อะแดปทีฟอัลกอริทึมสำหรับการดีเทคสัญญาณซายน์คลื่นเดียว โดยใช้ฟิลเตอร์แบบปรับค่าคิวแฟคเตอร์

โดย

นายพิพัฒน์ เอื้องเรืองโรจน์

นายสัมฤทธิ์ เตชะวงศ์ธรรม

วัน เดือน ปี ๑๑ ก.ค ๒๕๔๐
เลขทะเบียน ๐๓๘๙๕
เลขเรียกหนังสือ T ๓๘๐๑๘ พ ๔๗ อ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาอุตสาหกรรมศาสตรบัณฑิต

ภาควิชาเทคนิคอุตสาหกรรม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2538

036925

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	อะแดปทีฟอัลกอริทึมสำหรับการดีเทคสัญญาณซายน์คลื่นเดียวโดยใช้ฟิลเตอร์แบบปรับค่าคิวแฟคเตอร์ Adaptive Algorithm For A Single Sinusoid Detection Using A Variable Q Factor Filter			
โดย	นายพิพัฒน์ เอื้องเรืองโรจน์	รหัส	37012016	2N
	นายสัมฤทธิ์ เตชะวงศ์ธรรม	รหัส	37012033	2N
ภาควิชา	เทคนิคอุตสาหกรรม			
อาจารย์ที่ปรึกษา	อาจารย์ชวลิต เบญจางคประเสริฐ			

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง อนุมัติให้นับปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาอุตสาหกรรมศาสตรบัณฑิต

คณะกรรมการสอบปริญญานิพนธ์

_____อาจารย์ที่ปรึกษา

(_____)

_____กรรมการ

(_____)

_____กรรมการ

(_____)

_____กรรมการ

(_____)

_____กรรมการ

(_____)

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	อะแดปทีฟอัลกอริทึมสำหรับการดีเทคสัญญาณซายน์คลื่นเดียวโดยใช้ฟิลเตอร์แบบปรับค่าคิวแฟคเตอร์
นักศึกษา	นายพิพัฒน์ เอื้องเรืองโรจน์ 37012016
	นายสัมฤทธิ์ เตชะวงศ์ธรรม 37012033
อาจารย์ที่ปรึกษา	อาจารย์ชวลิต เบญจางคประเสริฐ
ระดับการศึกษา	อุตสาหกรรมศาสตร์บัณฑิต
ปีการศึกษา	พ.ศ.2538

บทคัดย่อ

ปริญญานิพนธ์นี้เป็นการนำเสนอทฤษฎีเบื้องต้น และแนวความคิดในการดำเนินการของอะแดปทีฟฟิลเตอร์ เพื่อใช้สำหรับการดีเทคสัญญาณซายน์คลื่นเดียว โดยใช้ฟิลเตอร์แบบปรับค่าคิวแฟคเตอร์ ด้วยวงจรกรองความถี่กลางผ่านแบบไอไอาร์ การทำงานของอัลกอริทึมนี้ จะเป็นการติดตามสัญญาณตามความถี่ที่ใช้งาน จากผลการทดลองโดยการ ซิมมูเลชั่น (simulation) ด้วยโปรแกรมคอมพิวเตอร์ จะเห็นได้ว่าอัลกอริทึมใหม่ สามารถติดตามสัญญาณซายน์ได้เร็วกว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PROJECT Adaptive Algorithm For A Single Sinusoid Detection Using A Variable Q Factor Filter

NAME PIPAT AUANGRUANGROD 37012016
SAMRIT TAECHAWONGTHAM 37012033

ADVISOR CHAWALIT BENJANGKAPRASERT
LEVEL OF STUDY BECHELOR DEGREE IN INDUSTRIAL ELECTRONIC
ACADEMIC YEAR 1995

ABSTRACT

THIS PAPER PROPOSED A NEW ADAPTIVE ALOGITHM USING VARIABLE Q FACTOR FILTER BASED ON IIR BANDPASS FILTER FOR A SINGLE SINUSOID DETECTION. THE ALGORITHM WORKS SUCH THAT THE FILTER CENTER FREQUENCY TRACKS THE OBJECT FREQUENCY WITH Q-FACTOR VARIED ADAPTIVELY. IT HAS THE REDUCED AMOUNT OF COMPUTATION. SIMULATION RESULTS ARE GIVEN TO DEMONSTRATE THE CONVERGENCE PERFORMANCE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อ	II
ABSTACT	III
กิติกรรมประกาศ	V
บทนำ	VI
บทที่ 1 ทฤษฎีของ Adaptive filter เบื้องต้น	1
- Filters	1
- Adaptivity	2
- Classifications of sampled-Datas and Digital Filters	3
- Examples of Adaptivity	6
บทที่ 2 การออกแบบ	20
บทที่ 3 ผลการทดลอง	31
บทที่ 4 สรุปผลการทดลอง	52
เอกสารอ้างอิง	54
ภาคผนวก	56

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ขอกราบขอบพระคุณบิดา มารดาที่ให้ความช่วยเหลือและอุปการะในทุกๆด้าน

ขอขอบคุณท่านอาจารย์ชวลิต เบญจางคประเสริฐ ที่ได้ประสิทธิประสาทความรู้ ตลอดจนให้คำปรึกษาแนะแนวทางและวิธีในการแก้ปัญหาต่างๆที่เกิดขึ้น และเอื้อเฟื้อหนังสือและเอกสารต่างๆ จนปริญญานิพนธ์ฉบับนี้เสร็จสมบูรณ์

ขอบคุณเพื่อนๆทั้งหลายที่ช่วยเป็นกำลังใจในการทำงาน

คณะผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทนำ

ในเครื่องมือที่มีการใช้ความถี่ของสัญญาณเพื่อนำไปประมวลผล เช่น เครื่องตรวจสอบการทำงานของหัวใจ หรือเครื่องมือวัดความถี่ต่างๆ ความสามารถของวงจรกรองความถี่ทั่วไปของเครื่องมือเหล่านี้ยังคงยอมให้มีสัญญาณรบกวนต่างๆผ่านไปได้

ปริญญานิพนธ์นี้จึงสร้างวงจรเพื่อใช้ควบคุมวงจรกรองความถี่ให้สามารถลดทอนสัญญาณรบกวนได้มากขึ้น โดยการออกแบบวงจรกรองความถี่ที่ใช้อะแดปทีฟอัลกอริทึมประยุกต์เข้ากับวงจรกรองความถี่กลางแบบไอไออาร์ ผลของการออกแบบนี้จะได้รับการพิสูจน์จากโปรแกรมที่เขียนมาเพื่อซิมูเลชั่นสมการต่างๆ

แต่ละบทภายในปริญญานิพนธ์ฉบับนี้เป็นดังนี้

บทที่ 1 เป็นทฤษฎีเบื้องต้นและตัวอย่างการประยุกต์ใช้งานของอะแดปทีฟฟิลเตอร์ คำจำกัดความของฟิลเตอร์ ประวัติโดยย่อ

บทที่ 2 เป็นการดำเนินงานหาสมการต่างๆที่จะใช้นำมาสร้างเป็นวงจร

บทที่ 3 กราฟแสดงการทดลอง

บทที่ 4 สรุป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
ทฤษฎีของ Adaptive Filter เบื้องต้น

1. FILTERS

คำว่า 'filter' บ่อยครั้งใช้ในการอธิบายอุปกรณ์ในรูปแบบของอุปกรณ์ Hardware ทางกายภาพ หรือในรูปของโปรแกรมคอมพิวเตอร์ และถูกใช้กับกลุ่มของข้อมูลที่มี noise มาก เพื่อที่จะดึงข้อมูลของกลุ่มที่สนใจในช่วงหนึ่งออกมา โดยที่ noise อาจจะเกิดจากแหล่งจ่ายหลากหลาย ตัวอย่างเช่น ข้อมูลอาจจะได้มาจากอุปกรณ์ตรวจจับ noise ซึ่งอาจจะแสดงส่วนของสัญญาณที่มีประโยชน์มาก แต่มีการสูญเสียโดยการส่งผ่านช่องทางการสื่อสาร ในเหตุการณ์ใดๆ เราอาจจะใช้ filter เพื่อทำปฏิบัติขบวนการเกี่ยวกับข่าวสารพื้นฐานสามอย่างคือ:

1. *Filtering* (การกรอง) ซึ่งหมายถึงการดึงเอาข่าวสารเกี่ยวกับกลุ่มที่สนใจในเวลา t โดยการใช้ข้อมูลที่ถูกวัดจนถึงเวลา t
2. *Smoothing* (การทำให้เรียบ) ซึ่งต่างจากการกรองโดยที่กลุ่มข่าวสารที่สนใจไม่จำเป็นที่จะมีอยู่ในเวลา t และข้อมูลที่ถูกวัดหลังเวลา t สามารถถูกใช้เพื่อให้ได้ข่าวสารนี้มาได้ หมายความว่าในกรณีของการทำให้เรียบ (Smoothing) จะมีการหน่วงเวลาการผลิตผลของสิ่งที่สนใจ และในเมื่อขบวนการการทำให้เรียบ เราสามารถใช้ข้อมูลที่จะได้มาจากเวลา t และได้มาจากหลังเวลา t ด้วย ทำให้คาดได้ว่ามันน่าจะมีความแม่นยำมากกว่าขบวนการ filter
3. *Prediction* (การทำนาย) เป็นการทำนายของขบวนการข่าวสาร เป้าหมายของมันคือการหากลุ่มข่าวสารที่สนใจในเวลา t + T ในอนาคต สำหรับ T > 0 โดยการใช้ข้อมูลที่ถูกวัดจนถึงเวลา t

โดยสถิติในการออกแบบวงจร filters ที่ดีที่สุด, เราสมมติการมีอยู่ของคุณสมบัติทางสถิติที่คงที่ของสัญญาณที่มีประโยชน์มากและสัญญาณ noise ที่เพิ่มเข้ามา, และปัญหาก็คือออกแบบวงจร filter ซึ่งข้อมูลที่มี noise เป็น input และให้ลดผลของ noise ให้ได้มากที่สุดเท่าที่เป็นไปได้ Wiener และ Kolmogorov เป็นกลุ่มแรก, ในปี 1940s, ที่ให้การแก้ไขปัญหานี้สำหรับกรณีของขบวนการคงที่สำหรับคุณสมบัติทางสถิติของสัญญาณที่ต้องการและขบวนการ noise ไม่ได้เปลี่ยนแปลงตามเวลา ผลของการแก้ไขปัญหานี้, ขึ้นอยู่กับค่าผิดพลาดวิกฤตเฉลี่ยกำลังสอง (mean-squared) ต่ำสุด, ซึ่งเป็นที่รู้กันโดยทั่วไปว่า Wiener filter

อย่างไรก็ตาม ทฤษฎีที่พัฒนาโดย Wiener และ Kolmogorov ไม่เพียงพอที่จะแก้ปัญหาในสถานการณ์ที่สัญญาณและ/หรือ noise ไม่ได้คงที่ ซึ่งเป็นปัญหาจริงๆ เพื่อที่จะขนะข้อจำกัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ของทฤษฎี Wiener-Kolmogorov, Kalman ได้พัฒนาในปี 1960s ด้วยทฤษฎีการกรองใหม่ซึ่งสามารถใช้ได้กับขบวนการที่ไม่ได้คงที่ ผลของการแก้ปัญหาอยู่ในรูปของตัวแปรเวลาเชิงเส้น เป็นที่รู้กันโดยทั่วไปว่า Kalman filter ทฤษฎีของ Kalman filters สัมพันธ์อย่างใกล้ชิดกับ method of least squares แบบเก่า, ซึ่งอยู่ในช่วงของ Gauss ในปี 1800s

2. ADAPTIVITY

การออกแบบ filter ของ Wiener ต้องใช้ข่าวสารอย่างมากเกี่ยวกับสถิติของข้อมูลที่จะเข้าขบวนการ filter ชนิดนี้จะดีที่สุดเมื่อคุณสมบัติทางสถิติของข้อมูล input เข้ากับข่าวสารซึ่งใช้ในการออกแบบ filter นี้ อย่างไรก็ตาม เมื่อไม่ได้รับข่าวสารนี้อย่างครบถ้วนแล้ว filter จะไม่ดีที่สุดอีกต่อไป ในสถานการณ์เช่นนี้ เราอาจจะใช้ adaptive filter, ซึ่งกลายมาเป็นหลักสำคัญในปีที่ผ่านมาเร็วๆนี้ เรื่องเกี่ยวกับ การสื่อสาร, การควบคุม, ตรวจจับแผ่นดินไหว, ฯลฯ โดย adaptive filter เราหมายถึงอุปกรณ์ซึ่งถูกออกแบบตัวมันเองในความคิดที่ว่า มันจะบรรจุกลุ่มของ parameters ที่ปรับค่าได้ และค่านั้นจะถูกกำหนดโดยอัตโนมัติขึ้นอยู่กับการประมาณค่าคุณสมบัติทางสถิติของสัญญาณที่เกี่ยวข้อง ฉะนั้นทฤษฎีของ adaptive filter จะมีความสัมพันธ์ใกล้ชิดกับการออกแบบ filters ที่ดีที่สุด ในปัญหาการออกแบบต้องการที่จะหากลุ่มที่ดีที่สุดของ filter parameters จากความรู้ของคุณสมบัติของสัญญาณที่เกี่ยวข้อง ให้เหมาะสมกับบรรทัดฐาน หรืออีกอย่างหนึ่ง, ปัญหาของ adaptive filter คือ ความต้องการที่จะหา algorithm (ขบวนการในการแก้ปัญหา) สำหรับปรับค่า filter parameters ในสถานการณ์ที่ความรู้เกี่ยวกับคุณสมบัติของสัญญาณที่เกี่ยวข้องไม่สมบูรณ์, ดังนั้นการทำงานของ adaptive filter จะลู่เข้าสู่ลักษณะ filter ที่ดีที่สุด หลังจากผ่านขบวนการซ้ำๆเกี่ยวกับจำนวนตัวเลขมากมายของ algorithm

Adaptive filter อาจจะเป็นหนึ่งในสองชนิดนี้:

1. Adaptive filter ลักษณะ open-loop หรือไม่กระทำซ้ำ, ประกอบด้วยขบวนการสองขั้น โดยที่ขบวนการแรกคือการเรียนรู้สถิติของสัญญาณที่เกี่ยวข้อง แล้วนำผลที่ได้ใส่เข้าไปใน algorithm ที่ไม่มีการกระทำซ้ำ (non-recursive algorithm) ลักษณะเช่นนี้ปกติจะมีข้อเสียที่ต้องการความละเอียดมากและ hardware ราคาแพง
2. Adaptive filter ลักษณะ closed-loop หรือการกระทำซ้ำ ที่เรื่องของสถิติไม่ได้ถูกประมาณอย่างเด่นชัด แต่ว่าการออกแบบ adaptive filter จะทำได้ในขบวนการเดียวโดยการเฉลี่ยของ algorithm ที่กระทำซ้ำ (recursive algorithm) ซึ่งจะเปลี่ยนค่า filter parameters โดยอัตโนมัติด้วยข้อมูลที่สุ่มเข้ามาใหม่ในแต่ละครั้ง ในแต่ละการกระทำซ้ำของ algorithm,

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

adaptive filter จะค่อยๆเรียนรู้เกี่ยวกับสถิติของสัญญาณที่เกี่ยวข้อง และการปรับปรุงกลุ่มของค่าปัจจุบันของ filter ที่เปลี่ยนค่า parameters ได้ จะถูกคำนวณโดยใช้ข่าวสารใหม่นี้ และ adjustable filter (filter ทีเปลี่ยนค่า parameters ได้) ค่า parameters ของมันจะเพิ่มขึ้น และการกระทำซ้ำการเรียนรู้ครั้งต่อไปจะมีฐานจากการปฏิบัติการของ filter ด้วยกลุ่มของค่าที่ปรับปรุงแล้วในลักษณะของ closed-loop, การเรียนรู้และปฏิบัติการคำนวณจะถูกรวมกันเป็นขบวนการเดียวกัน และเมื่อเป็นเช่นนั้น มันดูเหมือนว่าจะมีเครื่องมือที่ง่ายกว่าลักษณะ open-loop

3. CLASSIFICATIONS OF SAMPLED-DATA AND DIGITAL FILTERS

Adaptive filter อาจจะทำในรูปแบบของ เวลาที่ต่อเนื่อง (continuous-time) หรือ เวลาที่ไม่ต่อเนื่อง (discrete-time), ซึ่งในแต่ละกรณีความสัมพันธ์ของ input-output ของ filter อาจจะถูกอธิบายด้วยสมการ differential หรือสมการ difference, ตามลำดับ ในที่นี่เราใช้รูปแบบของเวลาที่ไม่ต่อเนื่อง พิจารณาสัญญาณ analog $\bar{u}(t)$ ซึ่งเปลี่ยนแปลงอย่างต่อเนื่องเทียบกับเวลา t ตัวอย่างเช่น, $\bar{u}(t)$ อาจจะแสดงแรงดันจากสัญญาณเสียงของการสื่อสารทางโทรศัพท์, output ของช่องสื่อสารถูกสร้างโดยข้อมูลคอมพิวเตอร์ดิจิตอลที่จ่ายให้กับช่องทาง input หรือ video output ของระบบ radar

โดยเหตุผลทางเทคนิค บ่อยครั้งที่พบว่า มันจะสะดวกกว่าที่จะแสดงสัญญาณ analog $\bar{u}(t)$ โดยอันดับของการสุ่มที่มีช่วงเท่าๆกัน $\{\bar{u}(nT)\}$, เมื่อ T เป็นช่วงของการสุ่ม ทฤษฎีการสุ่มที่นิยมกันกำหนดเงื่อนไขซึ่ง จะต้องทำให้ตรงเพื่อที่จะปรับขบวนการสุ่มนี้ ในเรื่องนี้จะให้อัตราการสุ่ม 1/T มีค่าเท่ากับหรือมากกว่าสองเท่าขององค์ประกอบความถี่สูงสุดของสัญญาณ analog $\bar{u}(t)$, ดังนั้นจะมีความตรงกันแบบหนึ่งต่อหนึ่งระหว่างสัญญาณ analog ดั้งเดิม $\bar{u}(t)$ และลำดับ $\{\bar{u}(nT)\}$, และฉะนั้น $\bar{u}(t)$ อาจจะถูกแปลงคืนได้จาก $\{\bar{u}(nT)\}$

filter จะทำงานบนลำดับของการสุ่ม $\{\bar{u}(nT)\}$ เพื่อสร้างลำดับอื่นๆของการสุ่ม $\{\bar{y}(nT)\}$ เรียกว่า filter ที่สุ่มข้อมูลแล้ว (sampled-data filter)

เมื่อตัวอย่างของลำดับ $\{\bar{u}(nT)\}$ ถูกสุ่มออกมา มันเป็นไปได้ที่จะแสดงความสูงของค่าพวกนี้โดยใช้รหัส binary ที่มีความยาวจำกัด การปฏิบัติการของ filter บนลำดับของ binary 1's และ 0's เพื่อสร้างลำดับ binary อื่นๆนั้นเรียกว่า digital filter ฉะนั้น round-off errors (ความผิดพลาดในการปัดเลข) จะถูกแนะนำเมื่อ filter แสดงอยู่ในรูปของ digital

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อความสะดวกในการวิเคราะห์, เราให้ $u(n) = \bar{u}(nT)$ และ $y(n) = \bar{y}(nT)$ ตามนี้, และใช้ $\{u(n)\}$ เพื่อแสดงลำดับของตัวอย่างที่ได้มาจากการสุ่มของสัญญาณ $u(t)$ และให้ลำดับ $\{y(n)\}$ ถูกคำนวณด้วยสูตร

$$y(n) = \sum_{k=0}^{M} a(k)u(n-k) + \sum_{k=1}^{N} b(k)y(n-k) \qquad (1.1)$$

เมื่อสัมประสิทธิ์ a(k) และ b(k) เป็นค่าคงที่ สมการ difference (1.1) นิยาม sampled-data หรือ digital filter จะเห็นว่า output y(n) ของ filter , ที่อยู่ในรูปทั่วไปนี้, เป็นเพียงการผสมกันของ input จากอดีตและปัจจุบัน u(n),u(n-1)....u(n-M) บวกกับการผสมกันแบบเชิงเส้นของ ตัวอย่าง output ในอดีต y(n-1),y(n-2).....y(n-N) เราอาจจะจำแนกชนิดของ filter ทั้งสอง ขึ้นอยู่กับค่าของสัมประสิทธิ์ในสมการ (1.1), ดังที่จะบรรยายดังนี้:

1. สัมประสิทธิ์ b(k) เป็นศูนย์หมด, ดังนั้น output y(n) จะขึ้นอยู่กับค่าตัวอย่าง input อดีตและปัจจุบันเท่านั้น, ดังแสดงได้ด้วย

$$y(n) = \sum_{k=0}^{M} a(k)u(n-k) \qquad (1.2)$$

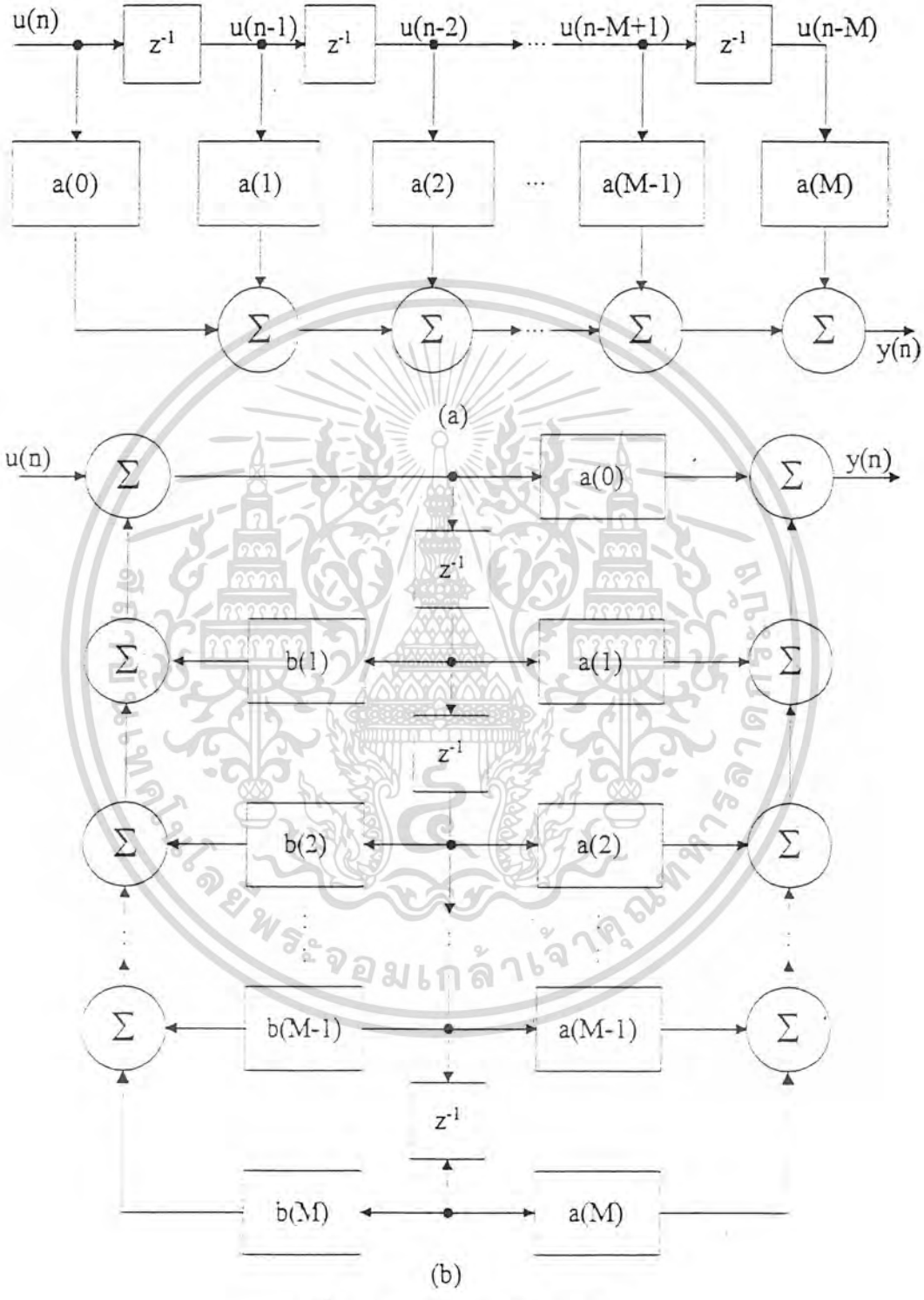
โครงสร้างของ filter อย่างเดียวกันนี้ ได้แสดงไว้ในรูปที่ 1.1(a) เมื่อ z^{-1} แสดงถึง unit-delay operator (ตัวปฏิบัติการหน่วยหนวงเวลา) และอ้างถึงโครงสร้างนี้ว่า finite-impulse response (FIR) filter (filter ที่มีการตอบสนอง impulse แบบจำกัด) หรือเรียกว่า tapped-delay-line filter หรือ transversal filter คุณสมบัติที่สำคัญของโครงสร้าง filter ของรูป 1.1(a) คือมันจะเสถียรโดยปกติ

2. มีสัมประสิทธิ์ b(k) อย่างน้อยหนึ่งตัวที่ไม่เป็นศูนย์ จะเป็นผลให้ค่าตัวอย่างในอดีตของ output มีอิทธิพลกับค่าตัวอย่างในปัจจุบัน y(n) ของ output เรียกโครงสร้างนี้ว่า infinite-impulse response (IIR) filter (filter ที่มีการตอบสนอง impulse ไม่จำกัด) มีหลายวิธีที่โครงสร้างนี้อาจจะเป็นจริงได้ในทางปฏิบัติ รูปที่ 1.1(b) แสดงถึงทางที่เป็นไปได้อันหนึ่งสำหรับกรณี N=M IIR filter ไม่เหมือน FIR filter ที่ IIR filter อาจจะไม่เสถียรภาพ, ขึ้นอยู่กับค่าซึ่งสมมติโดยสัมประสิทธิ์ที่สงกลับ b(k)

ในกรณีของ adaptive filters ใช้โครงสร้าง IIR, จะมีความยากอยู่สองอย่างใหญ่ๆซึ่งจะเกิดจากการ feedback (ส่งกลับ): (1) filter อาจจะไม่เสถียร,ถ้าไม่มีการระมัดระวังเป็นพิเศษ, และ (2) การปรากฏของ feedback อาจจะมีผลเสียต่อความแม่นยำด้วย ซึ่งค่าสัมประสิทธิ์ของ filter จะต้องระบุ และจากเหตุผลเหล่านี้ทำให้เครื่องใช้ในทางปฏิบัติต้องการ adaptive filter, และ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พบว่า adaptive FIR filters จะถูกใช้ค่อนข้างเฉพาะงาน ในที่นี้จะพูดถึง adaptive filters โดยใช้โครงสร้าง FIR



รูปที่ 1.1 (a) FIR filter (b) IIR filter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงสร้าง filter อื่นๆ ซึ่งจะพิจารณาได้แก่ multistage lattice filter (filter ที่มีโครงข่ายหลายๆตัว) ที่เรียกเช่นนี้เพราะว่า filter แต่ละตัวจะมีลักษณะเป็นโครงข่าย filter ชนิดนี้จะมีคุณสมบัติบางอย่างที่น่าสนใจ ซึ่งทำให้เป็นโครงสร้างอื่นที่น่าสนใจ กับโครงสร้างแบบ tapped-delay-line สำหรับเครื่องใช้ที่ใช้ adaptive filter

