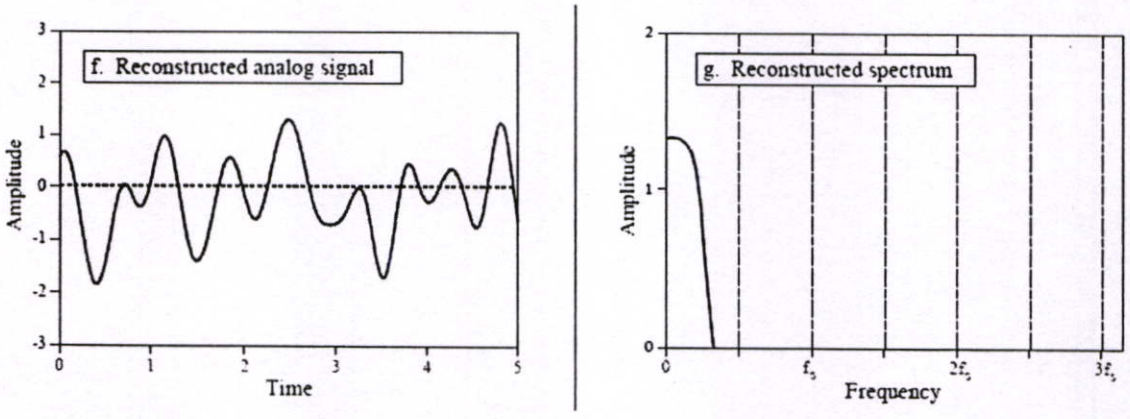
สัญญาณที่ได้จากตัวแปลงสัญญาณเชิงเลขเป็นสัญญาณเชิงอุปมานเป็นสัญญาณชั่วขณะหรือเป็นสัญญาณที่ถูกสุ่ม ดังนั้นมันจะมีองค์ประกอบทางความถี่แสดงดังรูปที่ 3.2 ในกรณีที่ไม่มีการคงค่าข้อมูล (Hold) ดังนั้นการที่จะทำให้ได้สัญญาณที่ถูกต้องจะต้องมีการกรองสัญญาณชั่วขณะดังกล่าวด้วยตัวกรองความถี่ต่ำผ่านเพื่อตัดองค์ประกอบของความถี่ที่มากกว่าครึ่งหนึ่งของความถี่สุ่มทิ้งไป ตัวกรองชนิดนี้เรียกว่า ตัวกรองสร้างสัญญาณคืน (Reconstruction Filter) แต่ในความเป็นจริงสัญญาณที่ได้จากตัวแปลงสัญญาณเชิงเลขเป็นสัญญาณเชิงอุปมานจะถูกคงค่าดังนั้นสเปกตรัมของสัญญาณจะมีความแตกต่างไปจากที่แสดงดังรูปที่ 3.2 เนื่องจากมันถูกคูณด้วยซิงค์ฟังก์ชั่น [11] ดังนั้นในการออกแบบตัวกรองจะต้องพิจารณาแยกเป็นแต่ละกรณีไป



รูปที่ 3.13 แสดงสัญญาณในโดเมนเวลาและความถี่ของสัญญาณก่อนและหลังการคงค่า



รูปที่ 3.14 แสดงผลตอบสนองความถี่ของตัวกรองสร้างสัญญาณคืนในอุดมคติ

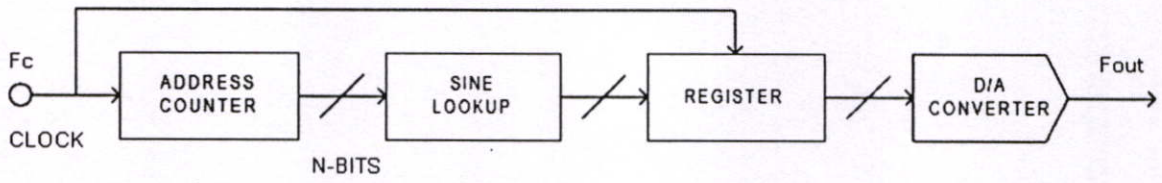


รูปที่ 3.15 แสดงสัญญาณในโดเมนเวลาและความถี่ของสัญญาณที่ได้จากตัวกรองสร้างสัญญาณคืนในอุดมคติ

3.4 Direct Digital Synthesis (DDS)

บางส่วนการทำงานของตัวเข้ารหัสฯที่กำลังนำเสนอต้องการสัญญาณรูปไซน์เพื่อการประมวลผลบางอย่าง เช่น เป็นพาหะรองของการมอดูเลตสัญญาณความแตกต่างสีหรือใช้ในการสังเคราะห์สัญญาณเบิรสต์ เทคนิคที่นำมาใช้ในการสังเคราะห์สัญญาณรูปคลื่นไซน์ได้แก่ ไดเร็คดิจิตอล

ซินทีซีส (Direct Digital Synthesis: DDS) [14], [15] ซึ่งเป็นวิธีการสร้างสัญญาณเชิงอุปมานด้วยวิธีการของระบบเชิงเลข ระบบดีดีแอสอย่างง่ายแสดงดังรูปที่ 3.16

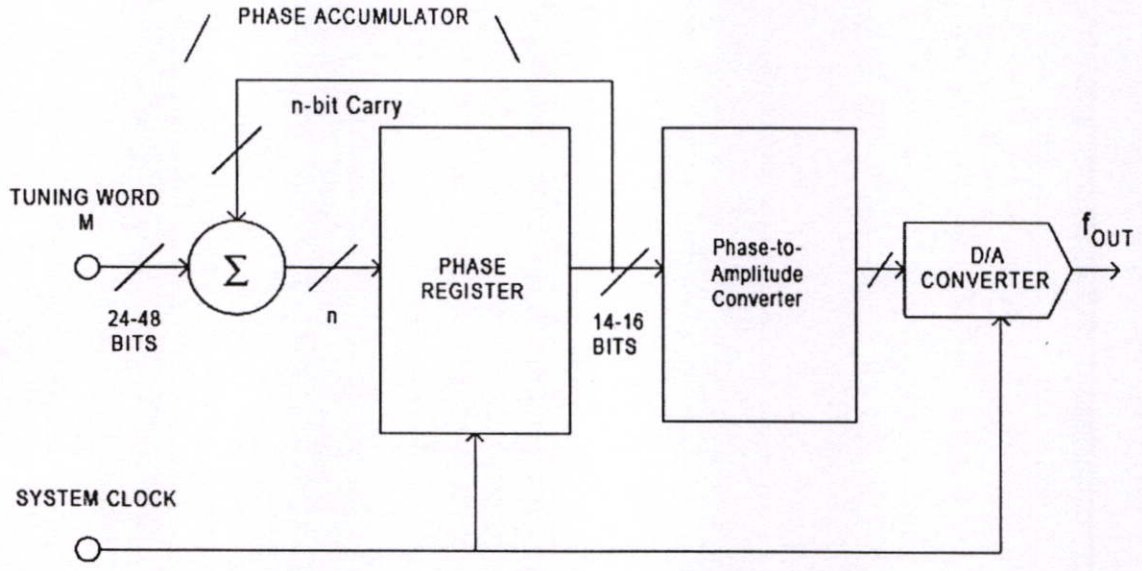


รูปที่ 3.16 แสดงระบบดีดีเอสอย่างง่าย

จากรูปที่ 3.16 ข้อมูลขนาดของสัญญาณไซน์หนึ่งไซเคิล (Cycle) จะถูกเก็บไว้ในหน่วยความจำพีรอม (PROM) ด้วยเหตุนี้ทำให้มันทำหน้าที่เป็นตารางเปิดดูของสัญญาณไซน์ (Sine Lookup Table: Sine LUT) ในการสร้างสัญญาณ ค่าเฟสของสัญญาณจะถูกสร้างจากวงจรนับตำแหน่ง (Address Counter) และส่งค่าตำแหน่งให้กับพีรอมเพื่อเปลี่ยนค่าเฟสให้เป็นค่าขนาดเชิงเลขของสัญญาณไซน์และส่งต่อให้รีจีสเตอร์ (Register) เพื่อทำการซิงโครไนส์ ขั้นตอนสุดท้ายของระบบคือแปลงสัญญาณเชิงเลขให้เป็นสัญญาณเชิงอุปมาน จากระบบอย่างง่ายนี้ ค่าความถี่ของสัญญาณขาออกที่ได้ขึ้นอยู่กับ 2 ค่าได้แก่ ค่าความถี่ของสัญญาณนาฬิกาอ้างอิง (Reference Clock) และค่าช่วงการเปลี่ยนขนาดของสัญญาณไซน์ที่เก็บในพีรอม ดังนั้นการเปลี่ยนค่าความถี่สามารถทำได้ 2 วิธีเช่นกันโดย วิธีแรกเปลี่ยนค่าความถี่ของสัญญาณนาฬิกาอ้างอิงหรือวิธีที่สองเขียนค่าพีรอมใหม่ เนื่องจากทั้งสองวิธีนี้ไม่สนับสนุนฟังก์ชั่นการย้ายความถี่อย่างเร็ว (High-speed Frequency Hopping) ของสัญญาณขาออก จึงได้มีการปรับปรุงระบบโดยการเพิ่มเฟสแอคคิวมูเลเตอร์ (Phase Accumulator) ไปแทนที่วงจรนับตำแหน่ง สถาปัตยกรรมแบบนี้เรียกว่า Numerical Control Oscillator (NCO) ซึ่งเป็นส่วนที่สำคัญมากในระบบดีดีเอสและในวิทยานิพนธ์นี้ก็ได้นำสถาปัตยกรรมแบบนี้มาประยุกต์ใช้งาน ระบบดีดีเอสที่ได้รับการปรับปรุงให้มีความสามารถในการปรับความถี่ได้ (Frequency Tunable) แสดงดังรูปที่ 3.17

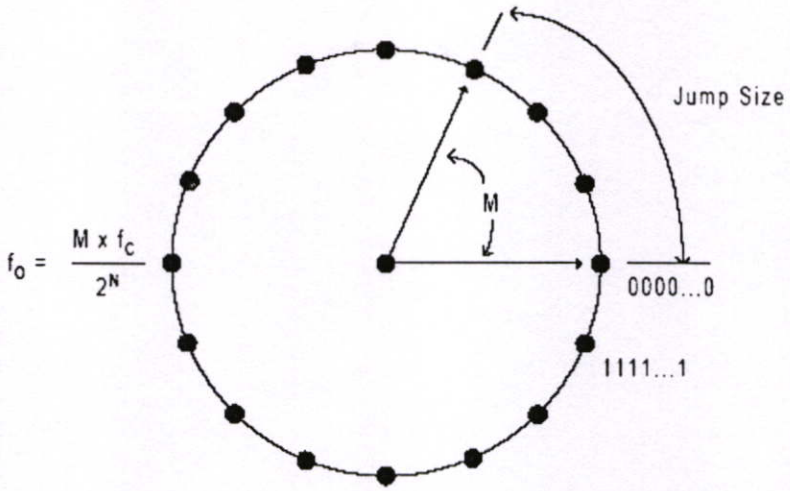
หน้าที่ของส่วนเฟสแอคคิวมูเลเตอร์เปรียบได้ว่าเป็นวงล้อของเฟส (Phase Wheel) ในระบบดีดีเอส เพื่อทำความเข้าใจกับฟังก์ชั่นการทำงานจะสมมติให้สัญญาณไซน์เกิดจากการหมุนของเวคเตอร์รอบวงกลมของเฟส (ดูรูที่ 3.18 ประกอบ) แต่ละจุดบนวงกลมของเฟสก็คือจุดที่เป็นขนาดเชิงเลขของสัญญาณไซน์ หนึ่งรอบการหมุนของเวคเตอร์ด้วยความเร็วคงที่จะทำให้ได้สัญญาณไซน์หนึ่งลูก (Cycle) เฟสแอคคิวมูเลเตอร์ถูกใช้งานให้เสมือนว่าเป็นเวคเตอร์ที่หมุนอยู่รอบวงล้อเฟสอย่างเชิงเส้นแต่สัญญาณขาออกของมันไม่สามารถนำไปใช้ผลิตสัญญาณได้โดยตรงเนื่องจากสัญญาณที่จุดนี้เกิดจาก

การบวกเพิ่มมันจึงได้สัญญาณอยู่ในรูปแรมป์ (Ramp) ที่ข้อมูลถูกตัดปลาย (Truncate) แต่จะใช้เพื่อเป็นตัวชี้ (Pointer) ให้กับพีรอมที่เก็บขนาดเชิงเลขของสัญญาณไซน์



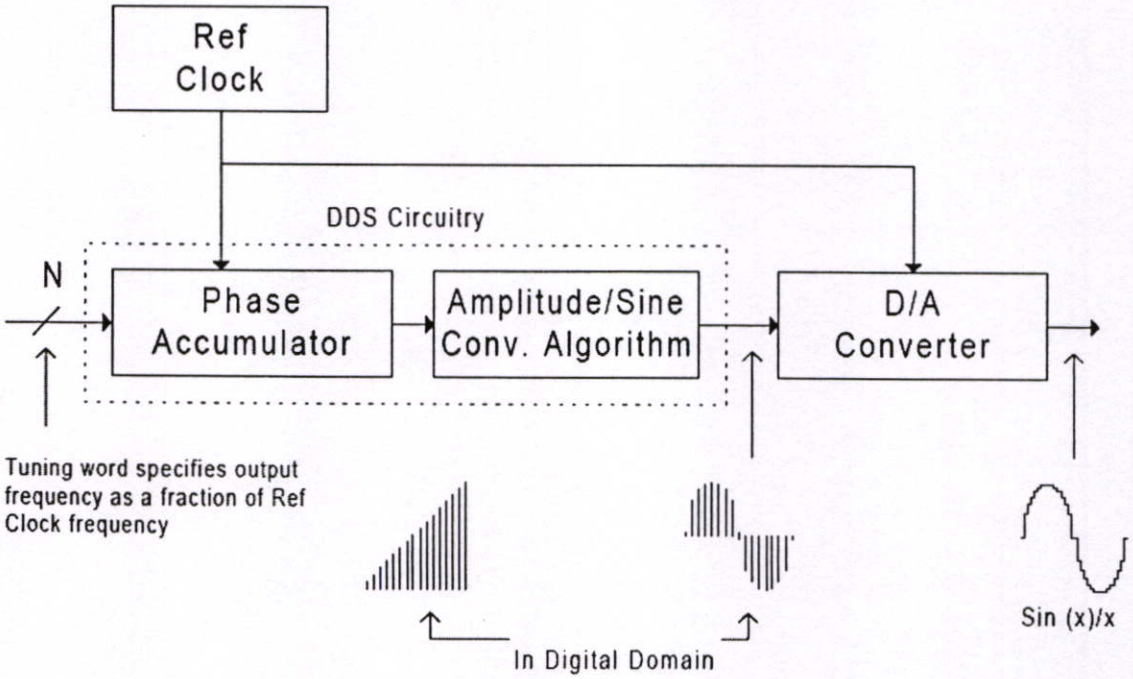
รูปที่ 3.17 แสดงระบบดีดีเอสที่สามารถปรับความถี่ได้

ในความเป็นจริงเฟสแอคคิวมูเลเตอร์ก็คือตัวนับแบบโมดูลัส M (Modulus) ซึ่งมันจะบวกค่าเพิ่มจากค่าที่เก็บไว้ทุกๆคาบของสัญญาณนาฬิกาอ้างอิง ค่าของการบวกเพิ่มกำหนดโดยค่า M ซึ่งเป็นตัวกำหนดค่าการกระโดดของเวคเตอร์ในแต่ละคาบของสัญญาณนาฬิกาหรืออีกนัยหนึ่งก็คือมันเป็นค่าที่บอกจำนวนที่ถูกข้ามในวงล้อเฟส ค่าการกระโดดที่มากจะทำให้การโอเวอร์โฟล์วเกิดเร็วขึ้นหรือใช้เวลาต่อรอบน้อยผลคือได้สัญญาณขาออกไซน์ที่มีความถี่สูง ยกตัวอย่างเช่นสมมติให้เฟสแอคคิวมูเลเตอร์มีขนาด N = 32 บิต และให้ค่า M = 1 กรณีนี้การเกิดโอเวอร์โฟล์วจะต้องใช้เวลาเท่ากับ 2^{32} ลูกของสัญญาณนาฬิกา แต่ถ้าใช้ค่า M = $2^{31}-1$ จะใช้เวลาเพียงสองลูกของสัญญาณนาฬิกาเท่านั้นก็จะทำให้เกิดโอเวอร์โฟล์ว ค่าของการกระโดดนี้เองเป็นตัวบอกถึงความละเอียดของการปรับความถี่ของสถาปัตยกรรมดีดีเอส



n	NUMBER OF POINTS
8	256
12	4096
16	65535
20	1048576
24	16777216
28	268435456
32	4294967296
48	281474976710656

รูปที่ 3.18 แสดงวงล้อเฟสเชิงเลข (Digital Phase Wheel)



รูปที่ 3.19 แสดงสัญญาณที่จุดต่างๆของระบบดีดีเอส

การหาค่าความถี่ที่ได้จากระบบดีดีเอสอาศัยความสัมพันธ์ของเฟสแอคคิวมูเลเตอร์และค่า M ดังนี้

$$Fout = \frac{M * REFCLOCK}{2^N} \tag{3.13}$$

โดยที่

Fout คือ ความถี่สัญญาณขาออกที่ได้จากระบบดีดีเอส

M คือ ค่าของการควบคุมการปรับ (The binary tuning word)

REFCLOCK คือ ความถี่ของสัญญาณนาฬิกาอ้างอิง

N คือ จำนวนบิตของเฟสแอคคิวมูเลเตอร์

บทที่ 4

ภาษาวีเอชดีแอล

4.1 บทนำ

จากทฤษฎีการประมวลผลสัญญาณเชิงเลขที่ได้ทำการศึกษาไปแล้วในบทที่ 3 จะพบว่าการสร้างระบบการประมวลผลสัญญาณเชิงเลขสามารถทำได้สองวิธีคือ วิธีแรกสร้างจากซอฟท์แวร์ วิธีนี้ใช้การเขียนโปรแกรมเพื่อทำการประมวลผลเพื่อให้ได้ผลตามอัลกอริทึม (Algorithm) ที่ต้องการ สำหรับตัวประมวลผลอาจจะใช้ตัวประมวลผลที่สร้างขึ้นมาโดยเฉพาะสำหรับการประมวลผลสัญญาณเชิงเลข เช่น Blackfin Processor, TigerSHARC Processor ของบริษัท Analog Device ฯลฯ หรืออาจจะใช้ตัวประมวลผลที่ใช้งานทั่วไป เช่น ตัวประมวลผลตระกูล X86 ที่ใช้ในเครื่องคอมพิวเตอร์ส่วนบุคคลหรืออาจจะใช้ตัวประมวลผลที่ออกแบบมาสำหรับระบบฝังตัว (Embedded System) เช่น PowerPC, XScale, ARM เป็นต้น วิธีที่สองสร้างจากฮาร์ดแวร์ วิธีที่สองนี้จะแตกต่างจากวิธีแรกคือ ใช้การแปลงอัลกอริทึมไปเป็นวงจรเชิงเลขเพื่อทำการประมวลผล ในอดีตการออกแบบจะใช้การต่อวงจรด้วยไอซีลอจิกเกทมาตรฐาน แต่ปัจจุบันอุปกรณ์ประเภทพีแอลดี (Programmable Logic Device: PLD) เช่น ซีพีแอลดี (Complex Programmable Logic Device: CPLD) หรือ เอฟพีจีเอ (Field Programmable Gate Array: FPGA) ได้ถูกพัฒนาให้มีขีดความสามารถสูงขึ้น เช่น จำนวนลอจิกเกทภายในที่มากขึ้น, ค่าการหน่วงเวลาของลอจิกเกท (Propagation Delay Time: t_{pd}) ที่น้อยลงในขณะที่ราคาต่อจำนวนของลอจิกเกทที่ลดลงอย่างมากรวมไปถึงมันมีความสามารถในการโปรแกรมใหม่ (Re-Program) เมื่อการออกแบบมีการปรับปรุงแก้ไข ดังนั้นการออกแบบวงจรเชิงเลขในปัจจุบันจึงเปลี่ยนไปใช้อุปกรณ์ประเภทพีแอลดีเสียเป็นส่วนใหญ่รวมถึงงานวิจัยที่กำลังนำเสนอนี้ด้วย

การออกแบบวงจรอิเล็กทรอนิกส์ในปัจจุบันไม่ว่าจะเป็นวงจรเชิงอุปมานหรือวงจรเชิงเลขได้มีการนำเครื่องคอมพิวเตอร์มาช่วยในการออกแบบที่เรียกว่า Computer Aided Design (CAD) หรือ Electronic Design Automation (EDA) เพื่อลดต้นทุนของการทำเครื่องต้นแบบและลดเวลาของการพัฒนาผลิตภัณฑ์เพื่อออกวางตลาด สำหรับซอร์ฟแวร์ที่ถูกพัฒนาขึ้นเพื่อช่วยในการออกแบบถูกแบ่งออกเป็น 2 กลุ่มตามรูปแบบของการติดต่อกับผู้ใช้ (User Interface: UI) ได้แก่ แบบแรกโหมดรูปภาพ (Graphic Mode) เช่น การออกแบบวงจรเชิงเลขด้วยการเขียนวงจร (Circuit Diagram) และ แบบที่สอง

โหมดตัวหนังสือ (Text Mode) เช่น การออกแบบวงจรเชิงเลขด้วยการเขียน ภาษาบรรยายพฤติกรรมของฮาร์ดแวร์ (Hardware Description Language: HDL) ซึ่งปัจจุบันมีหลายภาษาด้วยกัน เช่น เวรีล็อก (Verilog), วีเอชดีแอล (Very high speed integrated circuit Hardware Description Language: VHDL) สำหรับงานวิจัยที่กำลังนำเสนอได้เลือกใช้ภาษาวีเอชดีแอลในการออกแบบเนื่องจากมีข้อดีหลายประการซึ่งได้กล่าวไปแล้วในบทที่ 1 การออกแบบวงจรเชิงเลขโดยใช้ภาษาวีเอชดีแอลรวมไปถึงภาษาบรรยายพฤติกรรมของฮาร์ดแวร์อื่นๆ สามารถรองรับหลักการออกแบบที่เรียกว่า "การออกแบบจากบนลงล่าง" (Top Down Design) ซึ่งจะกล่าวถึงรายละเอียดในหัวข้อต่อไป

4.2 การออกแบบจากบนลงล่าง (Top Down Design)

แนวความคิดของการออกแบบจากบนลงล่างคือ การออกแบบระบบที่เริ่มจากการเขียนรูปแบบ (Model) จากแนวความคิดโดยสังเขปซึ่งมีรายละเอียดทางไฟฟ้าหรือรายละเอียดเกี่ยวกับการผลิตน้อยมาก จากนั้นจะผ่านกระบวนการหลายขั้นตอนที่ให้ข้อมูลทางไฟฟ้ามากขึ้นเช่น เป็นวงจรหรือลาย (Layout) ที่ใช้ในการผลิตไอซี (IC Fabrication) ไปจนถึงขั้นสุดท้ายคือการประกอบอุปกรณ์ลงแผ่นวงจรพิมพ์ (Printed Circuit Board: PCB) รวมกับอุปกรณ์ชนิดอื่นเพื่อสร้างเป็นผลิตภัณฑ์ การออกแบบจากบนลงล่างสามารถประยุกต์ใช้ได้กับหลายเทคโนโลยีของการผลิตแต่ละขั้นตอนที่กำลังจะกล่าวถึงเป็นวิธีการกลางซึ่งในรายละเอียดอาจจะมีความแตกต่างกันเล็กน้อยทั้งนี้เนื่องจากเทคโนโลยีเป้าหมายที่แตกต่างกันดังกล่าว ขั้นตอนของการออกแบบจากบนลงล่างมีรายละเอียดดังนี้

- System Specification and Analysis เริ่มการออกแบบด้วยการสร้างข้อกำหนดของความต้องการ (Specification) และวิเคราะห์ระบบเพื่อหาแนวความคิดและหลักการในการแก้ปัญหา
- Modeling and Simulation เขียนรูปแบบ (Model) ของระบบที่ต้องการซึ่งได้จากขั้นตอนแรกโดยใช้ภาษาบรรยายพฤติกรรมของฮาร์ดแวร์และจำลองการทำงานเปรียบเทียบผลที่ได้กับคุณสมบัติที่ต้องการ
- Logic and Test Synthesis รูปแบบ (Model) ที่ทำงานถูกต้องจะถูกสังเคราะห์ (Synthesize) ให้อยู่ในรูปของวงจรเชิงเลข (Gate Level) ที่ประกอบด้วยเกทชนิดต่างๆเชื่อมต่อกันหรืออาจจะอยู่ในรูปของเน็ทลีสท์ (Net list) ในขั้นตอนนี้กับการผลิตของบางเทคโนโลยีเช่น Gate Array, Standard Cell หรือ Full Custom IC อาจมีความจำเป็นที่จะต้องสร้างโครงสร้างของวงจรใหม่หลังจากที่ได้สังเคราะห์ไปแล้วในครั้งแรกเช่น อาจจะสร้างวงจรเพื่อทำการทดสอบตัวเอง (Build-In Self Test:

BIST) รวมถึงข้อมูลในการทดสอบ (Test Pattern) หลังการผลิตเป็นเครื่องต้นแบบและเรียกการออกแบบด้วยวิธีการนี้ว่า "Design for Test" (DFT)

- Pre-Timing Verification หลังจากได้ทำการสังเคราะห์ระบบให้อยู่ในระดับของเกทหรือเน็ทลีสท์แล้วในขั้นตอนนี้เป็นขั้นตอนของการจำลองการทำงานซึ่งต้องใช้ข้อมูลของเทคโนโลยีเป้าหมายที่ต้องการใช้เช่น เทคโนโลยีเอฟพีจีเอหรือเทคโนโลยีซีพีแอลดีของบริษัท Xilinx, Altera, Actel ฯลฯ ซึ่งประกอบด้วยข้อมูลของการทำงานในระดับฟังก์ชั่นและข้อมูลทางเวลาอันเนื่องมาจากความจริงที่ว่าอุปกรณ์อิเล็กทรอนิกส์ทุกชนิดจะมี Propagation Delay เสมอถึงแม้ว่ามันจะมีค่าน้อยมากคืออยู่ในระดับนาโนวินาที (10^{-9} Second) แต่ถ้าระบบที่ออกแบบเป็นระบบใหญ่ประกอบด้วยเกทจำนวนมากตั้งแต่ 10,000 จนถึงระดับล้านเกท ค่าการหน่วงเวลาสะสมอาจจะทำให้ระบบทำงานผิดฟังก์ชั่นที่ต้องการหรือไม่สามารถทำงานกับความถี่สัญญาณนาฬิกา (Clock) ที่ต้องการได้
- Physical Design and Analysis เป็นขั้นตอนของการผลิตเป็นวงจรจริงเช่น การผลิตเป็นเอสิก, การโปรแกรมข้อมูลของการออกแบบลงเอฟพีจีเอ, ซีพีแอลดี หรือการประกอบอุปกรณ์ลงแผ่นวงจรพิมพ์
- Post-Timing Verification หลังจากได้วงจรจริงจากขั้นตอนที่ผ่านมาขั้นตอนนี้เป็นขั้นตอนทดสอบการทำงานที่คำนึงถึงเวลาและความถูกต้องของฟังก์ชั่นที่ได้จากวงจรจริง เนื่องจากในขั้นตอนนี้จะได้วงจรที่ประกอบด้วย 2 จุดใหญ่ๆจะต้องทำการทดสอบคือ สัญญาณขาเข้าและสัญญาณขาออกซึ่งเป็นจุดที่จะต้องนำไปเชื่อมต่อกับส่วนอื่นๆเพื่อสร้างเป็นระบบหรือผลิตภัณฑ์ที่พร้อมใช้งาน
- System Level Verification ระบบที่กำลังออกแบบเพื่อสร้างเป็นผลิตภัณฑ์อาจจะประกอบด้วยวงจรหลายๆวงจรและอุปกรณ์ชนิดอื่นๆอีกหลายชนิดเช่น อาจจต้องใช้ทั้งอุปกรณ์ดีสครีต (Discrete Device) และวงจรรวม (IC) และอาจจะประกอบด้วยเอสิกและหรือพีแอลดีหลายตัว เป็นต้น ดังนั้นหลังจากการรวมอุปกรณ์ต่างๆเหล่านี้เข้าด้วยกันจึงต้องมีการทดสอบการทำงานอีครั้งหนึ่งซึ่งถือว่าเป็นขั้นตอนการควบคุมคุณภาพของผลิตภัณฑ์อันเป็นขั้นตอนสุดท้ายของการออกแบบจากบนลงล่าง

ประโยชน์ของการออกแบบจากบนลงล่างคือการแก้ไขข้อผิดพลาดหรือการเพิ่มคุณสมบัติใหม่ให้กับระบบที่ออกแบบสามารถทำได้ง่ายเนื่องจากทุกขั้นตอนมีการตรวจสอบความถูกต้องก่อนการออกแบบในระดับต่อไป การตรวจสอบแก้ไขข้อผิดพลาดรวมทั้งการเพิ่มคุณสมบัติใหม่ให้กับระบบจะทำในระดับที่สูงกว่าซึ่งเป็นการง่ายเมื่อเปรียบเทียบกับการแก้ไขหรือเพิ่มเติมในระดับล่าง ยกตัวอย่างเช่น การออกแบบไมโครคอนโทรลเลอร์ ถ้าผู้ออกแบบต้องการสร้างชุดคำสั่งใหม่ การเริ่มแก้ไขในระดับเกทเพื่อให้ได้ชุดคำสั่งใหม่ดังกล่าวจะทำได้ยากมากเนื่องจากมีส่วนเกี่ยวข้องกันหลายส่วน แต่ถ้า

ผู้ออกแบบเริ่มการแก้ไขตั้งแต่ระดับบนสุด (Top Level) เช่นแก้ไขระดับฟังก์ชั่นและทำตามขั้นตอนเดิมเหมือนการออกแบบครั้งแรกจะสามารถทำได้ง่ายกว่า

ในปัจจุบันบริษัทผู้ผลิตซอร์ฟแวร์รวมไปถึงบริษัทผู้ผลิตไอซีได้ทำการพัฒนาโปรแกรมที่เรียกว่า "เครื่องมือสังเคราะห์" (Synthesis Tool) เพื่อรองรับการออกแบบจากบนลงล่าง กล่าวคือมันจะทำการสังเคราะห์จากระดับบนสุดของการออกแบบและให้ไฟล์วัตถุที่สามารถนำไปสร้างเอสิกหรือนำไปโปรแกรมอุปกรณ์ประเภทพีแอลดี ทำให้ผู้ออกแบบไม่ต้องสนใจรายละเอียดในระดับล่างหรือระดับต่ำกว่าระดับที่ทำการออกแบบเช่น การออกแบบระบบเชิงเลขในระดับฟังก์ชั่นผู้ออกแบบไม่ต้องสนใจว่าต้องใช้เกทชนิดอะไรมาต่อเพื่อให้ได้การทำงานตามที่ต้องการหรือการออกแบบวงจรเชิงเลขในระดับโครงสร้าง (ระดับเกท) ผู้ออกแบบไม่ต้องสนใจว่าจะต้องต่อทรานซิสเตอร์อย่างไร (ในกรณีที่ใช้เทคโนโลยีทีทีแอล) จึงจะได้การทำงานตามที่ต้องการ นอกเสียจากว่าระบบที่ออกแบบต้องการ การออกแบบที่ได้ผลดีที่สุด (Optimum) เช่น ทำงานเร็วที่สุดและหรือใช้ทรัพยากรน้อยที่สุด บางโปรแกรมได้เปิดโอกาสให้ผู้ออกแบบสามารถแก้ไขการออกแบบในระดับล่างได้เช่นกัน เช่น โปรแกรมของบริษัท Xilinx หรือ Altera ได้เตรียมเครื่องมือที่เรียกว่า ฟลอร์แพลนเอดิเตอร์ (Floorplane Editor) ซึ่งผู้ออกแบบสามารถแก้ไขการเชื่อมต่อภายในของชิพได้ เป็นต้น

4.3 โครงสร้างและส่วนการออกแบบ

ภาษาวีเอชดีแอลเป็นภาษาบรรยายพฤติกรรมของฮาร์ดแวร์หรืออาจกล่าวได้ว่าเป็นภาษาที่ใช้ในการสร้างระบบหรือสังเคราะห์วงจร (ในยุคเริ่มต้นของการสร้างภาษามีจุดประสงค์เพื่อนำมาสร้างระบบเชิงเลขแต่ในปัจจุบันกำลังพัฒนาให้สามารถบรรยายพฤติกรรมของระบบเชิงอุปมานซึ่งถูกเรียกว่า VHDL-AMS) ซึ่งแตกต่างจากภาษาทั่วไปที่เขียนขึ้นเพื่อใช้งานกับตัวประมวลผล เช่น ภาษาซี, ภาษาแอสเซมบลี ซึ่งเป็นการเขียนโปรแกรมเพื่อจัดการกับข้อมูล ดังนั้นการเขียนจึงถูกเรียกว่าการ "โปรแกรม" (Programming) ในขณะที่การเขียนภาษาวีเอชดีแอลถูกเรียกว่าการ "โมเดล" (Modeling) โดยที่ภาษาทั้งสองชนิดดังกล่าวมีหลักภาษาหรือไวยากรณ์ (Grammar) แตกต่างกันหรืออาจจะมีบางส่วนคล้ายกันบ้าง ดังนั้นก่อนที่จะศึกษาถึงรายละเอียดของการโมเดล (Model) ภาษาวีเอชดีแอล จึงมีความจำเป็นอย่างยิ่งที่จะต้องศึกษารายละเอียดของไวยากรณ์เพื่อการโมเดลที่ถูกต้องตามกฎเกณฑ์ (Syntax) และมีประสิทธิภาพ

4.3.1 ลักษณะของรูปแบบ (Model Styles)

ลักษณะของการเขียนรูปแบบ (Model) ด้วยภาษาวีเอชดีแอลสามารถแบ่งออกได้ดังนี้

- Behavioral Model เรียกอีกอย่างหนึ่งว่า Algorithmic Description เป็นรูปแบบ (Model) ที่บรรยายพฤติกรรมของระบบเชิงเลขซึ่งแสดงความสำพันธ์ระหว่างสัญญาณขาเข้าและสัญญาณขาออก เช่น ถ้าสัญญาณขาเข้ามีค่าตามเงื่อนไขตามที่กำหนดจะให้สัญญาณขาออกเป็นอย่างไร มีรูปแบบคล้ายการเขียนโปรแกรมด้วยภาษาระดับสูงทั่วไปเช่น ภาษาซี, ภาษาปาสคาล ฯลฯ ในการโมเดลภาษาวีเอชดีแอลแบบนี้จะไม่ให้รายละเอียดทางด้านโครงสร้างของฮาร์ดแวร์

- Dataflow Model เรียกอีกอย่างหนึ่งว่า Register Transfer Level (RTL) เป็นรูปแบบ (Model) ที่ให้รายละเอียดทางโครงสร้างของฮาร์ดแวร์มากขึ้นกล่าวคือ การโมเดลจะแสดงการดำเนินการเพื่อให้ได้มาซึ่งสัญญาณขาออกที่ต้องการยกตัวอย่างเช่น Output = ((not InputC) and InputA) or (InputC and InputB) จากตัวอย่างจะเห็นว่าเป็นการสร้างรีจีสเตอร์ (Register) นั่นเองซึ่งในความเป็นจริงรีจีสเตอร์เกิดจากการต่อกันของเกทเพื่อให้ได้ผลอย่างใดอย่างหนึ่งของสัญญาณขาออกเช่น ท็อกเกิลฟลิปฟลอฟ (Toggle Flip Flop) ค่าสัญญาณขาออกของมันจะกลับสถานะทุกๆคาบของสัญญาณนาฬิกา เป็นต้น

- Structural Model เป็นรูปแบบที่แสดงการเชื่อมต่อกันระหว่างอุปกรณ์ต่างๆที่ประกอบกันขึ้นเป็นวงจรหรือระบบเชิงเลขและสามารถเรียกอีกอย่างหนึ่งว่า Netlist Representation

- Mixed Level Model เป็นการโมเดลที่ใช้รูปแบบทั้งสามชนิดแรกรวมกันในการออกแบบวงจรหรือระบบเชิงเลข ซึ่งเป็นความอ่อนตัวของภาษาวีเอชดีแอลที่เปิดโอกาสให้ผู้ออกแบบสามารถใช้รูปแบบผสมกับวงจรหรือระบบเดียวกันได้ โดยเฉพาะกับระบบเชิงเลขขนาดใหญ่ซึ่งบางครั้งแต่ละส่วนการทำงานต้องการรายละเอียดที่ต่างกันซึ่งอาจจะเกิดจากวัตถุประสงค์หรือข้อจำกัดของการออกแบบ

4.3.2 คำศัพท์ที่ใช้ในภาษาวีเอชดีแอล (VHDL Term)

การเขียนรูปแบบของระบบเชิงเลขด้วยภาษาวีเอชดีแอลจะมีศัพท์เทคนิคเฉพาะ ดังนั้นในส่วนนี้จะอธิบายศัพท์บางคำที่เกี่ยวของกับการโมเดลภาษาวีเอชดีแอลกับงานวิจัยที่กำลังนำเสนอ

- Concurrency เป็นชุดคำสั่ง (Statements) ที่ทำงานพร้อมกัน (แข่งขนาน) คือ มันจะทำงานพร้อมกันในเวลาเดียวกันทุกครั้งเมื่อมีการเปลี่ยนแปลงค่าของสัญญาณและทำงานอิสระต่อกัน ซึ่งเป็นคุณสมบัติที่เป็นความจริงทางฟิสิกส์ของวงจรอิเล็กทรอนิกส์

- Sequential เป็นชุดคำสั่งที่ทำงานแบบเรียงลำดับขั้นจากบนลงล่าง คล้ายกับการเขียนโปรแกรมภาษาที่ใช้งานกับตัวประมวลผลทั่วไปเช่น ภาษาซีหรือแอสเซมบลี ชุดคำสั่งที่เป็น Sequential นี้จะใช้ในโปรแกรมย่อยและใน Process ซึ่งจะกล่าวถึงรายละเอียดในหัวข้อต่อไป

- Driver สัญญาณ (Signal) ต่างๆในภาษาวีเอชดีแอลจะถูกควบคุมด้วยตัวขับ (Driver) กล่าวคือสัญญาณจะได้รับค่าระดับของสัญญาณใหม่ได้ด้วยตัวขับนี้เอง

- Transaction เมื่อมีการกำหนดค่าๆหนึ่งให้กับสัญญาณโดยที่ค่าใหม่ที่สัญญาณได้รับอาจจะมีผลหรือไม่มีผลทำให้เกิดการเปลี่ยนแปลงระดับของสัญญาณก็ได้เช่น สัญญาณถูกเปลี่ยนจากลอจิก "0" เป็น ลอจิก "0" ถือว่าเกิด Transaction กับสัญญาณ

- Event คือการเปลี่ยนแปลงค่าระดับของสัญญาณจากระดับหนึ่งไปสู่ระดับหนึ่งเช่น เปลี่ยนจากลอจิก "0" เป็น ลอจิก "1" ถือว่าเกิด Event ฉะนั้นจะเห็นว่าการเกิด Event จะเกิด Transaction ด้วยเสมอ แต่การเกิด Transaction อาจไม่เกิด Event

- Sensitivity List คือรายชื่อของสัญญาณต่างๆที่ใช้กระตุ้นการทำงานของชุดคำสั่ง Concurrent เมื่อเกิด Event ขึ้นกับสัญญาณตัวใดตัวหนึ่งหรือหลายๆตัวพร้อมกันในรายชื่อของ Sensitivity List

- Object ในภาษาวีเอชดีแอลคำว่า Object ใช้เขียนเพื่อบ่งบอกถึงองค์ประกอบส่วนหนึ่งของรูปแบบ (Model) สามารถแบ่งออกได้เป็นสามชั้น (Class) ด้วยกันคือ

  - Constant หรือค่าคงที่ เป็น Object ประเภทหนึ่งที่เมื่อกำหนดค่าเริ่มต้นแล้วไม่สามารถเปลี่ยนแปลงแก้ไขได้ สามารถประกาศใช้ได้ในส่วนประกาศ (Declaration) ของรูปแบบ

  - Signal คือ Object ประเภทหนึ่งที่สามารถเปลี่ยนแปลงค่าได้ การเปลี่ยนแปลงค่าเกิดจากการขับของตัวขับ (Driver) แต่ Signal จะสามารถรับค่าได้เพียงค่าเดียวเท่านั้น ณ เวลาใดเวลาหนึ่ง Signal สามารถประกาศใช้ได้ในส่วนของ Concurrent Body ดังนั้นมันจึงสามารถถูกนำไปใช้ได้ตลอดโครงสร้างของรูปแบบ (Model) หรือที่เรียกว่า Global Object

  - Variable หรือตัวแปร เป็น Object ที่สามารถเปลี่ยนแปลงค่าได้แต่จะเก็บค่าได้เพียงค่าเดียวเท่านั้นในขณะเวลาหนึ่ง ตามมาตรฐานของ IEEE 1076-1987 ตัวแปรสามารถประกาศใช้ได้เฉพาะในส่วนของ Sequential Body เท่านั้นทำให้มันถูกสามารถใช้ได้เฉพาะในขอบเขตที่ประกาศเท่านั้น แต่มาตรฐาน IEEE 1076-1993 ซึ่งเป็นมาตรฐานใหม่ของภาษาวีเอชดีแอล สามารถประกาศเป็นตัวแปรชนิด Global ได้

- Data Type คือประเภทข้อมูล ได้แก่ Type ของ Object ซึ่งเป็นตัวกำหนดว่าค่าใดบ้างในกลุ่มของค่า (Set of Value) สามารถที่จะกำหนดให้กับ Object ได้ นอกจากนั้น Type ยังเป็นตัวกำหนดการทำงานในลักษณะต่างๆ ของ Object นั้นๆ ประเภทข้อมูลแบ่งออกเป็น 4 ประเภทคือ

  - Scalar ได้แก่ตัวเลขซึ่งในภาษาวีเอชดีแอลมี่ตัวเลขจำนวนเต็มบวก (Integer) เช่น 1, 30, 100 เป็นต้น และเลขจำนวนจริง (Real) เช่น 1.0, 1E2 เป็นต้น

- Enumeration กลุ่มของค่าประเภทนี้ได้แก่ ตัวหนังสือหรือชื่อต่างๆ
- Physical ได้แก่หน่วยทางฟิสิกส์ในระบบ SI
- File เป็นประเภทของข้อมูลภายนอกสามารถมีค่าได้หลายๆอย่าง

Predefined	Operators
Logical Operators:	NOT, AND, OR, NAND, NOR, XOR
Operand TYPE:	BIT, BOOLEAN
Result TYPE:	BIT, BOOLEAN
Relational Operators:	=, /=, <, <=, >, >=
Operand TYPE:	TYPE ใดๆ ก็ได้
Result TYPE:	BOOLEAN
Arithmetic Operator:	+, -, *, /, **, MOD, REM, ABS
Operand TYPE:	INTEGER, REAL, Physical
Result TYPE:	INTEGER, REAL, Physical
Concatenation Operators:	&
Operand TYPE:	array ของ TYPE ทุกประเภท
Result TYPE:	array ของ TYPE ทุกประเภท

รูปที่ 4.1 แสดงตัวดำเนินการและผลที่ได้จากการดำเนินการในภาษาวีเอชดีแอล

- Predefined Type คือประเภทของ Object ที่ได้กำหนดไว้แล้ว ได้แก่ Type ที่กำหนดไว้ใน Package ชื่อ Standard และกำหนดโดย IEEE ว่าจะต้องมีในระบบที่ใช้พัฒนาภาษาวีเอชดีแอลดังนั้นจึงไม่จำเป็นต้องประกาศใช้ในทุกๆรูปแบบที่เขียนขึ้น Type ประเภทนี้ได้แก่

- Boolean คือกลุ่มของค่า False และ True
- Bit คือกลุ่มของค่า “0” และ “1”
- Integer คือกลุ่มของค่าตั้งแต่ -214748347 ถึง 214748347
- Real คือกลุ่มของค่า -1.0E38 ถึง 1.05E38
- Character คือกลุ่มของค่า พยัญชนะ “A”-“Z”, “a”-”z” อักษรหรือเครื่องหมายพิเศษและตัวอักษรควบคุม
- Time ได้แก่หน่วยเวลาที่มีค่าพื้นฐานเป็นวินาที
- Severity Level คือกลุ่มของค่า Note, Warning, Error, Failure

- Subtype คือประเภทย่อยของข้อมูล สำหรับ Type ที่กำหนดไว้แล้วสามารถที่จะแบ่งออกเป็นกลุ่มย่อยลงไปได้อีก โดยที่องค์ประกอบของ Type ใหม่นี้จะเป็นส่วนหนึ่งของ Type เดิมหรือที่เรียกว่า Subtype

- Operator คือตัวดำเนินการเนื่องจาก Type ของ Object จะเป็นตัวบ่งบอกถึง Operator ที่ Object นั้นๆสามารถกระทำได้ ตลอดจน Type ของผลลัพธ์ที่ได้จากการทำงาน ในภาษาวีเอชดีแอลแบ่งตัวดำเนินการออกได้เป็นประเภทต่างๆแสดงดังรูปที่ 4.1

- Testbenches คือรูปแบบ (Model) ที่เขียนขึ้นเพื่อสร้างสัญญาณทดสอบเพื่อป้อนเป็นสัญญาณขาเข้าให้กับรูปแบบที่กำลังออกแบบในการจำลองการทำงาน

4.3.3 หน่วยของแบบ (Design Units)

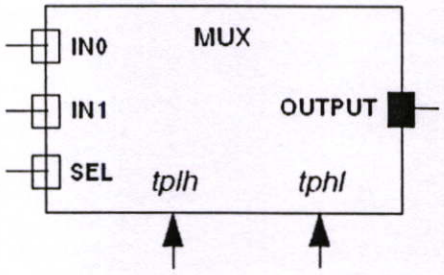
จากที่ได้กล่าวไปแล้วในข้างต้นว่าภาษาวีเอชดีแอลถูกออกแบบมาเพื่อสร้างระบบ มันจึงมีความแตกต่างในลักษณะของโครงสร้างเมื่อเปรียบเทียบกับภาษาคอมพิวเตอร์ระดับสูงทั่วไป ดังนั้นก่อนที่จะทำการศึกษาการเขียนรูปแบบ (Modeling) จึงมีความจำเป็นอย่างยิ่งที่จะต้องแนะนำให้รู้จักหน่วยต่างๆของแบบ (Design Units) ที่ใช้ในภาษาและถือได้ว่าเป็นขั้นตอนแรกที่สำคัญที่สุดของการเรียนรู้การใช้ภาษาวีเอชดีแอลในการเขียนรูปแบบบรรยายพฤติกรรมของระบบเชิงเลข ส่วนประกอบต่างๆที่สำคัญและเป็นพื้นฐานของการเขียนรูปแบบระบบเชิงเลขที่สำคัญมี 4 หน่วยคือ

- Entity Design Unit
- Architecture Design Unit
- Package Design Unit
- Configuration Design Unit

4.3.3.1 Entity Design Unit

หน่วยของแบบส่วนนี้เป็นส่วนที่ใช้กำหนดจุดต่อ, ประเภทของค่า, และทิศทางการไหลของสัญญาณว่าเป็นสัญญาณขาเข้าหรือขาออกของแต่ละจุดต่อหรือช่องทาง (Port) ของรูปแบบ เพื่อการติดต่อระหว่างโลกภายนอกกับรูปแบบที่เขียนขึ้น อธิบายให้เข้าใจง่ายๆก็คือสมมติให้รูปแบบที่กำลังออกแบบเป็นไอซีมัลติเพล็ก หน่วย Entity จะใช้ในการกำหนดชื่อของขาของไอซีที่กำลังออกแบบและกำหนดแต่ละขาว่าเป็นขาออกหรือขาเข้า นอกจากนั้นยังสามารถใส่ข้อมูลเกี่ยวกับเวลาเช่น Propagation Delay Time: t_{pd} ฯลฯ เพื่อใช้ในการจำลองการทำงานของระบบ

```
entity MUX is
    generic (tplh       :time:= 3 ns;
             tphl       :time:= 5 ns
             );
    port    (IN0        :in bit;
             IN1        :in bit;
             SEL        :in bit;
             OUTPUT     :out bit
             );
end MUX;
```



รูปที่ 4.2 (ซ้าย) แสดงโครงสร้างของหน่วย Entity (ขวา) แสดงมุมมองการเชื่อมต่อ

จากรูปที่ 4.2 เป็นรูปแบบ (Model) ของอุปกรณ์ชื่อมัลติเพล็กโดยชื่อ (MUX) จะเขียนอยู่หลังคำสั่ง entity คำสั่ง generic ใช้กำหนดค่าคงที่ tphl และ tplh ซึ่งเป็น Object ประเภทเวลามีค่าตายตัว (Default Value) อยู่ที่ 3 และ 5 nS. ตามลำดับ คำสั่ง port ใช้กำหนดชื่อ, ประเภทและทิศทางของสัญญาณสำหรับรูปแบบนี้จะเห็นว่ามีช่องทาง (Port) การเชื่อมต่อทั้งหมดสี่ช่องทางแบ่งออกเป็นสามช่องทางขาเข้า คือ IN0, IN1, SEL และ หนึ่งช่องทางขาออกคือ OUTPUT ทุกช่องทางเป็นชนิดบิทหรือมีขนาด 1 เส้นของสัญญาณดังนั้นค่าระดับของสัญญาณเป็นได้แค่สองค่าคือ "0" และ "1"

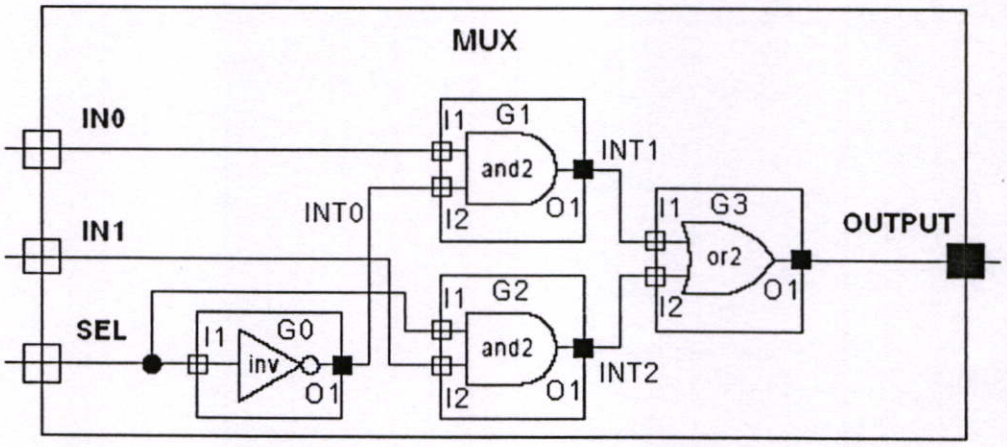
4.3.3.2 Architecture Design Unit

เป็นส่วนที่ใช้เขียนบรรยายกำหนดพฤติกรรมของรูปแบบซึ่งลักษณะของการเขียนรูป (Model Styles) แบบสามารถแบ่งออกได้เป็น 4 ลักษณะดังได้กล่าวไปแล้วในข้างต้น

```
architecture IDENTIFIER of COMPONENT_NAME is
    [Declaration]
begin
    Specification of the functionality of the
    component in terms of its input lines and as
    influenced by physical and other parameters
end [IDENTIFIER];
```

รูปที่ 4.3 แสดงโครงสร้างของหน่วย Architecture

การเขียนเริ่มต้นด้วยคำสั่ง architecture และตามด้วยชื่อของหน่วย architecture นั้นๆ ซึ่งอาจจะตั้งตามลักษณะของรูปแบบที่กำลังบรรยายเช่น BEHAVE, RTL, STRUCT หรือ MIXED เป็นต้น จากนั้นจะเป็นชื่ออุปกรณ์ซึ่งอยู่หลังคำสั่ง of เป็นตัวที่บอกว่า architecture นั้นๆใช้บรรยาย entity design unit ใด ส่วนที่อยู่ระหว่างคำสั่ง architecture และ begin เป็นส่วนเผื่อเลือก (Option) ใช้ประกาศกำหนดค่าต่างๆที่จะนำไปใช้ภายใน architecture นั้นๆเช่น สัญญาณ (Signal), ค่าคงที่ (Constant), โปรแกรมย่อย (ฟังก์ชั่นและโปรซีเยอร์) และ อุปกรณ์ (Component) ส่วนที่ใช้บรรยายความสัมพันธ์ระหว่างข้อมูลที่ไหลเข้าและออกของรูปแบบซึ่งถูกกำหนดโดยการใช้คำสั่ง port ในหน่วย entity จะถูกบรรยายในบริเวณเนื้อที่ระหว่างคำสั่ง begin กับ end และที่สำคัญชุดคำสั่งที่ใช้ภายในบริเวณนี้จะเป็นชุดคำสั่งแบบแข่งขนาน (Concurrent Statement) เท่านั้น หน่วย architecture จะถูกปิดท้ายด้วยคำสั่ง end ตามด้วยชื่อของหน่วยซึ่งเป็นส่วนเผื่อเลือก



รูปที่ 4.4 แสดงวงจรมัลติเพล็กขนาด 2:1

```
architecture RTL of MUX is
begin
    OUTPUT <= ((not SEL) and IN0) or (SEL and IN1);
end RTL;
```

รูปที่ 4.5 แสดงหน่วย Architecture ที่ใช้ Model Style แบบ Register Transfer Level (RTL)

รูปที่ 4.5 เป็นตัวอย่างการเขียนรูปแบบ (Model) ของหน่วย Architecture สำหรับวงจรมัลติเพล็กตามรูปที่ 4.4 ส่วนที่บรรยายความสัมพันธ์ระหว่างข้อมูลที่ไหลเข้า (IN0, IN1, SEL) และ

ข้อมูลที่ไหลออก (OUTPUT) ใช้ชุดคำสั่งแบบแข่งขนานเพียงชุดเดียว การบรรยายในลักษณะนี้เรียกว่า Data Flow Description หรือ Register Transfer Level (RTL)

```
architecture STRUCT of MUX is
    component inv
        port (I1 :in bit; O1 :out bit);
    end component;
    component and2
        port (I1,I2 :in bit; O1 :out bit);
    end component;
    component or2
        port (I1,I2 :in bit; O1 :out bit);
    end component;
    signal INT0, INT1, INT2          :bit;
begin
    G0: inv;
    port map (I1 => SEL, O1 => INT0);
    G1: and2;
    port map (I1 => IN0, I2 => INT0, O1 => INT1);
    G2: and2;
    port map (I1 => SEL, I2 => IN1, O1 => INT2);
    G3: or2;
    port map (I1 => INT1, I2 => INT2, O1 => OUTPUT);
end STRUCT;
```

รูปที่ 4.6 แสดงหน่วย Architecture ที่ใช้ Model Style แบบ Structural Description

รูปที่ 4.6 เป็นการเขียนรูปแบบ (Model) ในลักษณะ Structure โดยใช้สัญญาณ (Signal) ทั้งหมด 3 เส้นคือ INT0, INT1, INT2 ในการเชื่อมต่อระหว่างลอจิกเกทกับจุดต่อ (Port)

```
architecture BEHAVE of MUX is
begin
    process (IN0, IN1, SEL)
    begin
        if (SEL = '0') then
            OUTPUT <= IN0;
        else
            OUTPUT <= IN1;
        end if;
    end process;
end BEHAVE;
```

รูปที่ 4.7 แสดงหน่วย Architecture ที่ใช้ Model Style แบบ Behavioral Description

จากรูปที่ 4.7 จะเห็นว่าการบรรยายใช้การแสดงความสัมพันธ์ระหว่างสัญญาณขาเข้ากับขาออกโดยการเขียนเงื่อนไข ในการบรรยายไม่ได้ให้รายละเอียดของโครงสร้างทางฮาร์ดแวร์ การบรรยายรูปแบบในลักษณะนี้เรียกว่า Behavioral Model หรือ Algorithmic Description

4.3.3.3 Package Design Unit

ข้อมูลที่เป็นประโยชน์ในการเขียนรูปแบบรวมไปถึงโปรแกรมย่อยสามารถนำไปเก็บไว้ในส่วนที่เรียกว่า Package ได้ ข้อมูลที่ถูกเก็บไว้นี้สามารถเรียกใช้ได้จากหน่วยอื่นๆเช่น หน่วย Entity, หน่วย Architecture, หรือจากหน่วย Package อื่นๆ โดยใช้คำสั่ง Use ข้อมูลที่สามารถประกาศหรือเก็บไว้ในหน่วย Package ได้แก่

- Subprogram
- Types
- Constant
- Signal
- Aliases
- Attributes
- Component
- Disconnection Specification

หน่วย Package แบ่งออกเป็นสองส่วนคือ Package Declaration และ Package Body

- Package Body

เป็นส่วนที่ใช้บรรยายฟังก์ชั่นการทำงานของโปรแกรมย่อยซึ่งประกอบด้วยชุดคำสั่งลำดับ (Sequential Statement) รวมไปถึงการกำหนดค่าให้ Deferred Constants ที่ถูกประกาศไว้ในส่วนของ Package Declaration

- Package Declaration

เป็นส่วนที่ใช้ประกาศเพื่อทำให้หน่วยอื่นสามารถมองเห็นสิ่งที่เก็บไว้ในหน่วย Package และสามารถนำสิ่งต่างๆเหล่านั้นไปใช้งานได้ ถือได้ว่าเป็นส่วนที่สำคัญที่สุดของหน่วย Package กล่าวคือ รูปแบบที่กำลังเขียนอยู่จะไม่สามารถนำค่าหรือพฤติกรรมที่เก็บไว้ในส่วนของ Package Body ไปใช้งานได้ถ้ามันไม่ถูกประกาศในส่วนประกาศนี้ ซึ่งอาจจะเปรียบได้ว่าเป็นส่วนที่ใช้ในการติดต่อระหว่างหน่วย Package กับโลกภายนอกนั่นเอง

```
-- Package Declaration
package PACK_FUNC is
      function MEAN (A, B, C :real) return real;
end PACK_FUNC;
-- Package Body
package body PACK_FUNC is
      function MEAN (A, B, C :real) return real is
      begin
            return (A+B+C)/3.0;
      end MEAN;
end PACK_FUNC;
```

รูปที่ 4.8 แสดงหน่วย Package ซึ่งภายในประกอบด้วยโปรแกรมย่อยชื่อ Mean

4.3.3.4 Configuration Design Unit

ในหนึ่งระบบเชิงเลขที่ใช้ภาษาวีเอชดีแอลในการออกแบบสามารถมีหน่วย Entity ได้เพียงหนึ่งหน่วยเท่านั้น ในขณะที่ในหนึ่งหน่วยของ Entity สามารถมีหน่วย Architecture ที่เป็นหน่วยรองได้หลายหน่วยแต่สามารถเลือกใช้ได้เพียงหนึ่งรูปแบบเท่านั้น ดังนั้นในการออกแบบจะต้องใช้หน่วย Configuration เพื่อกำหนดว่าหน่วย Entity นั้นๆต้องการใช้หน่วย Architecture หน่วยใด

```
entity and2 is
    generic (ttl_delay      :time:= 3 ns);
    port     (IN1           :in bit;
              IN2           :in bit;
              OUTPUT        :out bit
             );
end and2;
architecture RTL of and2 is
begin
    OUTPUT <= IN1 and IN2 after ttl_delay;
end RTL;
architecture BEHAVE of and2 is
begin
    process (IN1,IN2)
        if (IN1 = '1' and IN2 = '1') then
            OUTPUT <= '1' after ttl_delay;
        else
            OUTPUT <= '0';
        end if;
    end process;
end BEHAVE;
configuration EXAMPLE of and2 is
    for RTL
    end for;
end EXAMPLE;
```

รูปที่ 4.9 แสดงรูปแบบ (Model) ที่มีหน่วย Architecture สองหน่วย

รูปที่ 4.9 เป็นรูปแบบ (Model) ของระบบเชิงเลขอย่างง่ายในการสร้างแอนเกตขนาดสองอินพุท ประกอบด้วยหน่วย Entity หนึ่งหน่วยชื่อ and2 และหน่วย Architecture สองหน่วยซึ่งหมายความว่ารูปแบบนี้มีลักษณะการเขียนรูปแบบ (Model Styles) สองแบบคือ แบบ Register Transfer Level (RTL) และแบบ Behavioral แต่ดังที่ได้กล่าวไปแล้วในข้างต้นว่าหนึ่งระบบจะสามารถเลือกใช้หน่วย Architecture ได้เพียงหนึ่งรูปแบบเท่านั้น ดังตัวอย่างตามรูปที่ 4.9 หน่วย Architecture ชื่อ RTL ได้ถูกกำหนดให้ใช้กับระบบนี้ด้วยคำสั่ง for RTL ซึ่งอยู่ภายในหน่วย Configuration

4.4 ตัวอย่างการออกแบบระบบเชิงเลขด้วยภาษาวีเอชดีแอล

การวิจัยคิดค้นอัลกอริทึมใหม่ๆต้องการระบบการคำนวณที่มีประสิทธิภาพมากขึ้นรวมไปถึงความต้องการเพื่อให้สามารถสร้างได้จริงในทางปฏิบัติ (Realization) เป็นผลทำให้การออกแบบระบบเชิงเลขมีความซับซ้อนมากขึ้น ในอดีตการออกแบบระบบเชิงเลขใช้การเขียนวงจร แต่เมื่อระบบที่ต้องการมีความซับซ้อนมากขึ้นการออกแบบโดยการเขียนวงจรที่ประกอบด้วยลอจิกเกตจำนวน 100,000 เกต คงทำได้ยากมาก การออกแบบด้วยภาษาบรรยายพฤติกรรมของฮาร์ดแวร์จึงได้มีการพัฒนาขึ้น อย่างไรก็ตามถึงแม้ว่าผู้ออกแบบระบบที่ใช้ภาษาบรรยายพฤติกรรมของฮาร์ดแวร์อาจไม่มีความจำเป็นที่จะต้องมีความรู้ในระดับโครงสร้าง (ระดับเกต) แต่ความเข้าใจพื้นฐานในระดับโครงสร้างจะมีส่วนช่วยให้การออกแบบระบบมีประสิทธิภาพมากขึ้น เช่น ใช้ทรัพยากรน้อย, ทำงานเร็ว ฯลฯ เป็นต้น ดังนั้นหัวข้อนี้จะกล่าวถึงตัวอย่างของการออกแบบระบบเชิงเลขพื้นฐานโดยใช้ภาษาวีเอชดีแอลเพื่อเป็นแนวทางสำหรับการศึกษาในระดับสูงต่อไป

4.4.1 การออกแบบวงจรเชิงเลขแบบ Combination

วงจรเชิงเลขแบบ Combination คือวงจรที่ให้ค่าของสัญญาณขาออกขณะนั้นขึ้นอยู่กับค่าของสัญญาณขาเข้าขณะนั้นโดยไม่สนใจสัญญาณขาเข้าสถานะก่อนหน้า [16] ซึ่งแตกต่างกับวงจรเชิงเลขแบบ Sequential ซึ่งจะกล่าวถึงรายละเอียดในหัวข้อต่อไป

4.4.1.1 ตัวถอดรหัสจาก 2 เป็น 4 (2 to 4 Decoder)

ใช้สำหรับการถอดรหัสสัญญาณที่ถูกเข้ารหัสมาในรูปแบบเดียวกันเช่น เลขฐานสองขนาด n บิตจะสามารถแทนค่าได้ทั้งหมดเท่ากับ 2^n ค่าเป็นต้น รูปที่ 4.10 แสดงรูปแบบขอตัวถอดรหัสซึ่งมีสัญญาณขาเข้าชื่อ A ขนาด 2 บิต (กำหนดโดยคำสั่ง 1 downto 0) มี Type เป็น std_ulogic_vector และสัญญาณขาออกชื่อ Z มีขนาด 4 บิต การทำงานใช้การตรวจสอบเงื่อนไขของสัญญาณขาเข้าโดยคำสั่ง when ถ้าสัญญาณขาเข้ามีเงื่อนไขตรงตามที่กำหนดสัญญาณขาออกจะถูกขับตามค่าที่กำหนดไว้ในเครื่องหมายคำพูดที่อยู่ด้านซ้าย แต่ถ้าสัญญาณไม่ตรงกับทั้งสี่เงื่อนไขที่กำหนดสัญญาณขาออกจะถูกขับด้วยค่าที่ไม่ทราบสถานะ (“XXXX”)

4.4.1.2 ตัวมัลติเพล็ก (Multiplexers)

ตัวมัลติเพล็กใช้สำหรับเลือกสัญญาณขาเข้าที่มีอยู่หลายๆสัญญาณเพื่อให้ออกที่สัญญาณขาออกที่มีอยู่เพียงสัญญาณเดียว ตัวอย่างของรูปแบบแสดงดังรูปที่ 4.11 เป็นมัลติเพล็กขนาด 4:1 (4 สัญญาณขาเข้าและ 1 สัญญาณขาออก) สัญญาณขาเข้า A, B, C, D และสัญญาณขาออก Y มี Type เป็น std_ulogic ในขณะที่สัญญาณชื่อ SEL ซึ่งใช้ในการเลือกสัญญาณขาเข้ามี Type เป็น std_ulogic_vector ขนาด 2 บิต การทำงานใช้การตรวจเงื่อนไขของสัญญาณ SEL ซึ่งอยู่ด้านขวาคำสั่ง when สัญญาณขา

ออกจะถูกขับด้วยสัญญาณขาเข้าเมื่อสัญญาณ SEL ตรงกับเงื่อนไขที่อยู่ในบรรทัดเดียวกัน แต่ถ้าสัญญาณไม่ตรงกับทั้งสี่เงื่อนไขที่กำหนดสัญญาณขาออกจะถูกขับด้วยค่าที่ไม่ทราบสถานะ ("X")

```
library ieee;
use ieee.std_logic_1164.all;
entity DECODER is
    port    (A      :in std_ulogic_vector(1 downto 0);
             Z      :out std_ulogic_vector(3 downto 0)
            );
end DECODER;
architecture BEHAVE of DECODER is
begin
    Z <= "0001" when A = "00" else
    Z <= "0010" when A = "01" else
    Z <= "0100" when A = "10" else
    Z <= "1000" when A = "11" else
    "XXXX";
end BEHAVE;
```

รูปที่ 4.10 แสดงรูปแบบของตัวถอดรหัสจาก 2 เป็น 4

4.4.1.3 ตัวบวก (Adder)

ตัวอย่างที่กำลังนำเสนอเป็นตัวบวกขนาด 1 บิต ใช้การเขียนรูปแบบในลักษณะของโครงสร้าง สัญญาณขาเข้าประกอบด้วย A, B, และ CIN มี Type เป็น std_ulogic และสัญญาณขาออกประกอบด้วย SUM และ COUT

4.4.2 การออกแบบวงจรเชิงเลขแบบ Sequential

ในความเป็นจริงระบบเชิงเลขส่วนใหญ่เป็นแบบซีเควนเชียล (Sequential) ซึ่งค่าของสัญญาณขาออกขึ้นอยู่กับค่าของสัญญาณขาเข้าปัจจุบันและค่าของสัญญาณขาเข้าก่อนหน้า ระบบจะถูกเรียกว่า อะซิงโครนัส (Asynchronous) เมื่อสัญญาณขาออกเปลี่ยนค่าทันทีเมื่อสัญญาณขาเข้าเปลี่ยนแปลงสถานะ แต่ถ้าระบบต้องใช้สัญญาณนาฬิกา (Clock) ในการควบคุมจังหวะของการทำงานจะถูกเรียกว่าเป็นระบบซิงโครนัส (Synchronous) อุปกรณ์ที่ใช้สัญญาณนาฬิกาในการควบคุมการทำงานจึงเป็นส่วนประกอบที่สำคัญมาก ดังนั้นก่อนที่จะกล่าวถึงการออกแบบระบบซีเควนเชียลจะแสดงตัวอย่างของการเขียนรูปแบบในการสร้างอุปกรณ์ที่จำเป็นในระบบเสียก่อน

```
library ieee;
use ieee.std_logic_1164.all;
entity MUX is
    port    ( A, B, C, D    :in std_ulogic;
              SEL           :in std_ulogic_vector(1 downto 0);
              Y             :out std_ulogic
            );
end MUX;
architecture BEHAVE of MUX is
begin
    with SEL select
    Y <= A when "00",
         B when "01",
         C when "10",
         D when "11",
         'X' when others;
end BEHAVE;
```

รูปที่ 4.11 แสดงรูปแบบของตัวมัลติเพล็กขนาด 4:1

4.4.2.1 เอส-อาร์ แลทช์ (SR Latch)

อุปกรณ์สองประเภทที่มีการทำงานคล้ายกันมากได้แก่ แลทช์ (Latch) และ ฟลิปฟล็อป แต่ในที่นี้จะกำหนดนิยามของแลทช์ว่าเป็นหน่วยความจำที่ทำงานโดยใช้ระดับของสัญญาณ (Level Sensitive) ส่วนฟลิปฟล็อปคือหน่วยความจำที่ทำงานโดยใช้ขอบของสัญญาณ (Edge Sensitive) รูปที่ 4.13 แสดงรูปแบบของเอสอาร์แลชท์

4.4.2.2 ดี-ฟลิปฟล็อป (D-Flip-Flop)

การทำงานของดีฟลิปฟล็อปคือ สัญญาณขาเข้า (D) จะถูกส่งออกไปที่สัญญาณขาออก (Q) เมื่อสัญญาณนาฬิกาเปลี่ยนสถานะหรืออาจกล่าวได้ว่ามันทำงานโดยใช้ขอบของสัญญาณ ตามรูปแบบแสดงดังรูปที่ 4.14 ใช้ขอบขาขึ้นในการกระตุ้นการทำงานโดยใช้ฟังก์ชั่น rising_edge

```
library ieee;
use ieee.std_logic_1164.all;
entity FULLADDER is
    port    (A, B, CIN      :in std_ulogic;
             SUM, COUT  :out std_ulogic
            );
end FULLADDER;
architecture STRUCT of FULLADDER is
begin
    SUM <= A xor B xor CIN;
    COUT <= (A and B) or (A and CIN) or (b and CIN);
end STRUCT;
```

รูปที่ 4.12 แสดงรูปแบบของตัวบวกขนาด 1 บิต

```
library ieee;
use ieee.std_logic_1164.all;
entity SR_LATCH is
    port    (S, R           :in std_ulogic;
             Q, QBAR        :buffer std_ulogic
            );
end SR_LATCH;
architecture RTL of SR_LATCH is
begin
    Q <= '1' when R = '0' else
         '0' when S = '0' else
         Q;
    QBAR <= '1' when S = '0' else
            '0' when R = '0' else
            QBAR;
end RTL;
```