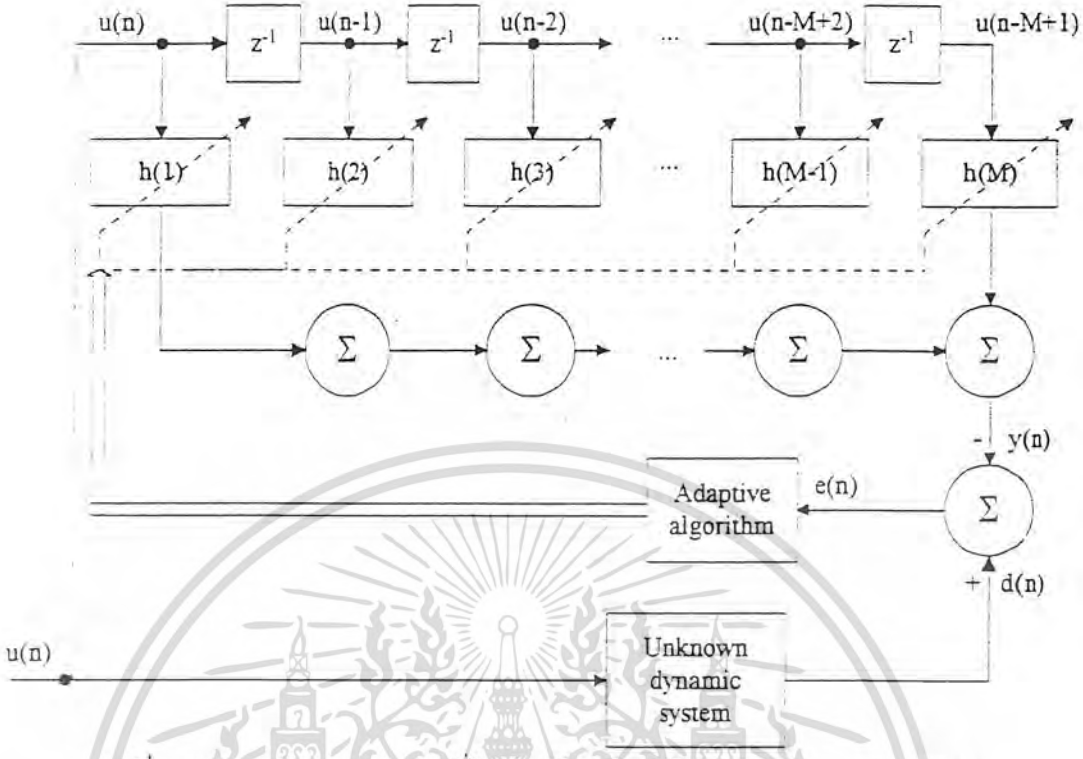
4 EXAMPLES OF ADAPTIVITY

ตัวอย่างการใช้ adaptive filter ในงานต่างๆ

ตัวอย่าง 1 Modeling of an Unknown Dynamic System

สมมติว่ามีระบบการเปลี่ยนแปลงที่ไม่เป็นที่รู้จัก, และมีกลุ่มของเครื่องมือวัด discrete-time ใช้พิจารณาความหลายหลากของสัญญาณ output ของระบบที่ตอบสนองกับสัญญาณคงที่ที่ป้อนให้กับ input ของระบบ สมมติว่าระบบนี้ไม่เปลี่ยนแปลงไปตามเวลาและมีคุณสมบัติเชิงเส้น และต้องการที่จะพัฒนา model (แบบจำลอง) สำหรับระบบนี้ในรูปแบบของ tapped-delay-line filter ซึ่งประกอบด้วยกลุ่มของอุปกรณ์ delay-line (delay-line แต่ละอันจะแสดงด้วยตัวปฏิบัติการ unit-delay z^{-1}) และมีกลุ่มของค่าสัมประสิทธิ์ที่ปรับค่าได้ที่มีการต่อภายในดังรูป 1.2 ที่เวลา n สัญญาณที่มีอยู่ประกอบด้วยกลุ่มของตัวอย่าง u(n),u(n-1),...,u(n-M+1) ตัวอย่างเหล่านี้ถูกคูณด้วยกลุ่มที่ตรงกันของสัมประสิทธิ์ที่ปรับค่าได้, มีชื่อว่า, h(1),h(2),....,h(M), เพื่อที่จะสร้างสัญญาณ output,y(n) ให้ output แท้จริงของระบบที่ไม่เป็นที่รู้จักมีสัญลักษณ์ d(n) output ของ adaptive filter y(n) ถูกเปรียบเทียบกับ output ของระบบที่ไม่เป็นที่รู้จัก d(n) เพื่อสร้างสัญญาณผิดพลาด (error signal) e(n), ซึ่งเป็นความแตกต่างระหว่างทั้งสอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 1.2 แบบจำลองของระบบที่ไม่เป็นที่รู้จักโดยใช้ tapped-delay-line filter

ถึงตอนนี้ อาจจะบอกได้ว่าสถานะการทำงานของ adaptive filter ได้ดังนี้ ด้วย output ของระบบที่ไม่เป็นที่รู้จัก d(n) จะเรียกว่าการตอบสนองที่ประสงค์ (desired response) ต้องการที่จะพัฒนาขบวนการ adaptive สำหรับการปรับค่าสัมประสิทธิ์ของ tapped-delay-line filter เพื่อที่จะทำให้สัญญาณผิดพลาด e(n) มีค่าน้อยที่สุดในลักษณะนี้ บรรทัดฐานซึ่งถูกใช้บ่อยๆในทางปฏิบัติสำหรับการทำให้น้อยที่สุดนี้คือบรรทัดฐานค่าผิดพลาดกำลังสองเฉลี่ย (mean-square-error), ซึ่งเหมาะสมกับ filter ที่ปรับค่าสัมประสิทธิ์ได้ เพื่อให้ได้ค่าเฉลี่ยกำลังสองน้อยที่สุดของสัญญาณผิดพลาด e(n) ในบางกรณีเมื่อสัญญาณ input และ output ของระบบที่ไม่เป็นที่รู้จักนั้นคงที่ (ซึ่งเป็นผลมาจากการสมมติให้ระบบไม่เปลี่ยนแปลงไปตามเวลา หรือ time-invariant) มันจะทำให้สัญญาณผิดพลาดนั้นคงที่ด้วย และค่าเฉลี่ยกำลังสองของ e(n) เป็น function อันดับสองของสัมประสิทธิ์ filter ค่าผิดพลาดเฉลี่ยกำลังสองที่ขึ้นอยู่กับสัมประสิทธิ์ unknown filter (filter ที่ไม่เป็นที่รู้จัก) นั้นจะอยู่ในรูปของ multidimensional paraboloid (รูปทรง parabola หลายมิติ) ที่มีจุดต่ำสุดที่ถูกกำหนดไว้เดียว หรือ minimum point (จุดที่มีค่าต่ำสุด) adaptive filter จะทำงานในการค้นหาจุดที่ต่ำที่สุดของพื้นที่นี้อย่างต่อเนื่อง จึงต้องทำให้การทำงานของมันดีที่สุดเท่าที่เป็นไปได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อระบบที่ไม่เป็นที่รู้จักนี้เปลี่ยนแปลงไปตามเวลา เป็นผลให้ output ของระบบ (รวมถึงการตอบสนองที่ต้องการที่ได้จาก adaptive tapped-delay-line filter) จะเป็นแบบไม่คงที่ ทำนองเดียวกัน การเปลี่ยนแปลงพื้นที่ของประสิทธิภาพการผิดพลาดเปลี่ยนไปตามเวลาด้วย ในกรณีนี้ adaptive algorithm ใช้ปรับค่าสัมประสิทธิ์ของ tapped-delay-line filter จะถูกเพิ่มงานสำหรับตามจุดต่ำสุดหรือ bottom ของประสิทธิภาพการผิดพลาดอย่างต่อเนื่อง

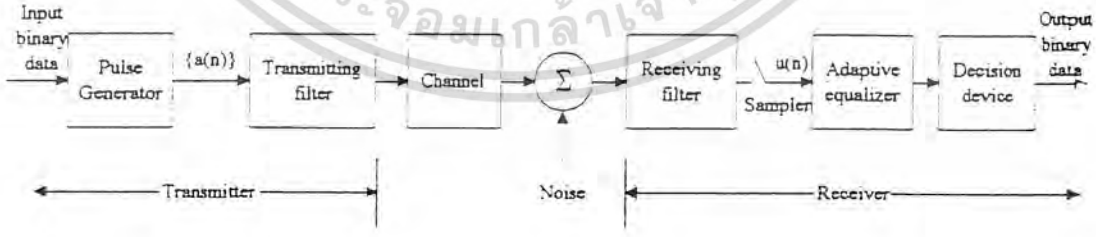
ตัวอย่างที่ 2 Adaptive Equalization for Data Transmission

เมื่อ 20 ปีก่อนได้มีการพยายามในการศึกษาระบบการส่งข้อมูลซึ่งใช้ประโยชน์จากช่องสัญญาณที่มีอยู่ให้มีประสิทธิภาพ เป้าหมายคือออกแบบระบบสำหรับการส่งข้อมูลด้วยอัตราสูงที่สุดเท่าที่เป็นไปได้ ภายใต้ความน่าเชื่อถือที่กำหนดซึ่งปกติถูกวัดอยู่ในรูปของ อัตราผิดพลาด (error rate) หรือความผิดพลาดความน่าจะเป็นเฉลี่ยของสัญลักษณ์ (average probability of symbol error) การส่งข้อมูล digital ผ่านช่องทางการสื่อสารแบบเชิงเส้นถูกจำกัดด้วยองค์ประกอบสองอย่าง:

1. *Intersymbol interference (ISI)*: เกิดจากการแพร่กระจายของรูป pulse ที่ถูกส่ง ซึ่งเป็นผลมาจากการเบี่ยงเบนของการตอบสนองความถี่ในช่องทาง จากลักษณะทางความคิดที่ความสูงคงที่และมี phase เชิงเส้น (ตัวอย่างเช่น constant delay)

2. *Additive thermal noise*: กำเนิดจากภาคแรกสุดของกลุ่มรับสัญญาณ

สำหรับช่องทางที่จำกัด bandwidth (เช่น ช่องทางเสียงโทรศัพท์) ปกติจะพบว่า Intersymbol Interference เป็นองค์ประกอบสำคัญที่ใช้พิจารณาในการออกแบบระบบอัตราส่งข้อมูลสูง



รูปที่ 1.3 Block diagram of a baseband data transmission system

รูปที่ 1.3 แสดงแบบจำลองเทียบเท่าของระบบข้อมูลสื่อสาร รูปแบบของลำดับข้อมูล {a(n)} ไม่เพียงขึ้นข้อมูล binary ที่จะส่งออกไปเท่านั้น แต่ยังขึ้นอยู่กับชนิดของการ modulate ที่ใช้ด้วย ดังตัวอย่าง, ในกรณีของ M-ary phase-shift keying, a(n) ได้มาจากค่า M จาก exp[j2p

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(m-1)/M], เมื่อ m=1,2,...,M โดยปกติ, M จะเท่ากับจำนวนเต็มยกกำลังสอง เมื่อไม่มี noise รบกวน, output ของ filter เครื่องรับในรูป 1.3 เท่ากับ

$$\tilde{u}(t) = \sum_n a_n \tilde{p}(t - nT) \tag{1.3}$$

เมื่อ T เป็นช่วงเวลาของคาบสัญญาณ และ $\tilde{p}(t)$ เป็นการตอบสนอง impulse ของจุดต่อลดหลั่นของ filter ภาคส่ง, ช่องสัญญาณ, และ filter ภาครับ โดยการสุ่มตัวอย่าง u(t) ให้เข้ากันกับภาคส่ง และนิยาม $\tilde{u}(n) = \tilde{u}(nT)$ และ $p(n) = \tilde{p}(nT)$ เราได้

$$\begin{aligned} u(k) &= \sum_n a_n p(k-n) \\ &= a_k p(0) + \sum_{\substack{n \\ n \neq k}} a_n p(k-n) \end{aligned} \tag{1.4}$$

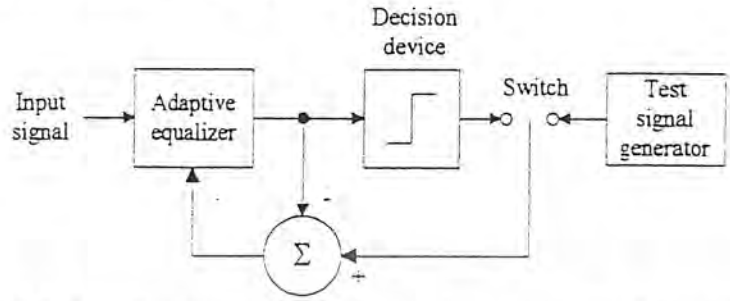
term แรกด้านขวามือของสมการ (1.4) กำหนดสัญลักษณ์ที่ต้องการ และอนุกรมที่เหลือแสดงถึง intersymbol interference ที่เป็นผลมาจากการกระทำรวมของ filter ภาคส่ง, ช่องสัญญาณ, และ filter ภาครับ Intersymbol interference นี้, ถ้าปล่อยไว้โดยไม่ตรวจสอบ, อาจจะเป็นผลลำหรับการตัดสินใจหาข้อผิดพลาดเมื่อสัญญาณที่ถูกสุ่มที่ filter ภาครับถูกเปรียบเทียบกับขอบเขตระดับย่อยที่กำหนดไว้บางตัวโดยได้มาจากอุปกรณ์ตัดสินใจ

เพื่อจะแก้ปัญหา Intersymbol interference, ต้องมีการควบคุม function ของเวลา $\tilde{p}(t)$ เฉพาะเรื่องนี้, ถ้าลักษณะของช่องส่งสัญญาณเป็นที่รู้อย่างแน่นอนแล้ว จะเสมือนว่ามีทางสำหรับออกแบบทั้ง filter ภาคส่งและภาครับ ซึ่งจะทำให้ผลของ intersymbol interference (ที่เวลาสุ่ม) ค่อนข้างน้อย, และที่เวลาเดียวกันจะจำกัดผลของ noise ภาครับที่เพิ่มเข้าไป โดยลดความผิดพลาดความเป็นไปได้เฉลี่ยของสัญลักษณ์ให้น้อยที่สุด อย่างไรก็ตาม, ในทางปฏิบัติ เราพบว่าช่องสื่อสารนั้นเป็นแบบสุ่มในกลุ่มของช่องสัญญาณที่เป็นไปได้ ฉะนั้น, การใช้คู่ตายตัวของ filter ภาคส่งและภาครับ, ที่ถูกออกแบบโดยคุณลักษณะช่องสัญญาณเฉลี่ย, อาจจะไม่เพียงพอที่จะลด Intersymbol Interference จึงเป็นสิ่งจำเป็นที่ต้องมี adaptive equalizer ซึ่งจะให้การควบคุมที่แม่นยำบนการตอบสนองของเวลาของช่องส่งสัญญาณ ด้วยเหตุนี้จึงจะทำให้ความสามารถของการส่งช่องสัญญาณให้เต็มที่ให้เป็นจริงได้ คำว่า "Equalizer" ถูกใช้อธิบาย filter ที่ใช้บนช่องสัญญาณโทรศัพท์เพื่อทำให้คุณลักษณะความสูงและการหน่วงเวลาของช่องส่งสัญญาณราบเรียบ

โดยพื้นฐานสำหรับ equalization (การทำให้เท่ากัน) ของระบบการส่งข้อมูลที่มีอยู่ คือ pre-equalization ที่ภาคส่ง และ post-equalization ที่ภาครับ และเนื่องจากวิธีอย่างแรกต้องใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทางย้อนกลับ (feedback path) เราจึงจะพิจารณา equalization ที่ภาครับ ที่ adaptive equalizer วางหลังจาก filter ภาครับ ดังรูป 1.3



รูปที่ 1.4 Adaptive filter with a decision-directed mode of operation

รูปที่ 1.4 แสดง block diagram ของ adaptive equalizer การทำงานที่เกี่ยวข้องกับ training mode แล้วถูกตามโดย tracking mode

ในช่วง training mode, สัญญาณทดสอบที่ทราบค่าจะถูกส่งเพื่อวัดช่องสัญญาณ สัญญาณทดสอบที่ใช้อย่างกว้างขวางประกอบด้วย maximal-length shift register หรือ pseudo-noise (PN) sequence พร้อมกับ spectrum ที่กว้างและลงตัว สัญญาณทดสอบจะต้องมีความยาวอย่างน้อยเท่ากับตัว equalizer เพื่อให้แน่ใจว่า spectrum ของสัญญาณที่ส่งมีความหนาแน่นเพียงพอใน bandwidth ของของส่งสัญญาณในเครื่องรับ, adaptive equalizer จะถูกจ่ายด้วยการตอบสนองที่ต้องการ output ของ equalizer ถูกลบออกจากการตอบสนองนี้เพื่อสร้างสัญญาณผิดพลาด ซึ่งใช้ในการปรับสัมประสิทธิ์ของ equalizer ให้ไปสู่ค่าที่ดีที่สุด ชนิดของ adaptive algorithms ที่เป็นที่นิยมมากที่สุดที่ใช้สำหรับการปรับนี้ เกี่ยวข้องกับการ update ของสัมประสิทธิ์แต่ละตัวของ equalizer ในแต่ละช่วงของ symbol, โดยเริ่มจากค่าเริ่มต้นที่ตั้งไว้

เมื่อช่วง training เริ่มต้นสำเร็จลง, ค่าสัมประสิทธิ์ของ adaptive equalizer อาจจะถูกเปลี่ยนแปลงต่อไปอีกในลักษณะของ decision-directed mode (mode ของการตัดสินใจโดยตรง) ใน mode นี้ สัญญาณผิดพลาดจะได้มาจากการประมาณของภาครับช่วงสุดท้าย (ซึ่งไม่จำเป็นต้องเป็นสัญญาณที่ถูกต้อง) ของลำดับการส่ง การประมาณของเครื่องรับได้มาจาก การให้ output ของ adaptive equalizer ไปยังอุปกรณ์ตัดสินใจ (decision device), ดังแสดงในรูปที่ 1.4 ในการปฏิบัติการปกติ, อุปกรณ์ตัดสินใจจะมีความถูกต้องที่มีความเป็นไปได้สูง ดังนั้นการประมาณของสัญญาณผิดพลาดจะถูกต้องบ่อยครั้ง เพียงพอที่จะยอมให้ adaptive equalizer สามารถปรับค่าสัมประสิทธิ์ของมันได้อย่างถูกต้อง ความสามารถที่น่าสนใจอื่นๆของ decision-directed adaptive equalizer คือ จริงๆแล้วมันสามารถตามรอยการเปลี่ยนแปลงอย่างช้าในคุณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะหรือการกระชากของช่องส่งสัญญาณในภาครับด้านหน้าสุด, อย่างเช่น การสั่นช้าๆของ phase ตัวอย่าง

ตัวอย่างที่ 3 Digital Representation of Speech

ในช่วง 20 ปีก่อนมีการใช้ขบวนการทาง digital มากขึ้นสำหรับการเข้ารหัสและการส่งสัญญาณเสียงอย่างมีประสิทธิภาพ องค์ประกอบหลักสองอย่างสำหรับงานนี้คือ อย่างแรก, ขบวนการ digital มีความเป็นไปได้ที่จะควบคุมลดผลของ noise และการรบกวนข้างเคียง ในช่วงของการสื่อสาร จึงเป็นการปรับปรุงความน่าเชื่อถือของระบบอย่างสำคัญ อย่างที่สอง, การปฏิวัติทางเทคโนโลยี digital [การสร้าง Very-Large-Scale Integration (VLSI)] ได้ลดทั้งขนาดและราคาของอุปกรณ์อย่างต่อเนื่อง

coders (ส่วนเข้ารหัส) สำหรับการแสดงสัญญาณเสียงแบบ digital แบ่งออกเป็น 2 ชนิดใหญ่: คือ source coders (ส่วนเข้ารหัสของแหล่งจ่าย) และ waveform coders (ส่วนเข้ารหัสของรูปคลื่น) Source coders นั้นขึ้นอยู่กับแบบจำลอง ซึ่งพวกนี้ใช้ความรู้ก่อนๆสำหรับสร้างสัญญาณเสียงที่แหล่งจ่าย Source coders สำหรับการพูดมักจะเรียกกันว่า vocoders (คำย่อของ voice coders) พวกนี้สามารถปฏิบัติการที่ 4.8 kbit/s หรือต่ำกว่า; อย่างไรก็ตาม, พวกมันให้คุณภาพที่ประกอบขึ้น, ด้วยสัญญาณเสียงที่ขาดความเป็นธรรมชาติ ในอีกทางหนึ่ง Waveform coders ต้องใช้สัญญาณรูปคลื่นเสียงที่ส่งมาสร้างขึ้นใหม่ ในจุดนี้, coders ตัวนี้จึงไม่ขึ้นอยู่กับสัญญาณ และอาจจะถูกออกแบบเพื่อสร้างคุณภาพโทรศัพท์สำหรับการพูดที่อัตราการเข้ารหัสต่ำๆเท่ากับ 16 kbit/s

แบบจำลองของขบวนการสร้างเสียง

รูปที่ 1.5 แสดง block diagram อย่างง่ายของแบบจำลองเก่า สำหรับการสร้างเสียง ซึ่งสมมติให้ขบวนการกำเนิดเสียงเป็นแบบเชิงเส้น (เช่น แหล่งของการกระตุ้น) และแยกออกจาก intelligence-modulating, vocal-tract filter ได้ รูปแบบที่แน่นอนของการกระตุ้นขึ้นอยู่กับว่าเสียงพูดเป็นเสียงคนพูดหรือไม่ ดังที่อธิบายดังนี้:

1. voiced speech sound (เช่นตัว อ.) ถูกกำเนิดจากเสียงของหลอดเสียงที่ลากต่อกัน ในแบบจำลองที่ 1.5 เครื่องกำเนิดขบวนของ impulse สร้างอันดับของ impulses (เป็น pulse ที่สั้นมากๆ) ซึ่งเว้นว่างด้วยช่วงพื้นฐาน เท่ากับช่วงของเสียงสูง ในสัญญาณนี้จะกระตุ้น filter เชิงเส้นที่มีการตอบสนอง impulse เท่ากับ pulse ของเสียงจากหลอดเสียง

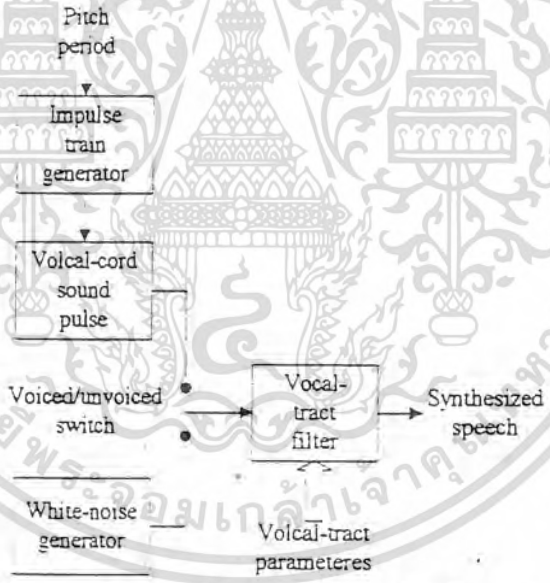
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. unvoiced speech sound (เช่นตัว ฟ) ถูกกำเนิดจากเสียงแบบสุ่มจากอากาศที่ไหลและสั่นอย่างรวดเร็ว ในกรณีนี้ตัวกระตุ้นประกอบด้วยง่ายๆเพียง white noise source (แหล่งจ่าย noise กว้างๆ) ความเป็นไปได้ในการกระจายของตัวอย่าง noise นี้ไม่เป็นจุดสำคัญเลย

การตอบสนองความถี่ของ vocal-tract filter สำหรับ unvoiced speech หรือ กลุ่มของเสียงนั้น ถูกคูณด้วย spectrum ของ vocal-cord sound pulses (pulses ของเสียงจากหลอดเสียง) และได้เป็น short-time spectral envelope ของสัญญาณเสียง

การเข้ารหัสคาดหมายแบบเชิงเส้น (Linear Predictive Coding)

ขบวนการของ linear predictive coding (LPC) เป็นตัวอย่างของ source coding (การเข้ารหัสแหล่งจ่าย) ขบวนการนี้สำคัญมากเพราะว่ามันไม่เพียงแต่ให้เทคนิคที่มีอำนาจมากสำหรับการส่งเสียงแบบ digital ที่อัตรา bit ต่ำแล้ว มันยังประมาณค่า parameters พื้นฐานของการพูดอย่างถูกต้องด้วย



รูปที่ 1.5 Block diagram ของแบบจำลองอย่างง่ายสำหรับขบวนการผลิตเสียงพูด

การพัฒนาของ LPC ขึ้นอยู่กับแบบจำลองของรูปที่ 1.5 สำหรับขบวนการผลิตเสียงพูด การตอบสนองความถี่ของ volcal tract (กลุ่มของเสียง) สำหรับ unvoiced speech หรือ vocal tract ที่ถูกคูณโดย spectrum ของ vocal-cord sound pulse สำหรับเสียงพูด สามารถเขียนเป็น transfer function

$$H(z) = \frac{G}{1 + \sum_{k=1}^{M} a(k) \cdot z^{-k}} \qquad (1.5)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อ G เป็น gain parameter และ z^{-1} เป็น unit-delay operator รูปแบบของตัวกระตุ้นที่ให้กับ filter นี้ ถูกเปลี่ยนโดย switching ระหว่าง voiced และ unvoiced sounds ฉะนั้น filter ที่มี transfer function H(z) ถูกกระตุ้นด้วยอันดับของ impulses เพื่อสร้างเสียงหรือ white-noise sequence เพื่อสร้าง unvoiced sounds

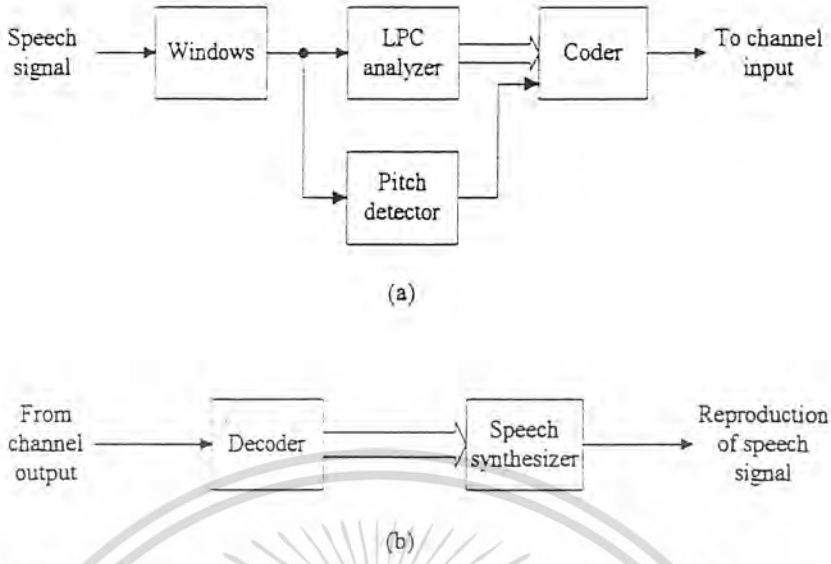
ใน LPC, linear prediction (การคาดหมายแบบเชิงเส้น) ถูกใช้เพื่อประมาณค่า parameters ของเสียง ให้กลุ่มของตัวอย่างในอดีตของสัญญาณเสียงให้ชื่อว่า, u(n-1),u(n-2),...,u(n-M), การคาดหมายเชิงเส้นของ u(n) ที่เป็นค่าตัวอย่างปัจจุบัน ถูกกำหนดโดย

$$\hat{u}(n|n-1,\ldots,n-M)=\sum_{k=1}^{M}h(k)\cdot u(n-k) \qquad (1.6)$$

สัมประสิทธิ์ของตัวคาดหมาย, h(1),h(2)...,h(M) ถูกทำให้ดีที่สุดโดยทำให้ค่าเฉลี่ยกำลังสองของความผิดพลาดในการคาดหมาย e(n) มีค่าต่ำที่สุด, โดยที่ e(n) กำหนดจากความแตกต่างระหว่าง $u(n)$ และ $\hat{u}(n|n-1,\ldots,n-M)$ การใช้บรรทัดฐานความผิดพลาดเฉลี่ยกำลังสองต่ำที่สุดสำหรับการทำให้ส่วนคาดหมายดีที่สุด อาจจะต้องถูกปรับเพื่อให้เข้ากับเหตุผลพื้นฐานสองประการ:

1. ถ้าสัญญาณเสียงพูดสามารถใช้ได้กับแบบจำลองที่อธิบายได้ด้วยสมการ (1.5) และถ้าค่าเฉลี่ยกำลังสองของสัญญาณผิดพลาด e(n) ถูกทำให้ต่ำที่สุดแล้ว เราพบว่า e(n) เท่ากับตัวกระตุ้น x(n) คูณกับ gain parameter G ในแบบจำลองในรูปที่ 1.5 และ a(k)=-h(k), k=1,2....,M ฉะนั้นสัญญาณผิดพลาด e(n) ประกอบด้วยขบวนของ impulses ในกรณีของ voiced sounds หรือเป็น white noise sequenced ในกรณีของ unvoiced sound ในกรณีทั้งคู่, สัญญาณผิดพลาด e(n) อาจจะมีขนาดเล็กมากตลอดเวลา
2. การใช้บรรทัดฐานผิดพลาดเฉลี่ยกำลังสองต่ำสุดจะนำไปสู่คณิตศาสตร์ที่หาได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 1.6 Block diagram of LPC vocoder: (a) transmitter, (b) receiver