รูปที่ 4.13 แสดงรูปแบบของเอสอาร์แลชท์

```
library ieee;
use ieee.std_logic_1164.all;
entity DFF is
    port    (D, CLOCK, RESET    :in bit;
             Q                  :out bit
            );
end DFF;
architecture RTL of DFF is
begin
P0:
process (D, CLOCK, RESET)
begin
    if (RESET = '0') then
            Q <= '0';
    elsif rising_edge (CLOCK) then
            Q <= D;
    end if;
end process
P0;
end RTL;
```

รูปที่ 4.14 แสดงรูปแบบของดีฟลิปฟล็อป

4.4.2.3 หน่วยความจำชนิดรอม (Read Only Memory: ROM)

รูปแบบของรอมสามารถสร้างได้จากอาร์เรย์ของค่าคงที่ ตัวอย่างแสดงดังรูปที่ 4.15 เป็นรอมที่มีขนาดของคำ (Word Size) เท่ากับ 8 บิตและมีความจุ 4 ไบท์ (Capacity) กำหนดโดยค่าคงที่ชื่อ ROM ซึ่งมี Type เป็น ROM_ARRAY ในการอ่านค่าของแต่ละตำแหน่งภายในทำได้โดยการส่งตำแหน่งเข้าไปที่ ADDRESS ซึ่งมี Type เป็น integer มีความกว้างเท่ากับ 2 บิต ข้อมูลจะถูกขับออกที่สัญญาณขาออก DATA ซึ่งมีขนาดเท่ากับขนาดของคำ

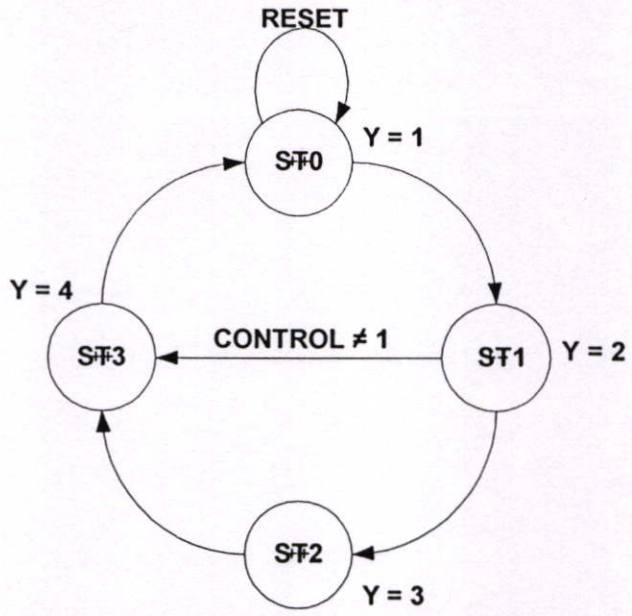
4.4.2.4 ไฟไนท์สเตทแมชีนส์ (Finite State Machines: FSM)

เป็นระบบเชิงเลขซึ่งมีการทำงานเป็นลำดับที่เฉพาะเจาะจงหรือมีจำนวนสถานะ (State) ที่จำกัด ไฟไนท์สเตทแมชีนส์ถูกนำมาประยุกต์ใช้งานอย่างกว้างขวางโดยเฉพาะกับตัวควบคุมที่ทำหน้าที่ควบคุมการทำงานของระบบเชิงเลขชนิดอื่นๆ [17] ยกตัวอย่างเช่น SDRAM Controller, IIC Bus Controller ฯลฯ ตัวอย่างรูปแบบของไฟไนท์สเตทแมชีนส์อย่างง่ายแสดงดังรูปที่ 4.17 ซึ่งบรรยาย

พฤติกรรมของ สเตทไดอะแกรมตามรูปที่ 4.16 ระบบประกอบด้วยสัญญาณขาเข้าที่เป็นสัญญาณควบคุมทั้งหมด 3 สัญญาณได้แก่ CLOCK, RESET, CONTROL และสัญญาณขาออกของระบบคือ Y ซึ่งเป็นสัญญาณขาออกชนิด Moore การทำงานของระบบมีดังนี้ ในกรณีที่สัญญาณ RESET มีค่าเป็น "1" สัญญาณขาออกจะมีค่าเท่ากับ 1 (ST0) แต่ในกรณีที่สัญญาณ RESET มีค่าเป็นค่าอื่นๆ สถานะ (State) จะถูกเปลี่ยนในช่วงขอบขาขึ้นของสัญญาณ CLOCK วนจาก ST0, ST1, ST2, ST3 และกลับมาที่ ST0 พิเศษที่ ST1 ระบบจะมีการตรวจสัญญาณ CONTROL ว่าเป็น "1" หรือไม่ ถ้าเป็น "1" สถานะจะเปลี่ยนไปเป็น ST2 แต่ถ้าเป็นค่าอื่นสถานะจะกระโดดไปที่ ST3 (โดยไม่ผ่าน ST2) รูปแบบแสดงดังรูปที่ 4.17 ถูกแบ่งออกเป็น 3 ส่วนได้แก่ ส่วนของ Next State, Current State, ทั้งสองส่วนนี้ทำงานแบบแข่งขนานเนื่องจากใช้ชุดคำสั่ง process และซิงโครนัส ส่วนที่สามเป็นส่วนของสัญญาณขาออก (Y) ซึ่งเป็นชนิด Combination

```
library ieee;
use ieee.std_logic_1164.all;
entity ROM_4x8 is
    port   (ADDRESS   :in integer range 0 to 3;
            DATA      :out std_ulogic_vector (7 downto 0)
           );
end ROM_4x8;
architecture RTL of ROM_4x8 is
    type ROM_ARRAY is array (0 to 3)
    of std_ulogic_vector (7 downto 0);
    constant ROM :ROM_ARRAY :=  (
                                "11001100",
                                "00110011",
                                "11110000",
                                "00001111",
                                );
begin
    DATA <= ROM (ADDRESS);
end RTL;
```

รูปที่ 4.15 แสดงรูปแบบของหน่วยความจำรอม



รูปที่ 4.16 แสดงสเตทไดอะแกรมของไฟไนท์สเตทแมชีนส์

```
library ieee;
use ieee.std_logic_1164.all;
entity FSM is
        port    ( CLOCK, RESET, CONTROL      :in std_logic;
                  Y                          :out integer range 0 to 4
                );
end FSM;
architecture RTL of FSM is
        type STATE_TYPE is (ST0, ST1, ST2, ST3);
        signal CURRENT_STATE, NEXT_STATE  :STATE_TYPE;
begin
```

รูปที่ 4.17ก แสดงรูปแบบของไฟไนท์สเตทแมชีนส์

```
--Next State Logic
COMB:process(CONTROL, CURRENT_STATE)
begin
        case CURRENT_STATE is
                when ST0 =>
                        NEXT_STATE <= ST1;
                when ST1 =>
                        if (CONTROL = '1') then
                                NEXT_STATE <= ST2;
                        else
                                NEXT_STATE <= ST3;
                        end if;
                when ST2 =>
                        NEXT_STATE <= ST3;
                when ST3 =>
                        NEXT_STATE <= ST0;
                when others =>
                        NEXT_STATE <= ST0;
        end case;
end process COMB;
```

รูปที่ 4.17ข แสดงรูปแบบของไฟไนท์สเตทแมชีนส์ (ต่อ)

```
--Current State Logic
SEQ:process(CLOCK, RESET)
begin
        if (RESET = '1') then
                CURRENT_STATE <= ST0;
        elsif rising_edge (CLOCK) then
                CURRENT_STATE <= NEXT_STATE;
        end if;
end process SEQ;
--Output Logic
with CURRENT_STATE select
        Y <= 1 when ST0,
                2 when ST1,
                3 when ST2,
                4 when ST3,
                1 when others;
end RTL;
```

รูปที่ 4.17ค แสดงรูปแบบของไฟไนท์สเตทแมชีนส์ (ต่อ)

บทที่ 5

ลำดับขั้นการดำเนินงาน

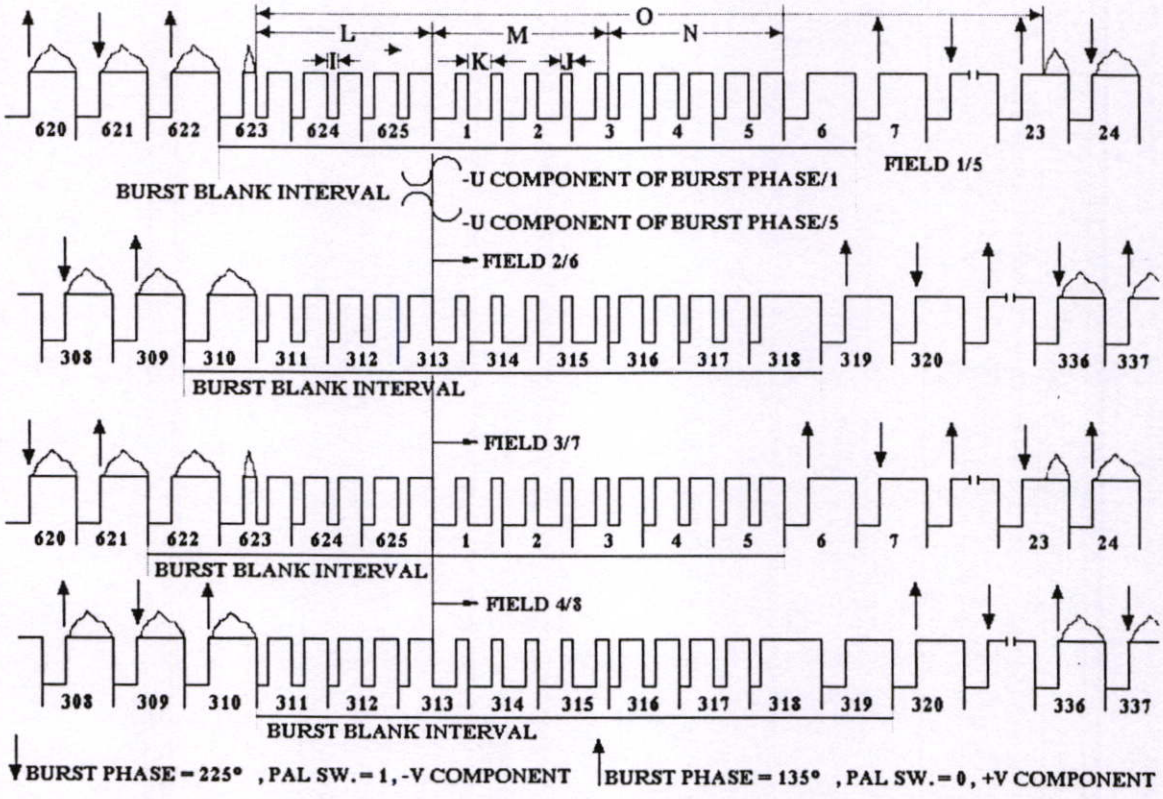
5.1 บทนำ

งานวิจัยที่กำลังนำเสนอเป็นการสร้างระบบเชิงเลขจากสัญญาณและระบบต้นแบบซึ่งเป็นระบบเชิงอุปมาน สิ่งที่สำคัญที่สุดในการออกแบบคือสัญญาณที่ได้จะต้องมีคุณสมบัติเป็นไปตามข้อกำหนดทั้งขนาด, เฟส, และความถี่หรือไทม์มิ่ง ในการออกแบบได้ใช้หลักการของการออกแบบจากบนลงล่างโดยเริ่มจากการศึกษาวิธีการกวาดเส้นภาพของโทรทัศน์ ส่วนประกอบและหน้าที่ของแต่ละส่วนประกอบของสัญญาณภาพสีเบ็ดเสร็จซึ่งในส่วนนี้ถือว่าเป็นส่วนที่สำคัญมากเนื่องจากเป็นจุดเริ่มต้นของการทำความเข้าใจสัญญาณและระบบที่จะใช้เป็นต้นแบบในการออกแบบเพื่อแปลงไปเป็นระบบเชิงเลขตามที่ต้องการ หลังจากทำความเข้าใจกับสัญญาณและระบบต้นแบบแล้วขั้นต่อไปคือการออกแบบระบบเชิงเลขโดยรวมในระดับของฟังก์ชั่นจากคุณสมบัติของสัญญาณภาพสีเบ็ดเสร็จและระบบเชิงอุปมานต้นแบบ ในขั้นตอนนี้จะทำการแบ่งระบบออกเป็นส่วนการทำงานย่อยซึ่งการแบ่งจะพิจารณาจากส่วนประกอบของสัญญาณที่ต้องการสร้าง ฟังก์ชั่นการทำงานและความเหมาะสมในมุมมองของการเชื่อมต่อเนื่องจากเป็นระบบเชิงเลขดังนั้นจึงไม่สามารถสร้างระบบให้มีความเหมือนกับระบบเชิงอุปมานได้ทุกส่วน ขั้นตอนต่อไปคือการออกแบบส่วนการทำงานย่อยที่ละส่วนซึ่งบางส่วนการทำงานที่ได้ออกแบบไปแล้วอาจต้องมีการเปลี่ยนแปลงตามความเหมาะสมเพื่อให้ได้จุดที่ดีที่สุดของการออกแบบ ขั้นตอนสุดท้ายของการออกแบบก็คือ การสร้างระบบฮาร์ดแวร์ในการทดสอบคุณสมบัติของสัญญาณโดยใช้เครื่องมือวัดทางอิเล็กทรอนิกส์เปรียบเทียบค่าที่วัดได้จริงจากระบบกับค่าจริงที่ควรจะเป็นตามทฤษฎี

5.2 ศึกษาคุณสมบัติของสัญญาณและการทำงานของระบบ

จากการศึกษาส่วนประกอบของสัญญาณแสดงดังรูปที่ 5.1-5.3 ทำให้ทราบว่าสัญญาณภาพสีเบ็ดเสร็จเกิดจากการรวมกันของสองสัญญาณคือ สัญญาณภาพเบ็ดเสร็จ (Composite Video Signal: CVS) และสัญญาณสี (Chrominance: C) ทั้งสองสัญญาณนี้ยังมีสัญญาณซึ่งเป็นส่วนประกอบย่อยแสดง

ดังรูปที่ 5.4 สัญญาณดังกล่าวจะต้องมีคุณสมบัติตามข้อกำหนดของซีซีไออาร์ ซึ่งแบ่งออกเป็น 3 กลุ่ม ตามหน่วยการวัดทางไฟฟ้าคือ ขนาด, เฟส, และความถี่หรือไทม์มิ่ง



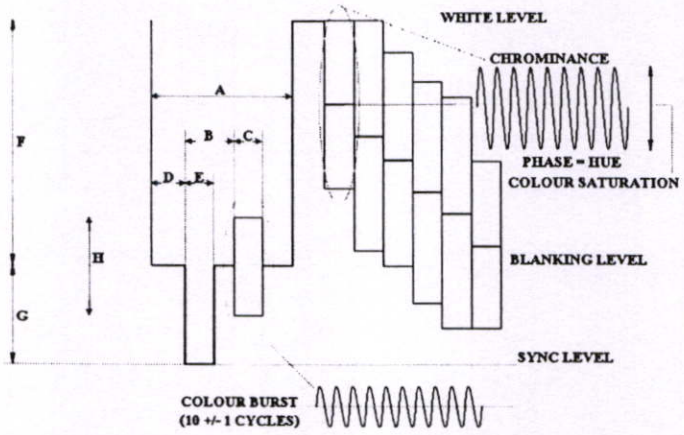
รูปที่ 5.1 แสดงส่วนประกอบของสัญญาณภาพสีเบ็ดเสร็จระบบพาลซึ่งมีคาบซ้ำทุกๆ 8 ฟิลด์

จากรูปที่ 5.1-5.3 ทำให้ทราบข้อมูลเกี่ยวกับสัญญาณภาพสีเบ็ดเสร็จว่ามีคุณสมบัติดังนี้

- สัญญาณมีคาบซ้ำทุกๆ 8 ฟิลด์ โดยเทียบกับเฟสสัญญาณพาหะรองของสัญญาณสี U
- สัญญาณในช่วง Vertical Blanking ระหว่างฟิลด์คู่และฟิลด์คี่มีความแตกต่างกัน ข้อมูลที่จุดนี้ใช้ในการสร้างสัญญาณคาบซ้ำของ Vertical Blanking เพื่อการประหยัดลอจิกเกท
- สัญญาณที่มีความละเอียดมากที่สุดอยู่ในช่วงของสัญญาณ Vertical Sync ข้อมูลที่จุดนี้เป็นประโยชน์ต่อการเลือกความถี่สัญญาณนาฬิกาของระบบ
- สัญญาณพาหะรองของสัญญาณสี U มีเฟสคงที่ตลอดแต่สัญญาณพาหะรองของสัญญาณสี V มีการกลับเฟส 180° เส้นเว้นเส้นและสัมพันธ์กับสัญญาณเบิรสต์ที่มีการสอดแทรกสลับเฟสเส้นเว้นเส้น 225° และ 135° เทียบกับเฟสสัญญาณพาหะรองของสัญญาณสี U

- การสอดแทรกสัญญาณเบิรสต์ในช่วง Vertical Blanking ระหว่างฟิลด์เส้นคู่และฟิลด์เส้นคี่มีตำแหน่งเส้นที่แตกต่างกันและเกิดคาบซ้ำทุกๆ 4 ฟิลด์ (Bruch Sequence)

คุณสมบัติอื่นๆนอกเหนือจากที่ได้กล่าวถึงใช้ข้อมูลอ้างอิงจาก [7], [18]



รูปที่ 5.2 แสดงส่วนประกอบของสัญญาณภาพสีเบ็ดเสร็จในช่วงเวลาการกวาดเส้นภาพทางแนวนอน

จากรูปที่ 5.2 A-H แสดงขนาดและช่วงเวลาของสัญญาณภาพสีเบ็ดเสร็จในช่วงเวลาการกวาดเส้นภาพทางแนวนอนซึ่งมีความหมายดังนี้

A = Line-blanking interval

B = Start of sub-carrier burst

C = Duration of sub-carrier burst

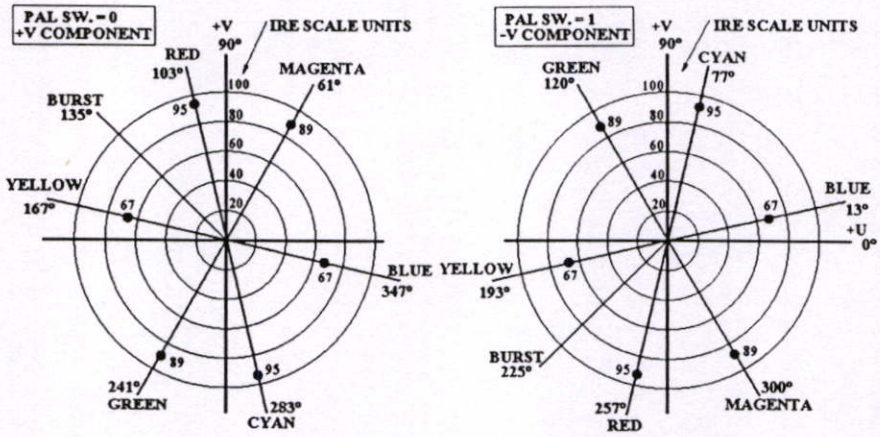
D = Front porch

E = Synchronizing pulse

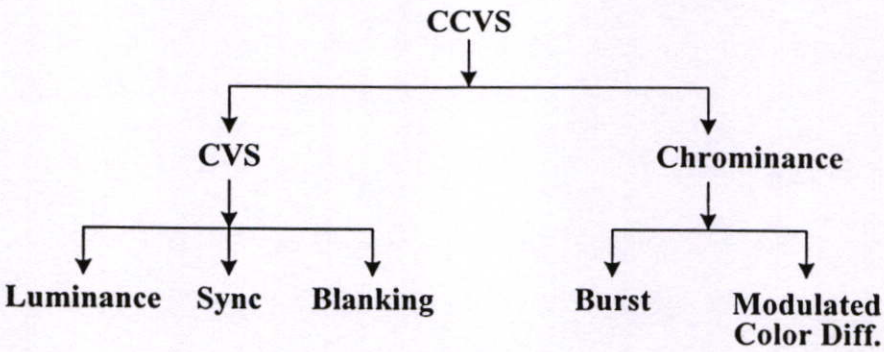
F = Peak white level

G = Synchronizing level

H = Peak-to-Peak value of burst



รูปที่ 5.3 แสดงเวคเตอร์สัญญาณความแตกต่างสีของสัญญาณแท่งสี 75 %



รูปที่ 5.4 แสดงการแบ่งกลุ่มสัญญาณที่ใช้ในการสร้างสัญญาณภาพสีเบ็ดเสร็จ

5.3 กำหนดคุณสมบัติเริ่มต้นของการออกแบบ

สิ่งที่ต้องการจะได้จากระบบที่กำลังออกแบบคือ สัญญาณ คุณสมบัติต่างๆของสัญญาณที่ต้องการจะเป็นตัวกำหนดว่าจะต้องออกแบบระบบอย่างไรจึงจะสามารถสร้างสัญญาณได้จริง ดังนั้นในการออกแบบจึงต้องกำหนดคุณสมบัติของสัญญาณเชิงอุปมานที่ต้องการจากระบบก่อนเป็นอันดับแรก หลังจากนั้นจะนำข้อมูลความต้องการที่ได้เป็นตัวกำหนดคุณสมบัติของระบบเชิงเลขที่จะต้องออกแบบ

5.3.1 คุณสมบัติของสัญญาณเชิงอุปมานที่ต้องการจากระบบ

- ระบบการเข้ารหัสสัญญาณสีแบบ PAL B/G
- ระบบการกวาดเส้นภาพแบบสอดแทรก (Interlace Scanning)
- สัญญาณขาออกแบบ สัญญาณภาพสีเบ็ดเสร็จ (Composite Color Video Signal: CCVS)
- สัญญาณขาออกแบบ ซุปเปอร์วีดีโอ (Supper Video: S-Video)

- ความละเอียด 768x576 Pixels

5.3.2 คุณสมบัติของระบบเชิงเลขที่ใช้ในการสร้างสัญญาณภาพสีเบ็ดเสร็จ

- ระบบตัวเลขและการประมวลผลแบบจำนวนโดยตรง (Fixed Point Arithmetic) เหตุผลของการเลือกใช้การประมวลผลแบบนี้เพราะต้องการลดความซับซ้อนของการออกแบบระบบการประมวลผล จากหัวข้อที่ 3.3.4.2 การประมวลผลแบบจำนวนอิงดัชนี (Floating Point) ในการออกแบบระบบจะต้องพิจารณาสองส่วนคือ ส่วนของค่าแมนทีสซา (Mantissa) และ ส่วนชี้กำลัง (Exponent) ซึ่งมีความซับซ้อนและต้องการทรัพยากรมากกว่าการประมวลผลแบบจำนวนโดยตรงอย่างมาก แต่อย่างไรก็ตามในการเลือกระบบการประมวลผลจะต้องพิจารณาถึงค่าความคลาดเคลื่อนของสัญญาณที่จะได้รับจากระบบเป็นสิ่งสำคัญ สำหรับงานวิจัยชิ้นนี้การใช้ระบบตัวเลขและการประมวลผลแบบจำนวนโดยตรงให้ผลของความคลาดเคลื่อนอยู่ในเกณฑ์ที่ยอมรับได้จึงไม่มีความจำเป็นที่จะต้องใช้ระบบตัวเลขและการประมวลผลแบบจำนวนอิงดัชนี

- สัญญาณขาเข้าเชิงเลขของสัญญาณสีแดง เขียวและน้ำเงิน มีความกว้างขนาด 8 บิต ไม่มีเครื่องหมาย (Unsigned) จำนวนบิตที่ใช้เลือกตามมาตรฐานการเข้ารหัสสัญญาณภาพโดยทั่วไปเช่น ระบบ MPEG-2 (YCbCr 4:2:2 บางระบบใช้ 10 บิต แต่ปัจจุบันส่วนใหญ่ใช้ 8 บิต) เพื่อเป็นมาตรฐานเดียวกันและง่ายสำหรับการนำไปเชื่อมต่อกับระบบประมวลผลสัญญาณภาพ (Video Processing) ซึ่งโดยส่วนใหญ่สัญญาณขาออกมีความกว้าง 8 บิต

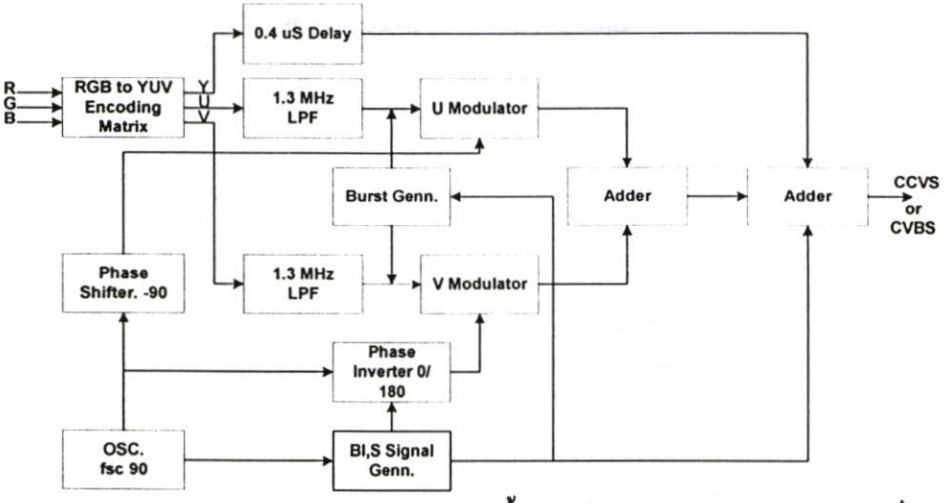
- สัญญาณขาออกเชิงเลขของสัญญาณภาพเบ็ดเสร็จ มีความกว้างขนาด 12 บิต มีเครื่องหมาย (Signed) มี 2 เหตุผลในการเลือกคือ ประการแรกระบบที่กำลังออกแบบต้องการให้มีความสามารถในการปรับแต่งเพื่อการสอบเทียบ (Calibration) โดยการปรับแต่งจะเป็นระบบเชิงเลขทั้งหมด จำนวนบิตพิจารณาจากช่วงของสัญญาณที่ยอมรับค่าความคลาดเคลื่อนของสัญญาณขาออกที่น้อยที่สุดซึ่งก็คือสัญญาณในช่วงซิงโครไนเซชั่นที่ยอมรับค่าความคลาดเคลื่อนได้อยู่ที่ $\pm 10mV$. ในขณะที่สัญญาณขาออกที่ต้องการจากระบบคือ $1.V_{P-P}$ ดังนั้น 1 LSB (Least Significant Bit) จะต้องเปลี่ยนแปลงค่าแรงดันน้อยกว่า $10mV$. การออกแบบในครั้งนี้ได้เลือกใช้ตัวแปลงสัญญาณเชิงเลขเป็นสัญญาณเชิงอุปมาน (DAC) ขนาด 12 บิต และสร้างวงจรเป็นแบบไบโพล่า (Bi-Polar) คือสามารถให้สัญญาณขาออกในช่วง $\pm 1.V_{P-P}$ (โดยการใช้ระบบตัวเลขแบบ Offset Two's Complement) ดังนั้นที่ 1 LSB จะให้ค่าแรงดันเปลี่ยนแปลงเท่ากับ 0.48828125 mV. $(2/2^{12})$ ซึ่งเพียงพอสำหรับการปรับละเอียดเพื่อการสอบเทียบ ประการที่สองพิจารณาจากความเป็นไปได้ที่จะสามารถหาซื้อไอซีที่ถูกเลือกมาจากเหตุผลข้อแรก นอกจากนี้การออกแบบได้เผื่อความละเอียดของจำนวนบิตไว้สำหรับอนาคตซึ่งมี

แนวโน้มว่าสัญญาณภาพจะมีความละเอียดที่สูงขึ้นตามขนาดของจอภาพแสดงผล (จากข้อมูลของการออกแบบของบริษัท Thomson Television Thailand ปี 2548)

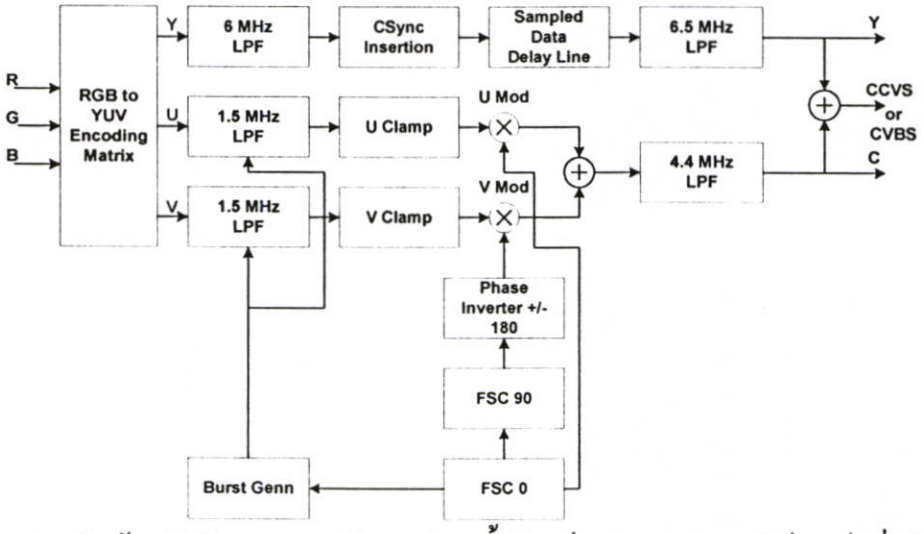
- สัญญาณขาออกเชิงเลขของสัญญาณสี มีความกว้างขนาด 12 บิต มีเครื่องหมาย (Signed) การเลือกจำนวนบิตพิจารณาจากตัวแปลงสัญญาณเชิงเลขเป็นสัญญาณเชิงอุปมาน (DAC) เหมือนกันกับส่วนของสัญญาณภาพเบ็ดเสร็จ และออกแบบเผื่อเช่นกัน (ใช้จำนวนบิตมากกว่าความจำเป็น)
- ความถี่สุ่มเท่ากับ 8 เท่าของสัญญาณพาหะรองของสัญญาณสี (4.43361875 MHz. x 8) มีเหตุผลสำคัญสองประการในการเลือกใช้ความถี่สุ่มค่านี้ เหตุผลประการแรกเพื่อให้ง่ายต่อการสร้างสัญญาณพาหะรองของสัญญาณสีและสัญญาณเบิรสต์ เนื่องจากตัวชี้เฟสสามารถสร้างจากวงจรนับเลขฐานสองแบบโมดูลัส (2^N) ขนาด 3 บิต ทำให้ง่ายต่อการออกแบบและประหยัดทรัพยากร ประการที่สองต้องการทำการสุ่มเกิน (Over Sampling) เพื่อลดความยุ่งยากของการออกแบบตัวกรองสร้างสัญญาณคืน (Reconstruction Filter) ซึ่งเป็นหลักการของการประมวลผลสัญญาณเชิงเลขหัวข้อที่สำคัญอีกหัวข้อหนึ่ง ซึ่งในความเป็นจริงบางระบบใช้การสุ่มเกินเพียง 4 เท่าของสัญญาณเบิรสต์แต่การควบคุมเฟสของสัญญาณเชิงอุปมานขาออกทำได้ยากกว่า

5.4 การออกแบบระบบโดยรวม

หลังจากที่ได้ศึกษาส่วนประกอบของสัญญาณที่ใช้ในการสร้างสัญญาณภาพสีเบ็ดเสร็จเสร็จและได้กำหนดคุณสมบัติที่ต้องการจากระบบแล้ว ขั้นตอนต่อไปคือการศึกษาวงจรเชิงอุปมานต้นแบบว่าใช้วงจรประเภทไหนในการสร้างสัญญาณแต่ละส่วน สำหรับงานวิจัยชิ้นนี้ใช้ระบบเชิงอุปมานต้นแบบสองระบบคือ ระบบแรกได้จากหนังสือของบริษัท ROHDE & SCHWARZ [5] ซึ่งเป็นบริษัทผลิตเครื่องมือวัดทางอิเล็กทรอนิกส์จากประเทศเยอรมัน ระบบที่สองได้จากสถาปัตยกรรมของไอซีเบอร์ AD722 ของบริษัท Analog Device Inc [19] ซึ่งเป็นบริษัทผู้ผลิตไอซีจากประเทศสหรัฐอเมริกา ทั้งสองระบบดังกล่าวแสดงดังรูปที่ 5.5 - 5.6

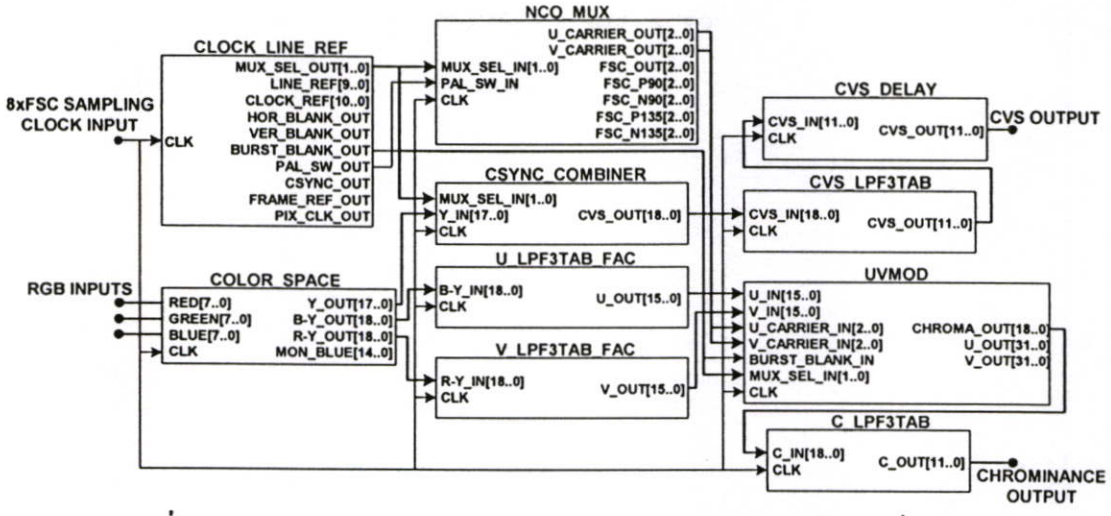


รูปที่ 5.5 แสดงตัวเข้ารหัสสัญญาณภาพสีแดง เขียว น้ำเงิน เป็นสัญญาณภาพสีเบ็ดเสร็จที่ได้จาก [5]



รูปที่ 5.6 แสดงตัวเข้ารหัสสัญญาณภาพสีแดง เขียว น้ำเงิน เป็นสัญญาณภาพสีเบ็ดเสร็จที่ได้จาก [19]

ระบบต้นแบบทั้งสองให้สัญญาณขาออกที่เหมือนกันถึงแม้ว่าจะมีสถาปัตยกรรมที่แตกต่าง ข้อมูลที่ได้จากการศึกษาคุณสมบัติของสัญญาณภาพสีเบ็ดเสร็จบวกกับรายละเอียดของระบบต้นแบบทั้งสองระบบนี้สามารถแปลงระบบให้อยู่ในระบบเชิงเลขได้แสดงดังรูปที่ 5.7 ซึ่งนอกเหนือจากกลุ่มสัญญาณที่ใช้ในการสร้างสัญญาณภาพสีเบ็ดเสร็จที่แสดงดังรูปที่ 5.4 ยังมีสัญญาณควบคุมอื่นๆที่จำเป็นต่อการทำงานของระบบ ยกตัวอย่างเช่น สัญญาณชี้เฟส (Phase Pointer) สำหรับการผลิตสัญญาณเบิรสต์ ฯลฯ เป็นต้น



รูปที่ 5.7 แสดงโครงสร้างระดับบนสุด (Top Level) ของตัวเข้ารหัสฯที่นำเสนอ

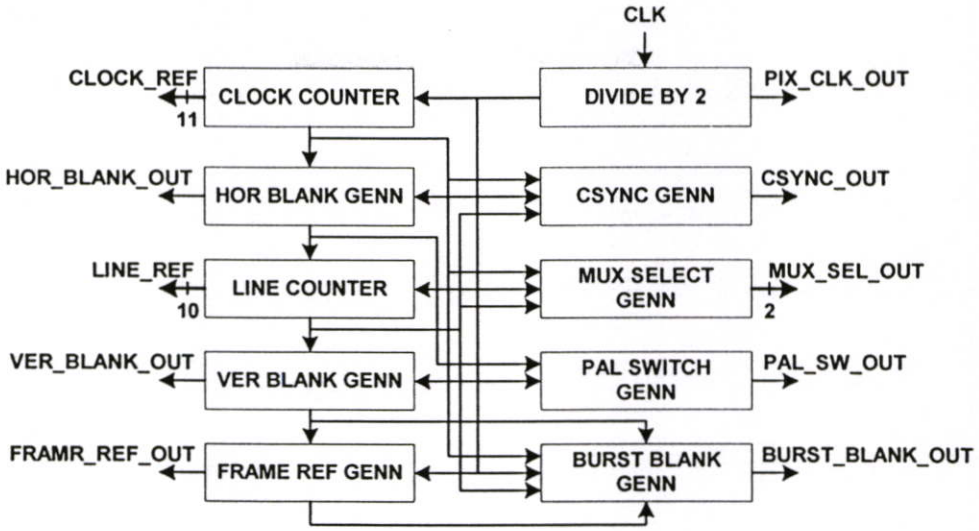
5.5 การออกแบบส่วนการทำงานย่อยของระบบ

หลังจากได้ออกแบบระบบโดยรวมซึ่งแสดงดังรูปที่ 5.7 ขั้นตอนต่อไปคือการเขียนรูปแบบ (Model) ด้วยภาษาวีเอชดีแอลโดยแบ่งการออกแบบออกเป็นส่วนการทำงานย่อยและแต่ละส่วนจะถูกออกแบบแยกอิสระต่อกัน ตามโครงสร้างของภาษาวีเอชดีแอลหนึ่งส่วนการทำงานย่อยคือหนึ่งหน่วย Entity ในขั้นตอนของการออกแบบ ทุกส่วนการทำงานจะต้องผ่านการจำลองการทำงานในระดับของฟังก์ชั่นและไทม์มิ่งเพื่อให้แน่ใจได้ว่าระบบทำงานถูกต้องตามวัตถุประสงค์ จากนั้นจึงจะนำแต่ละส่วนมาเชื่อมต่อกัน การแบ่งระบบออกเป็นส่วนการทำงานย่อยดังกล่าวมีข้อดีหลายประการเช่น การออกแบบ, แก้ไขและจำลองการทำงานสามารถทำแยกเฉพาะที่ละส่วนได้จึงง่ายต่อการตรวจสอบหรือแก้ไข

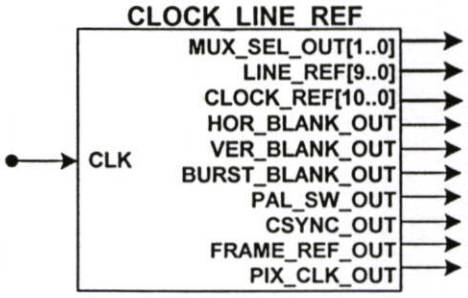
ระบบนี้ทำงานแบบซิงโครนัส (Synchronous) โดยใช้สัญญาณนาฬิกา (Clock: CLK) ที่มีความถี่เท่ากับ 8 เท่าของสัญญาณพาหะรองของสัญญาณสี (4.43361875 MHz.*8) สัญญาณนี้จะถูกใช้เป็นสัญญาณอ้างอิงของทุกส่วนการทำงาน การเลือกความถี่ค่านี้มีเหตุผลสองประการคือ ประการแรกเพื่อง่ายต่อการสร้างสัญญาณเบิรสต์ ประการที่สองคือต้องการลดความยุ่งยากของการออกแบบตัวกรองสร้างสัญญาณคืน (Reconstruction Filter)

5.5.1 ภาค CLOCK_LINE_REF

ทำหน้าที่สร้างสัญญาณอ้างอิงที่มีคาบเวลาเหมาะสมสำหรับการสร้างสัญญาณภาพสีเบ็ดเสร็จตามมาตรฐานของระบบ PAL และสร้างสัญญาณควบคุมให้กับส่วนการทำงานย่อยอื่นๆ รูปที่ 5.8 แสดงโครงสร้างในระดับ RTL และมุมมองของการเชื่อมต่อแสดงดังรูปที่ 5.9



รูปที่ 5.8 แสดงโครงสร้างในระดับ RTL ของภาค CLOCK_LINE_REF



รูปที่ 5.9 แสดงมุมมองการเชื่อมต่อของภาค CLOCK_LINE_REF

แต่ละบล็อกตามรูปที่ 5.8 ทำงานในลักษณะแข่งขนาน (Concurrent) รหัสต้นฉบับ (Source Code) ของภาคนี้ชื่อ CLOCK_LINE_REF.VHD หน้าที่ของแต่ละสัญญาณมีดังนี้

- CLOCK_REF เป็นสัญญาณขาออกขนาด 11 บิต เป็นสัญญาณอ้างอิงภายใช้ในการนับสัญญาณนาฬิกาเพื่อสร้างสัญญาณคาบซ้ำซึ่งมีเวลาเท่ากับ 1 เส้นกวาดทางแนวนอน ค่าของการนับเท่ากับ 1135 ซึ่งสามารถคำนวณได้จากสมการที่ (5.1)

$$CountingValue = \frac{64uS.}{(4.43361875 * 10^{6} * 8/2)^{-1}} \qquad (5.1)$$

- HOR_BLANK_OUT สัญญาณขาออกขนาด 1 บิต เป็นสัญญาณ Horizontal Blanking ตามมาตรฐานของระบบพาล เป็นสัญญาณคาบซ้ำมีค่าเป็น "1" หรือช่วงที่มีสัญญาณภาพเท่ากับ 52 uS. และ "0" หรือช่วงแบล้งกิ้งเท่ากับ 12 uS. หรือมีความถี่เท่ากับ 15,625 Hz.

- LINE_REF สัญญาณขาออกขนาด 10 บิต เป็นสัญญาณอ้างอิงภายใช้ในการนับเพื่อบอกตำแหน่งของเส้นภาพ ค่าของการนับจะเริ่มจาก 0 และสิ้นสุดที่ 624 ตามจำนวนเส้นภาพในหนึ่งเฟรมของระบบพาลคือ 625 เส้น การนำจะถูกกระตุ้นโดยขอบขาลงของสัญญาณ HOR_BLANK_OUT

- VER_BLANK_OUT สัญญาณขาออกขนาด 1 บิต เป็นสัญญาณ Vertical Blanking ตามมาตรฐานของระบบพาล เป็นสัญญาณคาบซ้ำมีค่าเป็น "1" หรือช่วงที่มีสัญญาณภาพจำนวนทั้งหมด 576 เส้นและเป็น "0" หรือช่วงแบล้งกิ้งจำนวนทั้งหมด 49 เส้น ต่อหนึ่งเฟรม สัญญาณนี้มีความถี่เท่ากับ 50 Hz.

- FRAME_REF_OUT สัญญาณขาออกขนาด 1 บิต เป็นสัญญาณใช้เพื่อบอกการสิ้นสุดการกวาดเส้นภาพทางแนวตั้งหนึ่งเฟรม สัญญาณจะถูกกลับเฟสทุกๆเส้นที่ 625 ของแต่ละเฟรม สัญญาณนี้ใช้ในการควบคุมการสอดแทรกสัญญาณเบิรสต์

- BURST_BLANK_OUT สัญญาณขาออกขนาด 1 บิต ใช้ควบคุมการสอดแทรกสัญญาณเบิรสต์ในช่วงของ Vertical Blanking เป็นสัญญาณคาบซ้ำ จากที่ได้กล่าวไปแล้วในข้างต้นว่าการกวาดเส้นภาพมีคาบซ้ำทุกๆ 8 ฟิลด์ การสอดแทรกสัญญาณเบิรสต์จะมีตำแหน่งเส้นตรงกันระหว่าง ฟิลด์ 1 กับ ฟิลด์ 5, ฟิลด์ 2 กับ ฟิลด์ 6, ฟิลด์ 3 กับ ฟิลด์ 7, และฟิลด์ 4 กับ ฟิลด์ 8

- PAL_SW_OUT สัญญาณขาออกขนาด 1 บิต เป็นสัญญาณที่ใช้ควบคุมเฟสของสัญญาณเบิรสต์และสัญญาณพาหะรองของสัญญาณสี V เป็นสัญญาณคาบซ้ำมีความถี่เท่ากับ 7812.5 Hz. เนื่องจากเฟสของสัญญาณทั้งสองนี้จะถูกกลับเฟสเส้นเว้นเส้น ซึ่งมีความสัมพันธ์กันกล่าวคือ เส้นที่ไม่มีการกลับเฟสสัญญาณพาหะรองของสัญญาณสี V เบิรสต์จะมีค่าเฟส 135 องศาและสำหรับเส้นที่มีการกลับเฟสเบิรสต์จะมีค่าเฟส 225 องศา

- MUX_SEL_OUT สัญญาณขาออกขนาด 2 บิต เป็นสัญญาณที่ใช้ควบคุมการสอดแทรกสัญญาณระดับ Blanking และ ระดับ Sync เข้ากับสัญญาณความส่องสว่างสำหรับการสร้างสัญญาณ CVS และควบคุมการสอดแทรกสัญญาณเบิรสต์เข้ากับสัญญาณความแตกต่างสีที่ผ่านการมอดูเลตแล้วเพื่อสร้างสัญญาณสี ค่าการนับตรงกับตำแหน่งของสัญญาณดังนี้ [7]

ค่า 0 เป็นคาบเวลาสำหรับสัญญาณ Sync

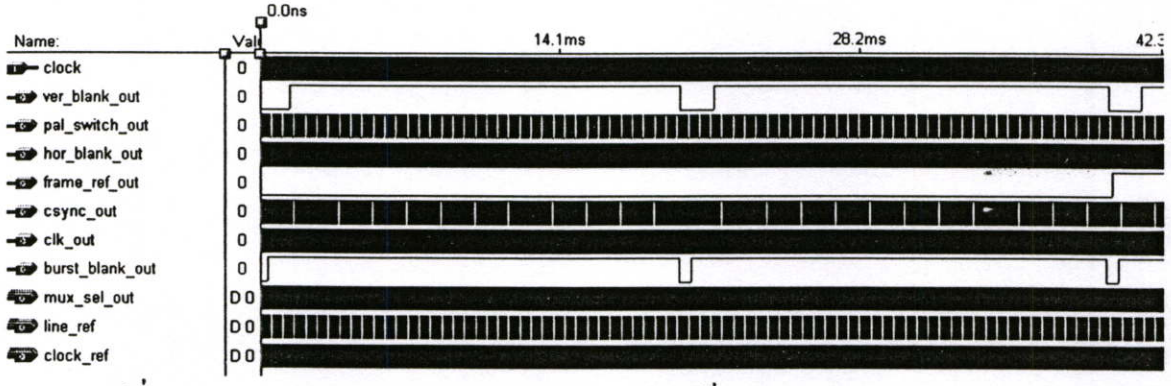
ค่า 1 เป็นคาบเวลาสำหรับสัญญาณ Front Porch (Blanking Level)

ค่า 2 เป็นคาบเวลาสำหรับสัญญาณ Active Video

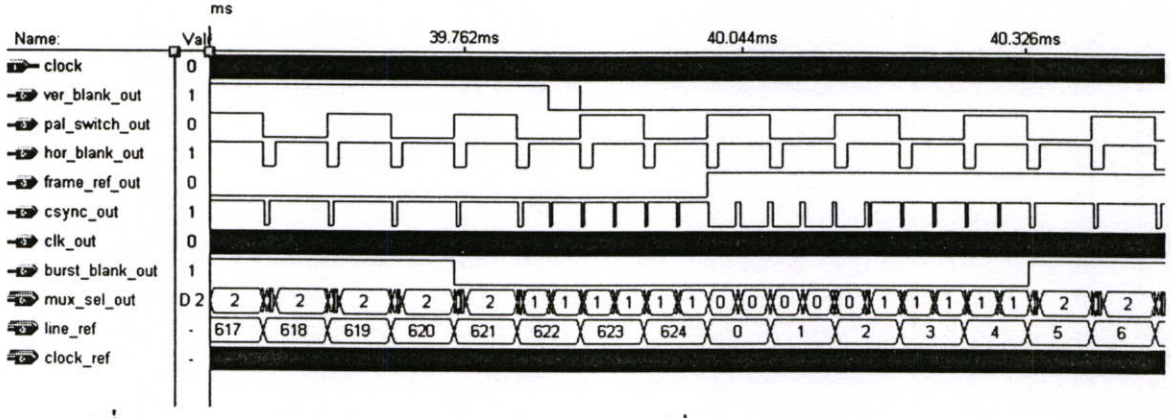
ค่า 3 เป็นคาบเวลาสำหรับสัญญาณ Burst

- CSYNC_OUT สัญญาณขาออกขนาด 1 บิต เป็นสัญญาณซิ้งค์เบ็ดเสร็จ (Composite Sync) ประกอบด้วยสัญญาณซิ้งค์ทางแนวตั้งและแนวนอน

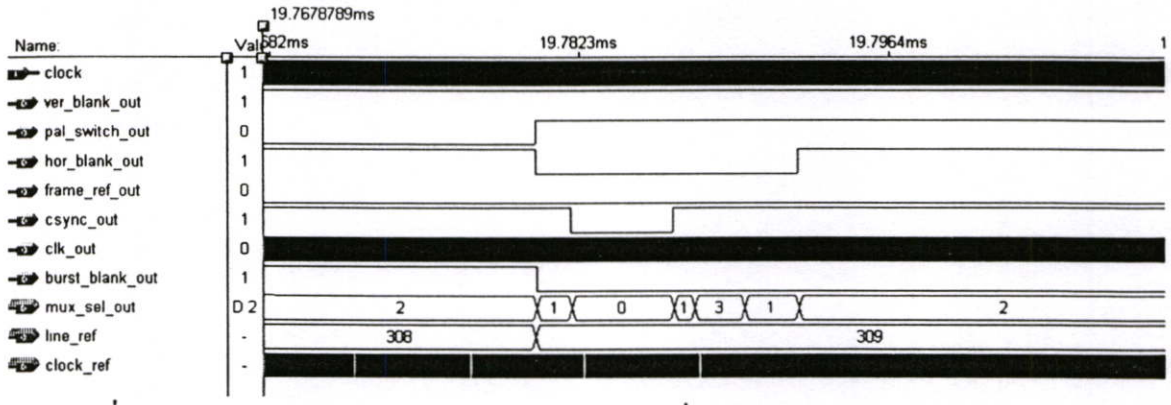
- PIX_CLK_OUT สัญญาณขาออกขนาด 1 บิต เป็นสัญญาณซึ่งถูกหารด้วยสองมาจากสัญญาณนาฬิกาของระบบดังนั้นมันจึงมีความถี่เท่ากับ 4.43361875 MHz.*4 ใช้เป็นสัญญาณอ้างอิงภายในของภาคนี้และเตรียมไว้เป็นสัญญาณนาฬิกาสำหรับการสร้างจุดภาพ (Pixel Clock)



รูปที่ 5.10 แสดงผลการจำลองการทำงานในระดับไทม์มิ่งของภาค CLOCK_LINE_REF



รูปที่ 5.11 แสดงผลการจำลองการทำงานในระดับไทม์มิ่งของภาค CLOCK_LINE_REF (ต่อ)



รูปที่ 5.12 แสดงผลการจำลองการทำงานในระดับไทม์มิ่งของภาค CLOCK_LINE_REF (ต่อ)

5.5.2 ภาค COLOR_SPACE

ทำหน้าที่แปลงสัญญาณสีแดง เขียวและน้ำเงินให้อยู่ในรูปของสัญญาณความส่องสว่าง (Luminance: Y) ซึ่งมีรูปแบบการคำนวณตามสมการที่ (5.2) และสัญญาณความแตกต่างสี (R-Y, B-Y) ออกแบบโดยใช้สถาปัตยกรรมแบบไปป์ไลน์ [20] ที่แบ่งงานการคำนวณออกเป็น 4 สเตจเพื่อทำให้ระบบสามารถทำงานได้ที่ความถี่สุ่ม 8 เท่าของความถี่พาหะรองของสัญญาณสี โครงสร้างในระดับ RTL แสดงดังรูปที่ 5.13

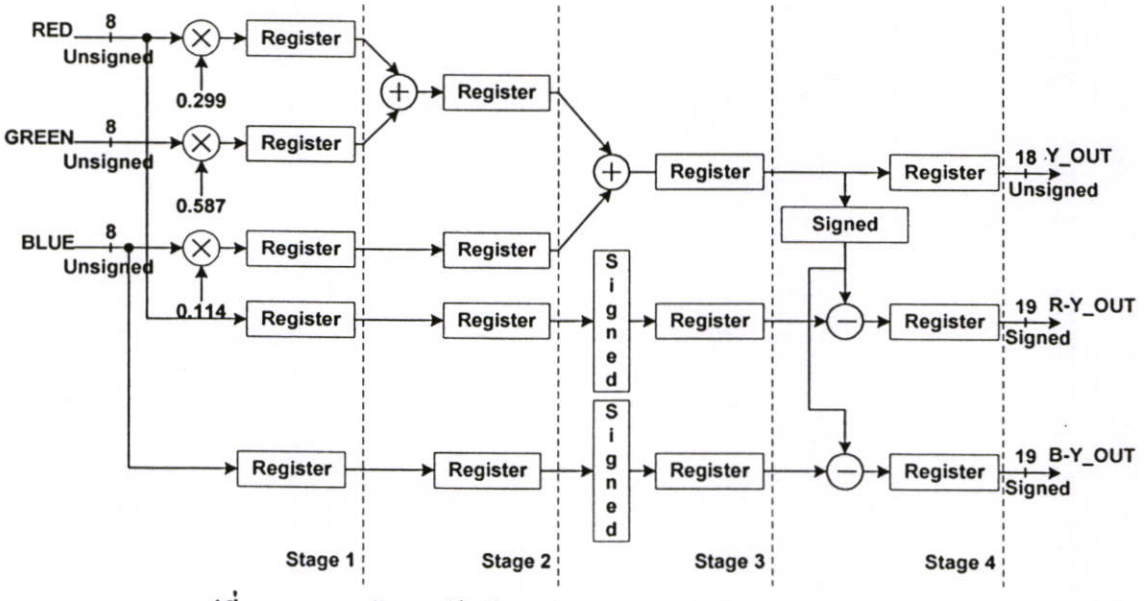
$$Y = 0.299R + 0.587G + 0.114B \tag{5.2}$$

รหัสต้นฉบับ (Source Code) ของภาคนี้ชื่อ COLOUR_SPACE.VHD หน้าที่ของแต่ละสัญญาณมีดังนี้

- RED สัญญาณขาเข้าขนาด 8 บิต เป็นข้อมูลของสัญญาณภาพสีแดงที่ใช้ระบบตัวเลขฐานสองแบบไม่มีเครื่องหมาย (Unsigned)
- GREEN สัญญาณขาเข้าขนาด 8 บิต เป็นข้อมูลของสัญญาณภาพสีเขียวที่ใช้ระบบตัวเลขฐานสองแบบไม่มีเครื่องหมาย (Unsigned)
- BLUE สัญญาณขาเข้าขนาด 8 บิต เป็นข้อมูลของสัญญาณภาพสีน้ำเงินที่ใช้ระบบตัวเลขฐานสองแบบไม่มีเครื่องหมาย (Unsigned)
- Y_OUT สัญญาณขาออกขนาด 18 บิต (Input + Coefficient = 8+10) เป็นข้อมูลของสัญญาณความส่องสว่าง (Luminance: Y) ที่ใช้ระบบตัวเลขฐานสองแบบไม่มีเครื่องหมาย (Unsigned)

- B-Y_OUT สัญญาณขาออกขนาด 19 บิต (Input + Coefficient + Sign = 8+10+1) เป็นข้อมูลของสัญญาณความแตกต่างสี (Color Difference) ที่ใช้ระบบตัวเลขฐานสองแบบมีเครื่องหมาย (Signed)

- R-Y_OUT สัญญาณขาออกขนาด 19 บิต (Input + Coefficient + Sign = 8+10+1) เป็นข้อมูลของสัญญาณความแตกต่างสี (Color Difference) ที่ใช้ระบบตัวเลขฐานสองแบบมีเครื่องหมาย (Signed)



รูปที่ 5.13 แสดงโครงสร้างในระดับ RTL ของภาค COLOR_SPACE

ในการทำไปป์ไลน์มีลำดับขั้นการคำนวณของแต่ละสเตจดังนี้

- Stage 1 สัญญาณสีแดง, เขียวและน้ำเงินจะถูกคูณด้วยสัมประสิทธิ์ซึ่งเป็นค่าคงที่ต่างๆกัน แต่มีความกว้างเท่ากันคือ 10 บิต สัมประสิทธิ์สามารถคำนวณได้จากสมการที่ (5.2)

$$BinCoeff = Round(2^{10} * C) \tag{5.2}$$

เมื่อ

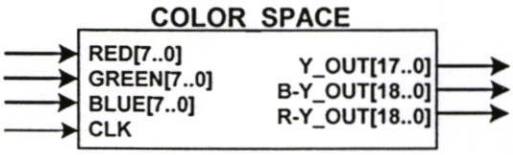
BinCoeff คือ ค่าของสัมประสิทธิ์ในระบบเลขฐานสอง

Round คือ ฟังก์ชั่นการปัดเศษ (น้อยกว่า 0.5 ปัดลง มากกว่าหรือเท่ากับ 0.5 ปัดขึ้น)

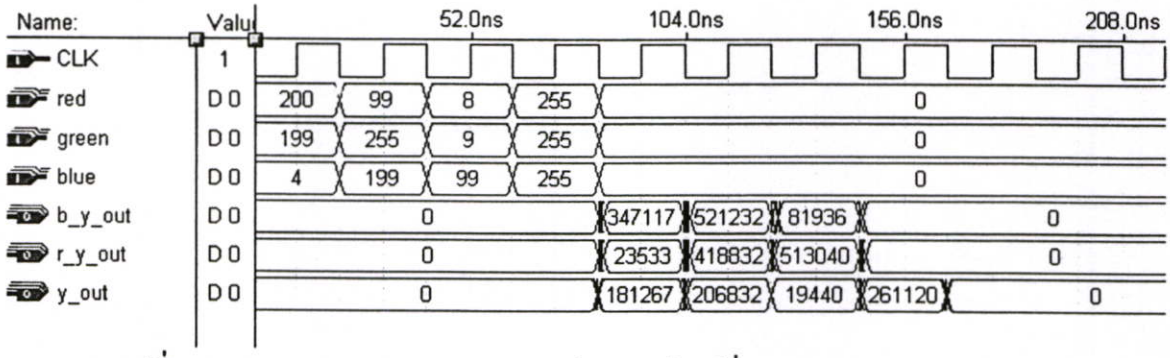
C คือ ค่า จริงของสัมประสิทธิ์ในระบบเลขฐานสิบ

สัญญาณหลังจากผ่านการคูณจะถูกเก็บไว้ในไปป์ไลน์รีจีสเตอร์พร้อมกับสัญญาณ RED และ BLUE เพื่อส่งให้สเตจต่อไป

- Stage 2 สัญญาณ RED ถูกนำมาบวกกับสัญญาณ GREEN และเก็บไว้ในไปป์ไลน์รีจีสเตอร์เพื่อเตรียมสร้างสัญญาณความส่องสว่างในขณะเดียวกันสัญญาณ RED และ BLUE ถูกหน่วงเวลาโดยการเก็บค่าในไปป์ไลน์รีจีสเตอร์เพื่อรอการประมวลผลในสเตจต่อไป
- Stage 3 สัญญาณผลบวกของ RED และ GREEN ถูกนำมาบวกกับสัญญาณ BLUE ทำให้ได้สัญญาณความส่องสว่างและเก็บค่าไว้ในไปป์ไลน์รีจีสเตอร์ อีกส่วนหนึ่งสัญญาณ RED และ BLUE ถูกแปลงจากแบบไม่มีเครื่องหมาย (Unsigned) ให้เป็นแบบแบบมีเครื่องหมาย (Signed) และเก็บค่าไว้ในไปป์ไลน์รีจีสเตอร์
- Stage 4 สัญญาณความส่องสว่างที่ได้จากสเตจที่แล้วถูกแบ่งออกเป็นสองทาง ทางแรกค่าถูกเก็บไว้ในไปป์ไลน์รีจีสเตอร์เพื่อรอการประมวลผลในส่วนของสัญญาณความแตกต่างสี ทางที่สองจะถูกแปลงจากแบบไม่มีเครื่องหมาย (Unsigned) ให้เป็นแบบแบบมีเครื่องหมาย (Signed) เพื่อนำไปลบกับสัญญาณ RED และ BLUE ในการสร้างสัญญาณสัญญาณความแตกต่างสี



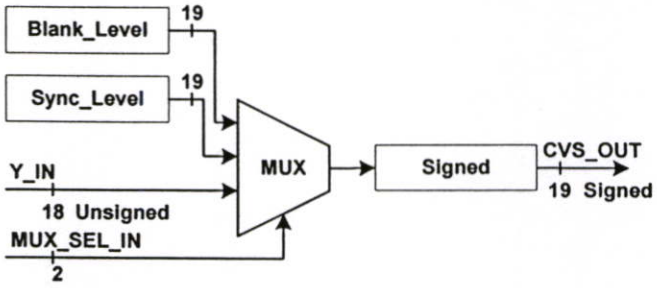
รูปที่ 5.14 แสดงมุมมองการเชื่อมต่อของภาค COLOR_SPACE



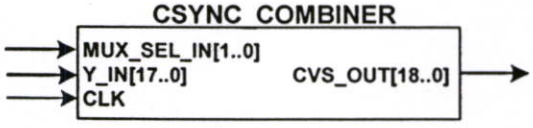
รูปที่ 5.15 แสดงผลการจำลองการทำงานในระดับไทม์มิ่งของภาค COLOR_SPACE

5.5.3 ภาค CSYNC_COMBINER

ทำหน้าที่มัลติเพล็กสัญญาณสามสัญญาณเข้าด้วยกัน ได้แก่ สัญญาณระดับแบล้งกิ้ง, สัญญาณระดับซิ้งค์และสัญญาณความส่องสว่างที่ส่งมาจากภาค COLOR_SPACE ทั้งสามสัญญาณนี้ใช้ในการสร้างสัญญาณภาพเบ็ดเสร็จ (Composite Video Signal: CVS) โครงสร้างในระดับ RTL และ มุมมองการเชื่อมต่อแสดงดังรูปที่ 5.16 และ 5.17 ตามลำดับ



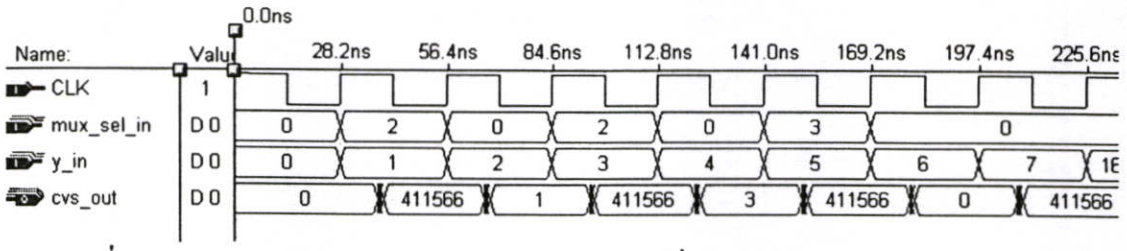
รูปที่ 5.16 แสดงโครงสร้างในระดับ RTL ของภาค CSYNC_COMBINER



รูปที่ 5.17 แสดงมุมมองการเชื่อมต่อของภาค CSYNC_COMBINER

รหัสต้นฉบับ (Source Code) ของภาคนี้ชื่อ CSYNC_COMBINER.VHD สัญญาณและค่าคงที่ มีหน้าที่ดังต่อไปนี้

- MUX_SEL_IN สัญญาณขาเข้าขนาด 2 บิต ถูกส่งมาจากภาค CLOCK_LINE_REF หน้าที่ควบคุมการมัลติเพล็กสัญญาณ รายละเอียดดังได้กล่าวไปแล้วในส่วนของภาค CLOCK_LINE_REF
- Y_IN สัญญาณขาเข้าขนาด 18 บิตแบบไม่มีเครื่องหมาย เป็นสัญญาณความส่องสว่าง ถูกส่งมาจากภาค COLOR_SPACE
- Blank_LEVEL เป็นค่าคงที่มีขนาด 19 บิต (กำหนดให้มีขนาดเท่ากับสัญญาณ CVS_OUT) แบบมีเครื่องหมาย เป็นค่าของสัญญาณระดับแบล้งกิ้ง
- Sync_LEVEL เป็นค่าคงที่มีขนาด 19 บิต (กำหนดให้มีขนาดเท่ากับสัญญาณ CVS_OUT) แบบมีเครื่องหมาย เป็นค่าของสัญญาณระดับซิ้งค์
- CVS_OUT สัญญาณขาออกขนาด 19 บิต (Y_IN + Sign = 18+1) แบบมีเครื่องหมาย เป็นสัญญาณภาพเบ็ดเสร็จ (Composite Video Signal: CVS) ตามมาตรฐานของระบบพาล



รูปที่ 5.18 แสดงผลการจำลองการทำงานในระดับไทม์มิ่งของภาค CSYNC_COMBINER

สำหรับงานวิจัยชิ้นนี้ตัวกรองทั้งหมดได้เลือกใช้แบบเอฟไออาร์ (Finite Impulse Response: FIR) เนื่องจากการเข้ารหัสสัญญาณสีใช้การมอดูเลตเชิงแอมปลิจูดแบบแถบข้างคู่ขจัดคลื่นพาห์ (AM-DSBSC) ความเพี้ยนทางเฟสส่งผลให้ขนาดของสัญญาณขาออกหลังการดีมอดูเลตผิดไปจากความเป็นจริง (อธิบายโดยละเอียดในหัวข้อ 2.3.2.4) ดังนั้นจึงต้องใช้ตัวกรองที่มีผลตอบสนองเฟสเชิงเส้นเพื่อแก้ปัญหาดังกล่าว คุณสมบัติต่างๆของตัวกรองเช่น ωp, ωs ได้มาจากตัวกรองเชิงอุปมานจากระบบที่ใช้เป็นต้นแบบ แต่อย่างไรก็ตามสำหรับงานวิจัยชิ้นนี้ไม่สามารถออกแบบตัวกรองให้มีคุณสมบัติเหมือนตัวกรองที่เป็นต้นแบบได้ 100 % เนื่องจากต้องการประหยัดทรัพยากรแต่ยังไม่ส่งผลกระทบอย่างรุนแรงกับระบบกล่าวคือสัญญาณขาออกที่ได้อยู่ในเกณฑ์ที่ยอมรับได้

5.5.4 ภาค CVS_LPF3TAB

เป็นตัวกรองไม่ป้อนกลับเชิงเลข (FIR) ทำหน้าที่กรองความถี่ต่ำผ่านเพื่อจำกัดย่านความถี่การทำงานของสัญญาณภาพเบ็ดเสร็จ (CVS) [5], [19] ออกแบบโดยใช้วิธี Equiripple ซึ่งให้ผลตอบสนองเฟสเชิงเส้น [21], [22] มีรูปแบบการคำนวณตามสมการที่ (5.3) คุณสมบัติพื้นฐานของตัวกรองมีดังนี้

อันดับของตัวกรอง = 3 ได้จากการทดลองและการออกแบบซ้ำ (Iteration) เพื่อให้ได้คุณสมบัติใกล้เคียงกับความต้องการและประหยัดทรัพยากร

ความกว้างสัมประสิทธิ์ = 8 Bits. ได้จากการทดลองและการออกแบบซ้ำ (Iteration) เพื่อให้ได้คุณสมบัติใกล้เคียงกับความต้องการและประหยัดทรัพยากรโดยใช้การจำลองการทำงานแบบจำนวนโดยตรง (Fixed Point) เพื่อวิเคราะห์ค่าความผิดพลาดจากการแบ่งระดับ

ความกว้างของสัญญาณขาเข้า:x(n) = 19 Bits. เนื่องจากความกว้างสัญญาณขาออกจากภาคก่อนหน้า (CSYNC_COMBINER) มีขนาด 19 บิต

ความกว้างของสัญญาณขาออก:y(n) = 12 Bits. ตัดปลาย (Truncation) เพื่อให้มีขนาดเท่ากับความกว้างสัญญาณขาเข้าของตัวแปลงสัญญาณเชิงเลขเป็นสัญญาณเชิงอุปมาน (DAC)

Passband edge ripple:ωp = 6 MHz. ได้จากตัวกรองเชิงอุปมานจากระบบที่ใช้เป็นต้นแบบทดลองและออกแบบซ้ำ (Iteration) เพื่อให้ได้คุณสมบัติใกล้เคียงกับความต้องการและประหยัดทรัพยากร

Stopband edge ripple:ωs = 6.25 MHz. ได้จากตัวกรองเชิงอุปมานจากระบบที่ใช้เป็นต้นแบบ ทดลองและออกแบบซ้ำ (Iteration) เพื่อให้ได้คุณสมบัติใกล้เคียงกับความต้องการและประหยัดทรัพยากร

$$y(n) = \sum_{k=0}^{3} h(k).x(n-k) \tag{5.3}$$

เมื่อ

$y(n)$ คือ ลำดับสัญญาณขาออกเชิงเลขของตัวกรอง

$h(k)$ คือ ลำดับผลตอบสนองอิมพัลส์ของตัวกรอง (ซึ่งเท่ากับสัมประสิทธิ์ของตัวกรอง)