รูปที่ 1.6 แสดง block diagram ของ LPC vocoder มันประกอบด้วยเครื่องส่งและเครื่องรับ เครื่องส่งแรกสุดจะส่ง window (โดยปกติมีขนาด 10-30 ms) เข้ากับสัญญาณเสียง input, และกลายเป็น block ของตัวอย่างเสียงพูดสำหรับเข้าขบวนการ window นี้จะสั้นพอที่ทำให้รูปร่างกลุ่มเสียงเกือบจะคงที่, ดังนั้น parameters ของการสร้างเสียงพูดในแบบจำลองที่ 1.5 อาจจะถือว่าเป็นค่าคงที่ที่สำคัญสำหรับช่วงเวลาของ window จากนั้นเครื่องส่งจะวิเคราะห์สัญญาณเสียงพูด input ในลักษณะของ adaptive, block ต่อ block, โดยการกระทำการคาดหมายอย่างเชิงเส้นและการตรวจจับความสูงของเสียง ท้ายสุด มันจะสร้างรหัส parameters: (1) กลุ่มของสัมประสิทธิ์ของส่วนคาดหมาย, (2) ช่วงความสูงของเสียง, (3) gain parameter, และ (4) voiced/unvoiced parameter, สำหรับการส่งไปทั่วช่องส่งสัญญาณ เครื่องรับจะทำปฏิบัติการกลับกัน, โดยแรกสุดถอดรหัส parameter ที่เข้ามา ในจุดนี้, มันจะคำนวณค่าของสัมประสิทธิ์ส่วนคาดหมาย, ช่วงความสูงของเสียงพูด, และ gain parameter, และพิจารณาว่าส่วนที่สนใจเป็นเสียงแบบ voiced หรือ unvoiced สุดท้าย เครื่องรับจะใช้ค่า parameters เหล่านี้ในการสังเคราะห์สัญญาณเสียงพูดโดยการใช้แบบจำลองของรูป 1.5

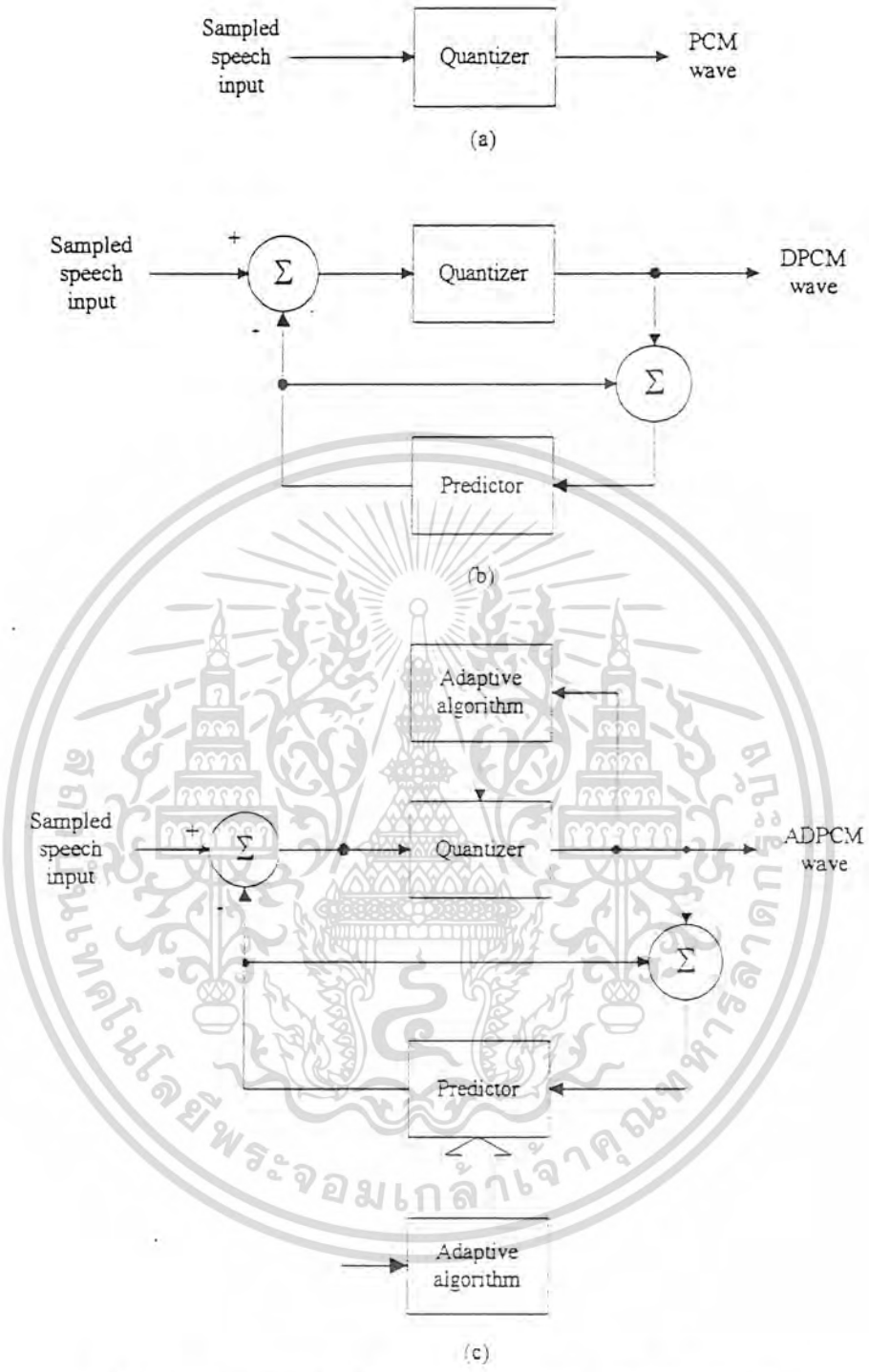
การเข้ารหัสรูปคลื่น (Waveform Coding)

การเข้ารหัสรูปคลื่น ปฏิบัติการที่กระทำบนสัญญาณเสียงถูกออกแบบเพื่อรักษารูปร่างของสัญญาณ โดยเฉพาะการะปฏิบัติการนี้รวมถึง sampling (การสุ่ม - time discretization) และ quantization (การประกอบ - amplitude discretization) อัตราส่วนของการสุ่มจะเป็นตามคุณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สมบัติพื้นฐานของสัญญาณเสียงพูดทั้งหมด,ที่มีชื่อว่า, bandlimited นี้หมายถึงว่าสัญญาณเสียงพูดสามารถสุ่มในเวลาอัตราจำกัด ที่เหมาะสมกับทฤษฎีการสุ่ม ตัวอย่างเช่น เครือข่ายธุรกิจโทรศัพท์ถูกออกแบบสำหรับส่งสัญญาณเสียงพูดที่ใช้ bandwidth จาก 200 ถึง 3200 Hz เพื่อจะเป็นไปตามทฤษฎี, อัตราการสุ่มที่คงไว้ 8 kHz จะถูกใช้โดยทั่วไปในทางปฏิบัติ Quantization ถูกปรับบนพื้นฐานเหล่านี้ และแม้ว่าสัญญาณเสียงพูดจะมีช่วงต่อเนื่องของความสูง (และแน่นอนตัวอย่างของเสียงนี้ก็มีช่วงความสูงที่ต่อเนื่อง), ไม่ว่าอย่างไร, มันก็ไม่จำเป็นที่จะส่งความสูงจริงๆ ของตัวอย่าง โดยพื้นฐานแล้ว, หูของคน (ที่เป็นเครื่องรับท้ายสุด) สามารถตรวจความแตกต่างของความสูงได้จำกัดเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 1.7 ตัวเข้ารหัสรูปคลื่น: (a) PCM, (b) DPCM, (c) ADPCM

ตัวอย่างของการเข้ารหัสรูปคลื่นรวมถึง pulse-code modulation (PCM) และ differential pulse-code modulation (DPCM) ใน PCM, ดังที่ถูกใช้ในการโทรศัพท์, สัญญาณเสียงพูด (หลังจากผ่าน low-pass filter แล้ว) จะถูกสุ่มที่อัตรา 8 kHz, เป็นแบบ quantized ไม่เชิงเส้น, และจากนั้นเข้ารหัสเป็น 8-bit words, ดังในรูป 1.7(a) ผลที่ได้จะเป็น signal-to-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

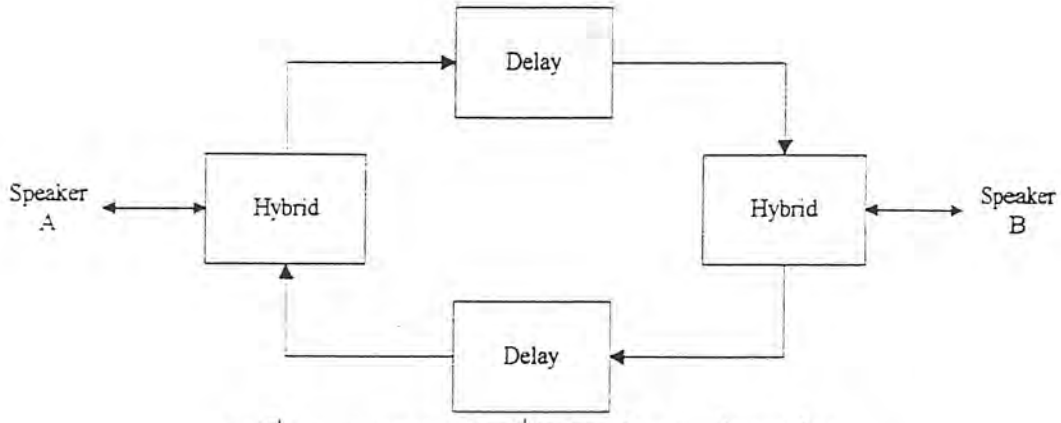
quantization-noise ratio ที่ดีบนช่วงเปลี่ยนแปลงของระดับสัญญาณ input DPCM จะเกี่ยวข้องกับการใช้ส่วนคาดหมายดังในรูป 1.7(b) ส่วนคาดหมายถูกออกแบบเพื่อใช้ประโยชน์จากความสัมพันธ์ซึ่งมีอยู่ระหว่างตัวอย่างข้างเคียงของสัญญาณเสียง, เพื่อที่จะทำให้การลดจำนวนของ bits ที่ถูกใช้สำหรับการส่งแต่ละตัวอย่างของสัญญาณเสียงเป็นจริง และยังคงคุณภาพของคุณสมบัติตามที่กำหนดไว้ สิ่งนี้สามารถทำได้โดย quantizing และจากนั้นเข้ารหัสความผิดพลาดของการคาดหมาย ซึ่งเป็นผลจากการลบกันจาก output ของส่วนคาดหมายจากสัญญาณ input ถ้าส่วนคาดหมายถูกทำให้ดีที่สุดแล้ว, การเปลี่ยนแปลงของความผิดพลาดของส่วนคาดหมายจะน้อยกว่าของสัญญาณ input อย่างมีนัยความสำคัญ, ดังนั้น quantizer และจำนวนของระดับที่ให้ไว้สามารถถูกปรับเพื่อลดความผิดพลาดของ quantizing ด้วยการเปลี่ยนแปลงที่เล็กกว่าที่จะเป็นไปได้ถ้าสัญญาณ input ถูก quantized โดยตรงในระบบ PCM มาตรฐาน เหมือนกัน,สำหรับความผิดพลาดของ quantizing ของการเปลี่ยนที่กำหนดไว้, DPCM ต้องการจำนวนของระดับ quantizing น้อยกว่า (และแน่นอนว่ามีอัตรา bit น้อยกว่า) PCM

DPCM ใช้ quantizer คงที่ และส่วนคาดหมายที่คงที่ การลดอัตราการส่งสามารถทำได้ต่อไปโดยใช้ adaptive quantizer และ adaptive predictor, ดังแสดงในรูป 1.7(c) การเข้ารหัสของรูปคลื่นชนิดนี้ถูกเรียกว่า adaptive differential pulse-code modulation (ADPCM) adaptive predictor ถูกใช้เพื่อที่จะบันทึกธรรมชาติที่ไม่คงที่ของสัญญาณเสียง

ตัวอย่าง 4 Echo Cancellation

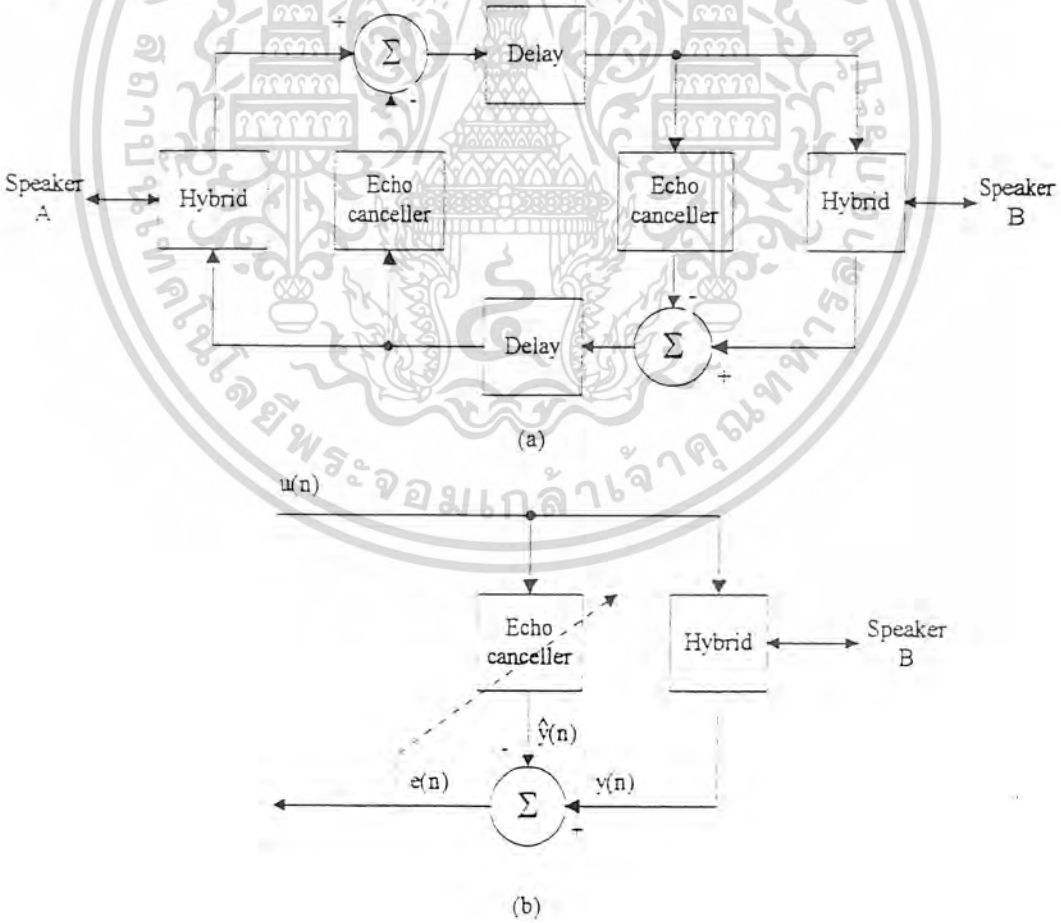
ในการต่อโทรศัพท์ซึ่งเกี่ยวข้องทั้งการส่งแบบสี่สายและส่งแบบสองสาย, echo (เสียงก้อง) นั้นกำเนิดที่ hybrid ซึ่งต่ำกับการส่งแบบสี่สายไปยังสองสาย เมื่อมีการเรียกโทรศัพท์ทางไกล (เช่น โดยการใช้ดาวเทียมประจำที่ geostationary satellites), echo จะแสดงถึงการลดทอนซึ่งอาจจะเป็นสิ่งที่น่ารำคาญมากกว่าเสียงเบาหรือมี noise รูปที่ 1.8 แสดงวงจรดาวเทียมซึ่งไม่มีการป้องกัน echo hybrids ที่ปลายทั้งสองข้างของวงจรเปลี่ยนการส่งแบบสองสายที่ใช้บน customer loops และกลุ่มของสายส่งแบบสี่เส้นเป็นสิ่งจำเป็นสำหรับส่งวงจรพาหะ ช่วงเวลาสูงสุดของดาวเทียม, การหน่วงเวลา 270 ms จะเกิดขึ้นในแต่ละทางบนสายแบบสี่สาย โดยแนวความคิด, เมื่อมีนาย A พูดอยู่ทางซ้าย, เสียงพูดของเขาควรจะตามทางสายส่งด้านบนไปยัง hybrid ทางด้านขวาและจะเข้าสู่วงจรสองสาย อย่างไรก็ตาม,ในทางปฏิบัติ พลังงานเสียงไม่ทั้งหมดที่จะตรงเข้าวงจรสองสาย, เป็นผลให้สัญญาณบางส่วนย้อนกลับทางสี่สายด้านล่างทำให้ได้ยินโดยบุคคลด้านซ้ายจึงเป็น echo ที่มีการหน่วงเวลาเท่ากับ 540 ms

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 1.8 วงจรดาวเทียมที่ไม่มีการป้องกันเสียงสะท้อน

เพื่อจะขนะปัญหานี้, echo cancellers (เครื่องตัดเสียงสะท้อน) จึงถูกติดตั้งไว้ในเครือข่ายเป็นคู่ดังแสดงในรูป 1.9(a) การตัดนี้จะทำได้โดยการประมาณค่าของการสะท้อนและลบมันจากสัญญาณที่กลับมา การสมมติข้างล่างนี้เป็นทางกลับของเสียงสะท้อน, จากจุดที่ canceller ต่อที่จุดที่ค่าประมาณเสียงสะท้อนถูกลบ, เป็นเชิงเส้นและไม่ขึ้นอยู่กับเวลา



รูปที่ 1.9 (a) วงจรดาวเทียมพร้อมกับคู่ของเครื่องกำจัดเสียงสะท้อน (b) นิยามของสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฉะนั้น, โดยการอ้างอิง canceller เดี่ยวในรูป 1.9(b) เมื่อกำหนด, สัญญาณที่ส่งกลับที่เวลา n อาจจะแสดงได้เป็น

$$y(n) = \sum_{k=0}^{\infty} h(k) \cdot u(n-k) + v(n)$$

เมื่อ u(n),u(n-1),...,เป็นตัวอย่างของเสียงพูดที่ไกลสุด (จากผู้พูด A), v(n) เป็นเสียงพูดที่ใกล้ที่สุด (จากผู้พูด B) บวกเข้ากับ noise ที่เพิ่มเข้ามาที่เวลา n และ {h(k)} เป็นการตอบสนอง impulse ของทางเสียงสะท้อน เครื่องตัดเสียงสะท้อนจะทำหน้าที่ประมาณ {h(k)} ของการตอบสนอง impulse ของเส้นทางสะท้อน, และจากนั้นประมาณเสียงสะท้อนออกมาเป็น convolution sum (ผลรวมผสม)

$$\hat{y}(n) = \sum_{k=0}^{M} \hat{h}(k) \cdot u(n-k)$$

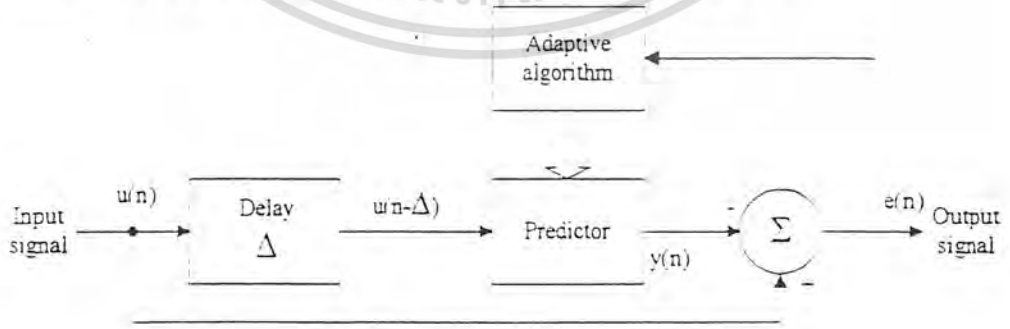
ซึ่งสามารถทำได้โดยใช้ tapped-delay-line filter กับสัมประสิทธิ์ $\hat{h}(0), \hat{h}(1), \ldots, \hat{h}(M)$ สัญญาณผิดพลาด e(n) จะถูกสร้างโดยการลบค่าประมาณ $\hat{y}(n)$ จากสัญญาณย้อนกลับ $y(n)$ ดังแสดงโดย

$$e(n) = y(n) - \hat{y}(n)$$

สัญญาณผิดพลาด e(n) ใช้สำหรับควบคุมการเปลี่ยนแปลงของสัมประสิทธิ์ของ canceller $\hat{h}(0), \hat{h}(1), \ldots, \hat{h}(M)$ ดังนั้นหลังจากที่ผ่านการคำนวณซ้ำๆเล็กน้อยแล้ว จะรู้สึกว่าผลของการสะท้อนจะน้อยที่สุด

ตัวอย่างที่ 5 Adaptive Line Enhancer

Adaptive line enhancer (ALE) เป็นอุปกรณ์ที่สามารถใช้ตรวจสอบคลื่น sine ระดับต่ำที่ถูกฝังไว้ในส่วนหลังของ noise ที่เพิ่มขึ้นมาบน spectrum ที่มี band กว้าง



รูปที่ 10 Adaptive line enhancer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
1036925

ดังรูปภาพที่ 1.10, ALE ประกอบด้วยอุปกรณ์หน่วงเวลาและ linear predictor (ส่วนคาดหมายเชิงเส้น) output ของ predictor y(n) จะถูกลบกับสัญญาณ u(n) เพื่อสร้างสัญญาณผิดพลาด e(n) สัญญาณผิดพลาดนี้จะถูกใช้เพื่อควบคุมการเปลี่ยนแปลงของสัมประสิทธิ์ของ predictor ค่า input ของ predictor เท่ากับ u(n-D), สัญญาณ input ดั้งเดิม u(n) ถูกหน่วงด้วย D วินาที, เมื่อ D นั้นเท่ากับหรือมากกว่าช่วงของการสุ่ม หน้าที่หลักของ delay parameter D คือ ตัดความสัมพันธ์ซึ่งอาจจะมีอยู่ระหว่างองค์ประกอบของ noise ในสัญญาณ input ดั้งเดิม u(n) และองค์ประกอบ noise ใน predictor input ที่หน่วงเวลาแล้ว u(n-D) สำหรับเหตุผลนี้, ค่า delay parameter D จึงถูกเรียกอีกชื่อว่า decorrelation parameter ของ ALE ฉะนั้น ALE อาจจะถูกมองเป็น adaptive filter ซึ่งถูกออกแบบเพื่อตัดองค์ประกอบแบบ band กว้าง (เช่น White noise) ที่บรรจุอยู่ใน input และในเวลาเดียวกันองค์ประกอบที่มีแบนแคบ (เช่น sine waves) ถูกส่งด้วยการลดทอนเล็กน้อย ในพูดได้อีกอย่างหนึ่งคือ มันสามารถถูกใช้เพื่อเพิ่มความประสิทธิภาพการปรากฏของ sine waves (ซึ่ง spectrum ประกอบด้วย harmonic lines) ในลักษณะของการเปลี่ยนแปลง - จึงเป็นที่มาของชื่อนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
การออกแบบ

โครงงานนี้เป็นการประยุกต์ adaptive algorithm เข้ากับ IIR Digital Bandpass filter อันดับสองที่มีค่า Q-factor คงที่เพื่อออกแบบฟิลเตอร์ที่มีความสามารถปรับค่า Q-factor ได้

สมมติสัญญาณทางด้านอินพุทเป็น

$$x(k) = A \cdot \cos(k \cdot \omega) + n(k) \tag{1}$$

ซึ่ง ω มีคาบเวลา T (วินาที) เป็น 1 เสมอ ตัวแปร k เป็นเลขจำนวนเต็มของเวลาการ sampling A $\cos(k\omega)$ เป็นสัญญาณ input ที่ไม่ทราบค่า (A คือค่าคงที่, ω เป็นความเร็วเชิงมุม (rad/sec) มีค่าอยู่ในช่วง $0 < \omega < \pi$) และ n(k) เป็น Gaussian noise

Transfer function ของวงจรกรองความถี่กลางผ่านแบบ IIR อันดับสองคือ

$$H(z) = \frac{1-\alpha_0}{2} \cdot \frac{1-z^{-2}}{1-\alpha_1(k)\cdot(1+\alpha_0)\cdot z^{-1}+\alpha_0 \cdot z^{-2}} \tag{2}$$

โดยมี ω_0 เป็นความถี่กลางและ $\alpha_1(k)$ มีค่าอยู่ระหว่าง $-1 < \alpha_1(k) < 1$ เป็นสัมประสิทธิ์แปรค่าที่ใช้สำหรับกำหนด ω_0

$$\alpha_1 = \cos(\omega_0) \tag{3}$$

α_0 มีค่าอยู่ระหว่าง $0 < \alpha_0 < 1$ เป็นสัมประสิทธิ์คงที่สำหรับกำหนดค่า Q-factor ของวงจรกรองความถี่

เพื่อนำ adaptive algorithm มาใช้งาน จึงต้องมีการศึกษาค่า parameters แต่ละตัวของ $H(z)$ โดยแทน z ด้วย $z=e^{j\omega}$ เมื่อ ω มีค่าระหว่าง $0 \le \omega \le \pi$ แล้ว plot กราฟของ $|H(e^{j\omega})|$ เทียบกับแกน ω พบว่าค่าของ α_0 (มีค่าระหว่าง $0 \le \alpha_0 \le 1$) จะมีผลต่อการเปลี่ยนแปลงของ Q-factor โดยมีความสัมพันธ์ที่แปรผันต่อกัน คือเมื่อค่า α_0 เข้าใกล้ศูนย์แล้ว ค่าของ Q-factor มีน้อยลง กราฟของ $|H(e^{j\omega})|$ จะมีความกว้างมาก และเมื่อค่าของ α_0 มีค่าเข้าใกล้หนึ่งแล้ว ค่าของ Q-factor จะมีค่าสูงขึ้น (กราฟของ $|H(e^{j\omega})|$ มีความกว้างน้อยลง)

ค่าของ α_1 จะมีผลทำให้ความถี่กลาง ω_0 เบี่ยงเบนไป โดยที่ค่าของ $\alpha_1 = 0$ จะทำให้ ω_0 อยู่ตรงกลางพอดี และ α_1 เป็นลบ ความถี่ ω_0 จะเบี่ยงไปทางซ้าย และ α_1 เป็นบวก ความถี่ ω_0 จะเบี่ยงไปทางขวา

เพื่อหาการเปลี่ยนแปลงของ $\alpha_1(k)$ เมื่อมีสัญญาณอินพุทป้อนเข้ามา จึงต้องหาเอ้าท์พุท $y(k)$ เมื่อ $Y(z)$ เป็น Z-Transform ของ $y(k)$ ความสัมพันธ์ของ Transfer function จะเป็น $H(z)=Y(z)/X(z)$ หรือ $Y(z)=H(z)X(z)$ เมื่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จาก $$Y(z)=\left(\frac{1-\alpha_0}{2}\cdot\frac{1-z^{-2}}{1-\alpha_1(k)\cdot(1+\alpha_0)\cdot z^{-1}-\alpha_0\cdot z^{-2}}\right)\cdot X(z)$$

$$Y(z)\cdot\left[1-\alpha_1(k)\cdot(1+\alpha_0)\cdot z^{-1}+\alpha_0\cdot z^{-2}\right]=\left(\frac{1-\alpha_0}{2}\right)\cdot(1-z^{-2})\cdot X(z)$$

$$Y(z)-\alpha_1(k)\cdot(1+\alpha_0)\cdot Y(z)\cdot z^{-1}+\alpha_0\cdot Y(z)\cdot z^{-2}=\left(\frac{1-\alpha_0}{2}\right)\cdot\left(X(z)-X(z)\cdot z^{-2}\right) \quad (4)$$

Take Inverse Z-Transform

$$y(k)-\alpha_1(k)\cdot(1+\alpha_0)\cdot y(k-1)+\alpha_0\cdot y(k-2)=\left(\frac{1-\alpha_0}{2}\right)\cdot\left[x(k)-x(k-2)\right]$$

จะได้ $$y(k)=\left(\frac{1-\alpha_0}{2}\right)\cdot\left[x(k)-x(k-2)\right]+\alpha_1(k)\cdot(1+\alpha_0)\cdot y(k-1)-\alpha_0\cdot y(k-2) \quad (5)$$

ค่าเฉลี่ยกำลังสองของ $y^2(k)$ จะเป็นสัดส่วนโดยตรงกับ $|H(e^{j\cdot\omega})|^2$ ที่ $z=e^{j\omega}$ ค่า $\alpha_1(k)$ ที่ทำให้ $y^2(k)$ มีค่าสูงสุด กำหนดด้วยสมการ

$$\alpha_1(k+1)=\alpha_1(k)-\frac{\mu_1}{2}\cdot\frac{\partial y^2(k)}{\partial\alpha_1(k)}$$

$$=\alpha_1(k)-\mu_1\cdot y(k)\cdot\psi(k) \quad (6)$$

เมื่อ $\psi(k)=\frac{\partial y(k)}{\partial\alpha_1(k)}$ เป็นสัญญาณควบคุม adaptive ของ $\alpha_1(k)$

μ_1 เป็น step size parameter ถ้าให้ค่านี้เล็กพอจะทำให้การเปลี่ยนแปลงของ $y(k)$ ในแต่ละครั้งมีค่าน้อย เราสามารถประมาณสัญญาณควบคุม adaptive ได้ดังนี้

$$\psi(k-i)=\frac{\partial y(k-i)}{\partial\alpha_1(k-i)}\approx\frac{\partial y(k-i)}{\partial\alpha_1(k)} \quad \text{เมื่อ } i \text{ เป็นจำนวนเต็มบวกใดๆ} \quad (7)$$