$x(n)$ คือ ลำดับสัญญาณขาเข้าเชิงเลขของตัวกรอง

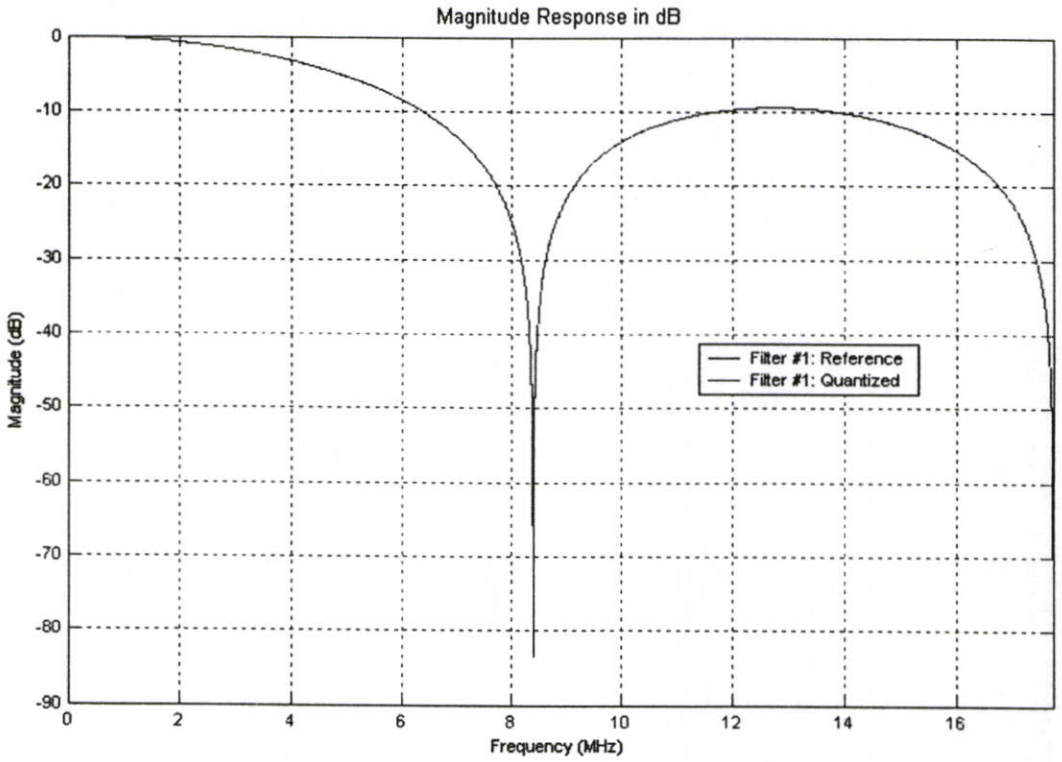
ตัวกรองทุกตัวที่ใช้ในงานวิจัยชิ้นนี้ใช้โครงสร้างและวิธีการออกแบบเดียวกัน แตกต่างกันเฉพาะคุณสมบัติที่ต้องการเท่านั้น หลังการออกแบบจะใช้การจำลองการทำงานเพื่อทดสอบคุณสมบัติ เนื่องจากการสร้างตัวกรองด้วยระบบตัวเลขและการประมวลผลแบบจำนวนโดยตรง ค่าสัมประสิทธิ์จะถูกแบ่งระดับทำให้คุณสมบัติของตัวกรองเปลี่ยนไป

จากคุณสมบัติในข้างต้นสามารถออกแบบตัวกรองจากการใช้อัลกอริทึม Parks-McClellan [21], [22] รูปที่ 5.19 แสดงค่าสัมประสิทธิ์ที่ยังไม่ได้แบ่งระดับ (Reference Coefficient) โดยการใช้ระบบตัวเลขในการคำนวณแบบจำนวนอิงดัชนี (Floating Point) และ ค่าสัมประสิทธิ์ที่แบ่งระดับแล้ว (Quantized Coefficient) โดยการใช้ระบบตัวเลขในการคำนวณแบบจำนวนโดยตรง (Fixed Point) ผลการจำลองการทำงานในระดับอัลกอริทึมที่เปรียบเทียบระหว่างการใช้ระบบตัวเลขในการคำนวณแบบจำนวนอิงดัชนี กับผลการทดลองที่พิจารณาผลการปัดเศษของสัมประสิทธิ์เมื่อใช้ระบบตัวเลขในการคำนวณแบบจำนวนโดยตรง แสดงดังรูปที่ 5.20 – 5.23

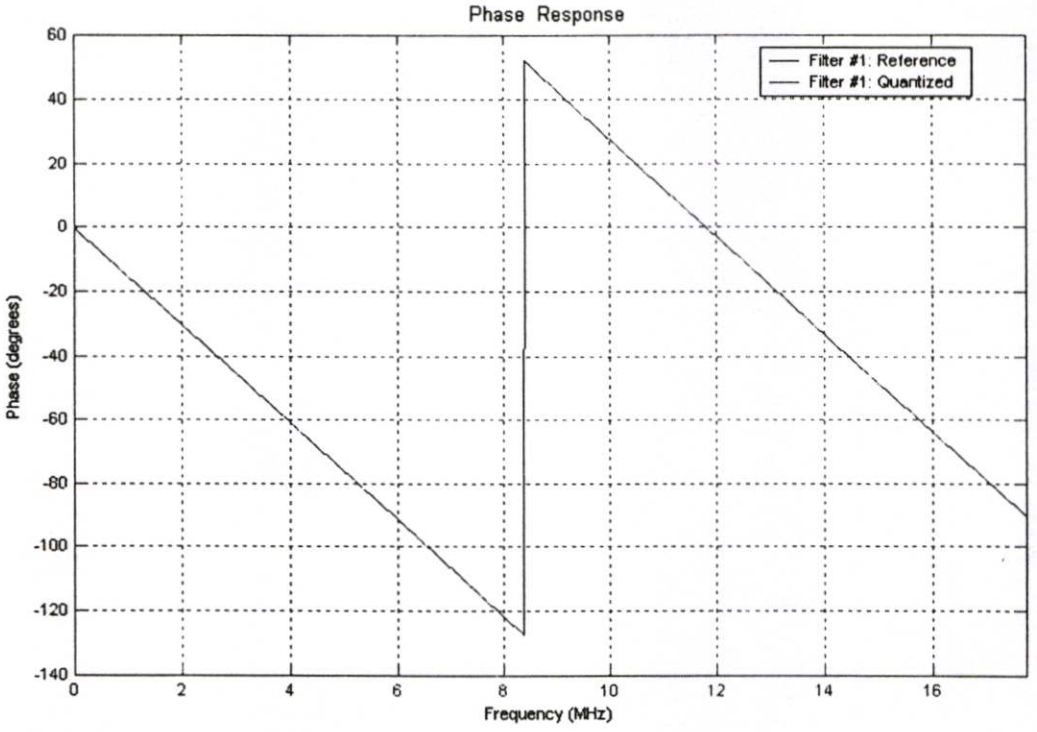
```
Quantized Direct form FIR filter
Numerator
    QuantizedCoefficients{1}    ReferenceCoefficients{1}
 (1)          0.27343750  0.273195991359514060
 (2)          0.22656250  0.226421847204183260
 (3)          0.22656250  0.226421847204183260
 (4)          0.27343750  0.273195991359514060

  FilterStructure = fir
     ScaleValues = []
 NumberOfSections = 1
 StatesPerSection = [3]
 CoefficientFormat = quantizer('fixed', 'round', 'saturate', [8  8])
     InputFormat = quantizer('fixed', 'floor', 'saturate', [16  15])
    OutputFormat = quantizer('fixed', 'floor', 'saturate', [16  15])
MultiplicandFormat = quantizer('fixed', 'floor', 'saturate', [16  15])
    ProductFormat = quantizer('fixed', 'floor', 'saturate', [32  30])
       SumFormat = quantizer('fixed', 'floor', 'saturate', [32  30])
```

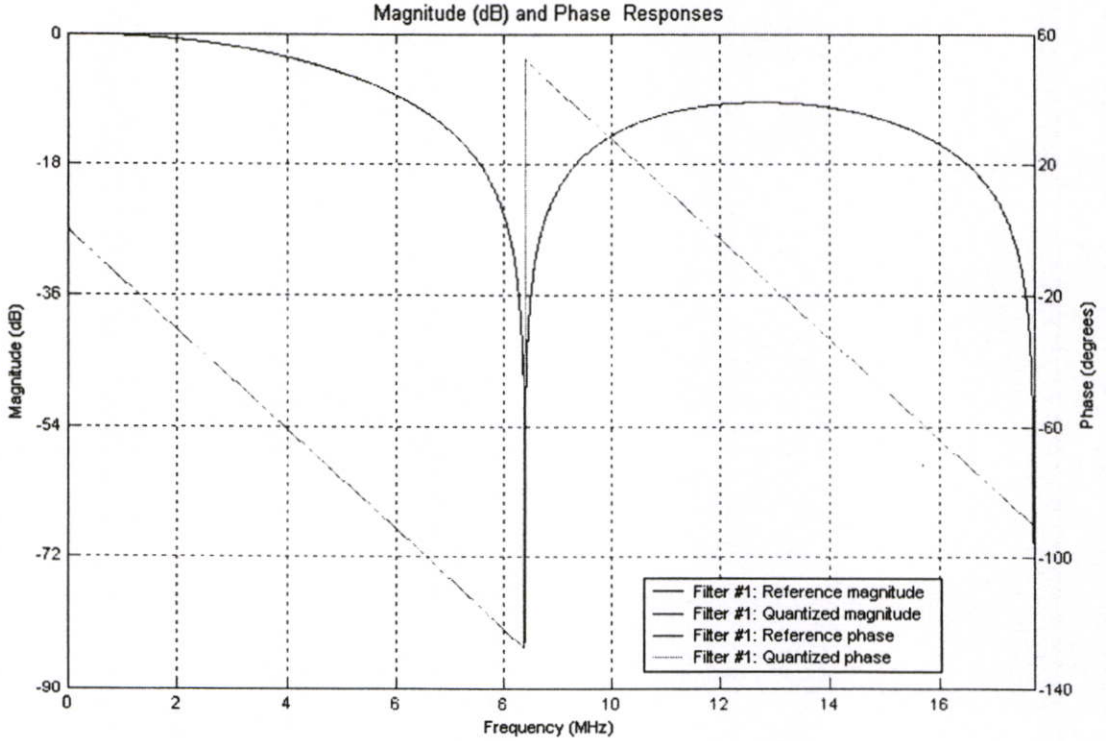
รูปที่ 5.19 แสดงค่าสัมประสิทธิ์ของตัวกรอง, ซ้าย (QuantizedCoefficient) เป็นค่าสัมประสิทธิ์ที่แบ่งระดับให้เหลือขนาดเป็น 8 บิต, ขวา (Reference Coefficient) เป็นค่าสัมประสิทธิ์ที่ยังไม่ได้แบ่งระดับ



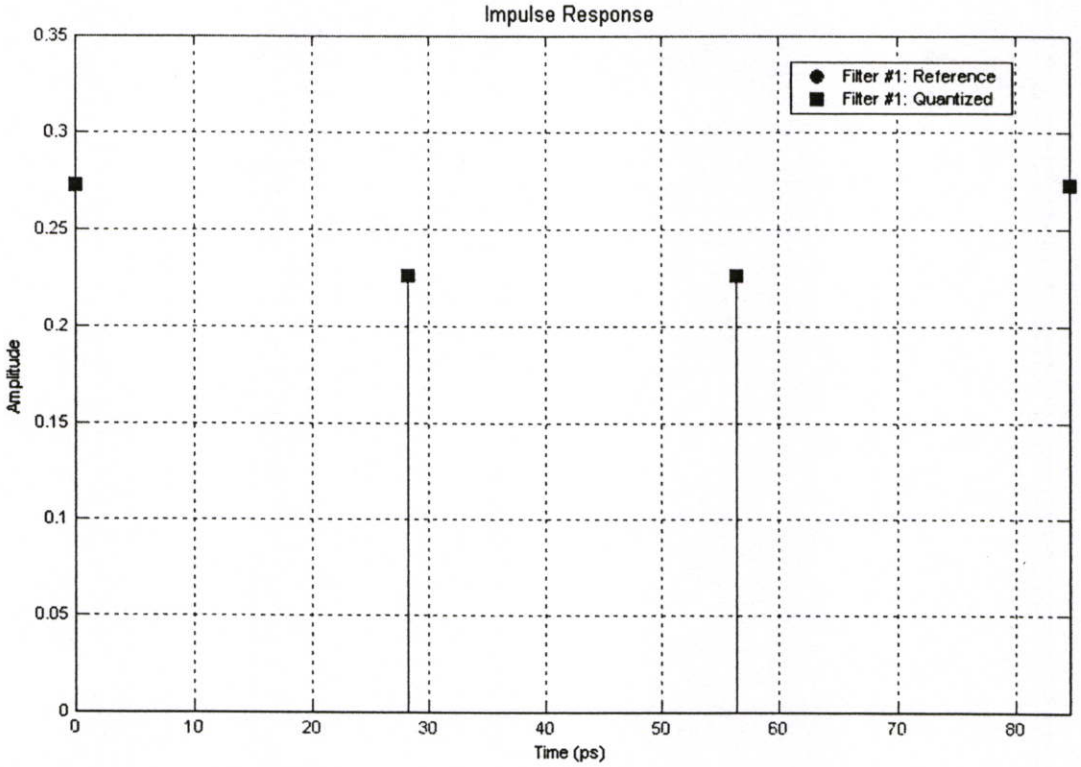
รูปที่ 5.20 แสดงผลตอบสนองความถี่ของตัวกรองภาค CVS_LPF3TAB



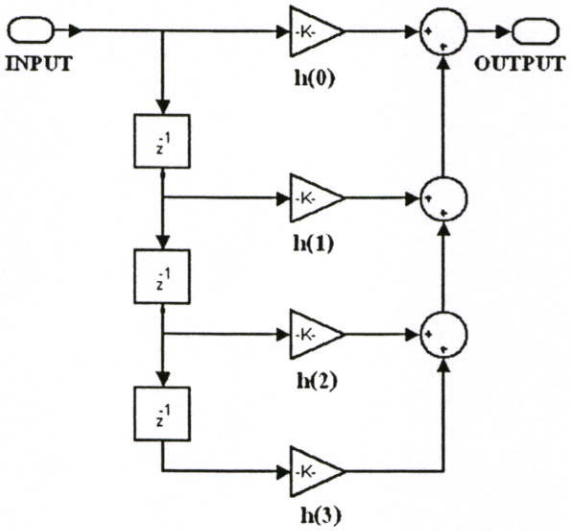
รูปที่ 5.21 แสดงผลตอบสนองเฟสของตัวกรองภาค CVS_LPF3TAB



รูปที่ 5.22 แสดงผลตอบสนองความถี่เทียบกับผลตอบสนองเฟสของตัวกรองภาค CVS_LPF3TAB



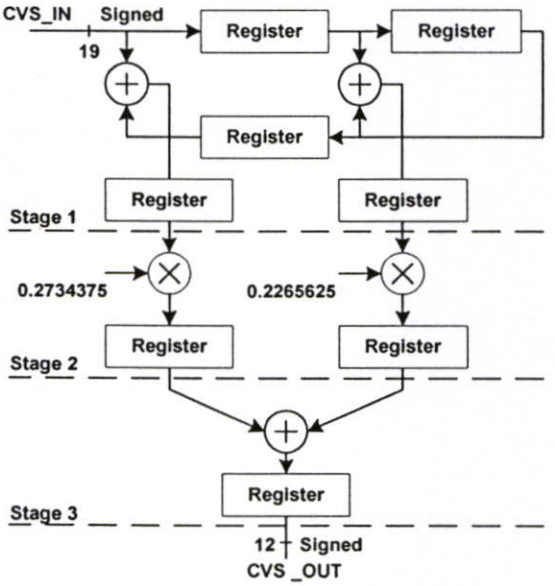
รูปที่ 5.23 แสดงผลตอบสนองอิมพัลส์ของตัวกรองภาค CVS_LPF3TAB



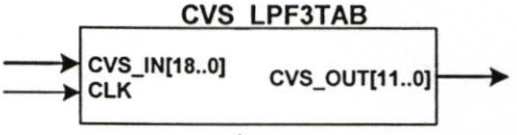
รูปที่ 5.24 แสดงโครงสร้างแบบโดยตรงของตัวกรองไม่ป้อนกลับเชิงเลขที่ใช้ในงานวิจัยนี้

การออกแบบได้ใช้โครงสร้างแบบโดยตรง (Direct form) ดังแสดงในรูปที่ 5.24 และเนื่องจากตัวกรองที่ออกแบบมีคุณสมบัติผลตอบสนองเฟสเชิงเส้นซึ่งมีผลตอบสนองอิมพัลส์ที่สมมาตร

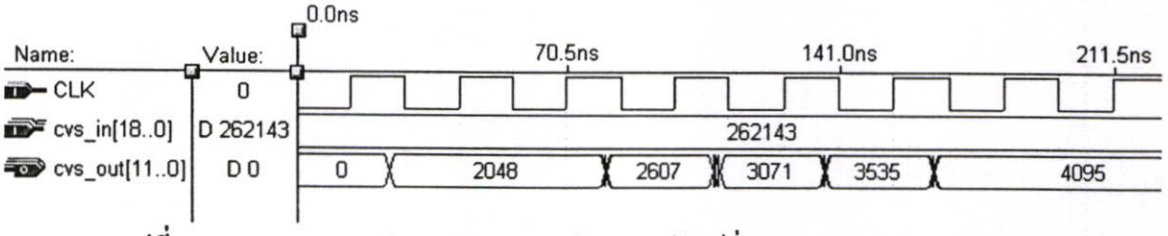
ทำให้สามารถลดขนาดของโครงสร้างให้เล็กลงได้ รูปที่ 5.25 แสดงโครงสร้างในระดับ RTL ที่ใช้สถาปัตยกรรมไปป์ไลน์ในการออกแบบ จะเห็นว่างาน (Task) ถูกแบ่งเป็นงานย่อย (Subtask) ทั้งหมด 3 สเตจ ดังนั้นผลของการคำนวณจึงล้าหลังสัญญาณขาเข้า (Latency) ทั้งหมด 3 คาบสัญญาณนาฬิกา รูปที่ 5.27 แสดงผลการจำลองการทำงานในระดับไทม์มิ่ง



รูปที่ 5.25 แสดงโครงสร้างในระดับ RTL ของตัวกรองภาค CVS_LPF3TAB



รูปที่ 5.26 แสดงมุมมองการเชื่อมต่อของภาค CVS_LPF3TAB



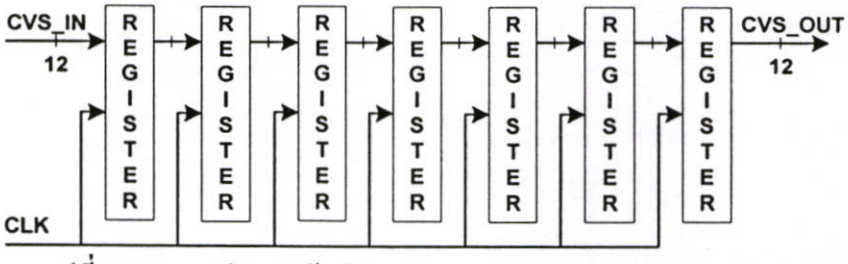
รูปที่ 5.27 แสดงผลการจำลองการทำงานในระดับไทม์มิ่งของภาค CVS_LPF3TAB

รหัสต้นฉบับ (Source Code) ของภาคนี้ชื่อ CVS_LPF3TAB.VHD หน้าที่ของแต่ละสัญญาณมีดังนี้

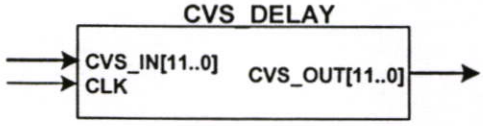
- CVS_IN สัญญาณขาเข้าขนาด 19 บิต ชนิดมีเครื่องหมายแบบ 2'Complement Signed Fractional เป็นสัญญาณภาพเบ็ดเสร็จที่ถูกส่งมาจากภาค CSYNC_COMBINER
- CVS_OUT สัญญาณขาออกขนาด 12 บิต ชนิดมีเครื่องหมายแบบ Offset 2'Complement เป็นสัญญาณภาพเบ็ดเสร็จขาออกที่ยังไม่ได้ชดเชยเวลา

5.5.5 ภาค CVS_DELAY

ทำหน้าที่หน่วงเวลาสัญญาณภาพเบ็ดเสร็จ เพื่อรอสัญญาณสีที่มีความซับซ้อนของการประมวลผลมากกว่า ผลการหน่วงเวลาของสัญญาณสีเกิดจากการใช้สถาปัตยกรรมไปป์ไลน์และผลตอบสนองของตัวกรองซึ่งการออกแบบจะใช้การชดเชยเฉพาะผลของการคำนวณแบบไปป์ไลน์เท่านั้นเนื่องจากตัวกรองที่ใช้มีอันดับต่ำดังนั้นจึงมีค่าการหน่วงเวลาน้อยจึงไม่นำมาพิจารณาในการออกแบบ โครงสร้างในระดับ RTLแสดงดังรูปที่ 5.28



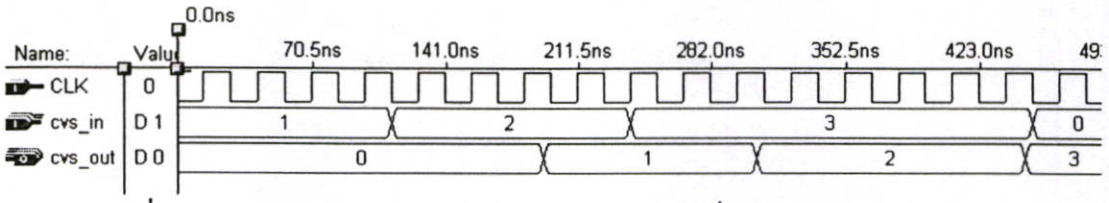
รูปที่ 5.28 แสดงโครงสร้างในระดับ RTL ของภาค CVS_DELAY



รูปที่ 5.29 แสดงมุมมองการเชื่อมต่อของภาค CVS_DELAY

รหัสต้นฉบับ (Source Code) ของภาคนี้ชื่อ CVS_DELAY.VHD แต่ละสัญญาณมีหน้าที่ดังต่อไปนี้

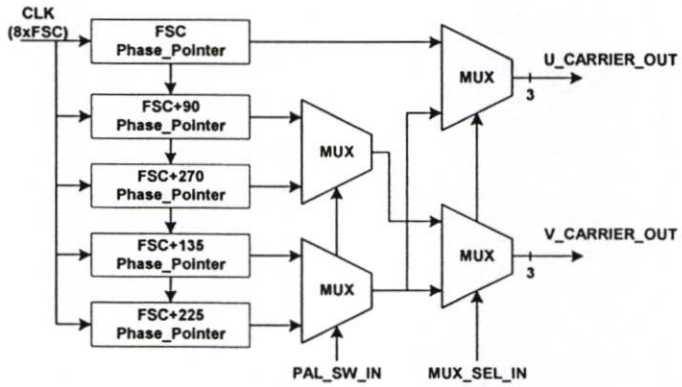
- CVS_IN สัญญาณขาเข้าขนาด 12 บิต มีขนาดเท่ากับความกว้างสัญญาณขาออกของภาคก่อนหน้า (CVS_LPF3TAB) เป็นสัญญาณภาพเบ็ดเสร็จขาออกที่ยังไม่ได้ชดเชยเวลา ถูกส่งมาจากภาค CVS_LPF3TAB
- CVS_OUT สัญญาณขาออกขนาด 12 บิต เป็นสัญญาณภาพเบ็ดเสร็จขาออกที่ผ่านการชดเชยเวลาแล้วและเป็นสัญญาณขาออกของระบบ



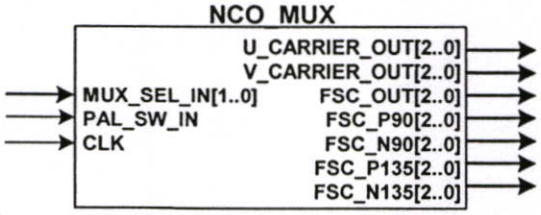
รูปที่ 5.30 แสดงผลการจำลองการทำงานในระดับไทม์มิ่งของภาค CVS_DELAY

5.5.6 ภาค NCO_MUX

ทำหน้าที่สร้างสัญญาณชี้เฟส (Phase Pointer) เพื่อควบคุมการสร้างสัญญาณพาหะรองของสัญญาณสีและสัญญาณเบิรสต์ให้กับภาค UVMOD โครงสร้างในระดับ RTL แสดงดังรูปที่ 5.31



รูปที่ 5.31 แสดงโครงสร้างในระดับ RTL ของภาค NCO_MUX



รูปที่ 5.32 แสดงมุมมองการเชื่อมต่อของภาค NCO_MUX

รหัสต้นฉบับ (Source Code) ของภาคนี้ชื่อ NCO_MUX.VHD การทำงานโดยสังเขปและหน้าที่ของแต่ละสัญญาณมีดังนี้

- PAL_SW_IN สัญญาณขาเข้าขนาด 1 บิต ใช้ควบคุมการมัลติเพล็กเพื่อสร้างสัญญาณเบิรสต์และสัญญาณพาหะของสัญญาณ V ในการสร้างสัญญาณเบิรสต์จะทำการมัลติเพล็กระหว่าง

FSC+135 กับ FSC+225 และในการสร้างสัญญาณพาหะของสัญญาณ V จะทำการมัลติเพล็กระหว่าง FSC+90 กับ FSC+270

- MUX_SEL_IN สัญญาณขาเข้าขนาด 2 บิต ที่ส่งมาจากภาค CLOCK_LINE_REF ใช้ควบคุมการมัลติเพล็กสัญญาณชี้เฟส ในกรณีของสัญญาณชี้เฟสของสัญญาณ U สัญญาณนี้จะเลือกระหว่าง สัญญาณอ้างอิง (FSC ซึ่งก็คือสัญญาณพาหะของ U นั่นเอง) กับสัญญาณเบิรสต์ (FSC+135 และ FSC+225) และในกรณีของสัญญาณชี้เฟสของสัญญาณ V สัญญาณนี้จะเลือกระหว่าง สัญญาณพาหะของสัญญาณ V (FSC+90 และ FSC+270 ได้จากการมัลติเพล็กซึ่งถูกควบคุมด้วยสัญญาณ PAL_SW) กับสัญญาณเบิรสต์ ค่าของสัญญาณ MUX_SEL_IN ตรงกับตำแหน่งของสัญญาณดังนี้

ค่า 0-1 ไม่พิจารณาและจะให้ค่าสัญญาณชี้เฟสเป็น 0

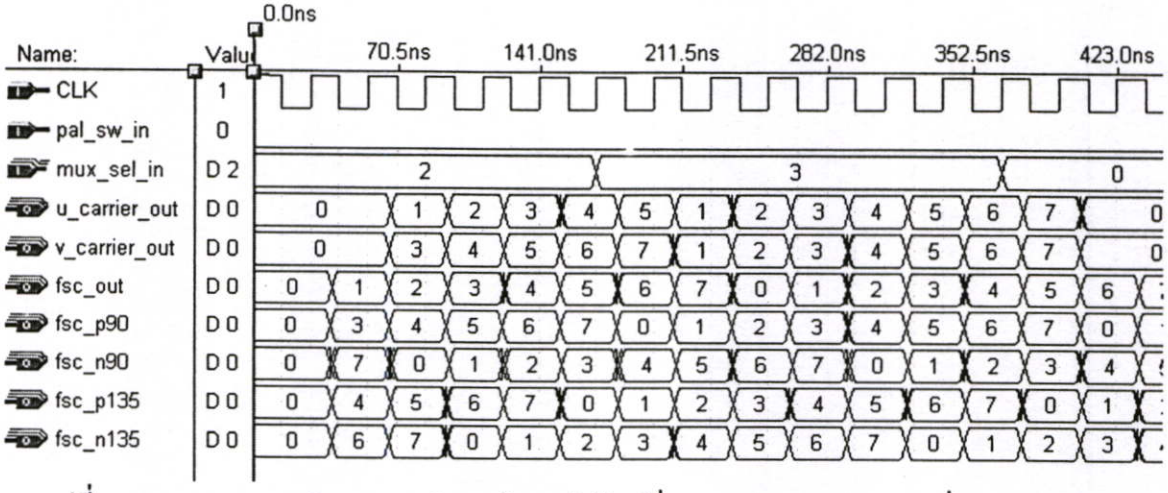
ค่า 2 ช่วงของสัญญาณภาพ (Active Video)

ค่า 3 ช่วงของสัญญาณเบิรสต์

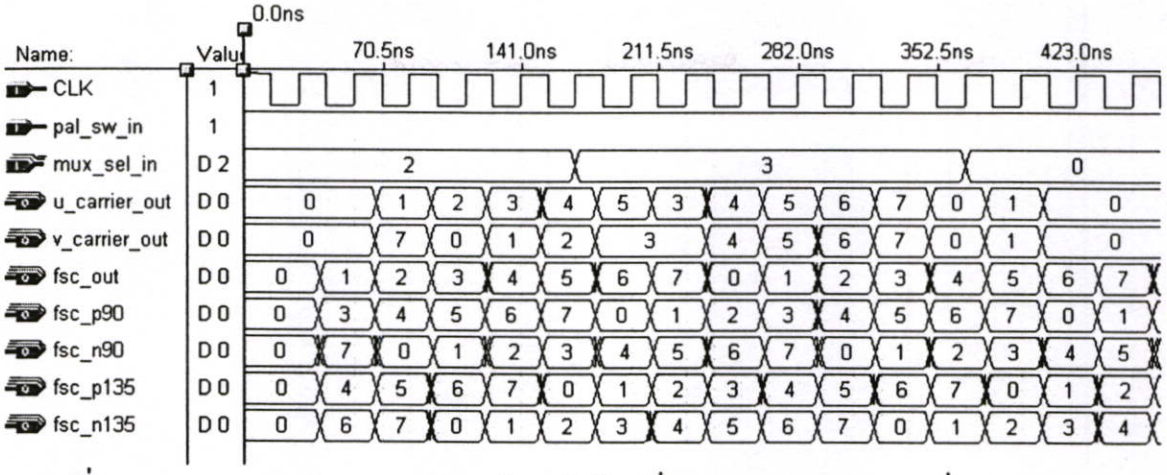
- U_CARRIER_OUT สัญญาณขาออกขนาด 3 บิต เป็นสัญญาณชี้เฟสสำหรับสัญญาณพาหะของสัญญาณ U ตำแหน่งการชี้มีค่าตั้งแต่ 0 ถึง 7 ตรงกับตำแหน่งของสัญญาณที่ 0, 45, 90, 135, 180, 225, 270, และ 315 องศา

- V_CARRIER_OUT สัญญาณขาออกขนาด 3 บิต เป็นสัญญาณชี้เฟสสำหรับสัญญาณพาหะของสัญญาณ V มีค่าตำแหน่งเหมือนกันกับสัญญาณ U_CARRIER_OUT

สัญญาณที่เหลือเป็นสัญญาณที่ไม่ได้ใช้งานใช้เพื่อทดสอบความถูกต้องของการออกแบบ



รูปที่ 5.33 แสดงผลการจำลองการทำงานในระดับไทม์มิ่งของภาค NCO_MUX เมื่อ PAL SW = 0



รูปที่ 5.34 แสดงผลการจำลองการทำงานในระดับไทม์มิ่งของภาค NCO_MUX เมื่อ PAL SW = 1

5.5.7 ภาค U_LPF3TAB_FAC

เป็นตัวกรองไม่ป้อนกลับเชิงเลขทำหน้าที่กรองความถี่ต่ำผ่านเพื่อป้องกันการเกิด Aliasing ที่ภาค UVMOD และลดทอนสัญญาณ B-Y ลง 0.493 เท่าเพื่อป้องกันการมอดูเลตเกิน (Over Modulation) [5] เรียกสัญญาณนี้ว่าสัญญาณ U ซึ่งโดยปกติตัวลดทอนจะถูกสร้างอยู่ในภาค COLOR_SPACE แต่เพื่อการประหยัดลอจิกเกตและลดการประมวลจึงใช้การลดสัมประสิทธิ์ของตัวกรองเพื่อให้ได้ผลการลดทอนดังกล่าว ออกแบบและใช้โครงสร้างเดียวกับภาค CVS_LPF3TAB คุณสมบัติพื้นฐานของตัวกรองมีดังนี้

อันดับของตัวกรอง = 3 ได้จากการทดลองและการออกแบบซ้ำ (Iteration) เพื่อให้ได้คุณสมบัติใกล้เคียงกับความต้องการและประหยัดทรัพยากร

ความกว้างของสัญญาณขาเข้า:X(n) = 19 Bits. เนื่องจากความกว้างสัญญาณขาออกจากภาคก่อนหน้า (COLOR_SPACE) มีขนาด 19 บิต

ความกว้างของสัญญาณขาออก:Y(n) = 16 Bits. ตัดปลาย (Truncation) เพื่อลดการใช้ทรัพยากรของภาคต่อไป (UVMOD)

Passband edge ripple:ωp = 6 MHz. ได้จากตัวกรองเชิงอุปมานจากระบบที่ใช้เป็นต้นแบบทดลองและออกแบบซ้ำ (Iteration) เพื่อให้ได้คุณสมบัติใกล้เคียงกับความต้องการและประหยัดทรัพยากร

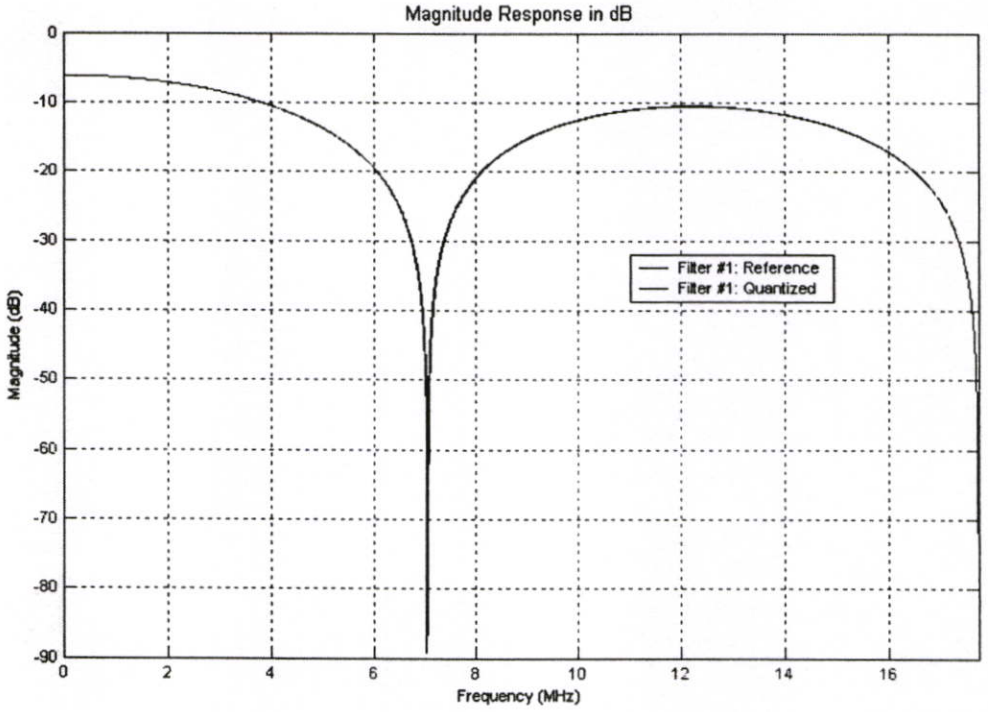
Stopband edge ripple:ωs = 6.25 MHz. ได้จากตัวกรองเชิงอุปมานจากระบบที่ใช้เป็นต้นแบบ ทดลองและออกแบบซ้ำ (Iteration) เพื่อให้ได้คุณสมบัติใกล้เคียงกับความต้องการและประหยัดทรัพยากร

h0 = h3 = 0.1796875 h1 = h2 = 0.06640625

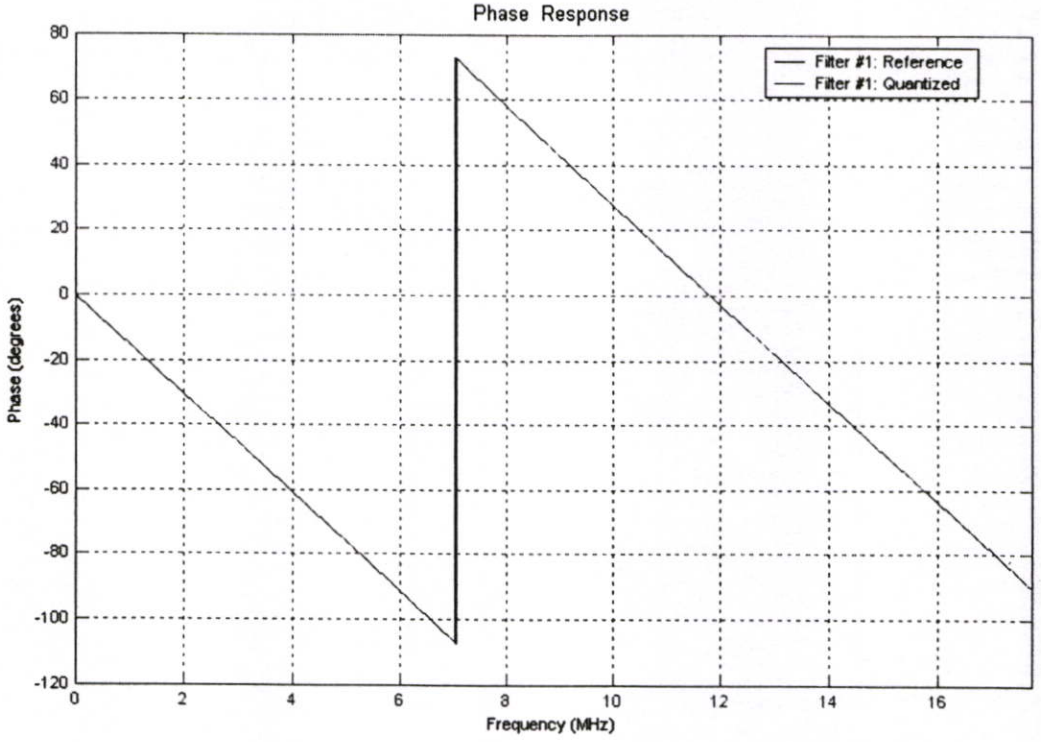
```
Quantized Direct form FIR filter
Numerator
    QuantizedCoefficients{1}    ReferenceCoefficients{1}
 (1)          0.17968750  0.180800163134391290
 (2)          0.06640625  0.065246870608314800
 (3)          0.06640625  0.065246870608314800
 (4)          0.17968750  0.180800163134391290

  FilterStructure = fir
     ScaleValues = []
  NumberOfSections = 1
  StatesPerSection = [3]
 CoefficientFormat = quantizer('fixed', 'round', 'saturate', [8  8])
     InputFormat = quantizer('fixed', 'floor', 'saturate', [16  15])
    OutputFormat = quantizer('fixed', 'floor', 'saturate', [16  15])
MultiplicandFormat = quantizer('fixed', 'floor', 'saturate', [16  15])
    ProductFormat = quantizer('fixed', 'floor', 'saturate', [32  30])
       SumFormat = quantizer('fixed', 'floor', 'saturate', [32  30])
```