หาค่า $\psi(k)$ จากสมการที่ 3 และ 5 ได้

$$\psi(k)=\alpha_1(k)\cdot(1+\alpha_0)\cdot\psi(k-1)-\alpha_0\cdot\psi(k-2)-(1+\alpha_0)\cdot y(k-1) \quad (8)$$

พิจารณาความเร็วในการ converge ของ $\alpha_1(k)$ โดยการ plot กราฟ $\alpha_1(k)$ เทียบกับแกนเวลา k จะพบว่า

- ความละเอียดของค่า μ_1 จะส่งผลให้ต่อการเกิด noise ถ้าค่านี้มีความละเอียดมาก noise จะลดลง แต่การ converge ก็จะช้าลงด้วย
- ความเร็วเชิงมุม ω_0 ที่มีค่ามากจะทำให้การ converge เร็วกว่าค่า ω_0 ที่มีค่าน้อย
- ความสูงของสัญญาณอินพุทที่เข้ามาก็มีผลต่อการ converge โดยที่ถ้าความแรงของสัญญาณอินพุทมีมากก็จะทำให้การ converge เร็วขึ้นด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- α_0 ค่ามาก (เข้าใกล้หนึ่ง) จะทำให้มีการแกว่งของสัญญาณ แต่การ converge จะเร็ว เมื่อเข้าใกล้สถานะคงที่ และเมื่อค่า α_0 ค่าน้อย (เข้าใกล้ศูนย์) การ converge ในช่วงเริ่มต้นเร็วและแทบไม่มีการแกว่งของสัญญาณ

การเปลี่ยนแปลงค่าของ $\alpha_1(k)$ จะเร็วขึ้นได้โดยการปรับค่า α_0 ให้เปลี่ยนแปลงไปด้วยวิธีการ Stochastic gradient เนื่องจากว่า $\alpha_1(k)$ จะติดตามความถี่ได้เร็วหรือไม่นั้นขึ้นอยู่กับค่าของ $y(k)^*\psi(k)$ และค่าของ $y(k)$ และ $\psi(k)$ ต่างก็มาจาก $H(z)$ ดังนั้น $[y(k)^*\psi(k)]^2$ ก็จะเป็นสัดส่วนโดยตรงกับ $\dfrac{\partial\left|H(\alpha_1(k))\right|^2}{\partial\alpha_1(k)}$

เพื่อใช้วิธีการ Stochastic gradient จะต้องพิสูจน์ว่าเทอม $\dfrac{\partial\left|H(\alpha_1(k))\right|^2}{\partial\alpha_1(k)}$ เมื่อ plot กับ α_0 แล้วจะต้องมียอดสูงสุดของกราฟเพียงยอดเดียวเท่านั้น จึงต้องหาค่าของเทอมนี้เพื่อนำมาเขียนกราฟ จาก

$$H(z)=\frac{1-\alpha_0}{2}\cdot\frac{1-z^{-2}}{1-\alpha_1(k)\cdot(1+\alpha_0)\cdot z^{-1}+\alpha_0\cdot z^{-2}}$$

แทน $z=e^{j\omega}$;

$$H(e^{j\omega})=\frac{1-\alpha_0}{2}\cdot\frac{1-\left(e^{j\omega}\right)^{-2}}{1-\alpha_1(k)\cdot(1+\alpha_0)\cdot\left(e^{j\omega}\right)^{-1}+\alpha_0\cdot\left(e^{j\omega}\right)^{-2}} \tag{9}$$

คูณพจน์บนและล่างด้วย $e^{j\omega}$ ทั้งเศษและส่วน

$$H(e^{j\omega})=\frac{1-\alpha_0}{2}\cdot\frac{e^{j\omega}-e^{-j\omega}}{e^{j\omega}-\alpha_1(k)\cdot(1+\alpha_0)+\alpha_0\cdot e^{-j\omega}} \tag{10}$$

จากกฎของ euler $e^{-j\theta}=\cos(\theta)\pm j\cdot\sin(\theta)$

$$H(e^{j\omega})=\frac{1-a_0}{2}\cdot\frac{\cos(\omega)+j\cdot\sin(\omega)-\cos(\omega)+j\cdot\sin(\omega)}{\cos(\omega)+j\cdot\sin(\omega)-\alpha_1(k)\cdot(1+\alpha_0)+\alpha_0\cdot\cos(\omega)-j\cdot\alpha_0\cdot\sin(\omega)} \tag{11}$$

Simplified

$$H(e^{j\omega})=\frac{j\cdot\sin(\omega)}{\left\{\cos(\omega)-\alpha_1(k)\right\}\left(\dfrac{1+\alpha_0}{1-\alpha_0}\right)+j\cdot\sin(\omega)} \tag{12}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หาค่าสัมบูรณ์

$$\left|H(e^{j\cdot\omega})\right| = \frac{\sin^2(\omega)}{\sqrt{\left[\left\{\cos(\omega)-\alpha_1(k)\right\}\left(\frac{1+\alpha_0}{1-\alpha_0}\right)\right]^2+\sin^2(\omega)}}$$
$$= \frac{1}{\sqrt{\left[\frac{\left\{\cos(\omega)-\alpha_1(k)\right\}}{\sin(\omega)}\left(\frac{1+\alpha_0}{1-\alpha_0}\right)\right]^2+1}} \quad (13)$$

ยกกำลังสองทั้งสองข้าง

$$\left|H(e^{j\cdot\omega})\right|^2 = \left[\left\{\frac{\left(\cos(\omega)-\alpha_1(k)\right)\left(1+\alpha_0\right)}{\sin(\omega)\cdot\left(1-\alpha_0\right)}\right\}^2+1\right]^{-1} \quad (14)$$

จะได้

$$H(\alpha_0) = \frac{\partial\left|H(e^{j\cdot\omega})\right|^2}{\partial\alpha_1(k)} = 2\cdot\frac{\left\{\frac{1+\alpha_0}{\sin(\omega)\cdot\left(1-\alpha_0\right)}\right\}^2\left\{\cos(\omega)-\alpha_1(k)\right\}}{\left[\left\{\frac{\left(\cos(\omega)-\alpha_1(k)\right)\left(1+\alpha_0\right)}{\sin(\omega)\cdot\left(1-\alpha_0\right)}\right\}^2+1\right]^2} \quad (15)$$

เมื่อนำฟังชันก์ (15) ไปเขียนกราฟ เมื่อ α_0 เปลี่ยนค่าจาก 0 ถึง 1 พบว่า กราฟที่ได้จะมียอดสูงที่สุดเพียงหนึ่งยอดเท่านั้น หมายความว่าสามารถใช้วิธี Stochastic gradient เพื่อหาสมการเปลี่ยนแปลงของ α_0 ได้ ดังนั้น α_0 ใหม่จะเป็น

$$\alpha_0(k+1) = \alpha_0(k) + \frac{\mu_0}{2}\cdot\frac{\partial\left(y(k)\cdot\psi(k)\right)^2}{\partial\alpha_0(k)} \quad (16)$$

$$\alpha_0(k+1) = \alpha_0(k) + \mu_0 y(k)\cdot\psi(k)\cdot\left[\psi(k)\cdot\psi_{01}(k)+\psi_{02}(k)\cdot y(k)\right] \quad (17)$$

โดยที่ $\psi_{01}(k) = \frac{\partial y(k)}{\partial\alpha_0(k)}$ และ $\psi_{02}(k) = \frac{\partial\psi(k)}{\partial\alpha_0(k)}$

เมื่อ μ_0 เป็น step size parameter

และ ψ_{01}, ψ_{02} เป็นสัญญาณควบคุมแบบ Adaptive ของ α_0(k)

จาก $y(k) = \left(\frac{1-\alpha_0}{2}\right)\cdot\left[x(k)-x(k-2)\right]+\alpha_1(k)\cdot\left(1+\alpha_0\right)\cdot y(k-1)-\alpha_0\cdot y(k-2)$

จะได้ $\psi_{01}(k) = \frac{\partial y(k)}{\partial\alpha_0(k)}$

$$= \frac{1}{2}\left[x(k-2)-x(k)\right]+\alpha_1(k)\cdot y(k-1)-y(k-2)+$$
$$\alpha_1(k)\cdot\left[1+\alpha_0(k)\right]\psi_{01}(k-1)-\alpha_0(k)\cdot\psi_{01}(k-2) \quad (18)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และจาก $$\psi(k)=\alpha_1(k)\cdot\left(1+\alpha_0\right)\cdot\psi(k-1)-\alpha_0\cdot\psi(k-2)+\left(1+\alpha_0\right)\cdot y(k-1)$$

จะได้ $$\psi_{02}(k)=\frac{\partial\psi(k)}{\partial\alpha_0(k)}$$

$$=\alpha_1(k)\cdot\left[1+\alpha_0(k)\right]\psi_{02}(k-1)+\alpha_1(k)\cdot\psi(k-1)-\alpha_0(k)\cdot\psi_{02}(k-2)-$$
$$\psi(k-2)+\left[1+\alpha_0(k)\right]\psi_{01}(k-1)+y(k-1) \quad (19)$$

เมื่อนำเอาทั้งสองฟังก์ชั่นไปแทนใน $\alpha_0(k)$ สำหรับค่าของ μ_0 ในฟังก์ชั่น $\alpha_0(k)$ จะต้องให้มีค่ามากกว่าค่า μ_1 ของฟังก์ชั่น $\alpha_1(k)$ เพื่อที่จะทำให้การลู่เข้า (Converge) ของฟังก์ชั่น $\alpha_0(k)$ เร็วกว่า $\alpha_1(k)$ แต่ต้องคำนึงถึงเสถียรภาพ เพราะถ้าค่า μ ทั้งสองแตกต่างกันมากเกินไปจะทำให้มีริปเปิล (ripple) เกิดขึ้นเมื่อฟังก์ชั่น $\alpha_1(k)$ converge และค่าของ $\alpha_0(k)$ ที่หาได้จากฟังก์ชั่นนี้นั้น จะมีค่าอยู่ระหว่าง $-1 \le \alpha_0(k) \le 1$ แต่ว่าค่า $\alpha_0(k)$ ที่ใช้กับแบนพาสฟิลเตอร์แบบไอไออาร์สำหรับโครงงานนี้ $\alpha_0(k)$ จะมีค่าอยู่ระหว่าง $0 < \alpha_0(k) < 1$ เท่านั้น ดังนั้นการใช้งานเราจะต้องควบคุมค่านี้ด้วย

สำหรับค่า $\alpha_0(k)$ ที่จะทำให้ $\alpha_1(k)$ มีค่าสูงสุด จะหาได้จาก $\alpha_{0,opt}(k)=\frac{\partial H_1(\alpha_0(k))}{\partial\alpha_0(k)}=0$

จาก $$H(\alpha_0)=\frac{\partial\left|H(e^{j\omega})\right|^2}{\partial\alpha_1(k)}=2\cdot\frac{\left\{\frac{1+\alpha_0}{\sin(\omega)\cdot\left(1-\alpha_0\right)}\right\}^2\left\{\cos(\omega)-\alpha_1(k)\right\}}{\left[\left\{\frac{\left(\cos(\omega)-\alpha_1(k)\right)\left(1+\alpha_0\right)}{\sin(\omega)\cdot\left(1-\alpha_0\right)}\right\}^2+1\right]^2}$$

กำหนดให้ $h(\alpha_0)=\frac{1+\alpha_0}{1-\alpha_0}$ $A=\frac{1}{\sin(\omega)}$ และ $B=\left(\cos(\omega)-\alpha_1(k)\right)$

จะได้ $$H(\alpha_0)=2\cdot\frac{A^2h^2(\alpha_0)\cdot B}{\left\{A^2B^2h^2(\alpha_0)+1\right\}^2}$$

ดังนั้น $$\frac{\partial}{\partial\alpha_0(k)}\left[\frac{A^2h^2(\alpha_0)\cdot B}{\left\{A^2B^2h^2(\alpha_0)+1\right\}^2}\right]=0$$

$$2\cdot A^2B\cdot\frac{\left[\begin{array}{l}\left(A^2B^2h^2(\alpha_0)+1\right)^2\left(2\cdot h(\alpha_0)\cdot h'(\alpha_0)\right)\\-h^2(\alpha_0)\cdot\left\{2\cdot\left(A^2B^2h^2(\alpha_0)+1\right)\left(A^2B^2\,2\cdot h(\alpha_0)\cdot h'(\alpha_0)\right)\right\}\end{array}\right]}{\left(A^2B^2h^2(\alpha_0)+1\right)^4}=0$$

$$2\cdot A^2B\frac{\left[\begin{array}{l}2\cdot\left(A^2B^2h^2(\alpha_0)+1\right)^2h(\alpha_0)\cdot h'(\alpha_0)\\-4\cdot\left(A^2B^2h^2(\alpha_0)+1\right)h^3(\alpha_0)\cdot A^2B^2h'(\alpha_0)\end{array}\right]}{\left(A^2B^2h^2(\alpha_0)+1\right)^4}=0$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$2 \cdot A^2 B \cdot \frac{\left[2 \cdot h(\alpha_0) \cdot h'(\alpha_0) \cdot \left\{A^2 B^2 h^2(\alpha_0) - 1 - 2 \cdot A^2 B^2 h^2(\alpha_0)\right\}\right]}{\left(A^2 B^2 h^2(\alpha_0) - 1\right)^3} = 0$$

$$\frac{4 \cdot A^2 B^2 h(\alpha_0) \cdot h'(\alpha_0) \cdot \left(1 - A^2 B^2 h^2(\alpha_0)\right)}{\left(A^2 B^2 h^2(\alpha_0) + 1\right)^3} = 0$$

จะมีอยู่ 2 กรณีคือ $4 \cdot A^2 B \cdot h(\alpha_0) \cdot h'(\alpha_0) = 0$ และ $1 - A^2 B^2 h^2(\alpha_0) = 0$

จาก $4 \cdot A^2 B \cdot h(\alpha_0) \cdot h'(\alpha_0) = 0$

$$h(\alpha_0) \cdot h'(\alpha_0) = 0$$

$$h'(\alpha_0) = \frac{\partial}{\partial \alpha_0} h(\alpha_0) = \frac{\partial}{\partial \alpha_0} \frac{\left(1+\alpha_0\right)}{\left(1-\alpha_0\right)}$$

$$\therefore \quad h'(\alpha_0) = \frac{2}{\left(1-\alpha_0\right)^2}$$

ได้ $$\left(\frac{1+\alpha_0}{1-\alpha_0}\right) \frac{2}{\left(1-\alpha_0\right)^2} = 0$$

$$\frac{\left(1-\alpha_0^2\right)2}{\left(1-\alpha_0\right)^3} = 0$$

$$1 - \alpha_0^2 = 0$$

$$\therefore \quad \alpha_0 = \pm 1$$

และจาก $1 - A^2 B^2 h^2(\alpha_0) = 0$

$$A^2 B^2 \left(\frac{1+\alpha_0}{1-\alpha_0}\right)^2 = 1$$

$$A \cdot B \cdot \left(1-\alpha_0\right) = \pm\left(1-\alpha_0\right)$$

<u>กรณีที่ 1</u> $A \cdot B \cdot \left(1-\alpha_0\right) = 1-\alpha_0$

$$A \cdot B - A \cdot B \cdot \alpha_0 = 1 - \alpha_0$$

$$\alpha_0 = \frac{1 - A \cdot B}{1 - A \cdot B}$$

$$A \cdot B = \frac{\cos(\omega) - \alpha_1(k)}{\sin(\omega)}$$

$$\alpha_0 = \frac{1 - \dfrac{\cos(\omega) - \alpha_1(k)}{\sin(\omega)}}{1 + \dfrac{\cos(\omega) - \alpha_1(k)}{\sin(\omega)}}$$

$$\therefore \quad \alpha_0 = \frac{\sin(\omega) - \cos(\omega) - \alpha_1(k)}{\sin(\omega) + \cos(\omega) - \alpha_1(k)}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<u>กรณีที่ 2</u>

$$A \cdot B \cdot \left(1+\alpha_0\right) = \alpha_0 - 1$$

$$\alpha_0 A \cdot B + A \cdot B = \alpha_0 - 1$$

$$\alpha_0 = \frac{1 + A \cdot B}{1 - A \cdot B}$$

$$= \frac{1 + \dfrac{\cos(\omega) - \alpha_1(k)}{\sin(\omega)}}{1 - \dfrac{\cos(\omega) - \alpha_1(k)}{\sin(\omega)}}$$

$$\therefore \quad \alpha_0 = \frac{\sin(\omega) + \cos(\omega) - \alpha_1(k)}{\sin(\omega) - \cos(\omega) - \alpha_1(k)}$$

ทดสอบโดยการกำหนดให้ $\alpha_1(k) > \cos(\omega)$ และ $\alpha_1(k) < \cos(\omega)$ พบว่า

$$\alpha_{0,opt}(k) = \frac{\sin(\omega) - \cos(\omega) - \alpha_1(k)}{\sin(\omega) + \cos(\omega) - \alpha_1(k)} \quad \text{เมื่อ} \quad \alpha_1(k) < \cos(\omega) \tag{20}$$

$$\alpha_{0,opt}(k) = \frac{\sin(\omega) + \cos(\omega) - \alpha_1(k)}{\sin(\omega) - \cos(\omega) - \alpha_1(k)} \quad \text{เมื่อ} \quad \alpha_1(k) > \cos(\omega) \tag{21}$$

ลักษณะฟังก์ชันของ $\psi(k)$, $\psi_{01}(k)$ และ $\psi_{02}(k)$ จะขึ้นอยู่กับฟังก์ชันหลายตัว ไม่ได้ขึ้นอยู่กับฟังก์ชันตัวใดตัวหนึ่ง ถ้าสามารถเปลี่ยนฟังก์ชันเหล่านี้ให้ขึ้นอยู่กับฟังก์ชันตัวใดตัวหนึ่งได้ ก็จะทำให้วงจรมีความง่ายขึ้น สามารถทำได้โดยการหาทรานสเฟอร์ฟังก์ชันของฟังก์ชันเหล่านี้

ทรานสเฟอร์ฟังก์ชันของ $\psi(k)$ เขียนสัญลักษณ์ด้วย $G(z)$ ที่อยู่ในรูปของ Z-transform หาได้จาก

$$\psi(k) = \alpha_1(k) \cdot \left(1 - \alpha_0(k)\right) \cdot \psi(k-1) - \alpha_0(k) \cdot \psi(k-2) + \left(1 - \alpha_0(k)\right) \cdot y(k-1)$$

Take Z-Transform

$$\psi(z) = \alpha_1(k) \cdot \left(1 - \alpha_0(k)\right) \cdot \psi(z) \cdot z^{-1} - \alpha_0(k) \cdot \psi(z) \cdot z^{-2} + \left(1 + \alpha_0(k)\right) \cdot y(z) \cdot z^{-1}$$

$$\psi(z) \cdot \left[1 - \alpha_1(k) \cdot \left(1 + \alpha_0(k)\right) \cdot z^{-1} - \alpha_0(k) \cdot z^{-2}\right] = \left(1 + \alpha_0(k)\right) \cdot z^{-1} \cdot y(z)$$

$$G(z) = \frac{\psi(z)}{y(z)} = \frac{\left(1 + \alpha_0(k)\right) \cdot z^{-1}}{1 - \alpha_1(k) \cdot \left(1 - \alpha_0(k)\right) \cdot z^{-1} + \alpha_0(k) \cdot z^{-2}} \tag{22}$$

ทรานสเฟอร์ฟังก์ชันของ $\psi_{01}(k)$ เขียนสัญลักษณ์ด้วย $G_{01}(z)$ หาได้จาก

$$\psi_{01}(k) = \frac{1}{2} \cdot \left[x(k-2) - x(k)\right] + \alpha_1(k) \cdot y(k-1) - y(k-2) + \alpha_1(k) \cdot \left[1 + \alpha_0(k)\right] \cdot \psi_{01}(k-1)$$
$$- \alpha_0(k) \cdot \psi_{01}(k-2) \tag{23}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เปลี่ยน $x(k)$ ให้อยู่ในรูปของ $y(k)$

$$y(k)=\left(\frac{1-\alpha_0(k)}{2}\right)\cdot\left[x(k)-x(k-2)\right]+\alpha_1(k)\cdot\left(1+\alpha_0(k)\right)\cdot y(k-1)-\alpha_0(k)\cdot y(k-2) \quad (24)$$

$$\frac{1}{2}\left[x(k-2)-x(k)\right]=-\frac{y(k)}{1-\alpha_0(k)}+\frac{1+\alpha_0(k)}{1-\alpha_0(k)}\alpha_1(k)\cdot y(k-1)-\frac{\alpha_0(k)}{1-\alpha_0(k)}\cdot y(k-2)+$$

$$_1(k)\cdot y(k-1)-y(k-2) \quad (25)$$

หา $\frac{\psi_{01}(k)}{y(k)}=G_{01}(z)$

$$\psi_{01}(k)-\alpha_1(k)\cdot\left(1+\alpha_0(k)\right)\cdot\psi_{01}(k-1)+\alpha_0(k)\cdot\psi_{01}(k-2)=-\frac{y(k)}{1-\alpha_0(k)}+$$

$$\frac{1+\alpha_0(k)}{1-\alpha_0(k)}\cdot\alpha_1(k)\cdot y(k-1)+\frac{\alpha_0(k)}{1-\alpha_0(k)}\cdot y(k-2)-\alpha_1(k)\cdot y(k-1)-y(k-2) \quad (26)$$

Take Z-Transform

$$\psi_{01}(z)\cdot\left[1-\alpha_1(k)\cdot\left(1+\alpha_0(k)\right)\cdot z^{-1}+\alpha_0(k)\cdot z^{-2}\right]=y(z)\cdot$$

$$\frac{1-\alpha_0(k)}{1-\alpha_0(k)}\cdot\alpha_1(k)\cdot z^{-1}-\frac{1-\alpha_0(k)\cdot z^{-1}}{1-\alpha_0(k)}-\frac{\alpha_1(k)\cdot\left(1-\alpha_0(k)\right)}{1-\alpha_0(k)}\cdot z^{-1}-\frac{1-\alpha_0(k)}{1-\alpha_0(k)}\cdot z^{-2}\Bigg]$$

$$G_{01}(z)=\frac{\psi_{01}(z)}{y(z)}=\frac{2\cdot\alpha_1(k)\cdot z^{-1}-z^{-2}-1}{\left(1-\alpha_0(k)\right)\cdot\left[1-\alpha_1(k)\cdot\left(1-\alpha_0(k)\right)\cdot z^{-1}+\alpha_0(k)\cdot z^{-2}\right]} \quad (27)$$

ทราบสเพอร์ฟังก์ชันของ $\psi_{02}(k)$ เขียนสัญลักษณ์ย่อย $G_{02}(z)$ หาได้จาก

$$\psi_{02}(k)=\alpha_1(k)\cdot\left[1-\alpha_0(k)\right]\cdot\psi_{02}(k-1)-\alpha_1(k)\cdot\psi(k-1)-\alpha_0(k)\cdot\psi_{02}(k-2)-\psi(k-2)$$

$$-\left[1-\alpha_1(k)\right]\cdot\psi_{01}(k-1)-y(k-1) \quad (28)$$

เปลี่ยน $\psi_{01}(k)$, $y(k)$ ให้อยู่ในเทอมของ $\psi(k)$

จาก $\psi(k)=\alpha_1(k)\cdot\left(1+\alpha_0(k)\right)\cdot\psi(k-1)-\alpha_0(k)\cdot\psi(k-2)+\left(1+\alpha_0(k)\right)\cdot y(k-1)$

จะได้ $y(k-1)=\frac{\psi(k)}{1-\alpha_0(k)}-\alpha_1(k)\cdot\psi(k-1)+\frac{\alpha_0(k)}{1-\alpha_0(k)}\cdot\psi(k-2)$ (29)

นำไปแทนลงใน $\psi_{02}(k)$ เป็น

$$\psi_{02}(k)=\alpha_1(k)\cdot\left[1+\alpha_0(k)\right]\cdot\psi_{02}(k-1)-\alpha_1(k)\cdot\psi(k-1)-\alpha_0(k)\cdot\psi_{02}(k-2)-\psi(k-2)$$

$$-\left[1+\alpha_0(k)\right]\cdot\psi_{01}(k-1)+\frac{\psi(k)}{1+\alpha_0(k)}-\alpha_1(k)\cdot\psi(k-1)+\frac{\alpha_0(k)}{1+\alpha_0(k)}\cdot\psi(k-2) \quad (30)$$

Take Z-Transform

$$\psi_{02}(z)\cdot\left[1-\alpha_1(k)\cdot\left(1+\alpha_0(k)\right)\cdot z^{-1}-\alpha_0(k)\cdot z^{-2}\right]=\alpha_1(k)\cdot\psi(z)\cdot z^{-1}-\psi(z)\cdot z^{-2}+$$

$$\frac{\psi(z)}{1-\alpha_0(k)}-\alpha_1(k)\cdot\psi(z)\cdot z^{-1}+\frac{\alpha_0(k)}{1+\alpha_0(k)}\cdot\psi(z)\cdot z^{-2}+\left(1+\alpha_0(k)\right)\cdot\psi_{01}(z)\cdot z^{-1} \quad (31)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\psi_{01}(z)=\frac{2\cdot\alpha_1(k)\cdot z^{-1}-z^{-2}-1}{\left(1-\alpha_0(k)\right)\left[1-\alpha_1(k)\cdot\left(1+\alpha_0(k)\right)z^{-1}+\alpha_0(k)\cdot z^{-2}\right]}\cdot y(z)$$

$$y(z)\cdot z^{-1}=\frac{\psi(z)}{1+\alpha_0(k)}-\alpha_1(k)\cdot\psi(z)\cdot z^{-1}+\frac{\alpha_0(k)}{1+\alpha_0(k)}\cdot\psi(z)\cdot z^{-2}$$

$$\psi_{01}(z)=\frac{\left(2\cdot\alpha_1(k)\cdot z^{-1}-z^{-2}-1\right)\cdot\left(1-\alpha_1(k)\cdot\left(1+\alpha_0(k)\right)\cdot z^{-1}+\alpha_0(k)\cdot z^{-2}\right)\cdot\psi(z)}{\left(1-\alpha_0^{2}(k)\right)\cdot\left[1-\alpha_1(k)\cdot\left(1+\alpha_0(k)\right)\cdot z^{-1}+\alpha_0(k)\cdot z^{-2}\right]z^{-1}}$$

$$\psi_{01}(z)=\frac{2\cdot\alpha_1(k)\cdot z^{-1}-z^{-2}-1}{\left(1-\alpha_0^{2}(k)\right)z^{-1}}\cdot\psi(z)$$

เอา ψ_{01} ที่ได้ไปแทนลงใน ψ_{02} จะได้

$$\psi_{02}(z)\cdot\left[1-\alpha_1(k)\cdot\left(1+\alpha_0(k)\right)\cdot z^{-1}+\alpha_0(k)\cdot z^{-2}\right]=\alpha_1(k)\cdot\psi(z)\cdot z^{-1}-\psi(z)\cdot z^{-2}+$$

$$\frac{\psi(z)}{1+\alpha_0(k)}-\alpha_1(k)\cdot\psi(z)\cdot z^{-1}+\frac{\alpha_0(k)}{1+\alpha_0(k)}\cdot\psi(z)\cdot z^{-2}+\frac{2\cdot\alpha_1(k)\cdot z^{-1}-z^{-2}-1}{\left(1-\alpha_0(k)\right)}\cdot\psi(z)$$

$$=\left[\alpha_1(k)\cdot z^{-1}-z^{-2}+\frac{1}{1+\alpha_0(k)}-\alpha_1(k)z^{-1}+\frac{\alpha_0(k)}{1+\alpha_0(k)}z^{-2}+\frac{2\cdot\alpha_1(k)z^{-1}-z^{-2}-1}{1-\alpha_0(k)}\right]\cdot\psi(z)$$

$$\frac{\psi_{02}(z)}{\psi(z)}=\frac{\left[\begin{array}{l}-\left(1-\alpha_0^{2}(k)\right)z^{-2}+1-\alpha_0(k)+\alpha_0(k)\left(1-\alpha_0(k)\right)z^{-2}+2\cdot\alpha_1(k)\left(1+\alpha_0(k)\right)z^{-1}-\\ \left(1+\alpha_0(k)\right)z^{-2}-1-\alpha_0(k)\end{array}\right]}{\left(1-\alpha_0^{2}(k)\right)\cdot\left[1-\alpha_1(k)\cdot\left(1+\alpha_0(k)\right)\cdot z^{-1}+\alpha_0(k)\cdot z^{-2}\right]} \quad (33)$$

$$G_{02}(z)=\frac{\psi_{02}(z)}{\psi(z)}=2\cdot\frac{-\alpha_0(k)+\alpha_1(k)\cdot\left(1+\alpha_0(k)\right)\cdot z^{-1}-z^{-2}}{\left(1-\alpha_0^{2}(k)\right)\cdot\left[1-\alpha_1(k)\cdot\left(1+\alpha_0(k)\right)\cdot z^{-1}+\alpha_0(k)\cdot z^{-2}\right]} \quad (34)$$