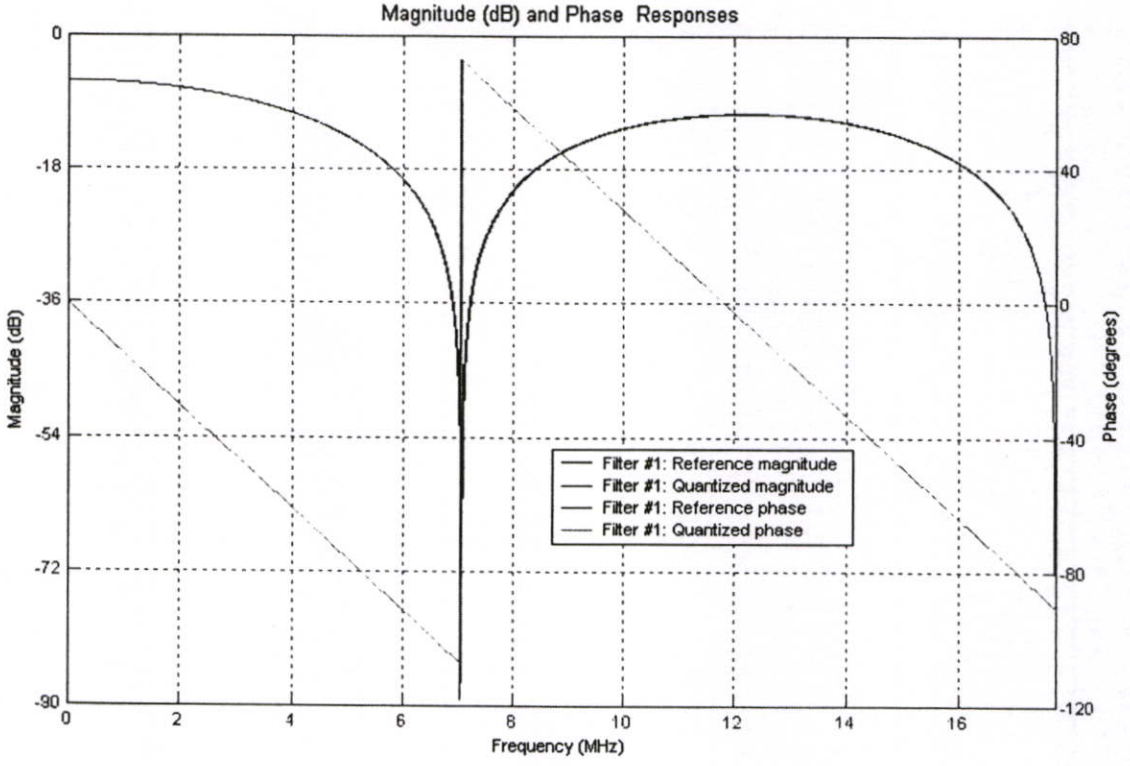
รูปที่ 5.35 แสดงค่าสัมประสิทธิ์ของตัวกรอง, ซ้าย (QuantizedCoefficient) เป็นค่าสัมประสิทธิ์ที่แบ่งระดับให้เหลือขนาดเป็น 8 บิต, ขวา (Reference Coefficient) เป็นค่าสัมประสิทธิ์ที่ยังไม่ได้แบ่งระดับ



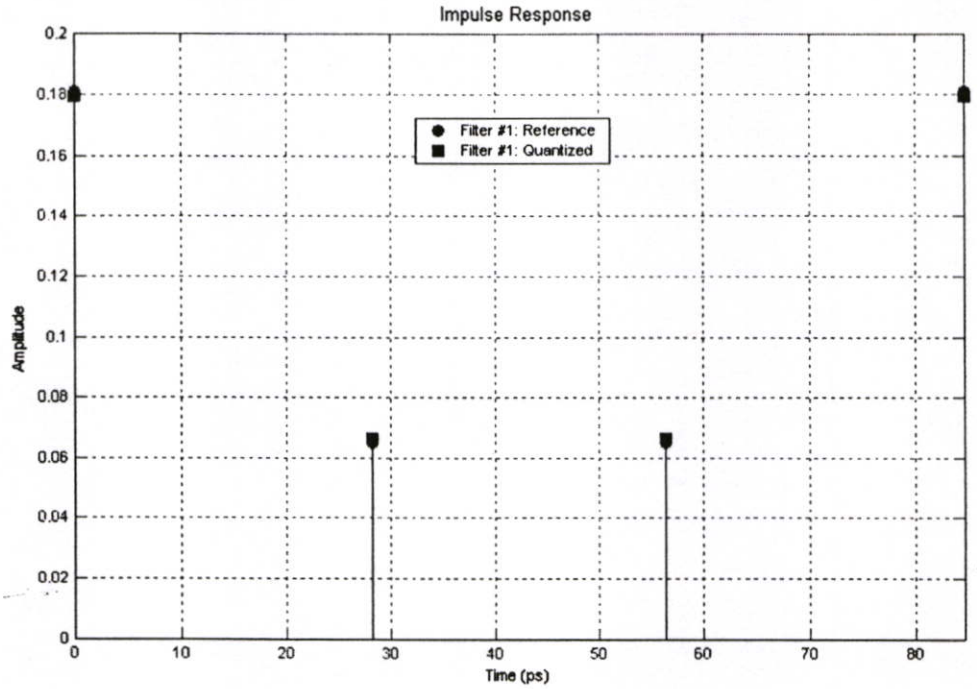
รูปที่ 5.36 แสดงผลตอบสนองความถี่ของตัวกรองภาค U_LPF3TAB_FAC



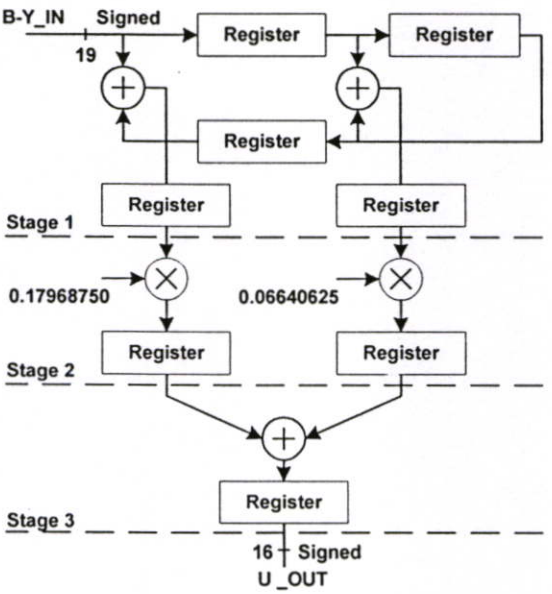
รูปที่ 5.37 แสดงผลตอบสนองเฟสของตัวกรองภาค U_LPF3TAB_FAC



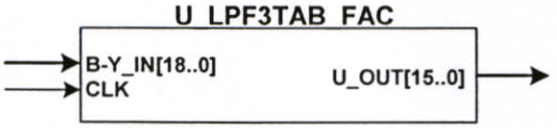
รูปที่ 5.38 แสดงผลตอบสนองความถี่เทียบกับผลตอบสนองเฟสของตัวกรองภาค U_LPF3TAB_FAC



รูปที่ 5.39 แสดงผลตอบสนองอิมพัลส์ของตัวกรองภาค U_LPF3TAB_FAC



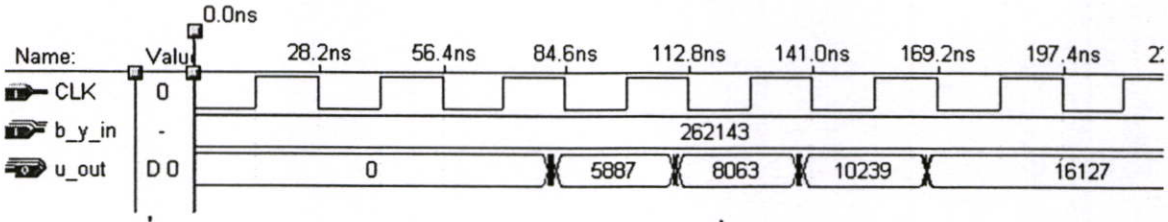
รูปที่ 5.40 แสดงโครงสร้างในระดับ RTL ของตัวกรองภาค U_LPF3TAB_FAC



รูปที่ 5.41 แสดงมุมมองการเชื่อมต่อของภาค U_LPF3TAB_FAC

รหัสต้นฉบับ (Source Code) ของภาคนี้ชื่อ U_LPF3TAB_FAC.VHD หน้าที่ของแต่ละสัญญาณมีดังนี้

- B-Y_IN สัญญาณขาเข้าขนาด 19 บิตชนิดมีเครื่องหมาย เป็นสัญญาณความแตกต่างสีที่ส่งมาจากภาค COLOR_SPACE
- U_OUT สัญญาณขาออกขนาด 16 บิตชนิดมีเครื่องหมาย เป็นสัญญาณความแตกต่างสีที่ถูกกรองความถี่ต่ำผ่านและลดขนาดลงโดยการคูณด้วย 0.493



รูปที่ 5.42 แสดงผลการจำลองการทำงานในระดับไทม์มิ่งของภาค U_LPF3TAB_FAC

5.5.8 ภาค V_LPF3TAB_FAC

ทำหน้าที่และใช้วิธีการออกแบบเหมือนกับภาค U_LPF3TAB_FAC ต่างกันที่อัตราการลดทอนเนื่องจากในการสร้างสัญญาณ V จะต้องลดทอนสัญญาณ R-Y ลง 0.877 เท่า [5] คุณสมบัติพื้นฐานของตัวกรองมีดังนี้

อันดับของตัวกรอง = 3 ได้จากการทดลองและการออกแบบซ้ำ (Iteration) เพื่อให้ได้คุณสมบัติใกล้เคียงกับความต้องการและประหยัดทรัพยากร

ความกว้างของสัญญาณขาเข้า:X(n) = 19 Bits. เนื่องจากความกว้างสัญญาณขาออกจากภาคก่อนหน้า (COLOR_SPACE) มีขนาด 19 บิต

ความกว้างของสัญญาณขาออก:Y(n) = 16 Bits. ตัดปลาย (Truncation) เพื่อลดการใช้ทรัพยากรของภาคต่อไป (UVMOD)

Passband edge ripple:ωp = 6 MHz. ได้จากตัวกรองเชิงอุปมานจากระบบที่ใช้เป็นต้นแบบทดลองและออกแบบซ้ำ (Iteration) เพื่อให้ได้คุณสมบัติใกล้เคียงกับความต้องการและประหยัดทรัพยากร

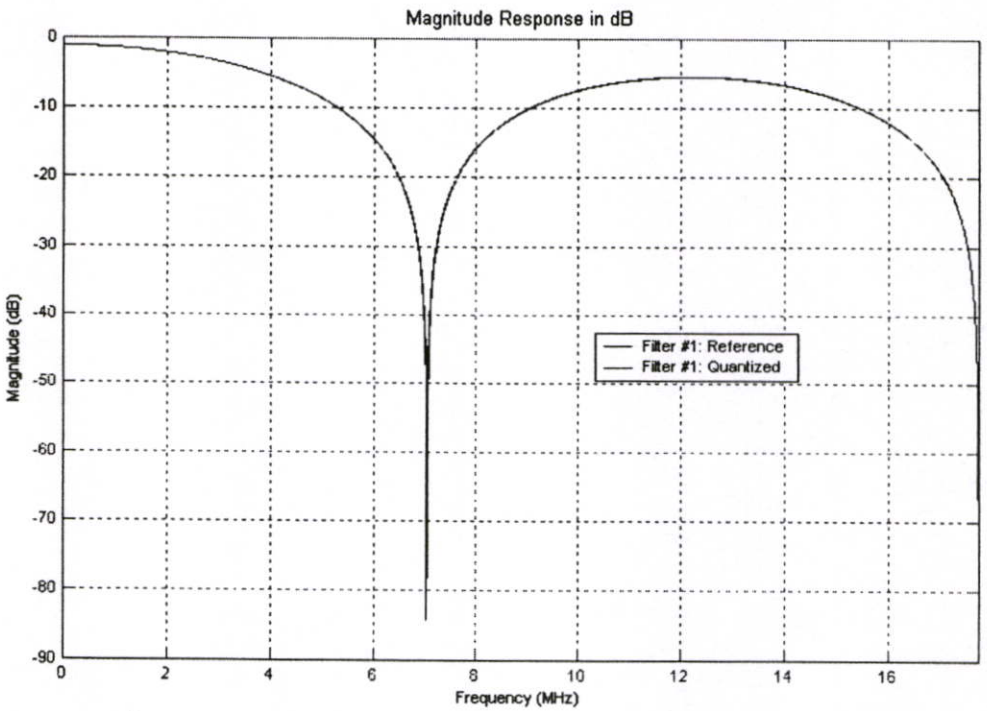
Stopband edge ripple:ωs = 6.25 MHz. ได้จากตัวกรองเชิงอุปมานจากระบบที่ใช้เป็นต้นแบบ ทดลองและออกแบบซ้ำ (Iteration) เพื่อให้ได้คุณสมบัติใกล้เคียงกับความต้องการและประหยัดทรัพยากร

h0 = h3 = 0.3203125 h1 = h2 = 0.1171875

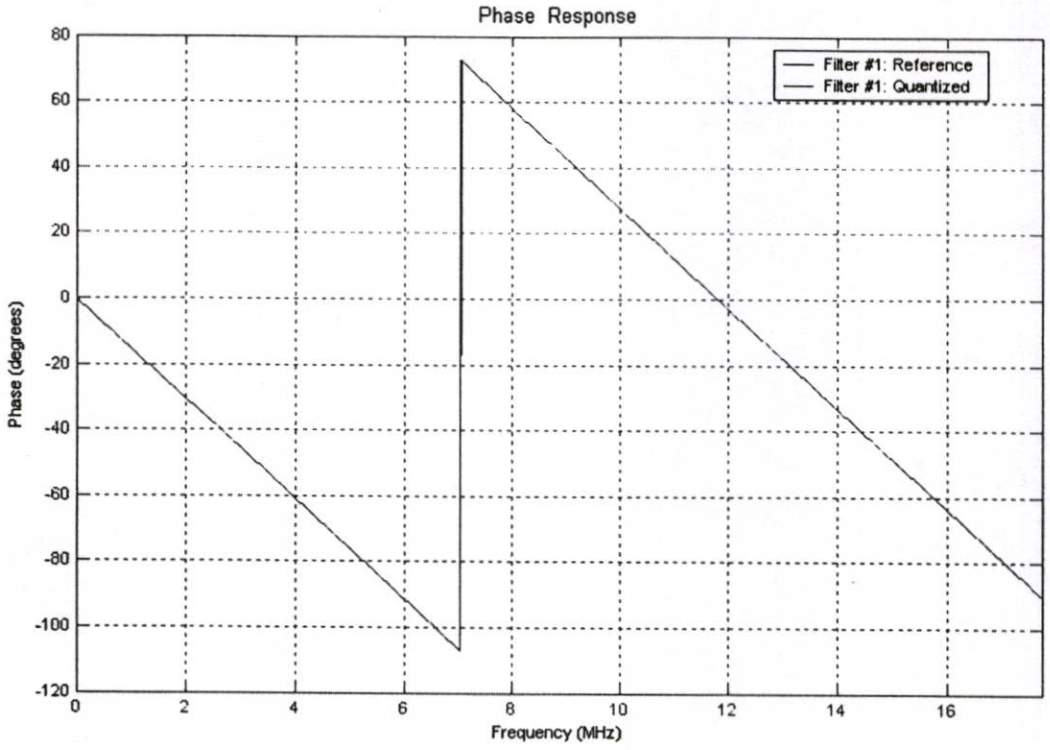
```
Quantized Direct form FIR filter
Numerator
    QuantizedCoefficients{1}    ReferenceCoefficients{1}
 (1)          0.32031250  0.321697841588329200
 (2)          0.11718750  0.116093797047549460
 (3)          0.11718750  0.116093797047549460
 (4)          0.32031250  0.321697841588329200

  FilterStructure = fir
     ScaleValues = []
 NumberOfSections = 1
 StatesPerSection = [3]
CoefficientFormat = quantizer('fixed', 'round', 'saturate', [8  8])
     InputFormat = quantizer('fixed', 'floor', 'saturate', [16  15])
    OutputFormat = quantizer('fixed', 'floor', 'saturate', [16  15])
MultiplicandFormat = quantizer('fixed', 'floor', 'saturate', [16  15])
    ProductFormat = quantizer('fixed', 'floor', 'saturate', [32  30])
       SumFormat = quantizer('fixed', 'floor', 'saturate', [32  30])
```

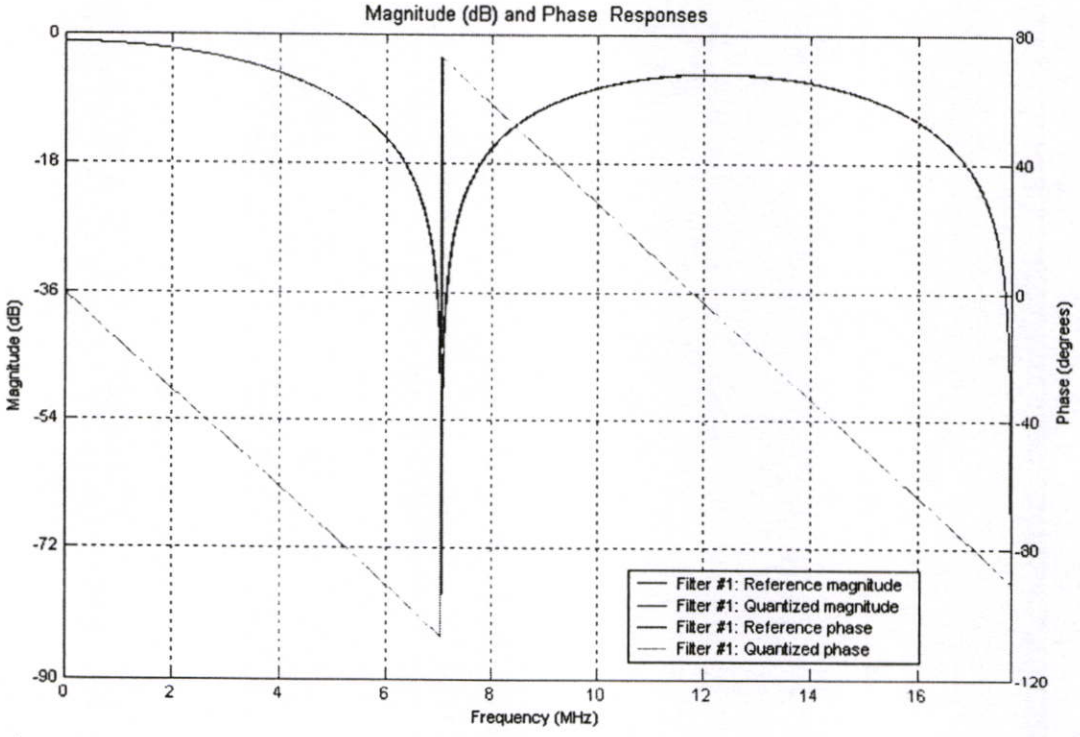
รูปที่ 5.43 แสดงค่าสัมประสิทธิ์ของตัวกรอง, ซ้าย (QuantizedCoefficient) เป็นค่าสัมประสิทธิ์ที่แบ่งระดับให้เหลือขนาดเป็น 8 บิต, ขวา (Reference Coefficient) เป็นค่าสัมประสิทธิ์ที่ยังไม่ได้แบ่งระดับ



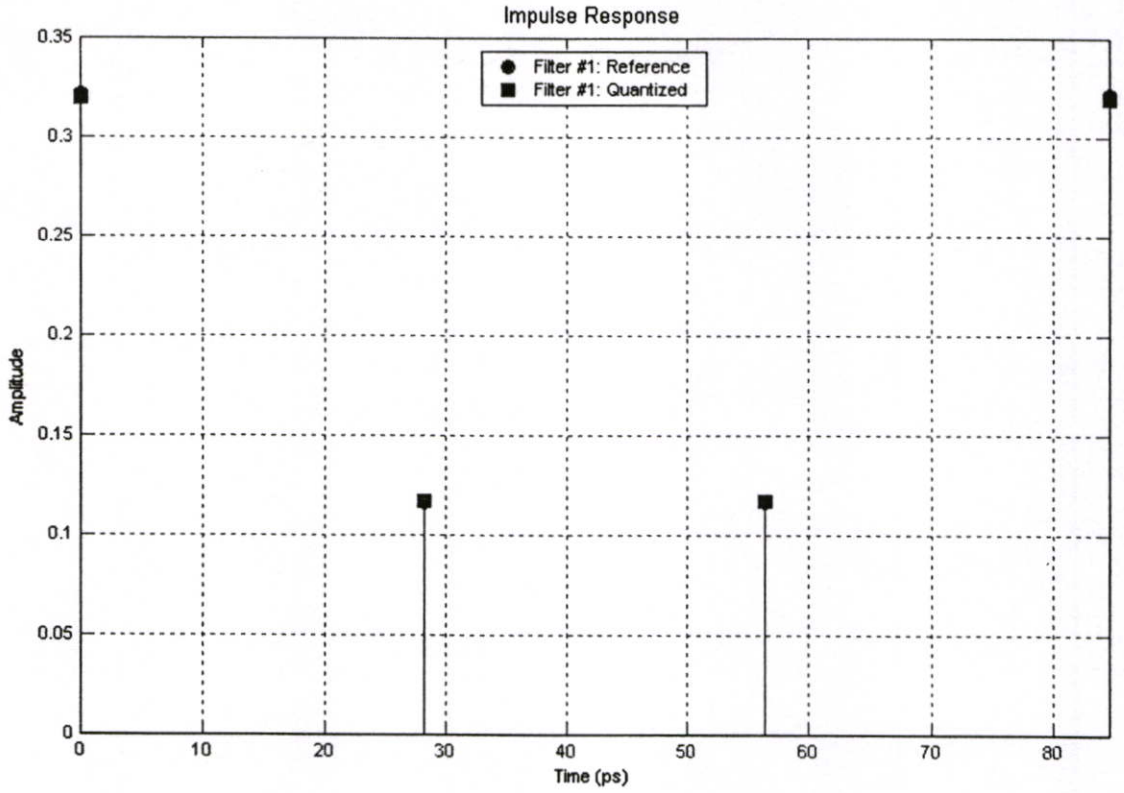
รูปที่ 5.44 แสดงผลตอบสนองความถี่ของตัวกรองภาค V_LPF3TAB_FAC



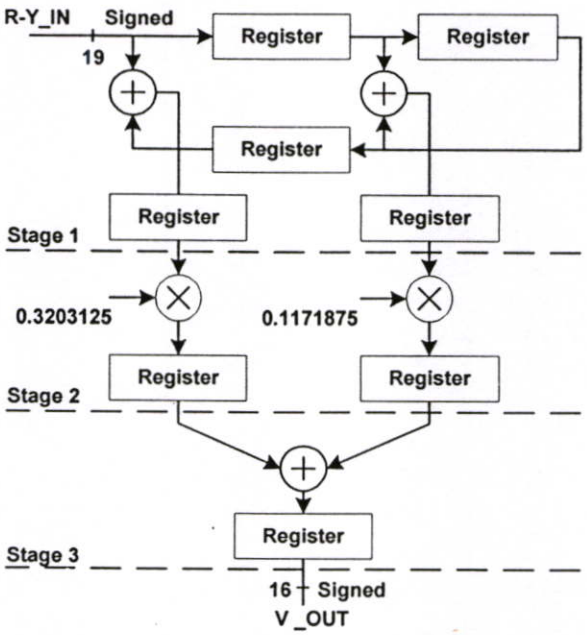
รูปที่ 5.45 แสดงผลตอบสนองเฟสของตัวกรองภาค V_LPF3TAB_FAC



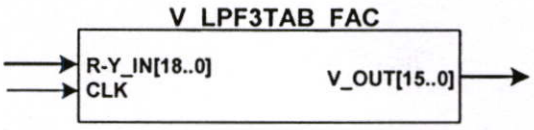
รูปที่ 5.46 แสดงผลตอบสนองความถี่เทียบกับผลตอบสนองเฟสของตัวกรองภาค V_LPF3TAB_FAC



รูปที่ 5.47 แสดงผลตอบสนองอิมพัลส์ของตัวกรองภาค V_LPF3TAB_FAC



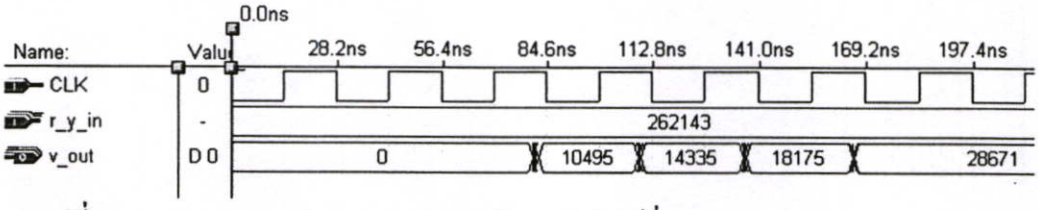
รูปที่ 5.48 แสดงโครงสร้างในระดับ RTL ของตัวกรองภาค V_LPF3TAB_FAC



รูปที่ 5.49 แสดงมุมมองการเชื่อมต่อของภาค V_LPF3TAB_FAC

รหัสต้นฉบับ (Source Code) ของภาคนี้ชื่อ V_LPF3TAB_FAC.VHD หน้าที่ของแต่ละสัญญาณมีดังนี้

- R-Y_IN สัญญาณขาเข้าขนาด 19 บิตชนิดมีเครื่องหมาย เป็นสัญญาณความแตกต่างสีที่ส่งมาจากภาค COLOR_SPACE
- V_OUT สัญญาณขาออกขนาด 16 บิตชนิดมีเครื่องหมาย เป็นสัญญาณความแตกต่างสีที่ถูกกรองความถี่ต่ำผ่านและลดขนาดลงโดยการคูณด้วย 0.877



รูปที่ 5.50 แสดงผลการจำลองการทำงานในระดับไทม์มิ่งของภาค U_LPF3TAB_FAC

5.5.9 ภาค UVMOD

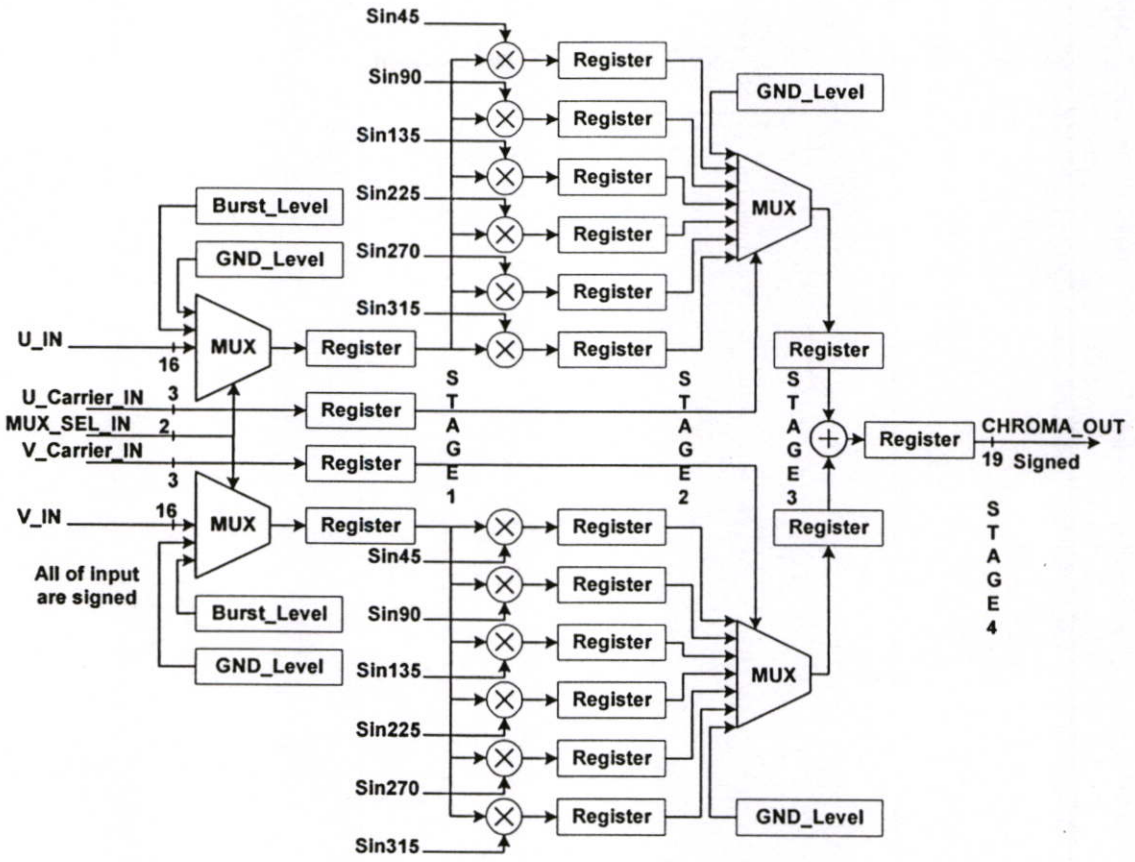
ทำหน้าที่มอดูเลตสัญญาณความแตกต่างสี (U,V) เชิงแอมปลิจูดแบบแถบข้างคู่ขจัดคลื่นพาห์ (AMDSBSC) และสอดแทรกสัญญาณเบิรสต์เพื่อสร้างเป็นสัญญาณสี (Chrominance: C) ตามมาตรฐานของระบบพาลซึ่งมีรูปแบบการคำนวณตามสมการที่ (5.4) ออกแบบโดยใช้สถาปัตยกรรมแบบไปป์ไลน์ขนาด 4 สเตจ รูปที่ 5.51 แสดงโครงสร้างในระดับ RTL

$$C(n) = U(n)Sin\ \frac{\pi n}{4} \pm V(n)Cos\ \frac{\pi n}{4} \qquad (5.4)$$

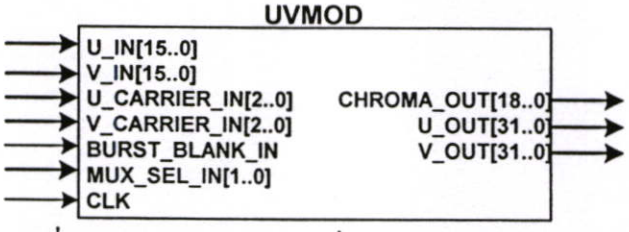
เมื่อ

$C(n)$ คือ สัญญาณเชิงเลขของสัญญาณสี

$U(n)$ และ $V(n)$ คือ สัญญาณเชิงเลขของสัญญาณความแตกต่างสี



รูปที่ 5.51 แสดงโครงสร้างในระดับ RTL ของภาค UVMOD



รูปที่ 5.52 แสดงมุมมองการเชื่อมต่อของภาค UVMOD

รหัสต้นฉบับ (Source Code) ของภาคนี้ชื่อ UVMOD.VHD หน้าที่ของแต่ละสัญญาณมีดังนี้

- U_IN และ V_IN สัญญาณขาเข้าขนาด 16 บิตชนิดมีเครื่องหมาย เป็นสัญญาณความแตกต่างสี ส่งมาจากภาค U_LPF3TAB_FAC และ V_LPF3TAB_FAC ตามลำดับ เพื่อทำการมอดูเลตเชิงแอมปลิจูดแบบแถบข้างคู่ขจัดคลื่นพาห์

- U_CARRIER_IN และ V_CARRIER_IN สัญญาณขาเข้าขนาด 16 บิต เป็นสัญญาณชี้เฟสของสัญญาณพาหะรองของสัญญาณความแตกต่างสี U และ V ตามลำดับ สัญญาณนี้ถูกส่งมาจากภาค NCO_MUX

- BURST_BLANK_IN สัญญาณขาเข้าขนาด 1 บิต ทำหน้าที่ควบคุมการสอดแทรกสัญญาณเบิรสต์ในช่วง Vertical Blanking

- MUX_SEL_IN สัญญาณขาเข้าขนาด 2 บิต ทำหน้าที่ควบคุมการมัลติเพล็กสัญญาณ U_IN, V_IN และ ค่าคงที่ Burst_Level และ GND_Level ก่อนนำไปคูณกับค่าคงที่ไซน์เพื่อสร้างสัญญาณสี

- CHROMA_OUT สัญญาณขาออกขนาด 19 บิต เป็นสัญญาณสี (Chrominance: C) ซึ่งประกอบด้วย สัญญาณเบิรสต์และสัญญาณความแตกต่างสีที่ผ่านการมอดูเลตแล้ว ความกว้างของสัญญาณถูกตัดปลาย (Truncation) จาก 32 บิตให้เหลือ 19 บิตเพื่อให้มีความละเอียด (Resolution) เท่ากับสัญญาณภาพเบ็ดเสร็จก่อนการกรองสัญญาณความถี่ต่ำผ่าน