เมื่อได้ทรานส์เฟอร์ฟังก์ชั่นทั้ง 3 แล้ว ก็จะใช้นำไปเขียนเป็นวงจรที่ลดรูป เนื่องจากทรานสเฟอร์ฟังก์ชันแต่ละตัวขึ้นอยู่กับฟังก์ชั่นใดฟังก์ชั่นหนึ่งเท่านั้น จึงทำให้เขียนรูปวงจรได้ง่าย จากสมการ

$$y(k)=\left(\frac{1-\alpha_0}{2}\right)\cdot\left[x(k)-x(k-2)\right]+\alpha_1(k)\cdot\left(1+\alpha_0\right)\cdot y(k-1)-\alpha_0\cdot y(k-2) \quad (5)$$

$$\alpha_1(k+1)=\alpha_1(k)+\mu_1\cdot y(k)\cdot\psi(k) \quad (6)$$

$$\psi(k)=\alpha_1(k)\cdot\left(1+\alpha_0\right)\cdot\psi(k-1)-\alpha_0\cdot\psi(k-2)+\left(1+\alpha_0\right)\cdot y(k-1) \quad (8)$$

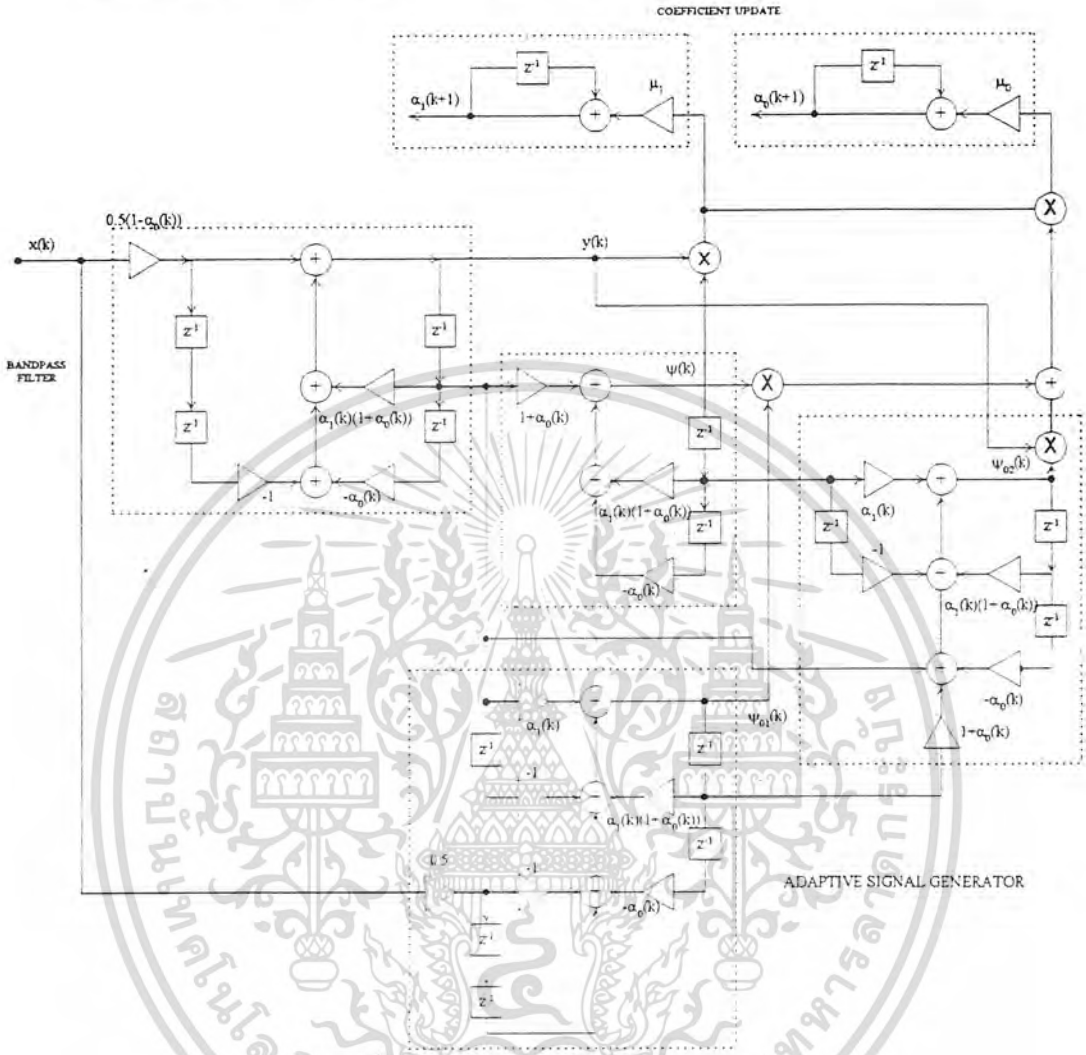
$$\alpha_0(k+1)=\alpha_0(k)+\mu_0 y(k)\cdot\psi(k)\cdot\left[\psi(k)\cdot\psi_{01}(k)+\psi_{02}(k)\cdot y(k)\right] \quad (17)$$

$$\psi_{01}(k)=\frac{1}{2}\left[x(k-2)-x(k)\right]+\alpha_1(k)\cdot y(k-1)-y(k-2)+$$
$$\alpha_1(k)\cdot\left[1+\alpha_0(k)\right]\psi_{01}(k-1)-\alpha_0(k)\cdot\psi_{01}(k-2) \quad (18)$$

$$\psi_{02}(k)=\alpha_1(k)\cdot\left[1+\alpha_0(k)\right]\psi_{02}(k-1)+\alpha_1(k)\cdot\psi(k-1)-\alpha_0(k)\cdot\psi_{02}(k-2)-$$
$$\psi(k-2)+\left[1+\alpha_0(k)\right]\psi_{01}(k-1)+y(k-1) \quad (19)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถนำมาใช้เขียนวงจร Adaptive Filter ได้ดังรูปที่ 1



รูปที่ 1 Unsimplified Adaptive Filter

จากวงจรนี้สามารถปรับตัวแปรเพื่อให้ง่ายต่อการนำไปใช้งานหรือเป็นการลดรูป สมการ $\psi_{01}(k)$ และ $\psi_{02}(k)$ ใหม่ได้มาจาก $G_{01}(k)$ และ $G_{02}(k)$

จาก $$G_{01}(z)=\frac{\psi_{01}(z)}{y(z)}=\frac{2\cdot\alpha_1(k)\cdot z^{-1}-z^{-2}-1}{\left(1-\alpha_0(k)\right)\cdot\left[1-\alpha_1(k)\cdot\left(1+\alpha_0(k)\right)\cdot z^{-1}+\alpha_0(k)\cdot z^{-2}\right]}$$

หา $\psi_{01}(z)$

$$\psi_{01}(k)-\alpha_1(k)\cdot\left\{1+\alpha_0(k)\right\}\cdot\psi_{01}(k-1)+\alpha_0(k)\cdot\psi_{01}(k-2)=-\frac{y(k)}{1-\alpha_0(k)}+\frac{\alpha_1(k)}{1-\alpha_0(k)}-\frac{y(k-2)}{1-\alpha_0(k)} \quad (35)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และจาก

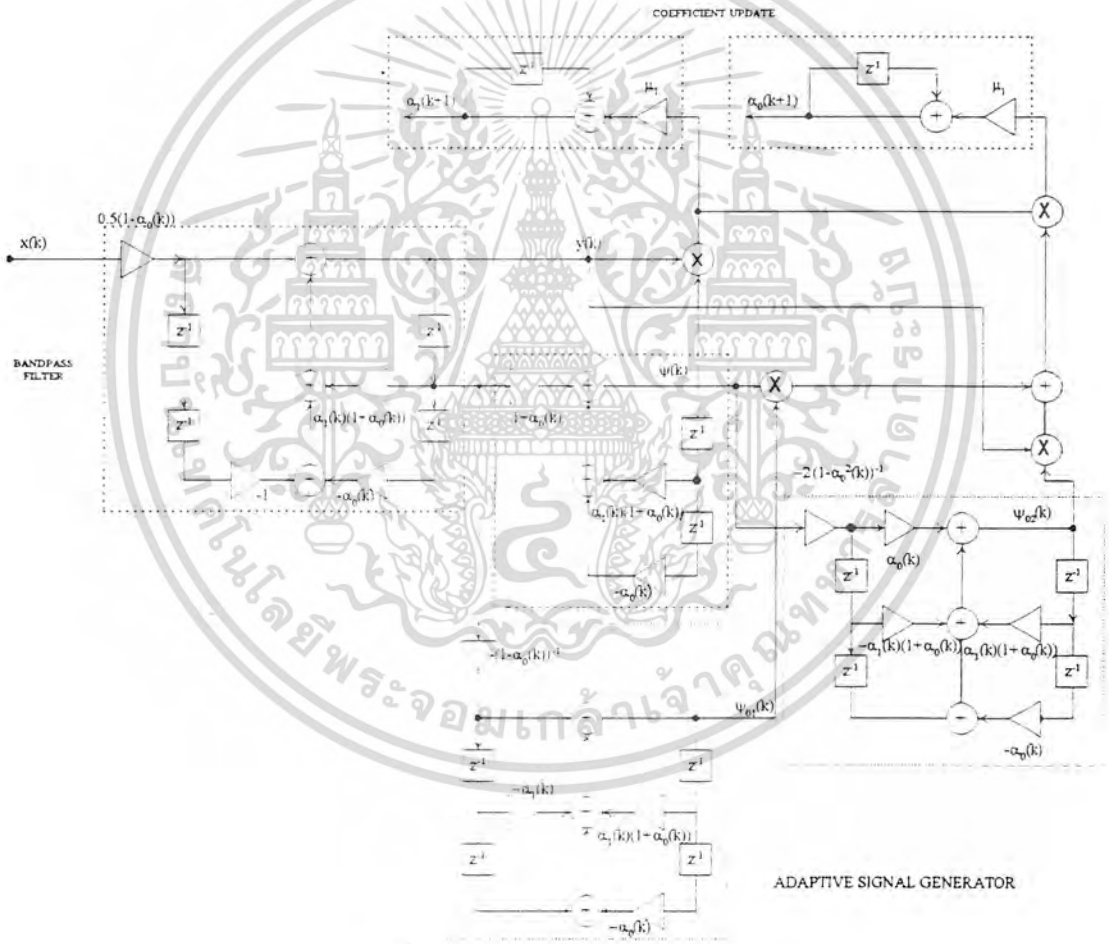
$$G_{02}(z) = \frac{\psi_{02}(z)}{\psi(z)} = 2 \cdot \frac{-\alpha_0(k) + \alpha_1(k) \cdot \left(1 + \alpha_0(k)\right) \cdot z^{-1} - z^{-2}}{\left(1 - \alpha_0^{\,2}(k)\right) \cdot \left[1 - \alpha_1(k) \cdot \left(1 + \alpha_0(k)\right) \cdot z^{-1} + \alpha_0(k) \cdot z^{-2}\right]}$$

หา $\psi_{02}(z)$

$$\psi_{02}(k) - \alpha_1(k) \cdot \left(1 + \alpha_0(k)\right) \cdot \psi_{02}(k-1) + \alpha_0(k) \cdot \psi_{02}(k-2) = \frac{2}{1 - \alpha_0^2(k)} (-\alpha_0(k) \cdot \psi(k) +$$

$$\alpha_1(\mathrm{k}) \cdot \left(1 + \alpha_0(\mathrm{k})\right) \cdot \psi(\mathrm{k}-1) - \psi(\mathrm{k}-2)) \qquad (36)$$

จากสมการ (5),(6),(8),(17),(35) และ (36) ที่ได้ จึงนำมาเขียนวงจรที่ลดรูปแล้วดังรูปที่ 2

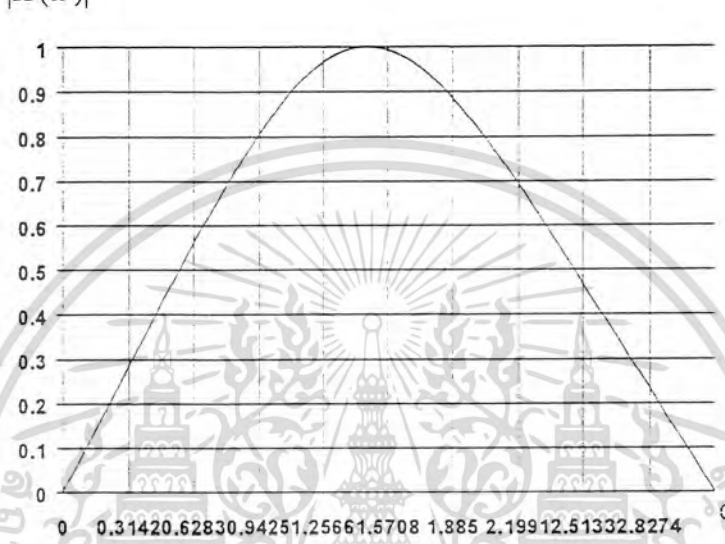


รูปที่ 2 Simplified Adaptive Filter

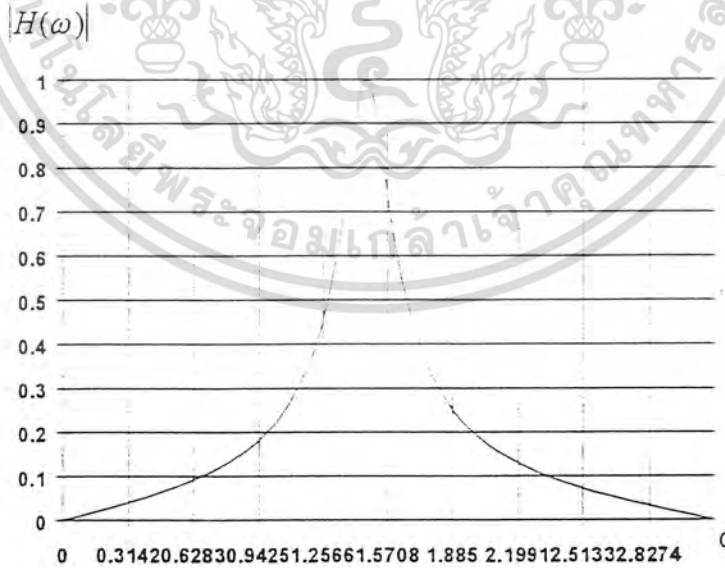
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
ผลการทดลอง

1. *กราฟคุณสมบัติของ* $|H(\omega)|$ *เทียบกับ* ω

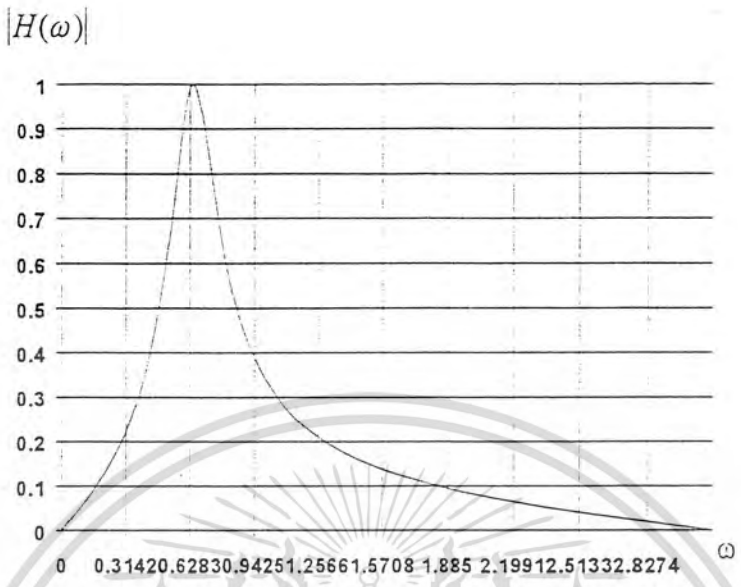


รูปที่ 1.1 α_0=0.1, α_1=0.1

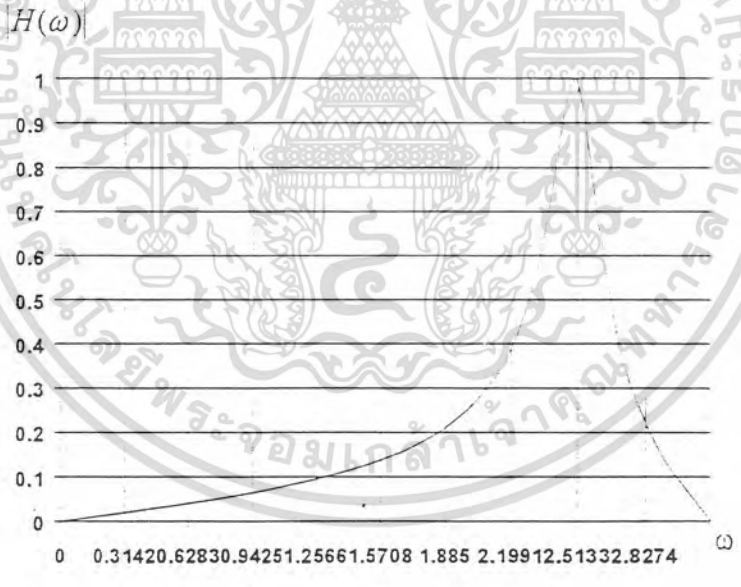


รูปที่ 1.2 α_0=0.8, α_1=0.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

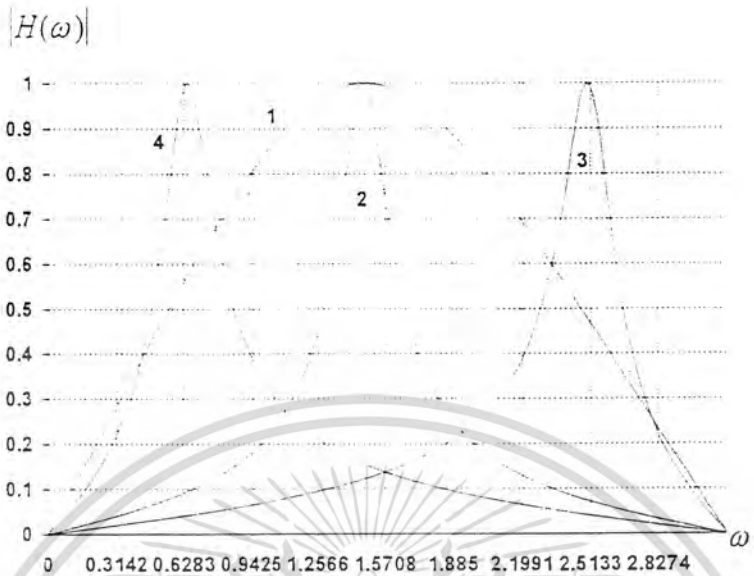


รูปที่ 1.3 α_0=0.8, α_1=0.8



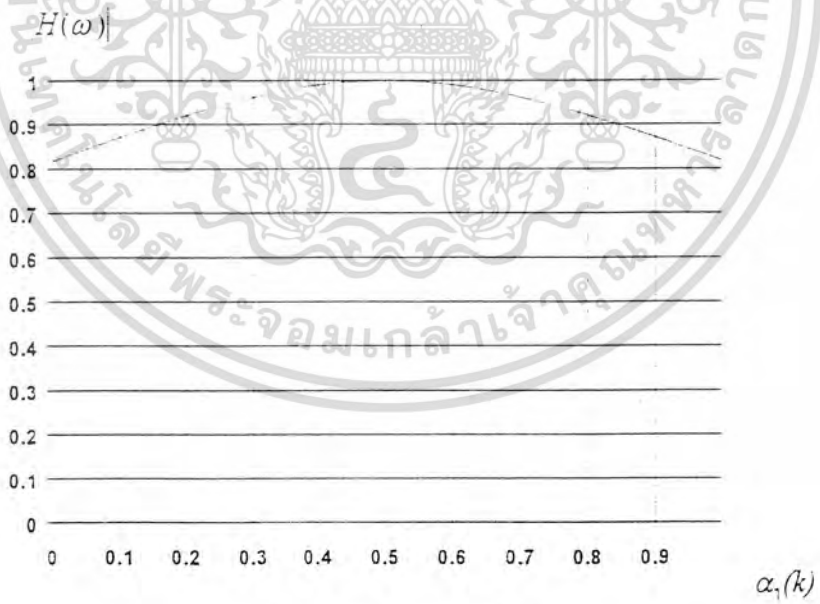
รูปที่ 1.4 α_0=0.8, α_1=-0.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



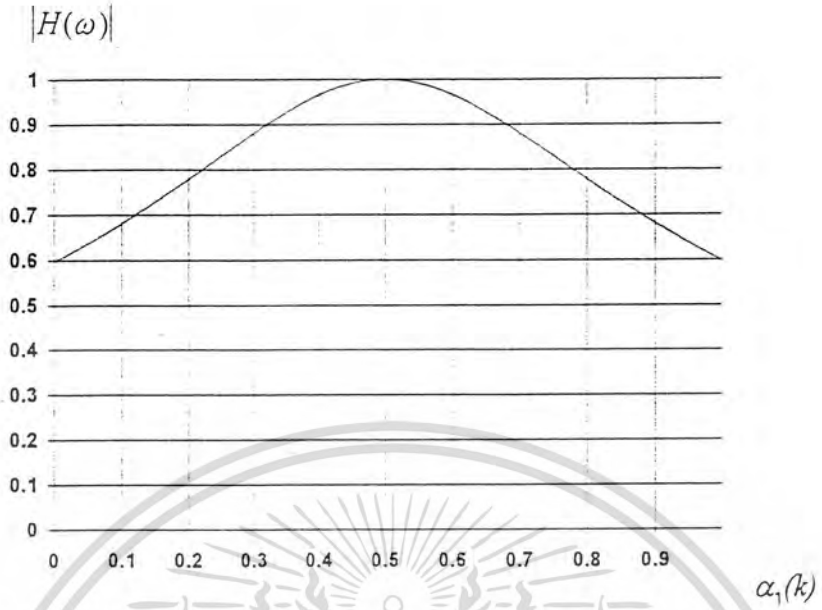
รูปที่ 1.5 1. α_0=0.1, α_1=0.1, 2. α_0=0.8, α_1=0.1
3. α_0=0.8, α_1=0.8, 4. α_0=0.8, α_1=-0.8

2. กราฟแสดงคุณสมบัติของ $|H(\omega)|$ เทียบกับ α_1

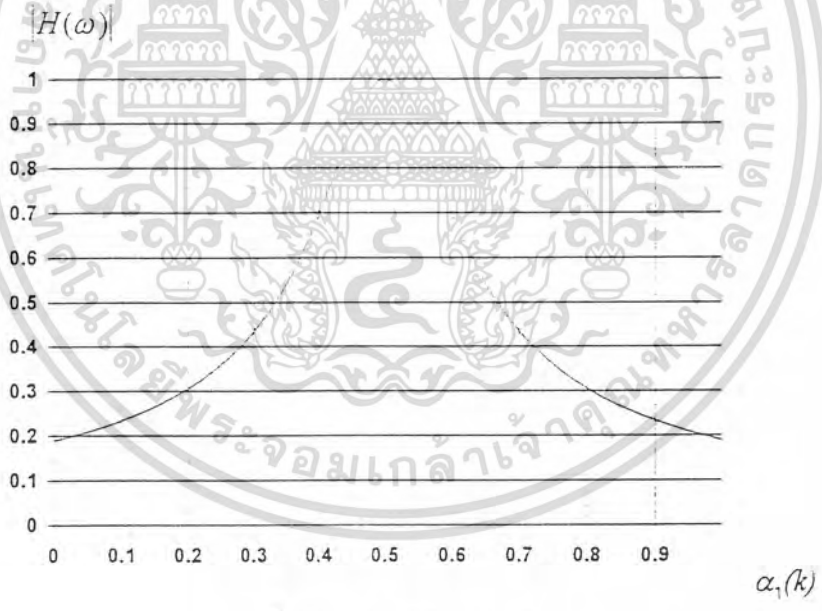


รูปที่ 2.1 $\omega_0=\pi/3$, α_0=0.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

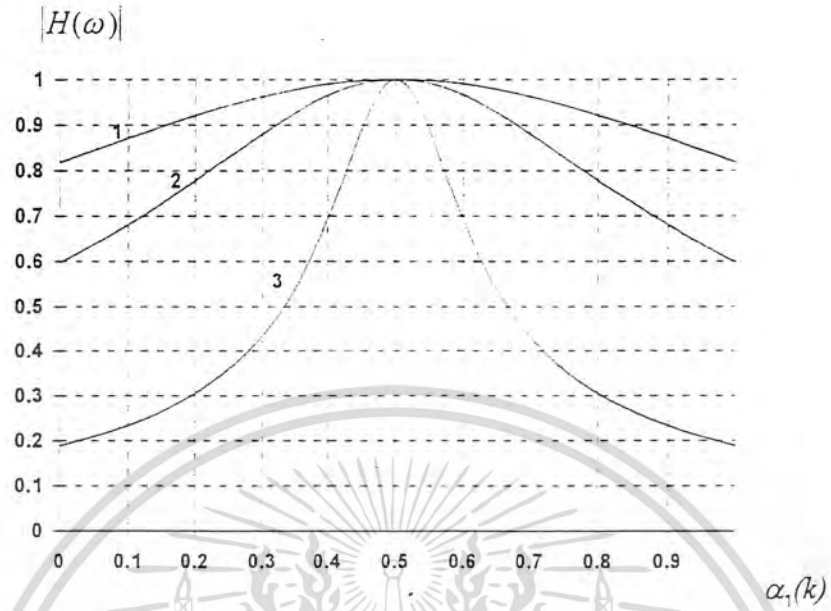


รูปที่ 2.2 $\omega_0=\pi/3$, $\alpha_0=0.4$

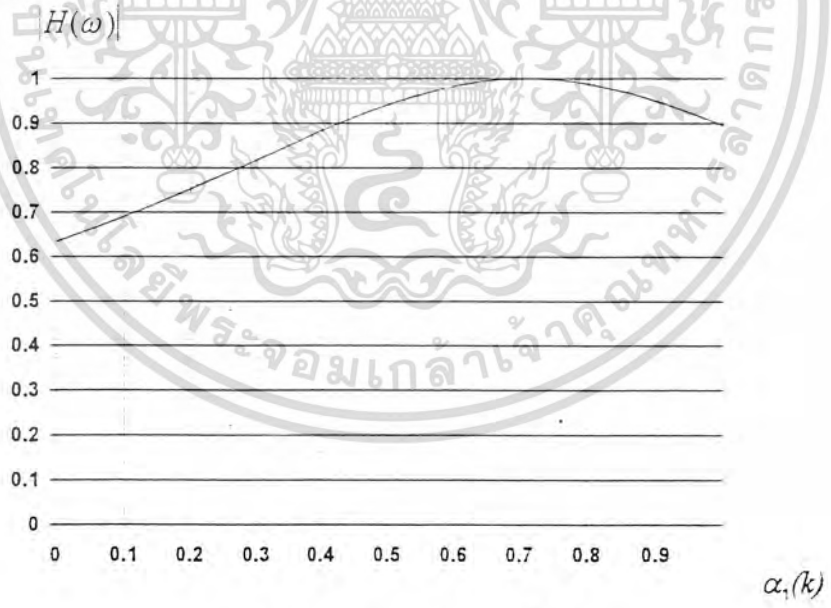


รูปที่ 2.3 $\omega_0=\pi/3$, $\alpha_0=0.8$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

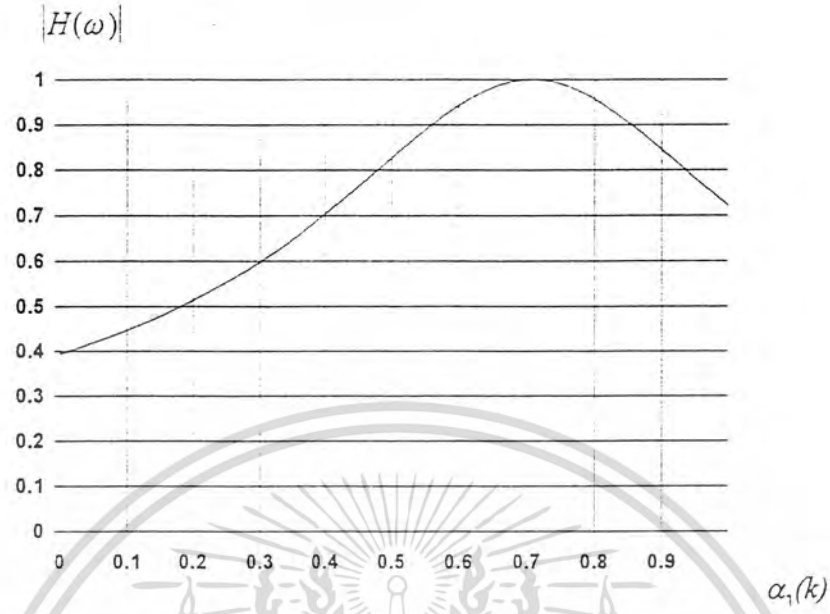


รูปที่ 2.4 1. $\omega_0=\pi/3$, $\alpha_0=0.1$, 2. $\omega_0=\pi/3$, $\alpha_0=0.4$
3. $\omega_0=\pi/3$, $\alpha_0=0.8$

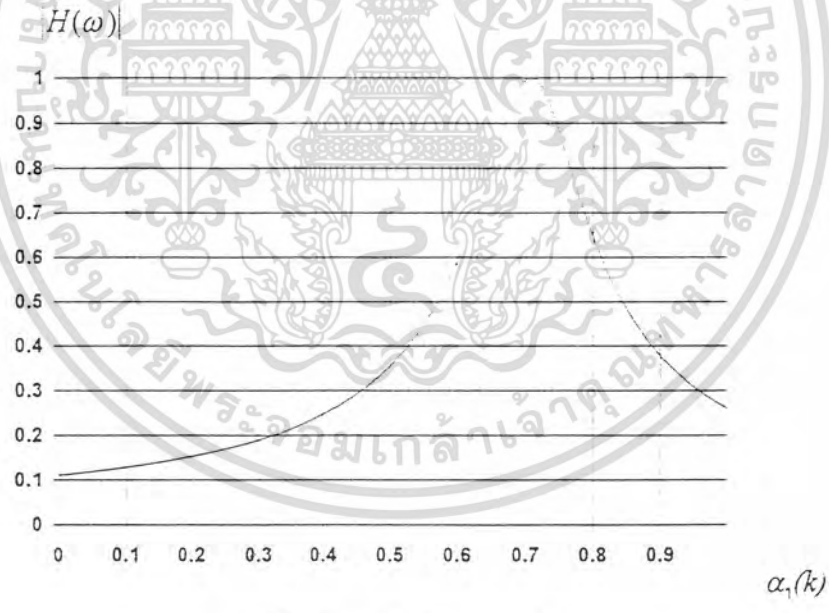


รูปที่ 2.5 $\omega_0=\pi/4$, $\alpha_0=0.1$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

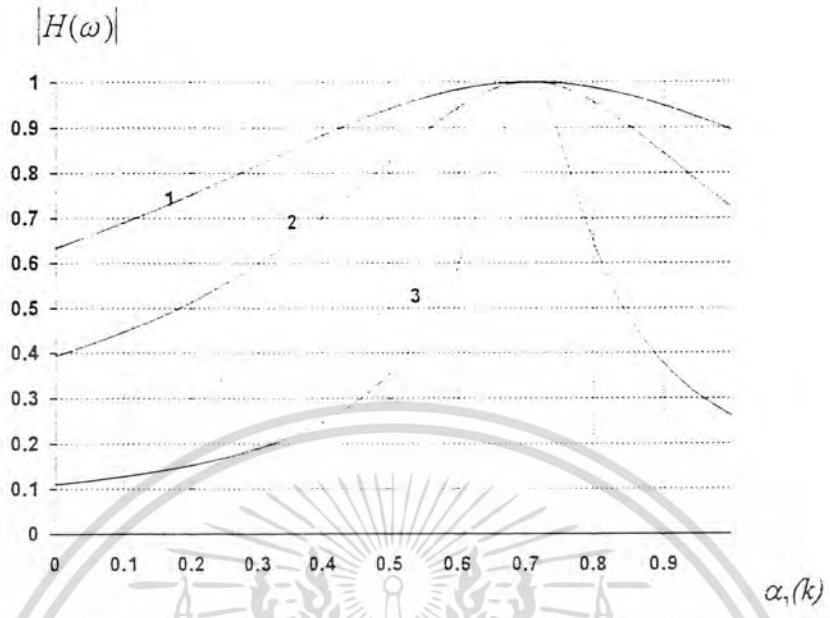


รูปที่ 2.6 $\omega_0=\pi/4$, $\alpha_0=0.4$



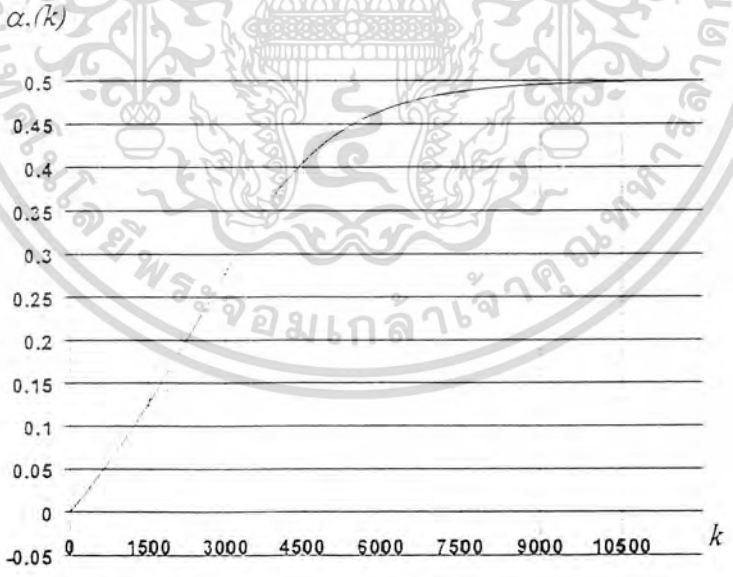
รูปที่ 2.7 $\omega_0=\pi/4$, $\alpha_0=0.8$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



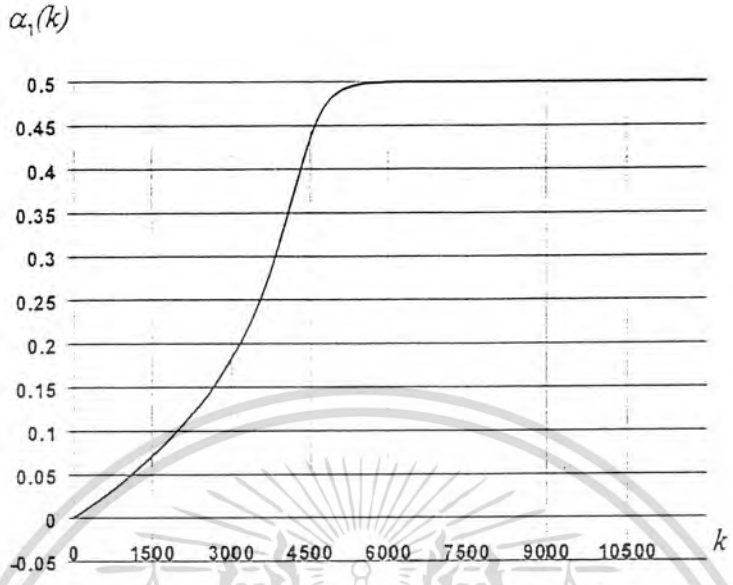
รูปที่ 2.3 1. $\omega_0=\pi/4$, $\alpha_0=0.1$, 2. $\omega_0=\pi/4$, $\alpha_0=0.4$
3. $\omega_0=\pi/4$, $\alpha_0=0.8$

3. *กราฟแสดงคุณสมบัติของ $\alpha_1(k)$ เทียบกับ k*

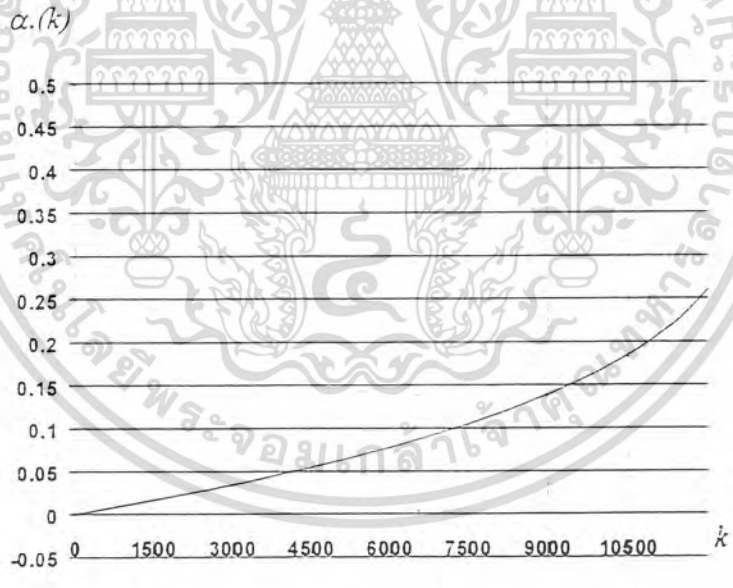


รูปที่ 3.1 $A=1$, $\mu=0.0003$, $\omega=\pi/3$, $\alpha_0=0.3$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

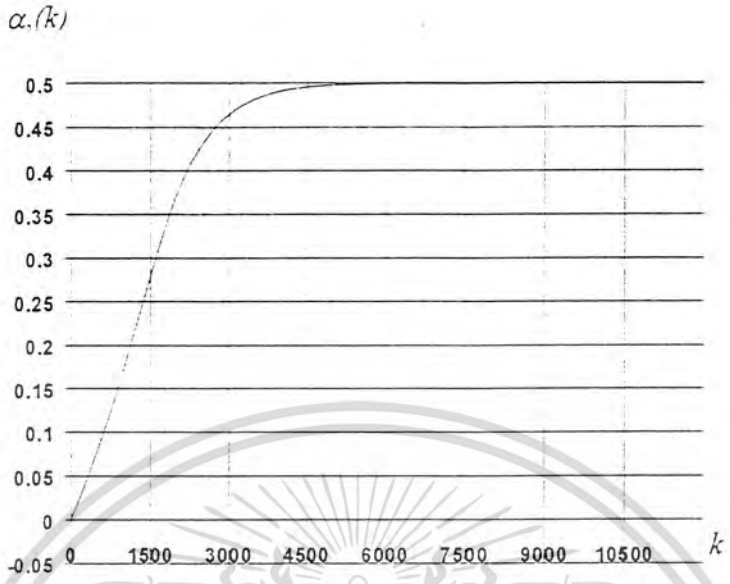


รูปที่ 3.2 $A=1$, $\mu=0.0003$, $\omega=\pi/3$, $\alpha_0=0.6$

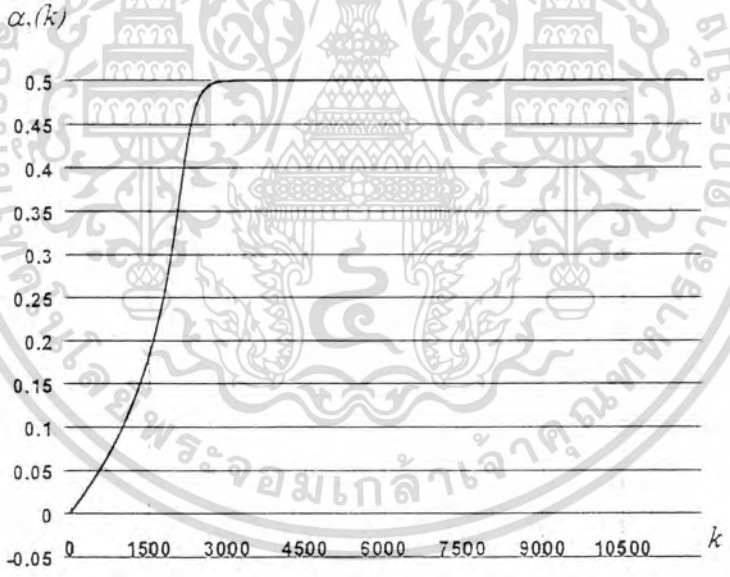


รูปที่ 3.3 $A=1$, $\mu=0.0003$, $\omega=\pi/3$, $\alpha_0=0.8$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

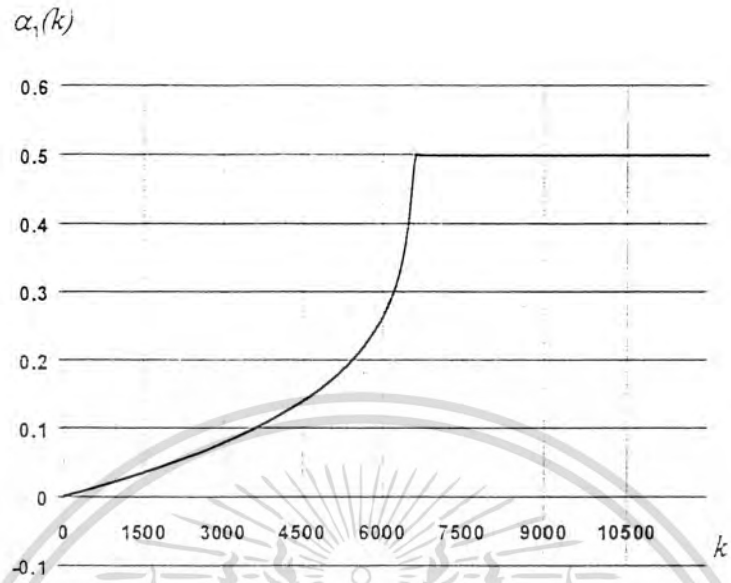


รูปที่ 3.4 $A=1$, $\mu=0.0006$, $\omega=\pi/3$, $\alpha_0=0.3$

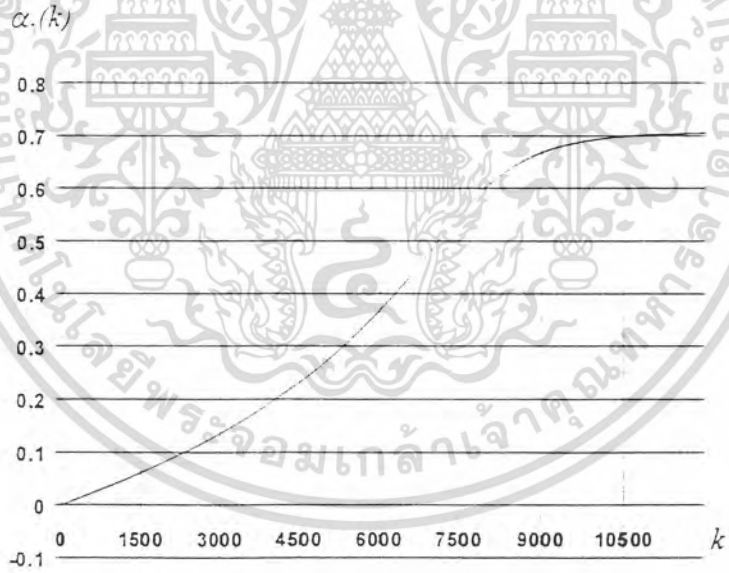


รูปที่ 3.5 $A=1$, $\mu=0.0006$, $\omega=\pi/3$, $\alpha_0=0.6$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

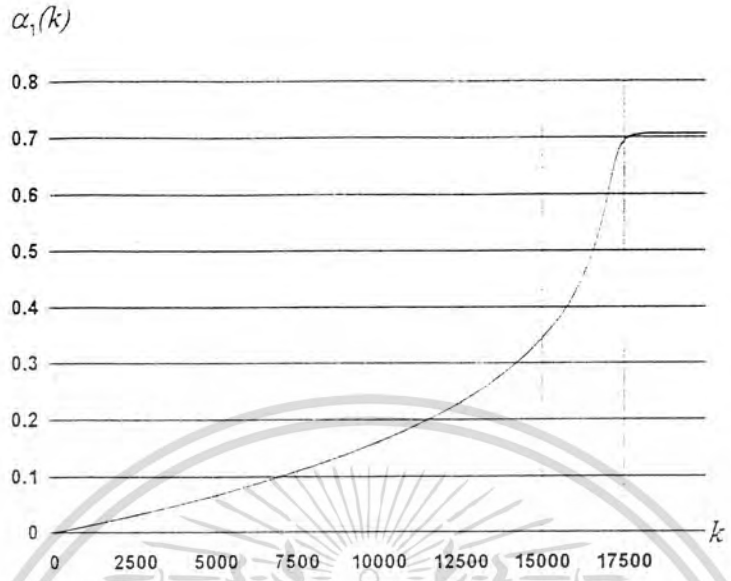


รูปที่ 3.6 $A=1$, $\mu=0.0006$, $\omega=\pi/3$, $\alpha_0=0.8$

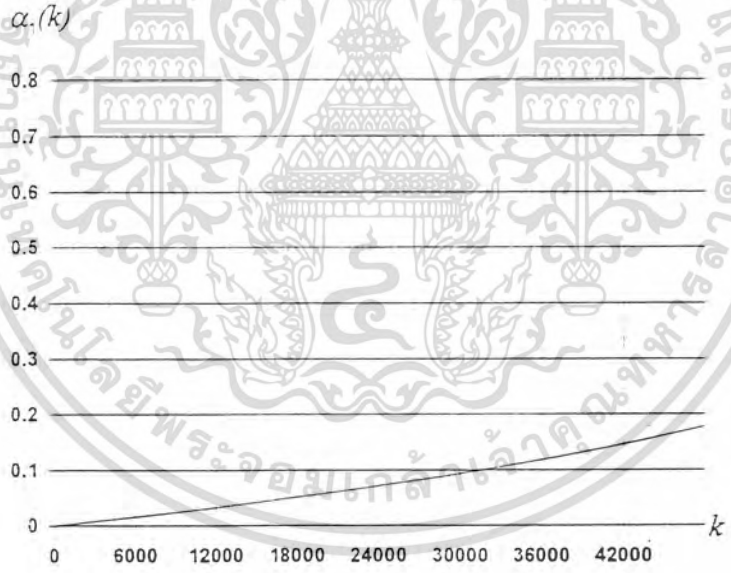


รูปที่ 3.7 $A=1$, $\mu=0.0003$, $\omega=\pi/4$, $\alpha_0=0.3$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



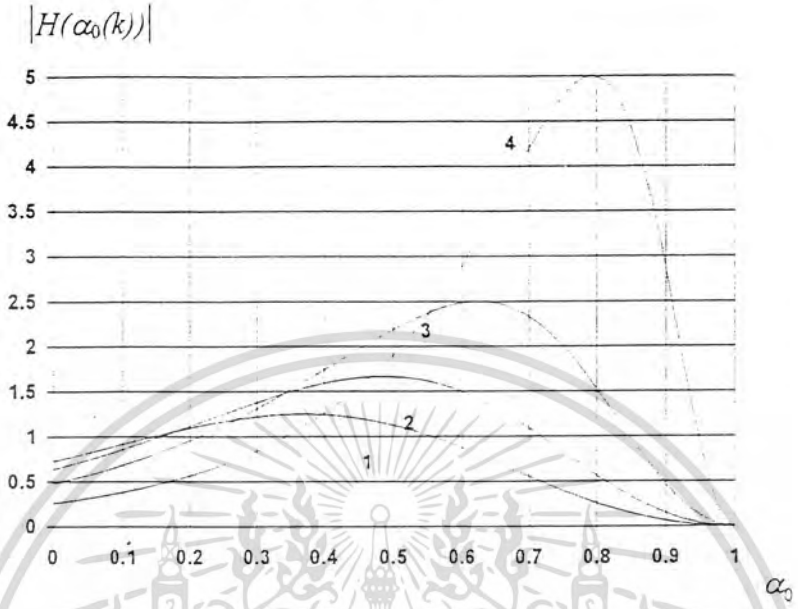
รูปที่ 3.8 A=1, μ=0.0003, $\omega=\pi/4$, α_0=0.6



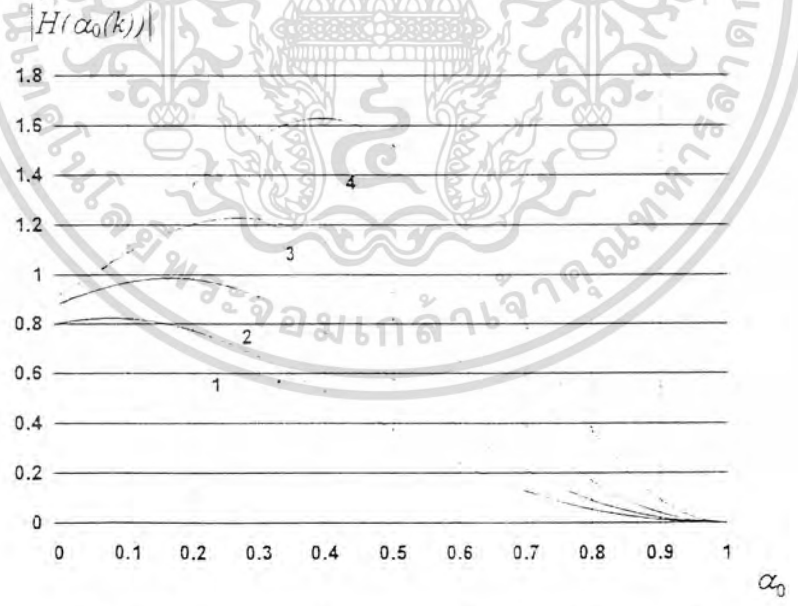
รูปที่ 3.9 A=1, μ=0.0003, $\omega=\pi/4$, α_0=0.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. กราฟแสดงคุณสมบัติของ $|H(\alpha_0(k))|$ เทียบกับ α_0



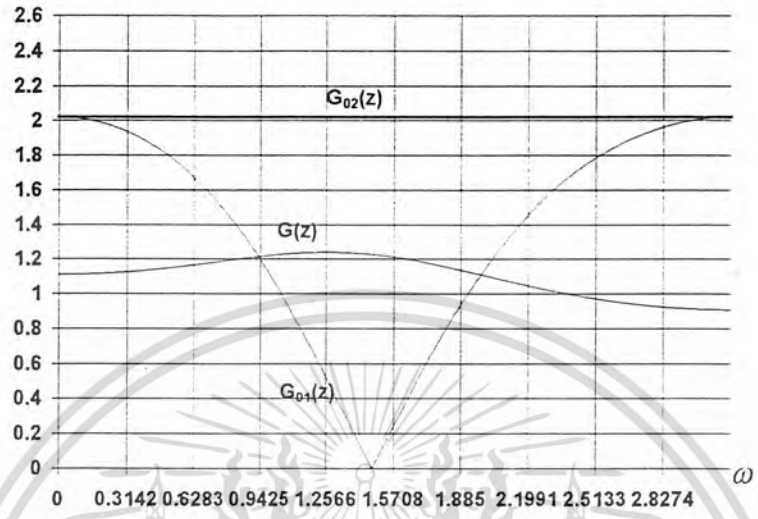
รูปที่ 4.1 1. α_1=0.1, $\omega=\pi/3$, 2. α_1=0.2, $\omega=\pi/3$
3. α_1=0.3, $\omega=\pi/3$, 4. α_1=0.4, $\omega=\pi/3$



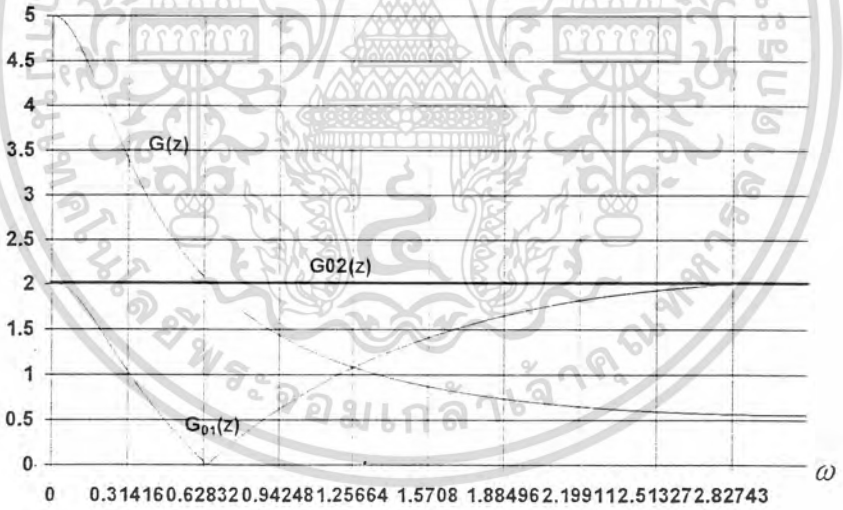
รูปที่ 4.2 1. α_1=0.1, $\omega=\pi/4$, 2. α_1=0.2, $\omega=\pi/4$
3. α_1=0.3, $\omega=\pi/4$, 4. α_1=0.4, $\omega=\pi/4$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. *กราฟแสดงคุณสมบัติของ* $|G(z)|, |G_{01}(z)|$, *และ* $|G_{02}(z)|$ *เทียบกับ* ω

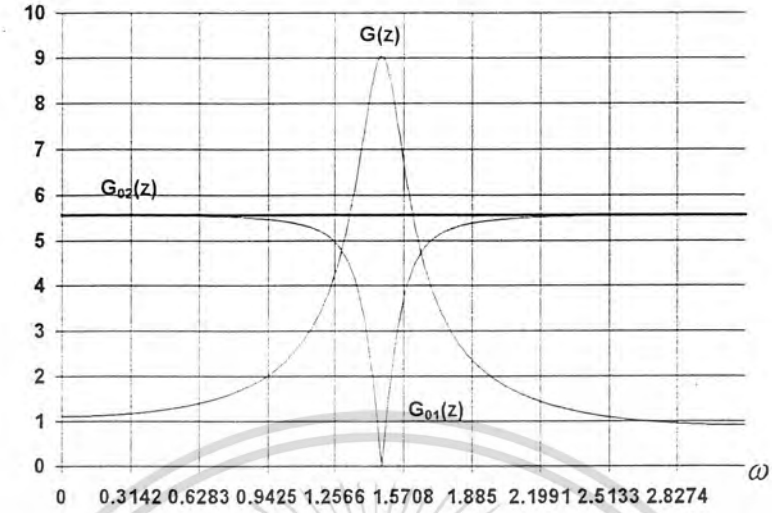


รูปที่ 5.1 α_0=0.1, α_1=0.1



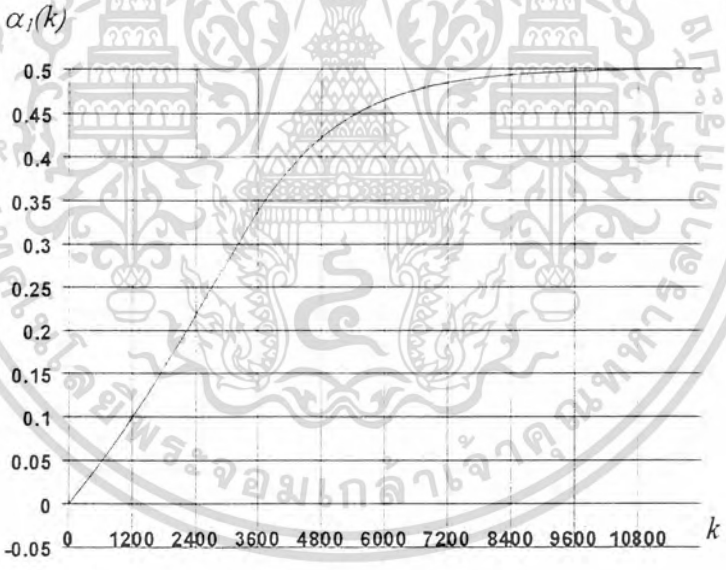
รูปที่ 5.2 α_0=0.1, α_1=0.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



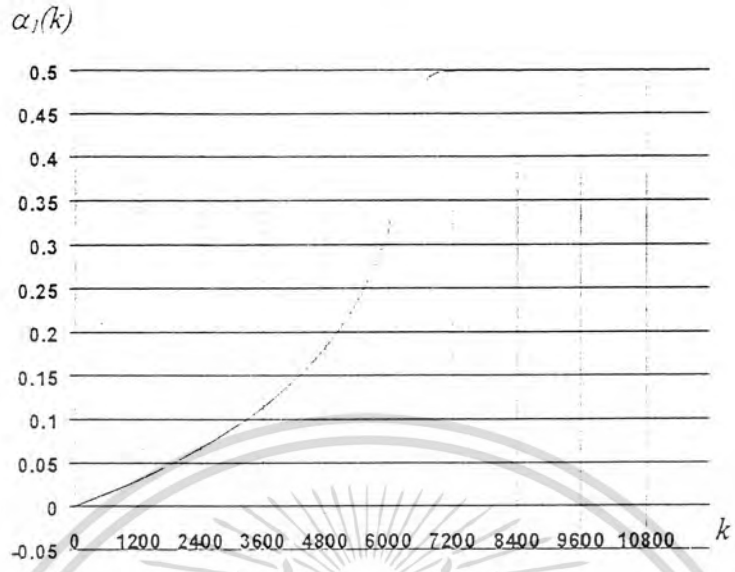
รูปที่ 5.3 α_0=0.8, α_1=0.1

6. *กราฟแสดงคุณสมบัติของ $\alpha_1(k)$ เทียบกับ k เมื่อ $\alpha_0(k)$ คงที่*

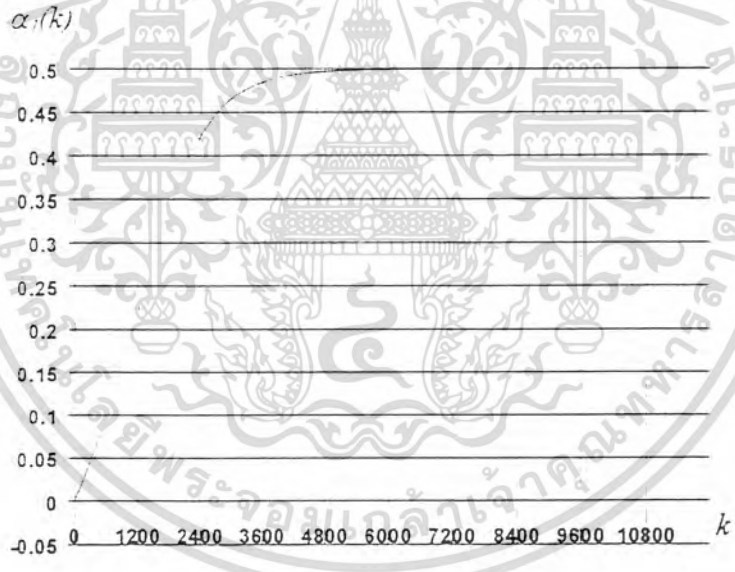


รูปที่ 6.1 α_0=0.3, μ=0.0003, $\omega=\pi/3$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