- U_OUT และ V_OUT เป็นสัญญาณที่ไม่ได้ใช้งานแต่ใช้ในการทดสอบการทำงานของระบบ

จากรูปที่ 5.51 จะสังเกตเห็นว่าได้ว่าระบบการประมวลผลสามารถแยกออกเป็นสองส่วน ได้แก่ ส่วนบนเป็นส่วนของการมอดูเลตสัญญาณ U และ ส่วนล่างเป็นส่วนของการมอดูเลตสัญญาณ V ในการทำไปป์ไลน์มีลำดับขั้นการคำนวณของแต่ละสเตจดังนี้

- Stage 1 สัญญาณ MUX_SEL_IN ถูกใช้ในการควบคุมการมัลติเพล็กระหว่างสัญญาณ U_IN, V_IN และค่าคงที่ Burst_Level, GND_Level ทั้งสี่สัญญาณนี้เป็นค่าของตัวคูณซึ่งจะถูกส่งไปเก็บในไปป์ไลน์รีจีสเตอร์ก่อนส่งให้สเตจต่อไป อีกทางหนึ่งสัญญาณชี้เฟส U_CARRIER_IN และ V_CARRIER_IN จะถูกส่งไปเก็บในไปป์ไลน์รีจีสเตอร์เพื่อชดเชยคาบเวลาการล้าหลัง (Latency)

- Stage 2 สัญญาณซึ่งเป็นค่าตัวคูณจะถูกส่งไปคูณ (แบบขนาน) กับค่าคงที่เฉพาะจุดของสัญญาณไซน์ด้วยตัวคูณแบบตารางเปิดดู (Look Up Table Multiplier) เนื่องจากการใช้ความถี่สุ่มเป็น 8 เท่าของสัญญาณพาหะรองของสัญญาณสี ดังนั้นมันจึงมีข้อมูลทั้งหมด 8 จุดต่อ 1 คาบของสัญญาณพาหะรองของสัญญาณสี แต่ตัวคูณจะมีแค่ 6 ชุดเนื่องจากไซน์ที่มุม 0° และ 180° มีค่าเท่ากับ 0 ดังนั้นจึงสามารถลดตัวคูณได้ทั้งหมด 4 ตัว และใช้การมัลติเพล็กค่าคงที่ GND_Level เพื่อให้ค่า 0 แทนการคูณในสเตจต่อไป

- Stage 3 สัญญาณที่ส่งมาจากตัวคูณและค่าคงที่ GND_Level ซึ่งเป็นตัวแทนสัญญาณไซน์ที่มุม 0° และ 180° จะถูกมัลติเพล็กด้วยการควบคุมของสัญญาณชี้เฟส U_CARRIER_IN และ

V_CARRIER_IN ที่จุดนี้สัญญาณได้ผ่านการมอดูเลตแล้วและยังมีสัญญาณเบิรสต์แทรกเข้ามาเป็นช่วงตามการควบคุมของสัญญาณ MUX_SEL_IN

- Stage 4 สัญญาณ U และ V ที่ผ่านการมอดูเลตและสอดแทรกสัญญาณเบิรสต์แล้วจะถูกนำมาบวกกันเพื่อสร้างเป็นสัญญาณสี (Chrominance: C) ที่สมบูรณ์

5.5.10 ภาค C_LPF3TAB

เป็นตัวกรองไม่ป้อนกลับเชิงเลขทำหน้าที่กรองความถี่ต่ำผ่านเพื่อลดฮาร์โมนิคส์ที่เกิดจากการมอดูเลตของภาค UVMOD [5] ออกแบบโดยใช้วิธีการเดียวกันกับภาค CVS_LPF3TAB คุณสมบัติพื้นฐานของตัวกรองมีดังนี้

อันดับของตัวกรอง = 3 ได้จากการทดลองและการออกแบบซ้ำ (Iteration) เพื่อให้ได้คุณสมบัติใกล้เคียงกับความต้องการและประหยัดทรัพยากร

ความกว้างสัมประสิทธิ์ = 8 Bits. ได้จากการทดลองและการออกแบบซ้ำ (Iteration) เพื่อให้ได้คุณสมบัติใกล้เคียงกับความต้องการและประหยัดทรัพยากร โดยใช้การจำลองการทำงานแบบจำนวนโดยตรง (Fixed Point) เพื่อวิเคราะห์ค่าความผิดพลาดจากการแบ่งระดับ

ความกว้างของสัญญาณขาเข้า:x(n) = 19 Bits. เนื่องจากความกว้างสัญญาณขาออกจากภาคก่อนหน้า (UVMOD) มีขนาด 19 บิต

ความกว้างของสัญญาณขาออก:y(n) = 12 Bits. ตัดปลาย (Truncation) เพื่อให้มีขนาดเท่ากับความกว้างสัญญาณขาเข้าของตัวแปลงสัญญาณเชิงเลขเป็นสัญญาณเชิงอุปมาน (DAC)

Passband edge ripple:ωp = 6.43361875 MHz. ได้จากตัวกรองเชิงอุปมานจากระบบที่ใช้เป็นต้นแบบ ทดลองและออกแบบซ้ำ (Iteration) เพื่อให้ได้คุณสมบัติใกล้เคียงกับความต้องการและประหยัดทรัพยากร

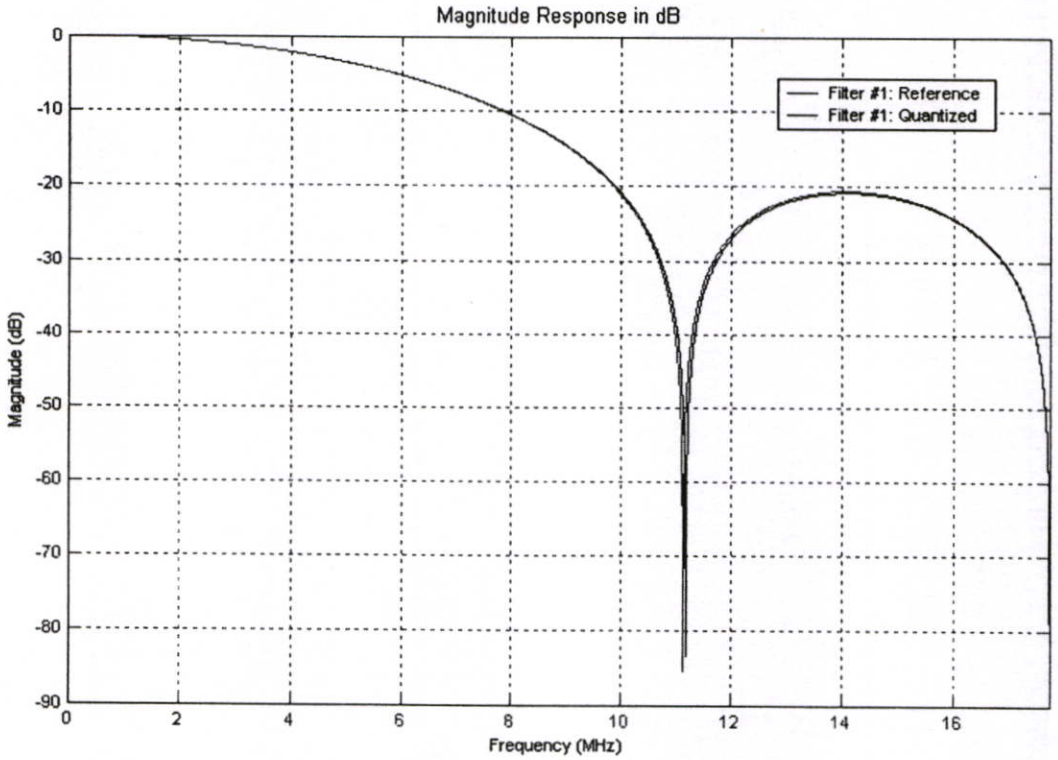
Stopband edge ripple:ωs = 10 MHz. ได้จากตัวกรองเชิงอุปมานจากระบบที่ใช้เป็นต้นแบบ ทดลองและออกแบบซ้ำ (Iteration) เพื่อให้ได้คุณสมบัติใกล้เคียงกับความต้องการและประหยัดทรัพยากร

h0 = h3 = 0.1796875 h1 = h2 = 0.3203125

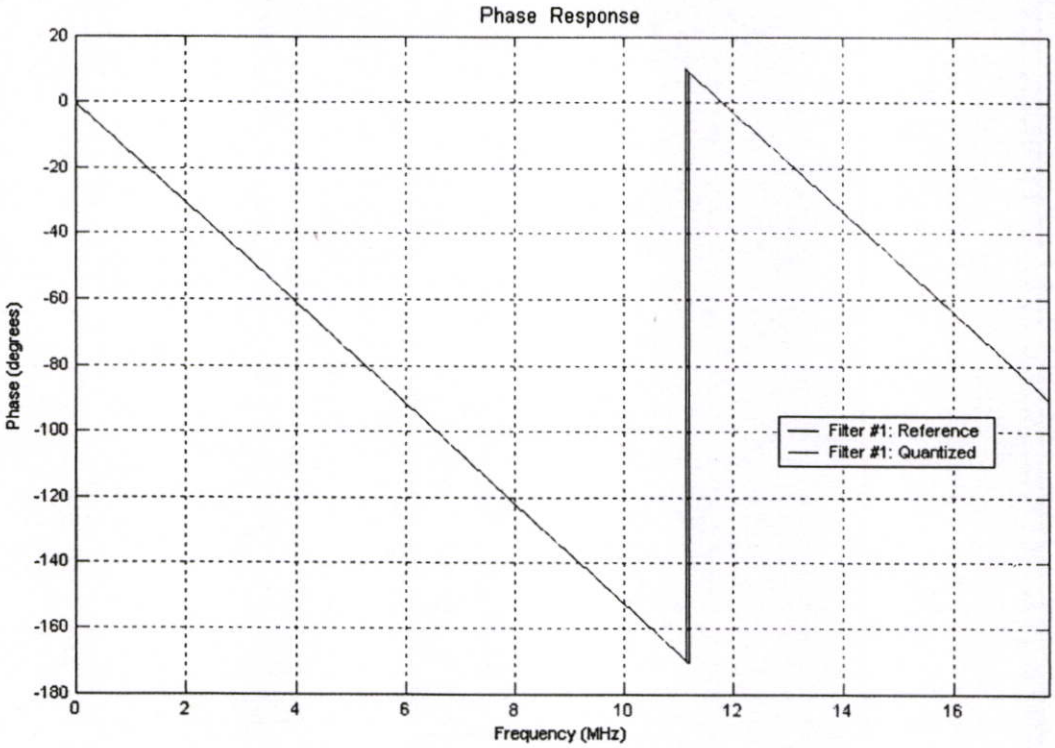
```
Quantized Direct form FIR filter
Numerator
    QuantizedCoefficients{1}    ReferenceCoefficients{1}
 (1)          0.17968750  0.178629761509659670
 (2)          0.32031250  0.321335512381658250
 (3)          0.32031250  0.321335512381658250
 (4)          0.17968750  0.178629761509659670

  FilterStructure = fir
     ScaleValues = []
 NumberOfSections = 1
 StatesPerSection = [3]
CoefficientFormat = quantizer('fixed', 'round', 'saturate', [8  8])
     InputFormat = quantizer('fixed', 'floor', 'saturate', [16  15])
    OutputFormat = quantizer('fixed', 'floor', 'saturate', [16  15])
MultiplicandFormat = quantizer('fixed', 'floor', 'saturate', [16  15])
    ProductFormat = quantizer('fixed', 'floor', 'saturate', [32  30])
       SumFormat = quantizer('fixed', 'floor', 'saturate', [32  30])
```

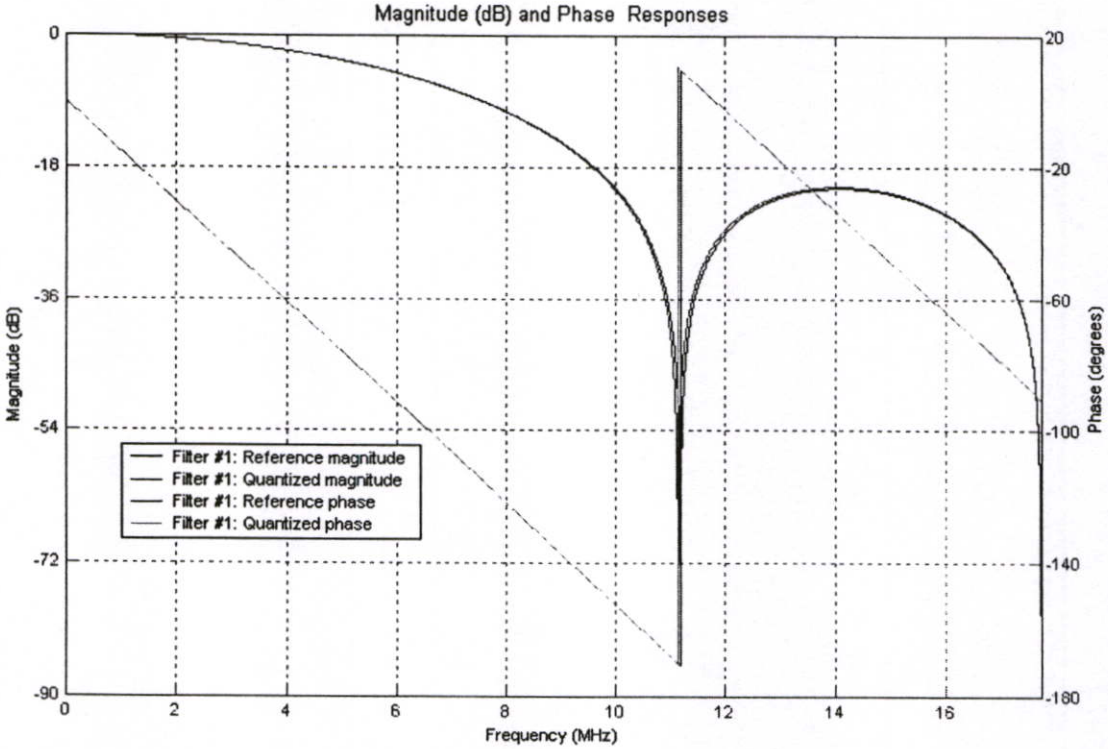
รูปที่ 5.53 แสดงค่าสัมประสิทธิ์ของตัวกรอง, ซ้าย (Quantized Coefficient) เป็นค่าสัมประสิทธิ์ที่แบ่งระดับให้เหลือขนาดเป็น 8 บิต, ขวา (Reference Coefficient) เป็นค่าสัมประสิทธิ์ที่ยังไม่ได้แบ่งระดับ



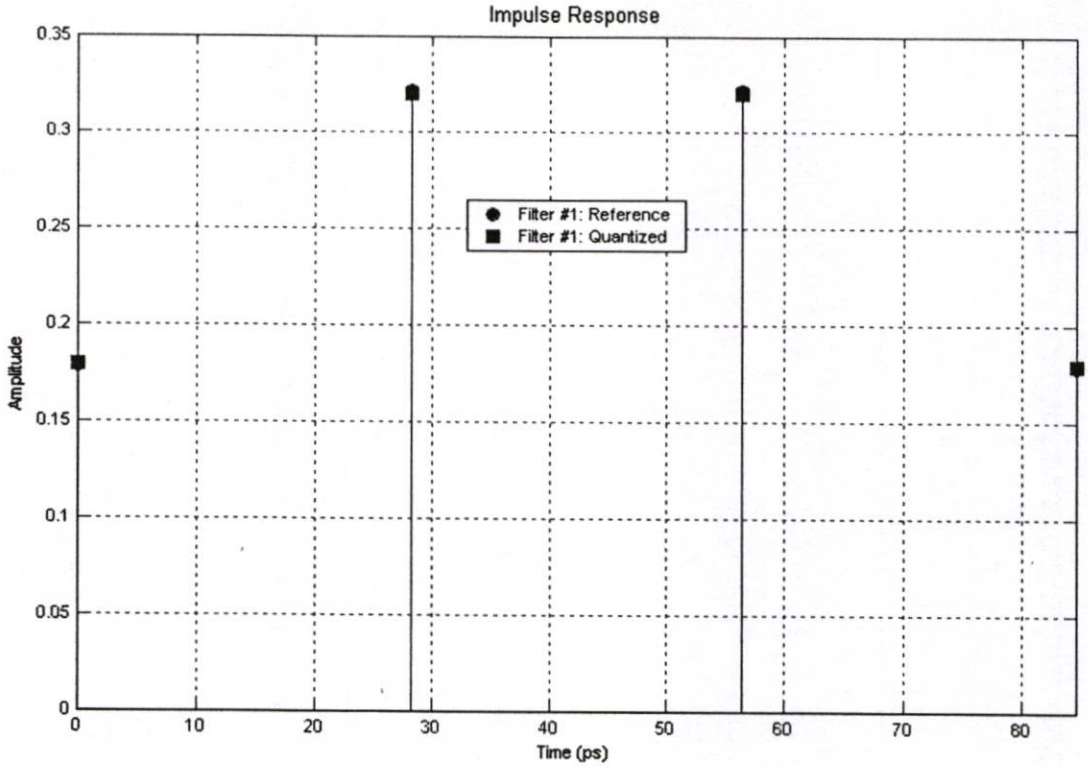
รูปที่ 5.54 แสดงผลตอบสนองความถี่ของตัวกรองภาค C_LPF3TAB



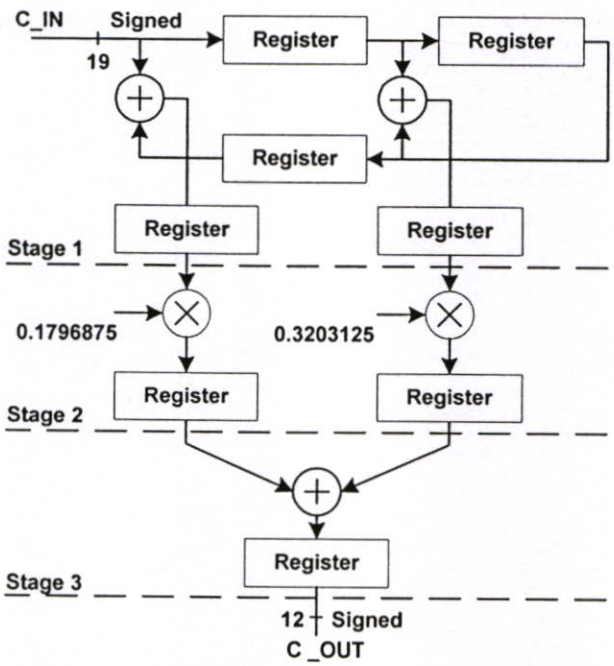
รูปที่ 5.55 แสดงผลตอบสนองเฟสของตัวกรองภาค C_LPF3TAB



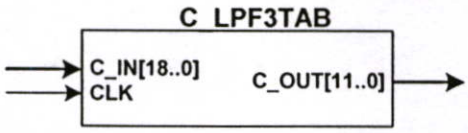
รูปที่ 5.56 แสดงผลตอบสนองความถี่เทียบกับผลตอบสนองเฟสของตัวกรองภาค C_LPF3TAB



รูปที่ 5.57 แสดงผลตอบสนองอิมพัลส์ของตัวกรองภาค C_LPF3TAB



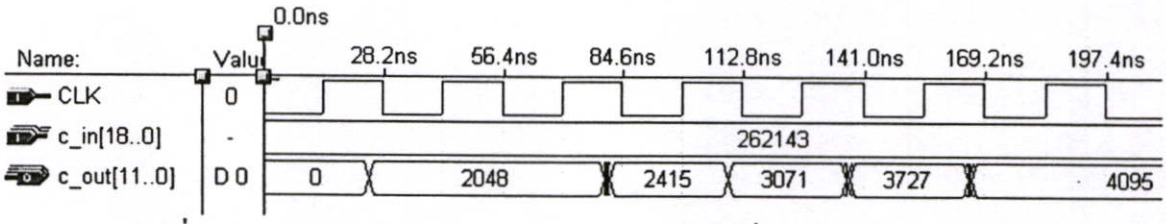
รูปที่ 5.58 แสดงโครงสร้างในระดับ RTL ของตัวกรองภาค C_LPF3TAB



รูปที่ 5.59 แสดงมุมมองการเชื่อมต่อของภาค C_LPF3TAB

รหัสต้นฉบับ (Source Code) ของภาคนี้ชื่อ C_LPF3TAB.VHD หน้าที่ของแต่ละสัญญาณมีดังนี้

- C_IN สัญญาณขาเข้าขนาด 19 บิตแบบ 2'Complement Fractional เป็นสัญญาณสีที่ส่งมาจากภาค UVMOD
- C_OUT สัญญาณขาออกขนาด 12 บิตแบบ Offset 2'Complement เป็นสัญญาณสีขาออกของระบบ



รูปที่ 5.60 แสดงผลการจำลองการทำงานในระดับไทม์มิ่งของภาค C_LPF3TAB

บทที่ 6

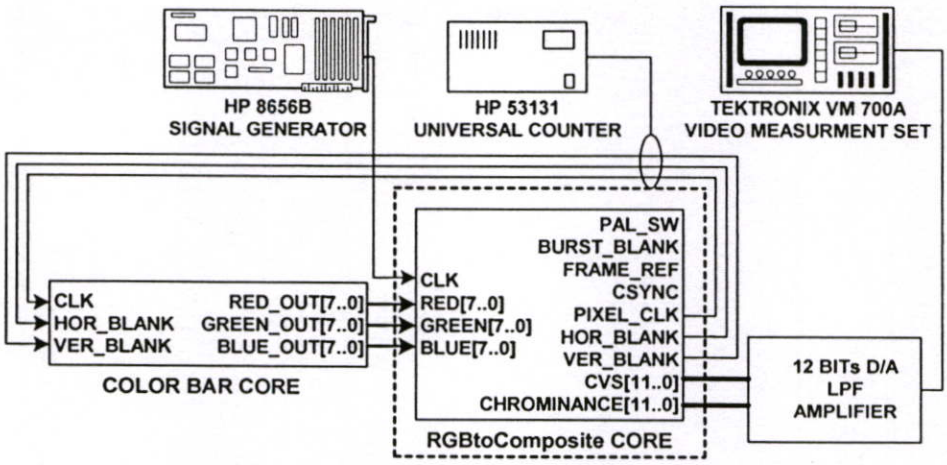
ผลการทดลอง

6.1 การทดสอบระบบซอฟท์แวร์

ในขั้นตอนของการออกแบบอัลกอริธึม (Algorithm) และการโมเดลวีเอชดีแอลได้ใช้โปรแกรมจำลองการทำงานเป็นเครื่องมือในการทดสอบคุณสมบัติของระบบซอฟท์แวร์ ตัวอย่างการจำลองการทำงานในระดับของอัลกอริธึม เช่น การจำลองการทำงานเปรียบเทียบระหว่างการใช้ระบบตัวเลขในการคำนวณแบบจำนวนโดยตรง (Fixed Point) กับการใช้ระบบตัวเลขในการคำนวณแบบจำนวนอิงดัชนี (Floating Point) ทำให้สามารถวิเคราะห์ผลของความคลาดเคลื่อนเพื่อหาจุดที่ดีที่สุดของการออกแบบ ดังจะเห็นได้จากการจำลองการทำงานเพื่อวิเคราะห์ผลของการใช้ระบบตัวเลขในการคำนวณแบบจำนวนโดยตรงกับตัวกรองไม่ป้อนกลับเชิงเลขซึ่งใช้สัมประสิทธิ์ขนาด 8 บิต ผลของความคลาดเคลื่อนมีค่าน้อยและอยู่ในเกณฑ์ที่ยอมรับได้อันเนื่องจากตัวกรองที่ออกแบบมีอันดับต่ำ

6.2 การทดสอบระบบฮาร์ดแวร์

หลังจากที่ได้ทดสอบความถูกต้องของการออกแบบด้วยการจำลองการทำงาน ขั้นตอนต่อไปคือ การสร้างระบบฮาร์ดแวร์และทดสอบคุณสมบัติของสัญญาณว่าอยู่ในเกณฑ์ที่ยอมรับได้หรือไม่ โดยใช้เกณฑ์ในการพิจารณาคุณสมบัติของสัญญาณตามมาตรฐานเดียวกันกับเครื่องมือวัดทดสอบซึ่งเป็นที่ยอมรับในงานอุตสาหกรรม [7], [18] ในการทดสอบได้ใช้เครื่องมือวัดซึ่งมีรูปแบบการเชื่อมต่อตามรูปที่ 6.1 ตัวเข้ารหัสฯที่นำเสนอ (ในกรอบ) ถูกทดสอบด้วยการป้อนสัญญาณภาพแท่งสี 100% ที่สร้างจากการโมเดลวีเอชดีแอลโดยรับข้อมูลและสัญญาณควบคุมมาจากตัวเข้ารหัสฯที่เตรียมไว้สำหรับการสร้างสัญญาณจุดภาพ (Pixel) โดยเฉพาะทำให้ในการใช้งานจริงสามารถสร้างสัญญาณภาพได้หลายรูปแบบจากสัญญาณที่เตรียมไว้ดังกล่าว ในการสังเคราะห์วงจรได้เลือกใช้ชิปเอฟพีจีเอของบริษัท Altera ตระกูล Acex เบอร์ EP1K50TC144-3 ผลการทดลองที่ได้สามารถแบ่งออกได้เป็นสามกลุ่มตามหน่วยการวัดได้แก่ ขนาด เฟสและไทม์มิ่งดังแสดงดังรูปที่ 6.3 – 6.12



รูปที่ 6.1 แสดงรูปแบบการเชื่อมต่อเครื่องมือวัดที่ใช้ในการทดสอบคุณสมบัติของระบบฮาร์ดแวร์

```
Chip/                          Input Output Bidir  Memory  Memory          LCs
POF       Device               Pins  Pins   Pins   Bits %  Utilized  LCs   % Utilized

rgbtocomposite_cir
      EP1K50TC144-3            25    24     0      0         0 %     2199    76 %

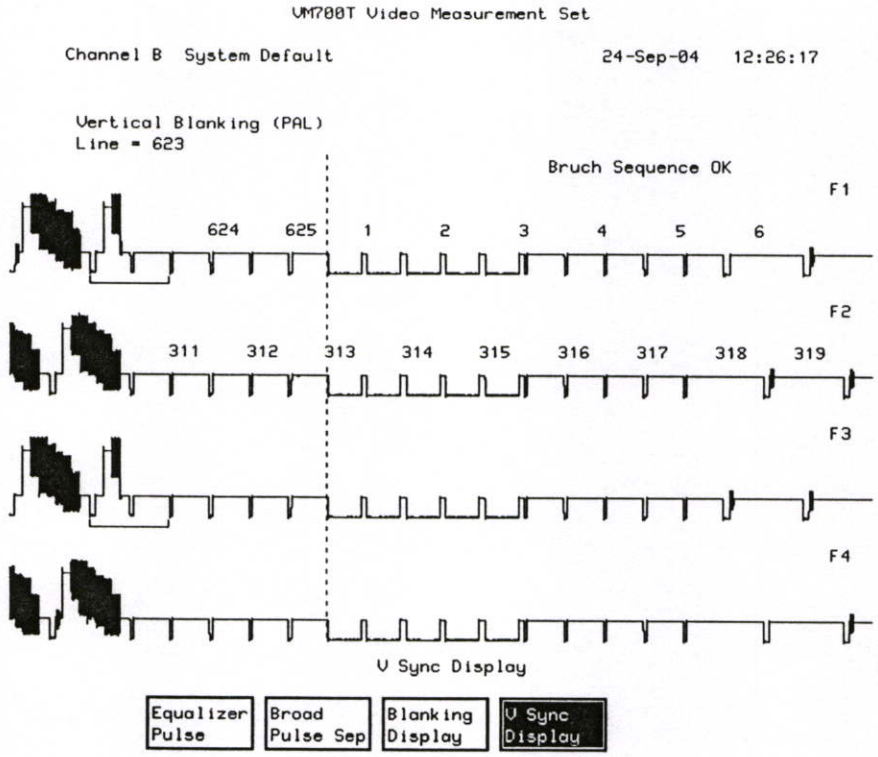
User Pins:                     25    24     0

  Total dedicated input pins used:              6/6      (100%)
  Total I/O pins used:                         43/96     ( 44%)
  Total logic cells used:                    2199/2880   ( 76%)
  Total embedded cells used:                    0/160    (  0%)
  Total EABs used:                              0/10     (  0%)
  Average fan-in:                            2.18/4      ( 54%)
  Total fan-in:                             4809/11520   ( 41%)

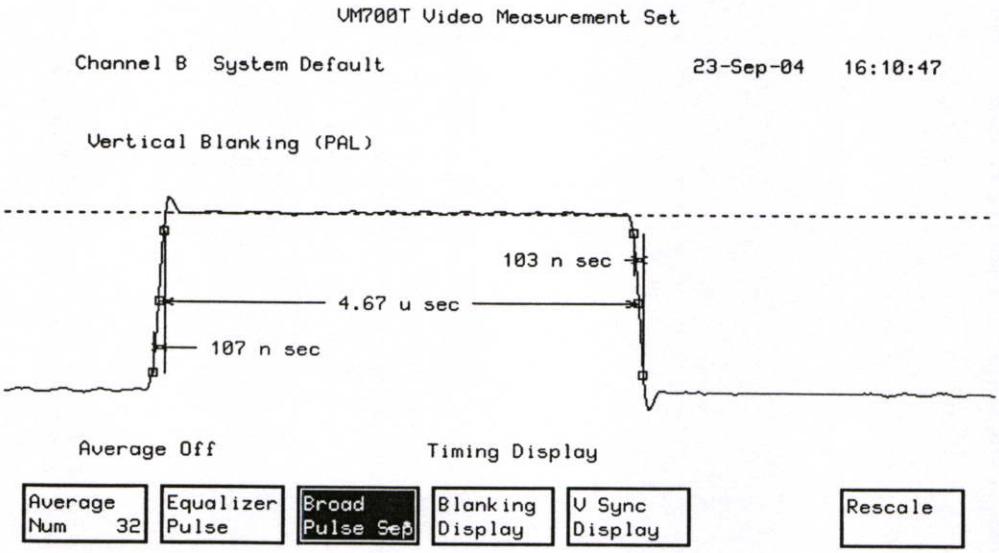
  Total input pins required:                   25
  Total input I/O cell registers required:      0
  Total output pins required:                  24
  Total output I/O cell registers required:     0
  Total buried I/O cell registers required:     0
  Total bidirectional pins required:            0
  Total reserved pins required                  0
  Total logic cells required:                2199
  Total flipflops required:                  1355
  Total packed registers required:              0
  Total logic cells in carry chains:         1053
  Total number of carry chains:                53
  Total number of carry chains of length  1-8 :  1
  Total number of carry chains of length  9-16:  7
  Total number of carry chains of length 17-24: 38
  Total number of carry chains of length 25-32:  7
  Total logic cells in cascade chains:        230
  Total number of cascade chains:             106
  Total single-pin Clock Enables required:      0
  Total single-pin Output Enables required:     0
  Logic cells inserted for fitting:             6

  Synthesized logic cells:                     12/2880   (  0%)
```

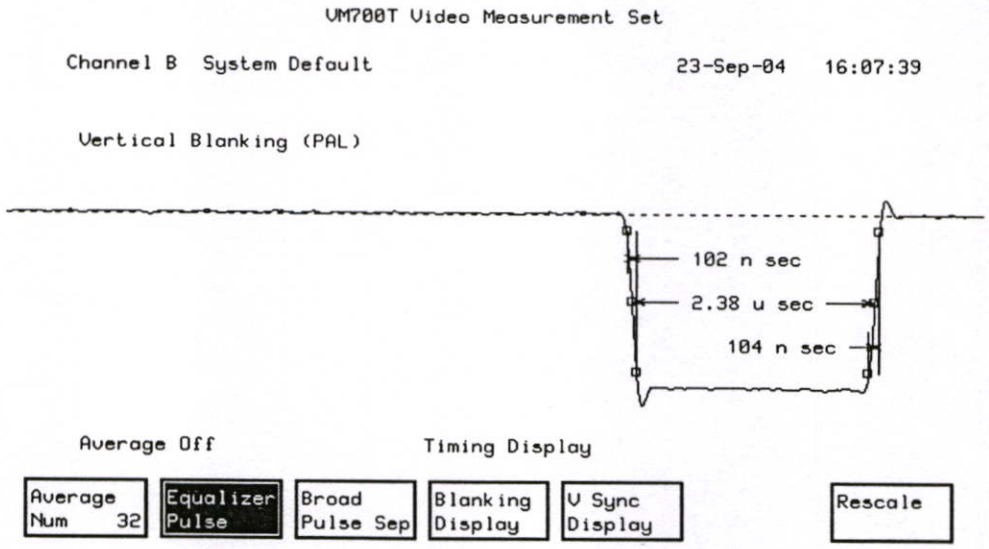
รูปที่ 6.2 แสดงผลสรุปการใช้ทรัพยากรภายในเอฟพีจีเอหลังการสังเคราะห์และเพลซแอนเร้าท์(Place & Route)



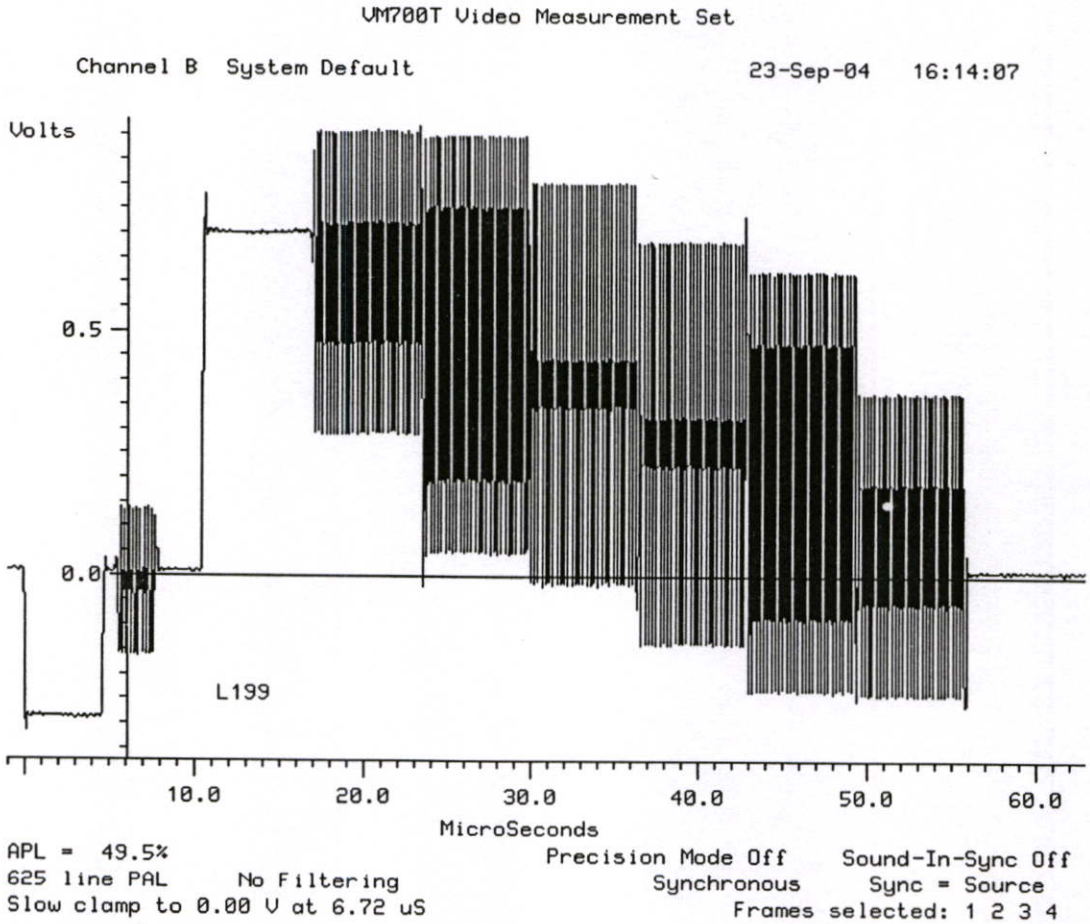
รูปที่ 6.3 แสดงสัญญาณในช่วงของการกวาดเส้นภาพกลับทางแนวตั้ง