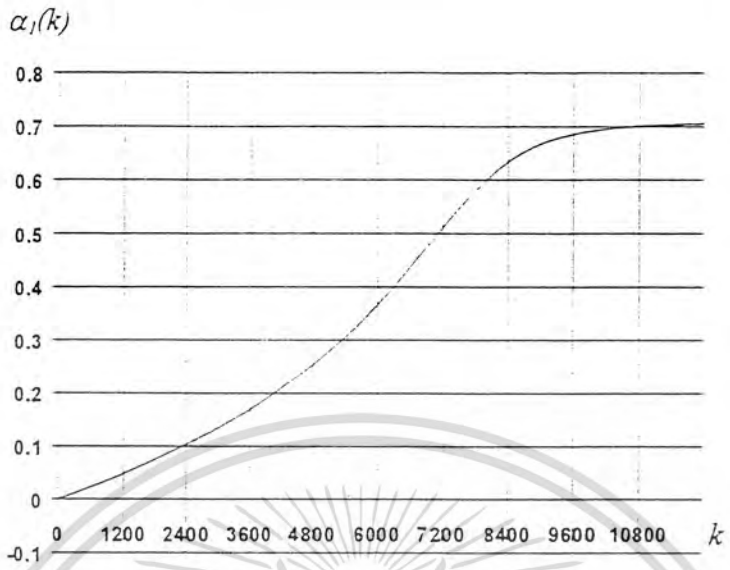


รูปที่ 6.2 α_0=0.7, μ=0.0003, $\omega=\pi/3$



รูปที่ 6.3 α_0=0.3, μ=0.0006, $\omega=\pi/3$

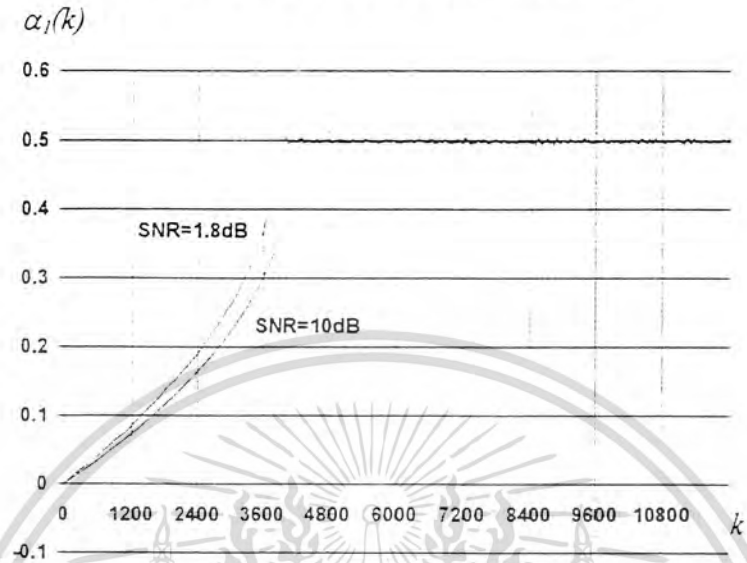
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



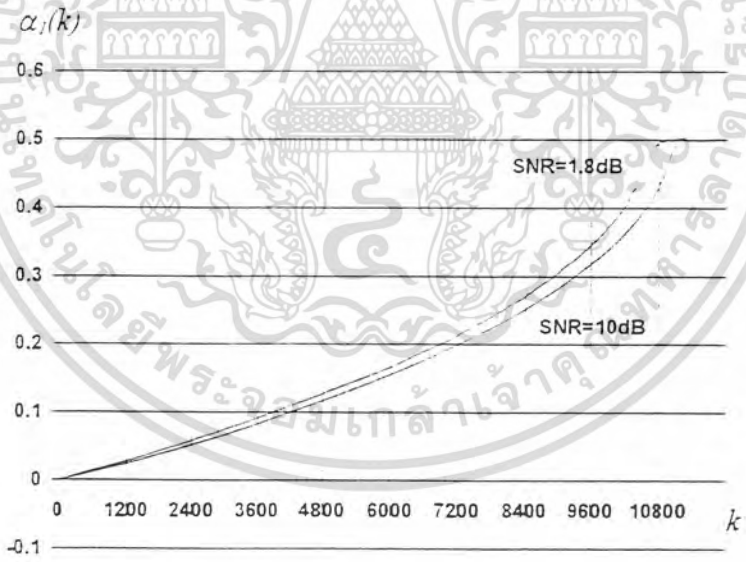
รูปที่ 6.4 α_0=0.3, μ=0.0003, $\omega=\pi/4$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. กราฟแสดงคุณสมบัติของ $\alpha_1(k)$ เทียบกับ k เมื่อ $\alpha_1(k)$ เปลี่ยนแปลง

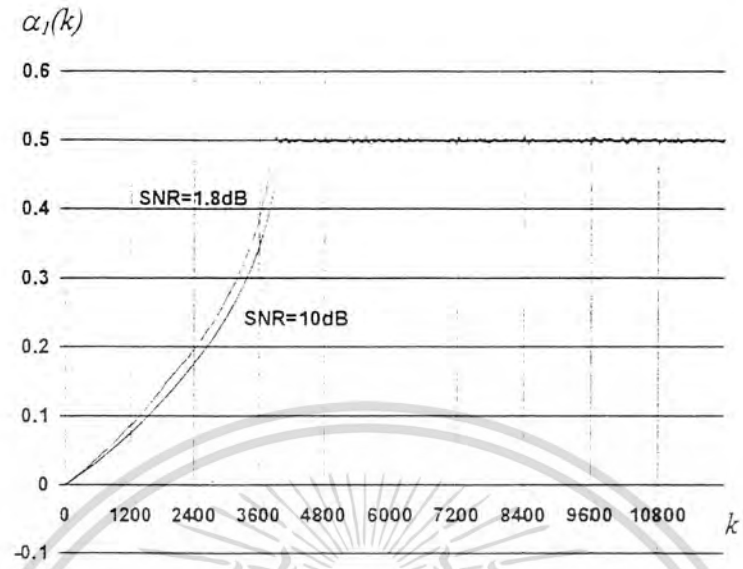


รูปที่ 7.1 $A=1$, $\mu=0.0003$, $\mu_0=0.0006$, $\omega=\pi/3$

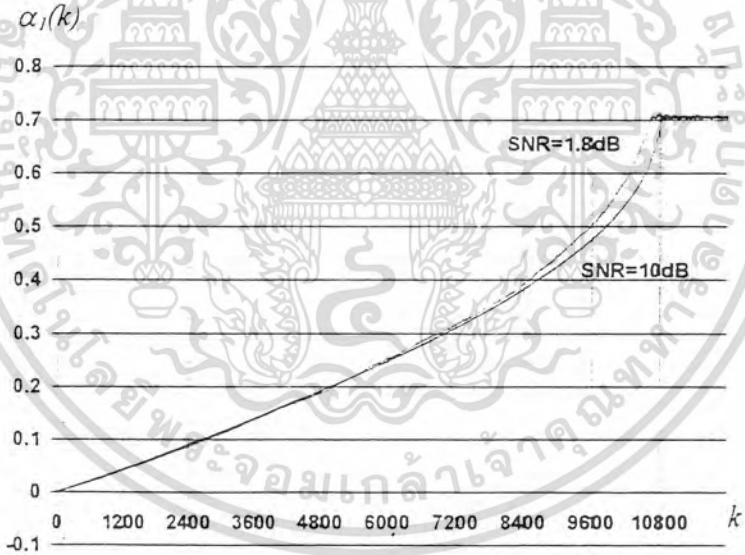


รูปที่ 7.2 $A=1$, $\mu=0.0001$, $\mu_0=0.0006$, $\omega=\pi/3$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

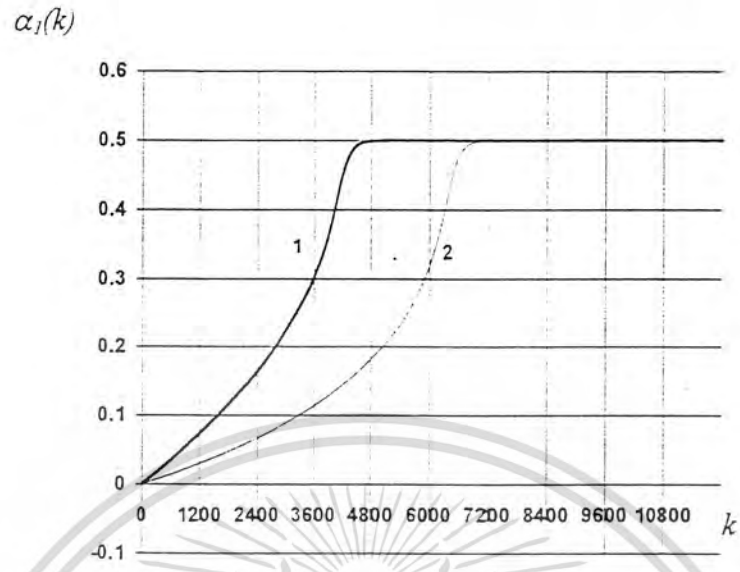


รูปที่ 7.3 A=1, μ=0.0003,μ_0=0.0009, $\omega=\pi/3$



รูปที่ 7.4 A=1, μ=0.0002,μ_0=0.0005, $\omega=\pi/4$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



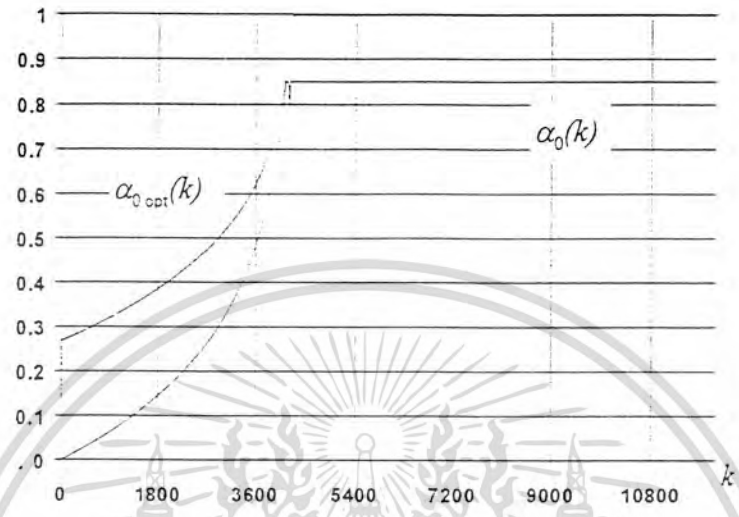
รูปที่ 7.5 A=1, μ=0.0003, $\omega=\pi/3$, SNR=10dB

1. α_0 เปลี่ยนแปลง μ_0=0.0006, $\alpha_{0,min}$=0, $\alpha_{0,max}$=0.7

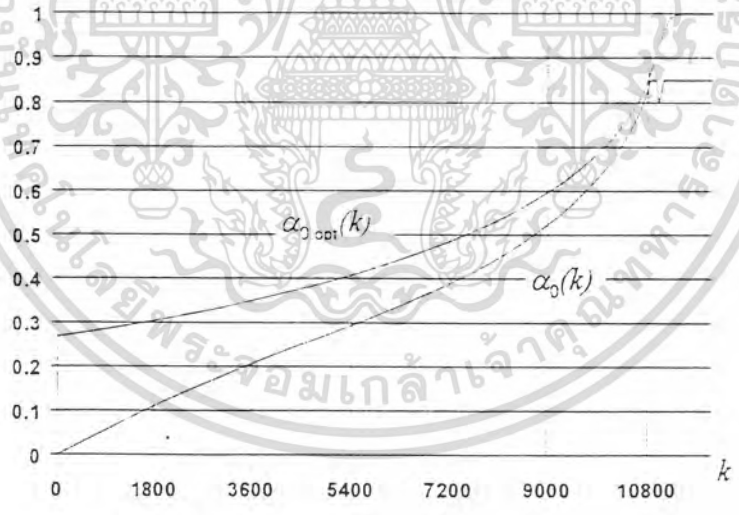
2. α_0 คงที่ α_0=0.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. *กราฟแสดงคุณสมบัติของ $\alpha_0(k), \alpha_{0,opt}(k)$ เทียบกับ k เมื่อ $\alpha_0(k)$ และ $\alpha_1(k)$ เปลี่ยนแปลง*

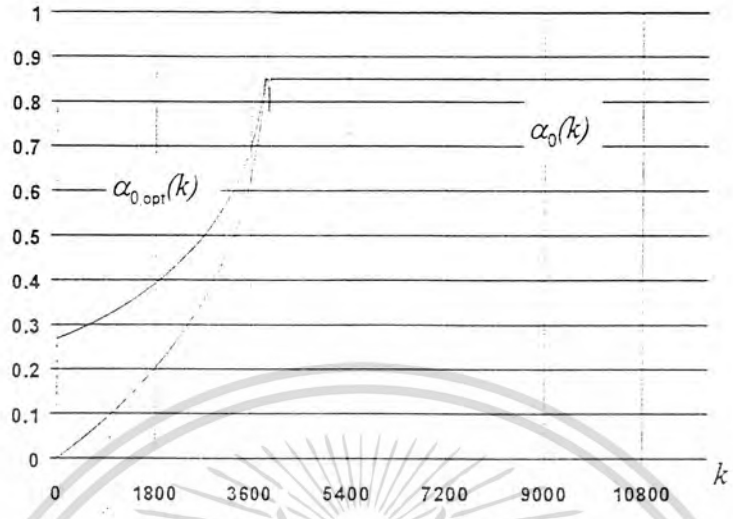


รูปที่ 8.1 μ=0.0003, μ_0=0.0006, $\omega=\pi/3$

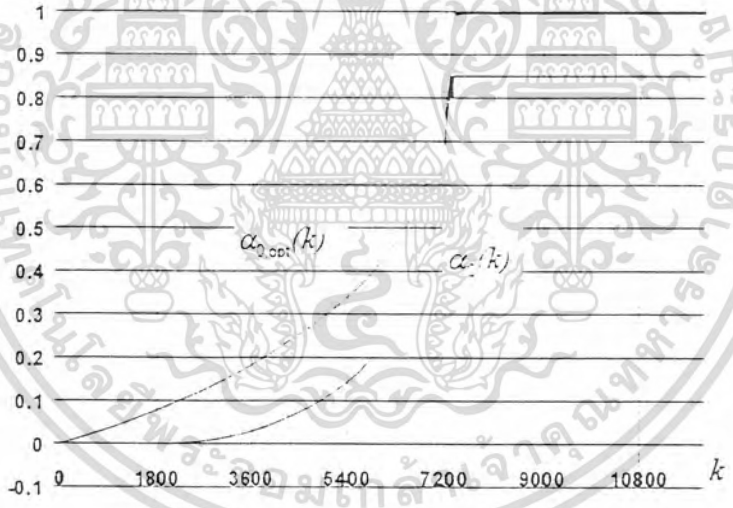


รูปที่ 8.2 μ=0.0001, μ_0=0.0006, $\omega=\pi/3$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 μ=0.0003, μ_0=0.0009, $\omega=\pi/3$



รูปที่ 3.4 μ=0.0003, μ_0=0.0006, $\omega=\pi/4$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
สรุปผลการทดลอง

จากกราฟที่ 1 เป็นการแสดงคุณสมบัติของสมการทรานสเฟอร์ฟังก์ชั่นแบนพาสต์เทียบกับมุม ω พบว่าค่าของ α_0 มีผลทำให้ค่า Q-factor หรือความกว้างของการตอบสนองของทรานสเฟอร์ฟังก์ชั่นเปลี่ยนแปลงไปโดยค่าค่า Q-factor จะแปรผันตรงกันค่าของ α_0 และค่าของ α_1 เป็นค่าที่กำหนดจุดการตอบสนองสูงสุดของทรานสเฟอร์ฟังก์ชั่น โดยจุดตอบสนองสูงสุดจะเบี่ยงเบนไปด้านซ้ายถ้า α_1 มีค่าน้อยกว่าศูนย์ และจะเบี่ยนเบนไปด้านขวาถ้า α_1 มีค่ามากกว่าศูนย์

กราฟที่ 2 เป็นการแสดงคุณสมบัติของทรานสเฟอร์ฟังก์ชั่นเทียบกับ α_1 พบว่าจุดตอบสนองสูงสุดของทรานสเฟอร์ฟังก์ชั่นจะอยู่ที่ $\alpha_1=\cos(\omega)$ เสมอ และค่า α_0 ก็ยังคงทำหน้าที่กำหนด Q-factor

กราฟที่ 3 เป็นการแสดงคุณสมบัติของ $\alpha_1(k)$ เทียบกับแกนเวลา k พบว่าค่า μ จะมีผลทำให้การลู่เข้าของกราฟเร็วขึ้น โดยที่ถ้าค่า μ มีคามากก็จะลู่เข้าเร็ว ถ้าค่าน้อย (เข้าใกล้ศูนย์) ก็จะลู่เข้าช้า ค่า μ เป็นความละเอียดของการลู่เข้าถ้ามีค่ามากจะทำให้ลู่เข้าได้เร็ว และถ้ามากเกินไปจะทำให้ลู่เข้าอย่างขาดเสถียรภาพ ค่า ω เป็นมุมที่ฟังก์ชั่น $\alpha_1(k)$ จะต้องลู่เข้าที่จุด $\alpha_1=\cos(\omega)$ และค่าของ α_0 จะทำให้การลู่เข้าช้าลงถ้าคาของ α_0 มาก (เข้าใกล้ศูนย์) และจะเร็วถ้ามีค่ามาก (เข้าใกล้หนึ่ง)

กราฟที่ 4 เป็นการแสดงคุณสมบัติของทรานสเฟอร์ฟังก์ชั่นเมื่อเทียบกับ α_0 จะพบว่าไม่ว่าจะเปลี่ยนค่าตัวแปร α_1 และ ω เป็นค่าใดก็ตาม กราฟการตอบสนองของทรานสเฟอร์ฟังก์ชั่นก็ยังคงมีความสูงอยู่เพียงยอดเดียวเท่านั้น

กราฟที่ 5 เป็นการแสดงคุณสมบัติทรานสเฟอร์ฟังก์ชั่นของวงจรที่ลดรูปแล้วเทียบกับแกนความถี่ ω

กราฟที่ 6 เป็นการแสดงคุณสมบัติของ $\alpha_1(k)$ เทียบกับแกนเวลา k เมื่อ α_0 คงที่ จะเหมือนกับกราฟที่ 3

กราฟที่ 7 เป็นการแสดงคุณสมบัติของ $\alpha_1(k)$ เทียบกับแกนเวลา k เมื่อ α_0 เปลี่ยนแปลง พบว่าค่าของ Signal to Noise Ratio ที่มีกับสัญญาณ input ถ้ามี SNR ต่ำ (noise สูง) จะทำให้การลู่เข้าของสัญญาณเร็วขึ้น แต่จะมีการกระเพื่อม (ripple) เมื่อลู่เข้าแล้วมากกว่าปกติ กราฟที่ 7.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นการแสดงความแตกต่างของ $\alpha_1(k)$ เมื่อ α_0 คงที่ และ α_0 เปลี่ยนแปลง จะพบว่าการลู่เข้าของ α_1 ที่มี α_0 เปลี่ยนแปลงจะเร็วกว่ากราฟ α_1 ที่ α_0 คงที่เสมอ

กราฟที่ 8 เป็นการแสดงคุณสมบัติของ $\alpha_0(k)$ กับ $\alpha_{0,opt}(k)$ เมื่อทั้ง $\alpha_1(k)$ และ $\alpha_0(k)$ เปลี่ยนแปลง $\alpha_{0,opt}(k)$ เป็นค่า α_0 optimum จะลู่เข้าเร็วกว่า α_0 ปกติเสมอ

ในปริญญานิพนธ์นี้ได้ศึกษาวิเคราะห์อะแดปทีฟอัลกอริทึมที่ใช้สำหรับการดีเทคสัญญาณซายห์คลื่นเดียว โดยใช้ฟิลเตอร์แบบปรับค่า Q-factor ด้วยวงจรกรองความถี่กลางผ่านแบบ IIR ซึ่งเราสามารถควบคุมค่า Q-factor เป็นไปอย่างต่อเนื่อง ในการควบคุมค่า Q นั้นทำโดยการใช้วิธีปรับพารามิเตอร์ของวงจรกรองอันได้แก่ α_0 และ α_1 ด้วยวิธี Stochastic Gradient จึงทำให้สามารถลดจำนวนครั้งในการคำนวณลงซึ่งยืนยันได้จากการซิมูเลชั่นด้วยโปรแกรมคอมพิวเตอร์

จากการซิมูเลชั่นจะเห็นว่าสัญญาณรบกวนที่ปนเข้ามากับสัญญาณอินพุทสามารถถูกกำจัดออกไปโดยวงจรกรองความถี่นี้และเอาท์พุทที่ได้ก็มีลักษณะสัญญาณใกล้เคียงกับสัญญาณอินพุทจริงเมื่อ α_1 ลู่เข้าสู่ค่าคงที่ จากคุณสมบัติเหล่านี้ เราสามารถนำวงจรนี้ไปใช้ลดสัญญาณรบกวนเช่น สัญญาณที่เกิด echo เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

โปรแกรมที่เขียน

โปรแกรมที่ใช้เขียนด้วยภาษา C แบบ Object-Oriented Programming ทดสอบและคอมไพล์โดยใช้โปรแกรม Borland C++ 4.5 บน Large Model และคอมไพล์บน DOS

โปรแกรมแยกเป็นสองส่วนใหญ่ๆ คือส่วนที่เขียนเพื่อสร้างเป็นไลบรารี (Library) กับส่วนที่เป็นตัวโปรแกรมซิมูเลชั่นจริงๆ ที่จะต้องลิงค์เข้ากับไลบรารีที่เขียนด้วย

โปรแกรมส่วนไลบรารี

ไลบรารีที่เขียนส่วนใหญ่ใช้ความสามารถทางด้านกราฟฟิคของ Borland C ไลบรารีด้วย (Borland Graphics Interface) ดังนั้นการคอมไพล์และลิงค์จะต้องรวมไลบรารีกราฟฟิคของ Borland ที่ชื่อ Graphics.Lib เข้าไปด้วย ไลบรารีที่ได้เขียนมานั้น ถูกคอมไพล์เป็น LLadG.Lib ซึ่งเป็นแบบ Large model ต้องลิงค์ไลบรารีตัวนี้ด้วยเวลาใช้งาน

โปรแกรมส่วนที่เป็นซิมูเลชั่น

Dsps.H

```
#include <alloc.h>
#include <bios.h>
#include <conio.h>
#include <dos.h>
#include <except.h>
#include <graphics.h>
#include <math.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <time.h>
#include "grapnbas.h"
#include "mousebas.h"
#include "winobj.h"
#include "pgraph.h"

/*_____.
 * FTP - FloaTing Point, ชนิดของเลขทศนิยมที่จะใช้ในโปรแกรม  *
 * TIME_DOMAIN, ชนิดของแกนเวลา ในที่นี้เราใช้เลขจำนวนเต็ม  *
 * MAXOUTPUT, ขนาด array ของ output                          *
 *_____.*/
// #define FTP      long double  ใช้ FTP ที่อยู่ใน PGraph.H แทน
#define TIME_DOMAIN int
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#ifndef __EXTENDTYPE__
#define __EXTENDTYPE__
#define TRUE  1
#define FALSE 0

typedef unsigned char BYTE;
typedef unsigned int  WORD;
typedef unsigned long DWORD;
#endif

#define PI        (FTP)(3.14159265358979323846264)
#define HIBYTE(X)  ((X)>>8)
#define LOWBYTE(X) ((X)&0xFF)
#define SQR(X)     ((X)*(X))

// Main window spec ----------------------
#define MBTHICK 1
#define METHICK 1
#define MTOTHICK (MBTHICK+METHICK*2)
#define MTOP     MTOTHICK
#define MLEFT    MTOTHICK
#define MRIGHT   (getmaxx()-MTOTHICK)
#define MBOTTOM  (getmaxy()-MTOTHICK)
#define MWIDTH   (MRIGHT-MLEFT+1)
#define MHEIGHT  (MBOTTOM-MTOP-1)

#define CMD_MOUSE (CMD_USERDEF+0)


Dsp.Cpp
#pragma hdrfile "Dsp.Sym"

#include "dsps.h"

#define BTOP    80
#define BLEFT   30
#define MAXBUTTON 10

static enum {
     OBJ_TOP=0,OBJ_EXIT=0,
     OBJ_1,OBJ_2,OBJ_3,OBJ_4,OBJ_5,OBJ_6,OBJ_7,OBJ_8,OBJ_9,OBJ_10,
     OBJ_NULL
     } currentobj;

/*********************** กำหนดตัวแปร global **********************/
nt mouseexist;
nt textw,texth;
int KeyboardCmd;
MOUSE_PARAM MStat;
pWindowBasic mainwin;

TextButton pbutt[MAXBUTTON];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
TextButton pbuttquit;

/************************** การประกาศ Functions **************************/
void ShiftArray(FTP *array);
int TextInputFunction(void);
void WaitInput(void);

static void InitAllObjects(void);
static void DisplayAllObjects(void);
static int ActivateInput(void);
static void MakeCursoronObj(int _cobj);

extern void TransferH(void);  /* DSP Expreiment I */
extern void TransferHvsA1(void);  /* DSP Experiment II */
extern void MaximumA0(void);  /* DSP Experiment V */
extern void A0Testing(void);  /* DSP Experiment III */
extern void AdaptivePlot(void);  /* DSP Experiment IV */
extern void AdaptiveSine(void);  /* DSP Experiment IX */
extern void TransferGvsz(void);  /* DSP Experiment VI */
extern void AdaptiveA1(void);  /* DSP Experiment VII */
extern void AdaptiveA1nA0(void);  /* DSP Experiment VIII */
extern void AdaptiveG(void);  /* DSP Experiment X */

/************************** Function bodies **************************/
//-------------------------------------
void InitAllObjects(void)
{
  int _k;

  // Init main window ---------------------
  mainwin.AdjustPosition(MLEFT,MTOP,MWIDTH,MHEIGHT);
  mainwin.AdjustThick(MBTHICK,MTHICK);
  mainwin.AdjustColor(BLUE,WHITE,DARKGRAY,LIGHTGRAY,WHITE);

  // กำหนด parameters ให้ button ต่างๆ ----------
  pbutt[0].AssignText(" 1. Plot the transfer fn H(z) vs. Omega w");
  pbutt[1].AssignText(" 2. Plot the transfer fn H(z) vs. Alpha 1 a1");
  pbutt[2].AssignText(" 3. Plot a1(k) convergence curve");
  pbutt[3].AssignText(" 4. Test a0(k) for the Stochastic gradient usage");
  pbutt[4].AssignText(" 5. Plot the adaptive parameters, a0(k) versus a1(k)");
  pbutt[5].AssignText(" 6. Plot All Sine-Functions");
  pbutt[6].AssignText(" 7. Plot the Transfer fn G(z) vs. z");
  pbutt[7].AssignText(" 8. Plot a1(k) from unsimplified function with a0-fix");
  pbutt[8].AssignText(" 9. Plot a1(k) from unsimplified function");
  pbutt[9].AssignText("10. Plot a1(k) from simplified function");
  pbuttquit.AssignText(" 0. quit");

  pbuttquit.AdjustJustify(LHJUSTIFY|CVJUSTIFY);
  for(_k=0;_k<MAXBUTTON;_k++)
  {
    pbutt[_k].AdjustJustify(LHJUSTIFY|CVJUSTIFY);
    if (!_k)
      pbutt[_k].AdjustPosition(BLEFT,BTOP,textw*70,texth+4);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    else
      pbutt[_k].AdjustPosition(BLEFT,pbutt[_k-1].Bottom()+10,textw*70,texth+4);
  }
  pbuttquit.AdjustPosition(BLEFT,pbutt[MAXBUTTON-1].Bottom()+16,textw*70,texth+4);
}

//-----------------------------------------
void DisplayAllObjects(void)
{
  int _k;

  if (mouseexist)  HideMouse();

  mainwin.Paint();
  mainwin.UseWindowScope(TRUE);

  mainwin.OutTextat(BLEFT,BTOP-(texth<<1),"Acaptive Filter Experiment");
  mainwin.OutTextat(BLEFT,BTOP-texth,"-------------------");

  for(_k=0;_k<MAXBUTTON;_k++)
    pbutt[_k].Paint();
  pbuttquit.Paint();

  // แสดง cursor ล้อมรอบ OBJ ---------------
  setcolor(YELLOW);  MakeCursoronObj(currentobj);

  if (mouseexist)  ShowMouse();
}

//-----------------------------------------
void MakeCursoronObj(int _cobj)
{
  switch(_cobj)
  {
  case OBJ_EXIT:
      rectangle(pbuttquit.Left()-3,pbuttquit.Top()-3,
            pbuttquit.Right()+3,pbuttquit.Bottom()+3);
      break;
  default:
      rectangle(pbutt[_cobj-1].Left()-3,pbutt[_cobj-1].Top()-3,
            pbutt[_cobj-1].Right()+3,pbutt[_cobj-1].Bottom()+3);
  }
}

/*-----------------------------------------*/
int ActivateInput(void)
{
  int _k;
  int _x,_y;

  if (KeyboardCmd == (int)CMD_MOUSE)
  {
    if (MStat.butt & MOUSELBIT)  /* ถ้า mouse กดปุ่มซ้าย */
```