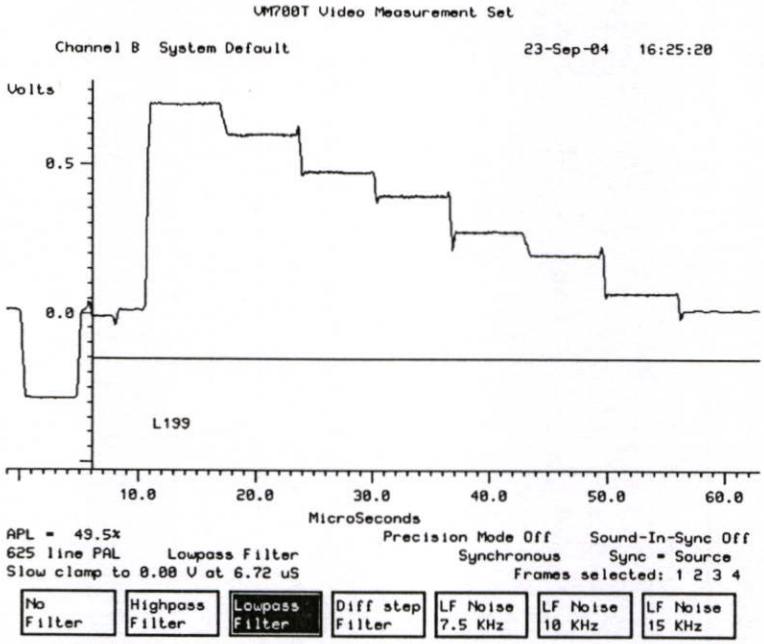
รูปที่ 6.4 แสดง Broad Pulse Separator ในช่วงของการกวาดเส้นภาพกลับทางแนวตั้ง



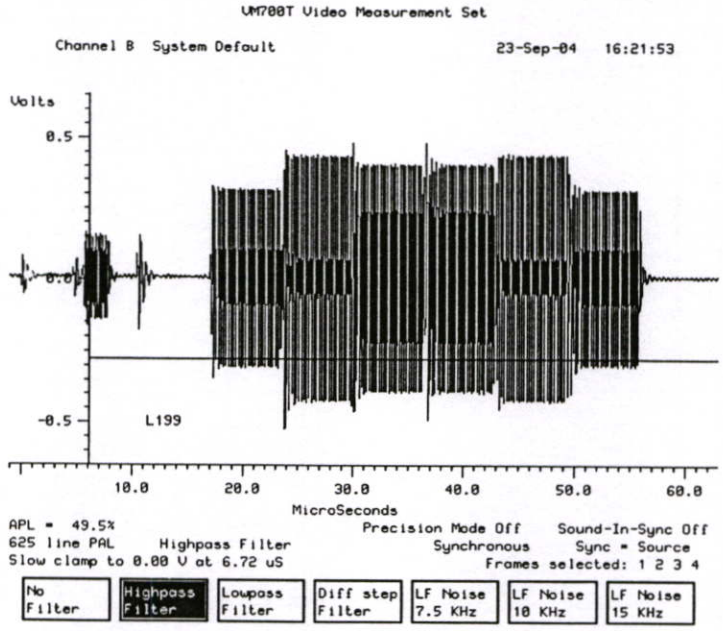
รูปที่ 6.5 แสดง Equalizer Pulse ในช่วงของการกวาดเส้นภาพกลับทางแนวตั้ง



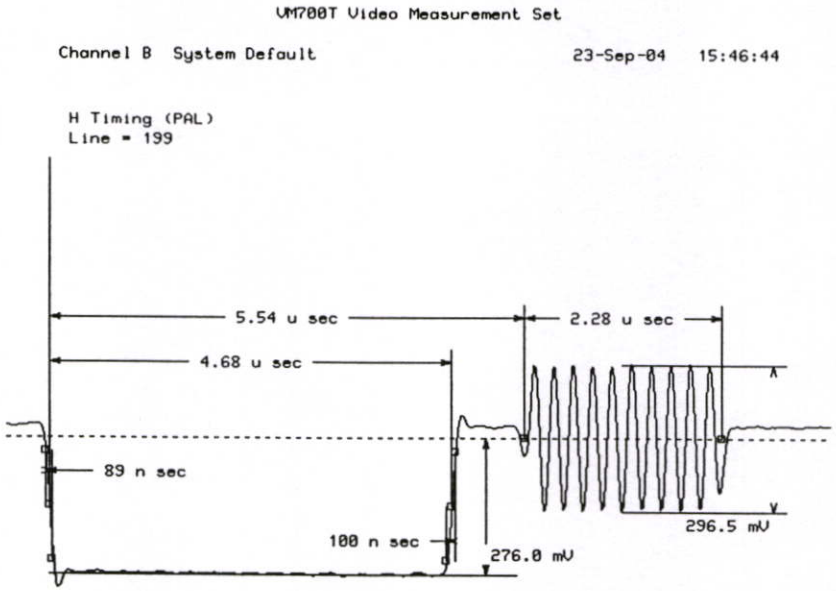
รูปที่ 6.6 แสดงสัญญาณภาพสีเบ็ดเสร็จของการกวาดเส้นภาพทางแนวนอน 1 เส้น



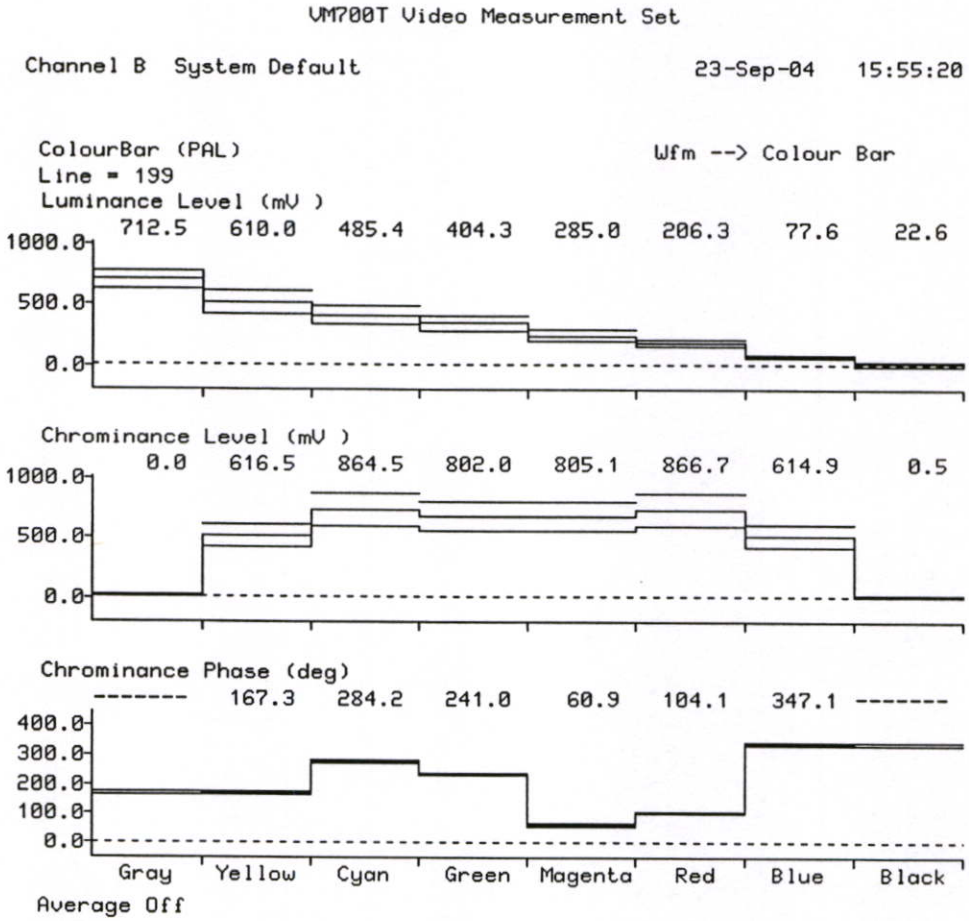
รูปที่ 6.7 แสดงสัญญาณภาพสีเบ็ดเสร็จของการกวาดเส้นภาพทางแนวนอนหลังจากผ่านวงจรกรองความถี่ต่ำผ่าน



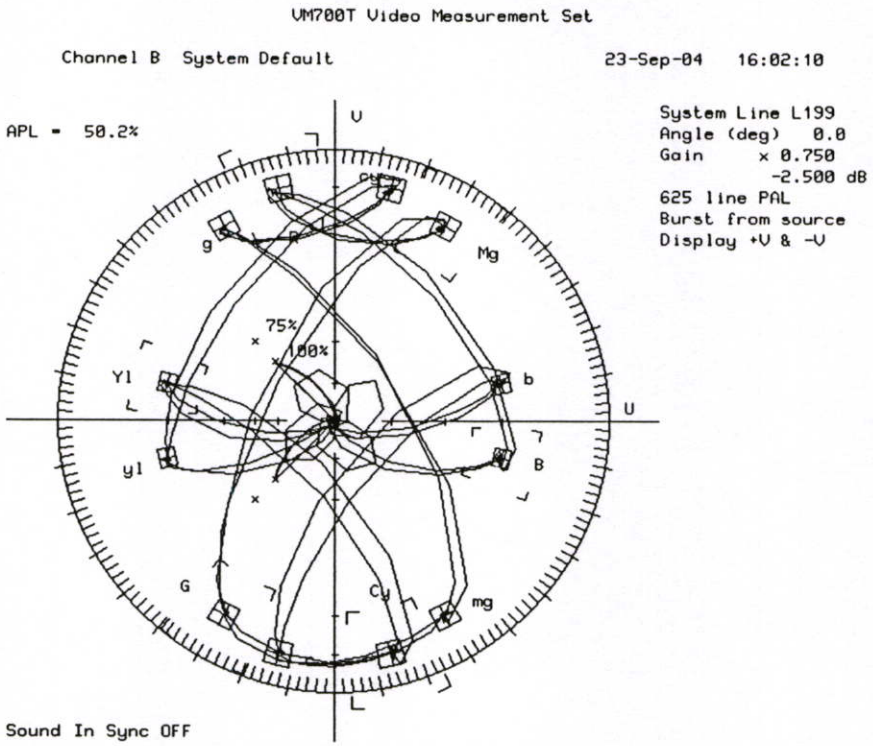
รูปที่ 6.8 แสดงสัญญาณภาพสีเบ็ดเสร็จของการกวาดเส้นภาพทางแนวนอนหลังจากผ่านวงจรกรองความถี่สูงผ่าน



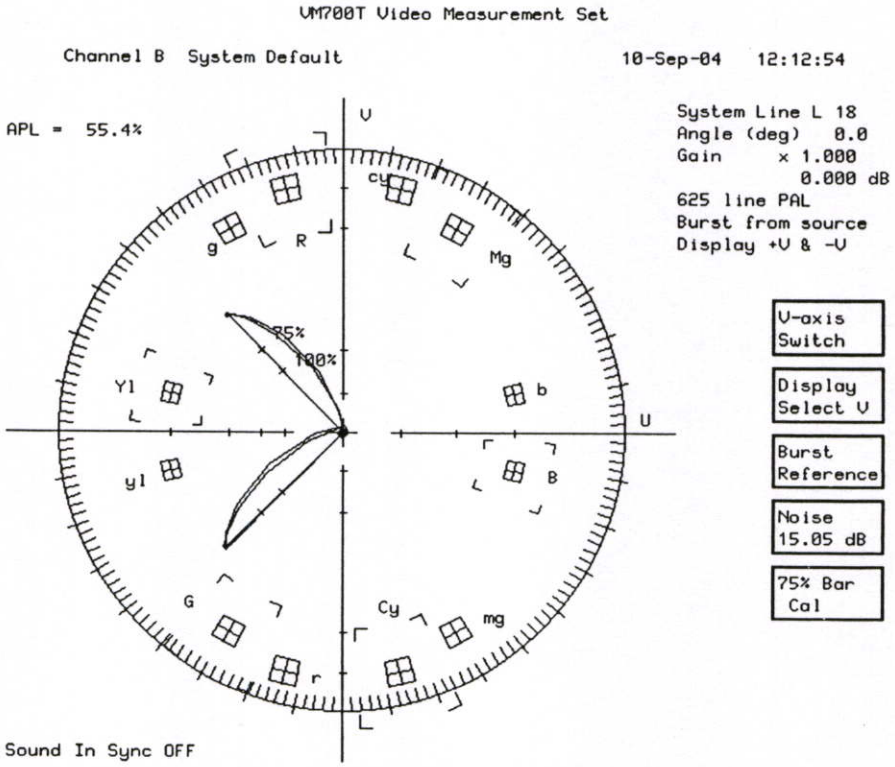
รูปที่ 6.9 แสดงสัญญาณซิ้งค์ทางแนวนอนและสัญญาณเบิรสต์



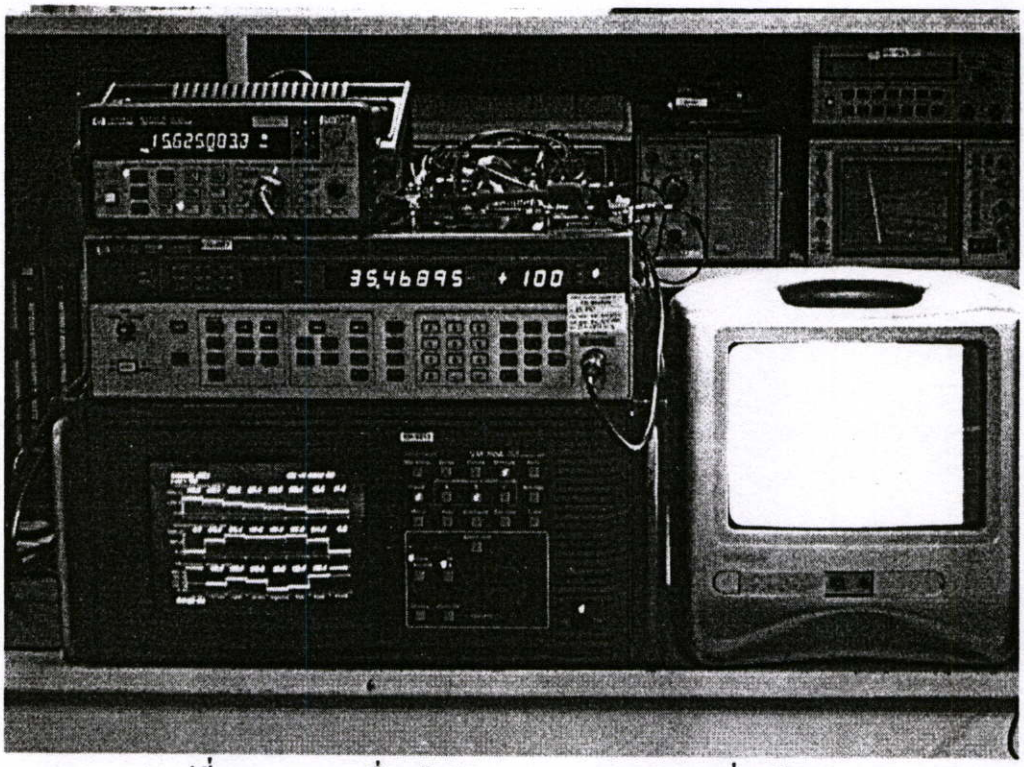
รูปที่ 6.10 แสดงระดับของสัญญาณความส่องสว่าง ระดับและเฟสของสัญญาณสี



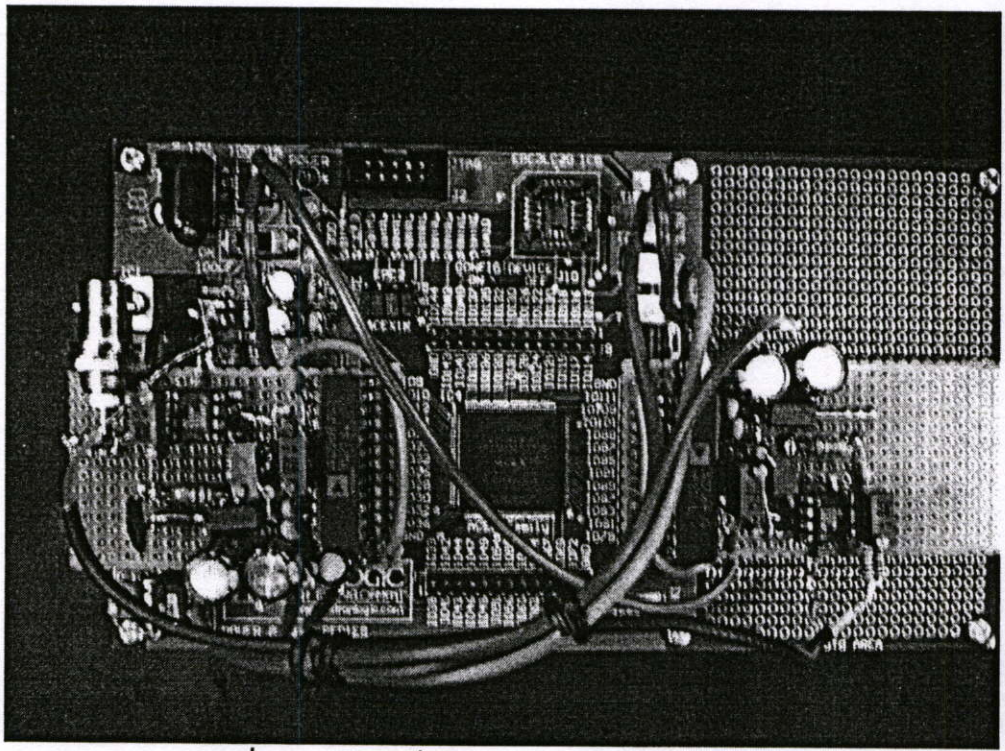
รูปที่ 6.11 แสดงเวคเตอร์สัญญาณความแตกต่างสีและสัญญาณเบิรสต์ของสัญญาณแท่งสี100%



รูปที่ 6.12 แสดงเวคเตอร์ของสัญญาณเบิรสต์



รูปที่ 6.13 แสดงเครื่องมือวัดขณะทำการทดสอบเครื่องต้นแบบ



รูปที่ 6.14 แสดงเครื่องต้นแบบตัวเข้ารหัสฯที่นำเสนอ

ตารางที่ 6.1 แสดงผลการวัดคาบเวลาและความถี่ของสัญญาณภาพสีเบ็ดเสร็จที่ได้จากระบบ

Description	Actual Value	Measured Value	Tolerance	Error
A of Fig.5.2	12 uS.	12.103 uS.	+/- 0.30 uS.	0.103 uS.
B of Fig.5.2	5.60 uS.	5.54 uS.	+/- 0.10 uS.	-0.06 uS.
C of Fig.5.2	2.25 uS.	2.28 uS.	+/- 0.23 uS.	0.03 uS.
D of Fig.5.2	1.65 uS.	1.633 uS.	+/- 0.10 uS.	-0.017 uS.
E of Fig.5.2	4.70 uS.	4.68 uS.	+/- 0.20 uS.	-0.02 uS.
I of Fig.5.1	2.35 uS.	2.38 uS.	+/- 0.10 uS.	0.03 uS.
J of Fig.5.1	4.70 uS.	4.67 uS.	+/- 0.10 uS.	-0.03 uS.
K of Fig.5.1	27.30 uS.	27.298 uS.	(1)	-0.002 uS.
L of Fig.5.1	2.5 Hor	2.5 Hor	0	0
M of Fig.5.1	2.5 Hor	2.5 Hor	0	0
N of Fig.5.1	2.5 Hor	2.5 Hor	0	0
O of Fig.5.1	25 Hor+A	25 Hor+A	0	0
Field Period	50 Hz.	50 Hz.	(1)	0
Line Period	15,625 Hz.	15,625 Hz.	(1)	0

ตารางที่ 6.2 แสดงผลการวัดเฟสของสัญญาณภาพสีเบ็ดเสร็จที่ได้จากระบบ

Description	Actual Value	Measured Value	Tolerance	Error
Burst Phase	135/225 Deg.	136/226 Deg.	+/- 1.5 Deg.	1/1 Deg.
Yellow	167.10 Deg.	167.30 Deg.	+/- 1.5 Deg.	0.2 Deg.
Cyan	283.50 Deg.	284.20 Deg.	+/- 1.5 Deg.	0.7 Deg.
Green	240.70 Deg.	241 Deg.	+/- 1.5 Deg.	0.3 Deg.
Magenta	60.70 Deg.	60.90 Deg.	+/- 1.5 Deg.	0.2 Deg.
Red	103.50 Deg.	104.10 Deg.	+/- 1.5 Deg.	0.6 Deg.
Blue	347.10 Deg.	347.10 Deg.	+/- 1.5 Deg.	0 Deg.

ตารางที่ 6.3 แสดงผลการวัดขนาดของสัญญาณภาพสีเบ็ดเสร็จที่ได้จากระบบ

Description	Actual Value	Measured Value	Error
White (F of Fig.5.2)	700 mV.	712.50 mV.	1.8 %
Yellow (CVS)	620.20 mV.	610 mV.	1.64 %
Cyan (CVS)	490.70 mV.	485.40 mV.	1.08 %
Green (CVS)	410.90 mV.	404.30 mV.	1.60 %
Magenta (CVS)	289.10 mV.	285 mV.	1.42 %
Red (CVS)	209.30 mV.	206.30 mV.	1.43 %
Blue (CVS)	79.80 mV.	77.60 mV.	2.76 %
Black (CVS)	0.0 mV.	22.60 mV.	(2)
White (Chrominance)	0.0 mV.	0 mV.	(2)
Yellow (Chrominance)	627.30 mV.	616.50 mV.	1.72 %
Cyan (Chrominance)	885.10 mV.	864.50 mV.	2.32 %
Green (Chrominance)	826.80 mV.	802 mV.	2.99 %
Magenta (Chrominance)	826.80 mV.	805.10 mV.	2.62 %
Red (Chrominance)	885.10 mV.	866.70 mV.	2.08 %
Blue (Chrominance)	627.30 mV.	614.90 mV.	1.98 %
Black (Chrominance)	0.0 mV.	0.50 mV.	(2)
G of Fig.5.2	300 mV.	298.40 mV.	0.53 %
H of Fig.5.2	300 +/- 10 mV.	296.50 mV.	1.17 %

หมายเหตุ

(1) เป็นช่วงที่มีคาบเวลากว้างและเนื่องจากระบบที่ใช้ในการสร้างส่วนใหญ่ใช้ความถี่สัญญาณนาฬิกาค่อนข้างสูง ดังนั้นคาบเวลาในช่วงนี้จะมีค่าความคลาดเคลื่อนน้อยมากจึงไม่ค่อยระบุในคู่มือการสอบเทียบ โดยส่วนใหญ่ถ้าคาบเวลาของสัญญาณในช่วงที่มีเวลาแคบ เช่น สัญญาณ Horizontal Sync., Equalizing Pulse อยู่ในเกณฑ์ คาบเวลานี้ก็จะอยู่ในเกณฑ์ด้วยเสมอ

(2) สัญญาณจริงมีค่าเท่ากับศูนย์ไม่สามารถคำนวณหาค่าร้อยละได้

บทที่ 7

สรุปการวิจัยและข้อเสนอแนะ

ตัวเข้ารหัสสัญญาณภาพสีแดง เขียว น้ำเงิน เป็นสัญญาณภาพสีเบ็ดเสร็จแบบใช้วงจรเชิงเลข ทั้งหมดที่นำเสนอออกแบบโดยการโมเดลวีเอชดีแอลบรรยายพฤติกรรมของฮาร์ดแวร์จากต้นแบบที่เป็นระบบเชิงอุปมาน มีวัตถุประสงค์เพื่อการนำไปประยุกต์ใช้กับระบบที่เป็นเชิงเลข โดยเฉพาะกับการสร้างสัญญาณภาพทดสอบสำหรับการปรับแต่งที่ใช้ในอุตสาหกรรมการผลิตเครื่องรับโทรทัศน์ ทั้งนี้ตัวเข้ารหัสฯที่นำเสนอยังสามารถนำไปใช้กับระบบอื่นๆที่ใช้ภาพในการแสดงการทำงานของระบบ เช่น ระบบดีซีเอส (Distributed Control System:DCS) หรือเครื่องใช้ไฟฟ้าเพื่อความบันเทิง เป็นต้น

ขั้นตอนของการออกแบบเริ่มจากการศึกษาการทำงานของระบบเชิงอุปมานต้นแบบจากนั้นแปลงให้อยู่ในรูปของระบบเชิงเลขที่แบ่งเป็นส่วนการทำงานย่อย โดยใช้ทฤษฎีของการประมวลผลสัญญาณร่วมกับทฤษฎีการออกแบบวงจรเชิงเลขรวมไปถึงทฤษฎีการสังเคราะห์สัญญาณโดยใช้ระบบเชิงเลขโดยตรง (Direct Digital Synthesis : DDS) [14], [15] การจำลองการทำงานในระดับของการออกแบบขั้นตอนการคำนวณ (Algorithm) ช่วยในการพิจารณาผลของความคลาดเคลื่อนของระบบเพื่อหาจุดที่เหมาะสมของการออกแบบคือประหยัดทรัพยากรและได้มาซึ่งความคลาดเคลื่อนที่ยอมรับได้ ส่วนการจำลองการทำงานในระดับฟังก์ชั่นและไทม์มิ่งช่วยในการตรวจสอบความถูกต้องของการโมเดลวีเอชดีแอล

การออกแบบด้วยสถาปัตยกรรมไปป์ไลน์ทำให้ระบบสามารถทำงานได้ที่ความถี่ในการประมวลผลสูงขึ้น ในขณะที่ใช้ทรัพยากรเท่าเดิม เนื่องจากสถาปัตยกรรมของอุปกรณ์เอฟพีจีเอที่ใช้เป็นชนิดตารางเปิดดู (Look Up Table:LUT) ซึ่งมีฟลิปฟลอปต่ออยู่กับทุกๆลอจิกเอเลเมนท์ (Logic Element : LE) การสร้างรีจิสเตอร์ซึ่งเป็นอุปกรณ์ที่สำคัญในการออกแบบตัวประมวลผลแบบไปป์ไลน์จึงไม่ต้องใช้ลอจิกเกตเพิ่ม ปัจจัยที่สำคัญอีกประการหนึ่งที่จะทำให้ประหยัดทรัพยากรก็คือการแบ่งงานย่อยให้มีความสมมาตร [23]

ขั้นตอนสุดท้ายของการออกแบบคือการวัดหาคุณสมบัติของสัญญาณที่ได้จากระบบฮาร์ดแวร์ด้วยเครื่องมือวัดทดสอบ โดยใช้เกณฑ์ในการพิจารณาคุณสมบัติของสัญญาณตามมาตรฐานเดียวกันกับเครื่องมือวัดทดสอบซึ่งเป็นที่ยอมรับในงานอุตสาหกรรม[8] จากผลการทดลองแสดงให้เห็นว่าเครื่องต้นแบบที่สร้างขึ้นสามารถทำงานได้ถูกต้องและสัญญาณที่ได้อยู่ในเกณฑ์ที่ยอมรับได้ โดยการ

เปรียบเทียบค่าความคลาดเคลื่อนที่ได้จากการวัดด้วยเครื่องมือวัดทดสอบกับค่าความคลาดเคลื่อนที่ยอมรับได้จากคู่มือการสอบเทียบ [7], [18]

ก่อนที่จะได้ผลการทดลองอันเป็นที่น่าพอใจ ในขั้นตอนของการออกแบบได้พบกับปัญหาบางประการเช่น ปัญหาของการศึกษาสถาปัตยกรรมภายในอุปกรณ์เอฟพีจีเอไม่ละเอียดพอ ในตอนแรกไม่ได้นำสถาปัตยกรรมแบบไปป์ไลน์มาใช้ในการออกแบบพบว่าการทำงานของระบบช้ามากจนไม่สามารถทำงานได้ที่ความถี่สุ่ม 8 เท่าของความถี่พาหะรองของสัญญาณสี เหตุผลที่ไม่ต้องการใช้สถาปัตยกรรมแบบไปป์ไลน์ในตอนแรกเนื่องจากเกรงว่าอาจจะเปลืองทรัพยากร แต่พอได้ศึกษา Data Sheet ของเอฟพีจีเอทำให้จึงทราบว่า การใช้สถาปัตยกรรมแบบไปป์ไลน์ไม่ทำให้สิ้นเปลืองทรัพยากร ดังนั้นในการออกแบบระบบที่ต้องการใช้อุปกรณ์ประเภทพีแอลดี (Programmable Logic Device: PLD) ควรศึกษาสถาปัตยกรรมภายในอย่างละเอียดจะช่วยให้การออกแบบมีประสิทธิภาพมากขึ้น ปัญหาต่อมาเป็นปัญหาของการใช้เครื่องมือ เนื่องจากระบบที่ออกแบบใช้สัญญาณนาฬิกาความถี่สูงและสัญญาณเชิงเลขเป็นระดับของทีทีแอล (TTL) ดังนั้นจึงมีสัญญาณรบกวนที่ค่อนข้างแรง การวัดสัญญาณจุดที่ไม่มีปัญหาเรื่อง Loading Effect ควรใช้สายวัด (Probe) ที่มีอิมพิแดนซ์ไม่สูงมากเพื่อป้องกันการรับ (Pickup) สัญญาณรบกวนของสายวัด ถ้าจำเป็นต้องใช้ความถี่สูงในการทำงาน ทางที่ดีควรเลือกใช้อุปกรณ์ชนิดแรงดันต่ำ เช่น อุปกรณ์ประเภทแอลวีดีเอส (Low Voltage Differential Signaling: LVDS) เป็นต้น

เอกสารอ้างอิง

[1] ประภากร สุวรรณะ นิยม ชาญสูงเนิน. 2544 **"เครื่องสร้างสัญญาณรูปแบบชนิดโปรแกรมได้."** หน้า 114. ใน การประชุมเสนอผลงานวิจัยระดับบัณฑิตศึกษาของประเทศไทยครั้งที่ 2 . กรุงเทพฯ : คณะวิทยาศาสตร์ มหาวิทยาลัยมหิดล.

[2] ศ.ดร. วัลลภ สุระกำพลธร. **การประมวลผลสัญญาณเชิงเลข.** บริษัทไดนาพริ้นท์ จำกัด. 2533

[3] พรชัย ภววงษ์ศักดิ์. **การประมวลผลสัญญาณดิจิตอลเบื้องต้น.** มหาวิทยาลัยเทคโนโลยีมหานคร. 2542

[4] Sanjit K.Mitra. **Digital Signal Processing** :A Computer-Based Approach. McGraw-Hill. 2001

[5] RUDOLF MAUSL. **Refresher Topics Television Technology**. ROHDE & SCHWARZ. 1990

[6] John Watkinson. **The Engineer's Guide to Decoding & Encoding**. Snell & Wilcox. 1994

[7] Philips Electronics. **Colour TV Pattern Generator Service Manual**. Philips GmbH – Hamberk – Germany. 1987

[8] รศ. ดร. วิวัฒน์ กิรานนท์. **วิศวกรรมการสื่อสาร.** อักษรสยามการพิมพ์. 2540

[9] Philips Electronics. **TV modulator PM 5671 Instruction Manual**. Philips Electronics. 1986

[10] Margaret Craig. **Television Measurements**. Tektronix Inc. 1991

[11] Steven W. Smith. **Digital Signal Processing**. California Technical Publishing. 1999

[12] Walk Kester. **High Speed Design Techniques**. Analog Device Inc. 1996

[13] Altera Corporation. **Binary Numbering System.** [Online]. Available: http://www.altera.com/literature/an/an083_01.pdf

[14] Analog Devices Inc. **A Technical Tutorial on Digital Signal Synthesis.** [Online]. Available: http://www.analog.com/UploadedFiles/Tutorials/450968421DDS_Tutorial_rev12-2-99.pdf

[15] Bar-Giora Goldberg. **Digital Techniques in Frequency Synthesis**. McGraw-Hill. 1996

[16] Mark Zwolinski. **Digital System Design with VHDL**. Prentice Hall. 2000

[17] Douglas Smith. **HDL Chip Design**. Doone Publications. 1996

[18] Tektronix Inc. **VM700A Programmer's Reference Manual**. Tektronix Inc. 1992

[19] Analog Devices Inc. **AD722 RGB to NTSC/PAL Encoder**. [Online]. Available: http://www.analog.com/UploadedFiles/Obsolete_Data_Sheets/710320677AD722.pdf

[20] Sajjan G. Shiva. **Pipelined and Parallel Computer Architectures**. Harper Collins College Publishers. 1996.

[21] Ricardo A. Losada. **Practical FIR Filter Design in MATLAB**. Mathworks Inc. [Online]. Available: https://tagteamdbserver.mathworks.com/ttserverroot/Download/14883_firdesign.pdf

[22] Mathworks Inc. **Filter Design Toolbox User's Guide**. [Online]. Available: http://www.mathworks.com/access/helpdesk/help/pdf_doc/filterdesign/filterdesign.pdf

[23] Altera Corporation. **Implementing FIR Filters in FLEX Devices**. [Online]. Available: http://www.altera.com/literature/an/an073_01.pdf

ผลงานวิจัยที่ได้รับการตีพิมพ์เผยแพร่

1. ประภากร สุวรรณะ นิยม ชาญสูงเนิน. 2548. "ตัวเข้ารหัสสัญญาณภาพสีแดง เขียว น้ำเงิน เป็นสัญญาณภาพสีเบ็ดเสร็จแบบใช้วงจรเชิงเลขทั้งหมด." **พระจอมเกล้าลาดกระบัง.** 13(1) : 12-20.

ประวัติผู้เขียน

นายนิยม ชาญสูงเนิน เกิดเมื่อวันที่ 24 ธันวาคม พ.ศ. 2516 ที่จังหวัดนครราชสีมา สำเร็จการศึกษาในระดับประกาศนียบัตรวิชาชีพเมื่อปี พ.ศ. 2534 จากวิทยาลัยเทคนิคนครราชสีมาและประกาศนียบัตรวิชาชีพชั้นสูงเมื่อปี พ.ศ. 2536 จากสถาบันเดียวกัน จากนั้นได้เข้าศึกษาต่อในระดับปริญญาตรี สาขาวิศวกรรมไฟฟ้า-สื่อสาร คณะครุศาสตร์อุตสาหกรรม สถาบันเทคโนโลยีพระจอมเกล้าธนบุรีและสำเร็จการศึกษาในปี พ.ศ. 2539 หลังจากสำเร็จการศึกษาได้เริ่มการทำงานกับบริษัท Thomson Television (Thailand) ในตำแหน่งวิศวกรทดสอบ (Test Engineer) จนถึงปี พ.ศ. 2548 และในปี พ.ศ. 2549 มีโอกาสได้ร่วมงานกับบริษัท Emerson Process Management ในตำแหน่งวิศวกรทดสอบตลอดช่วงการทำงานหลังจากสำเร็จการศึกษาในระดับปริญญาตรีจนถึงปัจจุบัน งานหลักที่รับผิดชอบคือการออกแบบและสร้างเครื่องมือทดสอบอัตโนมัติ (Automatic Test Equipment: ATE) ซึ่งระบบประกอบด้วยฮาร์ดแวร์และซอฟท์แวร์ สำหรับงานวิจัยชิ้นนี้ก็เป็นส่วนหนึ่งของระบบดังกล่าว