เอกสารนี้มีลิขสิทธิ์ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    {
      _x = MStat.x;  _y = MStat.y;
    }
    else
    {
      _x=_y=-1;
    }
  }
  else
  {
ChkKeyboardCmd:;
    _x=_y=-1;
    switch(KeyboardCmd)
    {
    case 0x010F: /* Shift-TAB */
    case 0x09:  /* TAB */
      // ลบ cursor ปัจจุบัน ————————————
      setcolor(BLUE);  /* ใช้สีพื้นของ mainwin */
      MakeCursoronObj(currentobj);
NextField:;
      if (KeyboardCmd == 0x09)
      {
        currentobj++;  /* เลื่อน cursor ไปตำแหน่งถัดไป */
        if (currentobj == OBJ_NULL)  currentobj = OBJ_TOP;
      }
      else
      {
        if (currentobj==OBJ_TOP)  currentobj = OBJ_NULL;
        currentobj--;
      }

      setcolor(YELLOW);
      MakeCursoronObj(currentobj);
      break;

   case 0x0D:  /* ENTER */
      switch(currentobj)
      {
      case OBJ_EXIT:
        _x = pbuttquit.Left();  _y = pbuttquit.Top();
        break;
      default:
        _x = pbutt[currentobj-1].Left();
        _y = pbutt[currentobj-1].Top();
        break;
      }
      break;
    }
  }

  // ตรวจว่าปุ่มไหนถูกกด ————————————
  for(_k=0; _k<MAXBUTTON; _k++)
  {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    if (pbutt[_k].Touch(_x,_y))
    {
      pbutt[_k].PushButton();  pbutt[_k].Paint();
      break;
    }
  }

  if (_k==MAXBUTTON)
  {
    if (pbuttquit.Touch(_x,_y))
    {
      pbuttquit.PushButton();  pbuttquit.Paint();
      return FALSE;  /* quit */
    }
  }
  else
  {
    switch(_k+1)
    {
    case OBJ_1:  TransferH();  break;
    case OBJ_2:  TransferHvsA1();  break;
    case OBJ_3:  MaximumA0();  break;
    case OBJ_4:  A0Testing();  break;
    case OBJ_5:  AdaptivePlot();  break;
    case OBJ_6:  AdaptiveSine();  break;
    case OBJ_7:  TransferGvsz();  break;
    case OBJ_8:  AdaptiveA1();  break;
    case OBJ_9:  AdaptiveA1nA0();  break;
    case OBJ_10:  AdaptiveG();  break;
    }
    pbutt[_k].ReleaseButton();
    DisplayAllObjects();
  }
  return TRUE;
}

/*------------------------------ Main Function ------------------------------*/
/*-------------------------------------------------------------------------*/
int main(void)
{
  int _drv,_mod;

  // เลือก graphics mode ----------------------
  _drv = VGA;  _mod = VGAHI;
  initgraph(&_drv,&_mod,"");
  textw = textwidth("M");  texth = textheight("M")+4;
  currentobj = OBJ_1;

  mouseexist = ResetMouseDrv(NULL);
  if (mouseexist)  ShowMouse();

  InitAllObjects();
  DisplayAllObjects();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  do {
    WaitInput();
  } while(ActivateInput());

  if (mouseexist)  HideMouse();
  closegraph();

  return 0;
}

/*-=-=-=-=-=-=-=-=-=-=-=-=-= Shared Functions =-=-=-=-=-=-=-=-=-=-=-=-=*/
//---------------------------------------
int TextInputFunction(void)
{
  int _ch;
  MOUSE_PARAM _ms;

  _ch = CMD_NOINPUT;
  if (kbhit())
  {
    if ((_ch = getch()) == 0)  _ch = getch()+256;
    switch(_ch)
    {
    case 0x0D:  /*  ENTER key  */
            KeyboardCmd = 0x0D;
            _ch = CMD_EOT;
            break;
    case 0x08:  _ch = CMD_BACKRUB;  break;
    case 0x09:  /*  TAB next field  */
    case 0x010F:/*  Shift-TAB back field  */
            KeyboardCmd = _ch;
            _ch = CMD_EOT;
            break;
    case 0x19:  _ch = CMD_DELLN;    break; /*  Ctrl-Y  */
    case 0x1B:  _ch = CMD_EOT;  /*  ESC  */
            KeyboardCmd = _ch;
            break;
    }
  }
  else
  if (mouseexist)
  {
    MouseStatus(_ms);
    if (_ms.x != MStat.x || _ms.y != MStat.y || _ms.butt != MStat.butt)
    {
      MStat.x = _ms.x;  MStat.y = _ms.y;  MStat.butt = _ms.butt;
      _ch = CMD_MOUSE;
    }
  }

  return _ch;
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
//-------------------------------------
void WaitInput(void)
{
  int _cmd;

  if (mouseexist)  ShowMouse();
  KeyboardCmd = CMD_NOINPUT;
  do {
    _cmd = TextInputFunction();
  } while(_cmd == CMD_NOINPUT);
  if (KeyboardCmd == CMD_NOINPUT)  KeyboardCmd = _cmd;
  if (mouseexist)  HideMouse();
}

/*-------------------------------------*/
void ShiftArray(FTP *array)
{
  array[2] = array[1];
  array[1] = array[0];
  array[0] = (FTP)0.0;
}


DspExp1.Cpp
/*-------------------------------------*/
/*                    -2
** H(z) = (1-w0)    1 - z
**       ---- * ----------------
**                   -1      -2
**        2     1 - a1(k)*(1+w0)*z  + w0*z
*/
/*-------------------------------------*/
#pragma hdrfile "DspExp1.Sym"
#include "dsps.h"

/*------------------- define Macros ------------------*/
#define DATAFILENAME "DspExp1"

#define MAXGRAPH 20
#define MAXOUTPUT 1000

#define BTHICK 2
#define MINX  50
#define MINY  50
#define WIDTH  (MWIDTH-100)
#define HEIGHT (MHEIGHT-80)
#define MAXGRAPHX (FTP)PI
#define MAXGRAPHY 1.1

#define SBTHICK 1
#define SWIDTH  (textw*60+(SBTHICK<<1))
```

เอกสารนี้ใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#define SHEIGHT  (texth*3+(SBTHICK<<1))
#define SMINX    ((getmaxx()-SWIDTH)>>1)
#define SMINY    ((getmaxy()-SHEIGHT)>>1)

static enum {
   OBJ_TOP=1,OBJ_PLOTGRAPH=1,OBJ_SAVE,OBJ_EXIT,OBJ_IA0,OBJ_IA1,OBJ_NULL
   } currentobj;

/*^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^ Global variables ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^*/
static FTP vA0,vA1;
static FTP *aaOutput[MAXGRAPH],*aOutput;
static int graphidx;

static pTextInput iA0,iA1;  /* input a0, a1 */
static TextButton bExit,bPlotGraph,bSave;
static mBaseGraphic gpx;

extern int textw,texth;
extern pWindowBasic mainwin;
extern int KeyboardCmd;
extern MOUSE_PARAM MStat;

/*-------------------- Function Prototypes --------------------*/
/* First part ----------------------------------------*/
void TransferH(void);
static FTP Function_H(FTP _w);
static void Plot_TransferH(void);
static void Save2File(void);
static void InitAllObjects(void);
static void ShowGetParametersDialog(void);
static int  ActivateInput(void);

extern void WaitInput(void);
extern int TextInputFunction(void);
#pragma hdrstop

/*^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^ Functions body ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^*/
/*----------------------------------------*/
FTP Function_H(FTP _w)
{
  FTP _B;
  FTP _cosw,_sinw;

  _cosw = cos(_w); _sinw = sin(_w);
  _B = vA1*SQR(1+vA0);

  return (1-vA0)*_sinw/sqrt(2*vA0*cos(2*_w)+2*_B*_cosw+SQR(vA0)+_B*vA1+1);
}

//----------------------------------------
void InitAllObjects(void)
{
  iA0.AdjustString("0",11);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
iA1.AdjustString("0",11);

// ให้รับ input ตั้งแต่เริ่มเข้าโปรแกรม ————————
KeyboardCmd = (int)CMD_MOUSE;
MStat.x = textw+100;  MStat.y = 5;  /* ให้รับ alpha 0 */
MStat.butt = MOUSELBIT;

iA0.AdjustPosition(textw+100, 5);  iA0.AdjustThick(3);
iA1.AdjustPosition(textw+100,25);  iA1.AdjustThick(3);
iA0.inputfn=iA1.inputfn=TextInputFunction;

bPlotGraph.AdjustPosition(MRIGHT-300, 5,80,texth+10);
bPlotGraph.AssignText("Plot graph");
bPlotGraph.AdjustJustify(CHJUSTIFY|CVJUSTIFY);
bPlotGraph.ReleaseButton();

bSave.AdjustPosition(MRIGHT-200, 5,80,texth+10);
bSave.AssignText("Save");
bSave.AdjustJustify(CHJUSTIFY|CVJUSTIFY);
bSave.ReleaseButton();

bExit.AdjustPosition(MRIGHT-100, 5,80,texth+10);
bExit.AssignText("Exit");
bExit.AdjustJustify(CHJUSTIFY|CVJUSTIFY);
bExit.ReleaseButton();

// ทำการ scale ให้พอเหมาะและเปลี่ยนจุด Origin ————*/
gpx.AdjustGraphSize(0,0,WIDTH,HEIGHT);
gpx.xzoomfactor = WIDTH/MAXGRAPHX;
gpx.yzoomfactor = HEIGHT/MAXGRAPHY;
gpx.MoveOrigin(0,0,20/gpx.yzoomfactor);
}

//————————————————————————
void ShowGetParametersDialog(void)
{
  EDGECOLORPACK _ep;

  _ep.Up = _ep.Left = DARKCOLOR;  _ep.Down = _ep.Right = HIGHCOLOR;

  // clear main window ————————————
  setfillstyle(SOLID_FILL,BLUE);
  bar(0,0,MWIDTH,MHEIGHT);

  // เขียน Legend ————————————————
  setcolor(YELLOW);
  outtextxy(getmaxx()/2,MINY+HEIGHT+textheight("w"),"w"); /* พิมพ์ชื่อตัวแปรที่แกน X */
  outtextxy(MINX-textwidth("|H(w)|"),getmaxy()/2, "|H(w)|"); /* พิมพ์ชื่อตัวแปรที่แกน Y */

  mainwin.OutTextat(textw, 5,"u0 [ 0, 1]");
  mainwin.OutTextat(textw,25,"u1 [-1, 1]");

  iA0.Paint();  iA1.Paint();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  bPlotGraph.Paint();  bSave.Paint();  bExit.Paint();

  // วาดพื้นที่สำหรับวาดกราฟ ————————————
  LineStrip(MINX,MINY,WIDTH,HEIGHT,_ep,BLACK,WHITE,LIGHTGRAY,BTHICK);

  mainwin.SetViewport(MINX,MINY,WIDTH,HEIGHT,TRUE);
  setcolor(GREEN);
  gpx.DrawGridxy(PI/20,0.1);  /* ความละเอียดของจอทหารด้วยจำนวนกริดที่ต้องการ */

  setcolor(BLACK);
  gpx.DisplayAxis();

  setcolor(RED);
  gpx.DrawNumberxy(PI/10,0.1,2,2);  /* เขียนตัวเลขที่กริด */

  mainwin.UseWindowScope(TRUE);
}

/*————————————————————————*/
int ActivateInput(void)
{
  int _x,_y;

  if (KeyboardCmd == (int)CMD_MOUSE)
  {
    if (MStat.butt & MOUSELBIT)  /* ถ้า mouse กดปุ่มซ้าย */
    {
      _x = MStat.x;  _y = MStat.y;
    }
    else
    {
      _x=_y=-1;
    }
  }
  else
  {
ChkKeyboardCmd:;
    _x=_y=-1;
    switch(KeyboardCmd)
    {
    case 0x010F: /* Shift-TAB */
    case 0x09:  /* TAB */
      // ลบ cursor ปัจจุบัน ————————————————
      setcolor(BLUE);  /* ใช้สีพื้นของ mainwin */
      switch(currentobj)
      {
      case OBJ_PLOTGRAPH:
        rectangle(bPlotGraph.Left()-3,bPlotGraph.Top()-3,
                  bPlotGraph.Right()+3,bPlotGraph.Bottom()+3);
        break;
      case OBJ_SAVE:
        rectangle(bSave.Left()-3,bSave.Top()-3,
                  bSave.Right()+3,bSave.Bottom()+3);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            break;
        case OBJ_EXIT:
            rectangle(bExit.Left()-3,bExit.Top()-3,
                    bExit.Right()+3,bExit.Bottom()+3);
            break;
        }
NextField:;
        if (KeyboardCmd == 0x09)
        {
          currentobj++;  /* เลื่อน cursor ไปตำแหน่งถัดไป */
          if (currentobj == OBJ_NULL)  currentobj = OBJ_TOP;
        }
        else
        {
          if (currentobj==OBJ_TOP)  currentobj = OBJ_NULL;
          currentobj--;
        }

        setcolor(YELLOW);
        switch(currentobj)
        {
        case OBJ_PLOTGRAPH:
            rectangle(bPlotGraph.Left()-3,bPlotGraph.Top()-3,
                    bPlotGraph.Right()+3,bPlotGraph.Bottom()+3);
            break;
        case OBJ_SAVE:
            rectangle(bSave.Left()-3,bSave.Top()-3,
                    bSave.Right()+3,bSave.Bottom()+3);
            break;
        case OBJ_EXIT:
            rectangle(bExit.Left()-3,bExit.Top()-3,
                    bExit.Right()+3,bExit.Bottom()+3);
            break;
        case OBJ_IA0:
            _x = textw+100;  _y = 5;  break;
        case OBJ_IA1:
            _x = textw+100;  _y = 25;  break;
        }
        break;

    case 0x0D:  /* ENTER */
        if (currentobj == OBJ_IA0 || currentobj == OBJ_IA1)
        {
          KeyboardCmd = 0x09;  /* ทำหน้าที่เป็น TAB แทน */

          goto NextField;
        }
        switch(currentobj)
        {
        case OBJ_PLOTGRAPH:
            _x = bPlotGraph.Left();  _y = bPlotGraph.Top();
            break;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            case OBJ_SAVE:
                _x = bSave.Left();   _y = bSave.Top();
                break;

            case OBJ_EXIT:
                _x = bExit.Left();   _y = bExit.Top();
                break;
            }
            break;
        }
    }

    if (iA0.Touch(_x,_y))
    {
        iA0.Getstring();
        sscanf((char *)iA0, "%" FTP_PREFIX "f", &vA0);

        goto ChkKeyboardCmd;
    }
    else
    if (iA1.Touch(_x,_y))
    {
        iA1.Getstring();
        sscanf((char *)iA1, "%" FTP_PREFIX "f", &vA1);

        goto ChkKeyboardCmd;
    }
    else
    if (bPlotGraph.Touch(_x,_y))
    {
        bPlotGraph.PushButton();   bPlotGraph.Paint();
        Plot_TransferH();
        bPlotGraph.ReleaseButton();   bPlotGraph.Paint();
    }
    else
    if (bSave.Touch(_x,_y))
    {
        bSave.PushButton();   bSave.Paint();
        Save2File();
        bSave.ReleaseButton();   bSave.Paint();
    }
    else
    if (bExit.Touch(_x,_y))
    {
        bExit.PushButton();
        bExit.Paint();
        return FALSE;
    }
    return TRUE;
}

//-----------------------------------------------------------
void Plot_TransferH(void)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น ทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
  int _k;
  FTP _w,_stpw;
  FTP _miny, _maxy;

  // กำหนดค่าเริ่มต้น ————————————————
  _stpw = PI/MAXOUTPUT;
  _miny = _maxy = 0.0;
  mainwin.SetViewport(MINX,MINY,WIDTH,HEIGHT,TRUE);
  gpx.Moveto(0,aOutput[0]=Function_H(0));

  setcolor(BLUE);
  for(_w=_k=0; _k<MAXOUTPUT; _k++,_w+=_stpw)
  {
    gpx.Lineto(_w,aOutput[_k]=Function_H(_w));
    if (_miny > aOutput[_k]) _miny = aOutput[_k];
    if (_maxy < aOutput[_k]) _maxy = aOutput[_k];
  }
  mainwin.UseWindowScope(TRUE);
}

//————————————————————————————
void Save2File(void)
{
  WORD _k,_i;
  FTP _w;
  FILE *_fhandle,*_hhandle;

  if ((_hhandle = fopen(DATAFILENAME ".Hdr","wt")) != NULL)
  {
    // เขียนจำนวนสูงที่สุด และจำนวนกราฟที่ส่วนหัว .Hdr ————
    fprintf(_hhandle,"%u %d\n", MAXOUTPUT, graphidx);
    fclose(_hhandle);

    // เขียนข้อมูลจริงๆไว้ที่ .Dat ————————————
    if ((_fhandle = fopen(DATAFILENAME ".Dat","wt")) != NULL)
    {
      for(_k=0,_w=0.0;_k<MAXOUTPUT;_k++,_w+=PI/MAXOUTPUT)
      {
        fprintf(_fhandle,"\n%" FTP_PREFIX "g ",_w);
        for(_i=0;_i<graphidx;_i++)
          fprintf(_fhandle,"%" FTP_PREFIX "g ",aaOutput[_i][_k]);
      }
      fclose(_fhandle);
    }
  }
}

/*————————————————————————————*/
void TransferH(void)
{
  currentobj = OBJ_IA0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    graphidx = 0;
    try {
      aOutput = new FTP [MAXOUTPUT];
      aaOutput[graphidx++]=aOutput;
    }
    catch (xalloc) {
      return;
    }

    InitAllObjects();
    ShowGetParametersDialog();

    while(ActivateInput())
      WaitInput();

    while(graphidx)
      delete[] aaOutput[--graphidx];

    mainwin.UseWindowScope(FALSE);
}


DspExp2.Cpp
/*-------------------------------------------------------*/
/*                    -2                                 */
** H(z) = (1+b0)          1 - z                          **
**        ----- * ------------------                     **
**                          -1    -2                     **
**          2     1 - b1(k)*(1+b0)*z   + b0*z           **
*/
/*-------------------------------------------------------*/
#pragma hdrfile "DspExp2.Sym"
#include "dsps.h"

/*-=-=-=-=-=-=-=-=-=-=-=-=-= define Macros =-=-=-=-=-=-=-=-=-=-=-=-=-=*/
#define DATAFILENAME "DspExp2"

#define MAXGRAPH 20
#define MAXOUTPUT 1000

#define BTHICK 2
#define MINX  50
#define MINY  50
#define WIDTH  (MWIDTH-100)
#define HEIGHT (MHEIGHT-70)
#define MAXGRAPHX (FTP)1.0
#define MAXGRAPHY 1.1

#define SBTHICK  1
#define SWIDTH   (textw*60+(SBTHICK<<1))
#define SHEIGHT  (texth*3+(SBTHICK<<1))
#define SMINX    ((getmaxx()-SWIDTH)>>1)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#define SMINY   ((getmaxy()-SHEIGHT)>>1)

static enum {
   OBJ_TOP=1,
   OBJ_PLOTGRAPH=1,OBJ_SAVE,OBJ_EXIT,
   OBJ_IW0,OBJ_IA0,
   OBJ_NULL
   } currentobj;

/*^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^ Global variables ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^*/
static FTP vA0,vW0;
static FTP *aaOutput[MAXGRAPH],*aOutput;
static int graphidx;

static pTextInput iW0,iA0;
static TextButton bExit,bPlotGraph,bSave;
static mBaseGraphic gpx;

extern int textw,texth;
extern pWindowBasic mainwin;
extern int KeyboardCmd;
extern MOUSE_PARAM MStat;

/*-------------------------- Function  Prototypes --------------------------*/
/* First part ------------------------------*/
void TransferHvsA1(void);
static FTP Function_H(FTP _w);
static void Plot_TransferH(void);
static void Save2File(void);
static void InitAllObjects(void);
static void ShowGetParametersDialog(void);
static int  ActivateInput(void);

extern void WaitInput(void);
extern int TextInputFunction(void);
#pragma hdrstop

/*^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^ Functions body ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^*/
/*-----------------------------------------*/
FTP Function_H(FTP _a1)
{
  FTP _B;
  FTP _cosw,_sinw;

  _cosw = cos(vW0);  _sinw = sin(vW0);
  _B = _a1*SQR(1+vA0);

  return (1-vA0)*_sinw/sqrt(2*vA0*cos(2*vW0)+2*_B*_cosw+SQR(vA0)+_B*_a1+1);
}

//-----------------------------------------
void InitAllObjects(void)
{
```

เอกสารนี้มีลิขสิทธิ์ สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  iW0.AdjustString("0",11);
  iA0.AdjustString("0",11);

  // ให้รับ input ตั้งแต่เริ่มเข้าโปรแกรม ————————
  KeyboardCmd = (int)CMD_MOUSE;
  MStat.x = textw+100;  MStat.y = 5;  /* ให้รับ alpha 0 */
  MStat.butt = MOUSELBIT;

  iW0.AdjustPosition(textw+100, 5);  iW0.AdjustThick(3);
  iA0.AdjustPosition(textw+100,25);  iA0.AdjustThick(3);
  iW0.inputfn=iA0.inputfn=TextInputFunction;

  bPlotGraph.AdjustPosition(MRIGHT-300, 5,80,texth+10);
  bPlotGraph.AssignText("Plot graph");
  bPlotGraph.AdjustJustify(CHJUSTIFY|CVJUSTIFY);
  bPlotGraph.ReleaseButton();

  bSave.AdjustPosition(MRIGHT-200, 5,80,texth+10);
  bSave.AssignText("Save");
  bSave.AdjustJustify(CHJUSTIFY|CVJUSTIFY);
  bSave.ReleaseButton();

  bExit.AdjustPosition(MRIGHT-100, 5,80,texth+10);
  bExit.AssignText("Exit");
  bExit.AdjustJustify(CHJUSTIFY|CVJUSTIFY);
  bExit.ReleaseButton();

  // ทำการ scale ให้พอเหมาะและเปลี่ยนจุด Origin ————————*/
  gpx.AdjustGraphSize(0,0,WIDTH,HEIGHT);
  gpx.xzoomfactor = WIDTH/MAXGRAPHX;
  gpx.yzoomfactor = HEIGHT/MAXGRAPHY;
  gpx.MoveOrigin(0.0,0.0);
  gpx.MoveOrigin(0.0,20/gpx.yzoomfactor);
}

//————————————————————————————
void ShowGetParametersDialog(void)
{
  EDGECOLORPACK _ep;

  _ep.Up = _ep.Left = DARKCOLOR;  _ep.Down = _ep.Right = HIGHCOLOR;

  // clear main window ————————————————
  setfillstyle(SOLID_FILL,BLUE);
  bar(0,0,MWIDTH,MHEIGHT);

  // เขียน Legend ————————————————
  setcolor(YELLOW);
  outtextxy(getmaxx()/2,MINY+HEIGHT+texthegnt("|1|"),"|1|"); /* พิมพ์ชื่อตัวแปรที่แกน X */
  outtextxy(MINX-textwidth("|H(w)|"),getmaxy()/2, "|H(w)|"); /* พิมพ์ชื่อตัวแปรที่แกน Y */

  mainwin.OutTextat(textw, 5,"w0");
  mainwin.OutTextat(textw,25,"0 [0, 1]");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    iW0.Paint();   iA0.Paint();
    bPlotGraph.Paint();   bSave.Paint();   bExit.Paint();

    // วาดพื้นที่สำหรับวาดกราฟ ————————————
    LineStrip(MINX,MINY,WIDTH,HEIGHT,_sp,BLACK,WHITE,LIGHTGRAY,BTHICK);

    mainwin.SetViewport(MINX,MINY,WIDTH,HEIGHT,TRUE);
    setcolor(GREEN);
    gpx.DrawGridxy(MAXGRAPHX/20,0.1);  /* ความละเอียดของจอหารด้วยจำนวนกริดที่ต้องการ */

    setcolor(BLACK);
    gpx.DisplayAxis();

    setcolor(RED);
    gpx.DrawNumberxy(MAXGRAPHX/10,0.1,2,2);  /* เขียนตัวเลขที่กริด */

    mainwin.UseWindowScope(TRUE);
}

/*————————————————————————*/
int ActivateInput(void)
{
  int _x,_y;

  if (KeyboardCmd == (int)CMD_MOUSE)
  {
    if (MStat.butt & MOUSELBIT)  /* ถ้า mouse กดปุ่มซ้าย */
    {
      _x = MStat.x;   _y = MStat.y;
    }
    else
    {
      _x=_y=-1;
    }
  }
  else
  {
ChkKeyboardCmd:;
    _x=_y=-1;
    switch(KeyboardCmd)
    {
    case 0x010F: /* Shift-TAB */
    case 0x09:  /* TAB */
      // ลบ cursor ปัจจุบัน ————————————
      setcolor(BLUE);  /* ใช้สีพื้นของ mainwin */
      switch(currentobj)
      {
      case OBJ_PLOTGRAPH:
        rectangle(bPlotGraph.Left()-3,bPlotGraph.Top()-3,
                  bPlotGraph.Right()+3,bPlotGraph.Bottom()+3);
        break;
      case OBJ_SAVE:
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            rectangle(bSave.Left()-3,bSave.Top()-3,
                    bSave.Right()+3,bSave.Bottom()+3);
            break;
        case OBJ_EXIT:
            rectangle(bExit.Left()-3,bExit.Top()-3,
                    bExit.Right()+3,bExit.Bottom()+3);
            break;
        }
NextField::
        if (KeyboardCmd == 0x09)
        {
          currentobj++;  /* เลื่อน cursor ไปตำแหน่งถัดไป */
          if (currentobj == OBJ_NULL)  currentobj = OBJ_TOP;
        }
        else
        {
          if (currentobj==OBJ_TOP)  currentobj = OBJ_NULL;
          currentobj--;
        }

        setcolor(YELLOW);
        switch(currentobj)
        {
        case OBJ_PLOTGRAPH:
            rectangle(bPlotGraph.Left()-3,bPlotGraph.Top()-3,
                    bPlotGraph.Right()+3,bPlotGraph.Bottom()+3);
            break;
        case OBJ_SAVE:
            rectangle(bSave.Left()-3,bSave.Top()-3,
                    bSave.Right()+3,bSave.Bottom()+3);
            break;
        case OBJ_EXIT:
            rectangle(bExit.Left()-3,bExit.Top()-3,
                    bExit.Right()+3,bExit.Bottom()+3);
            break;
        case OBJ_IW0:
            _x = textw+100;  _y = 5;  break;
        case OBJ_IA0:
            _x = textw+100;  _y = 25;  break;
        }
        break;

    case 0x0D:  /* ENTER */
        switch(currentobj)
        {
        case OBJ_PLOTGRAPH:
            _x = bPlotGraph.Left();  _y = bPlotGraph.Top();
            break;

        case OBJ_SAVE:
            _x = bSave.Left();  _y = bSave.Top();
            break;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        case OBJ_EXIT:
            _x = bExit.Left();   _y = bExit.Top();
            break;
        default:
            KeyboardCmd = 0x09;  /* ทำหน้าที่เป็น TAB แทน */
            goto NextField;
        }
        break;
    }
}

if (iW0.Touch(_x,_y))
{
    iW0.Getstring();
    sscanf((char *)iW0, "%" FTP_PREFIX "f", &vW0);

    goto ChkKeyboardCmd;
}
else
if (iA0.Touch(_x,_y))
{
    iA0.Getstring();
    sscanf((char *)iA0, "%" FTP_PREFIX "f", &vA0);

    goto ChkKeyboardCmd;
}
else
if (bPlotGraph.Touch(_x,_y))
{
    bPlotGraph.PushButton();  bPlotGraph.Paint();
    Plot_TransferH();
    bPlotGraph.ReleaseButton();  bPlotGraph.Paint();
}
else
if (bSave.Touch(_x,_y))
{
    bSave.PushButton();  bSave.Paint();
    Save2File();
    bSave.ReleaseButton();  bSave.Paint();
}
else
if (bExit.Touch(_x,_y))
{
    bExit.PushButton();
    bExit.Paint();
    return FALSE;
}
return TRUE;
}

//-----------------------------------------------------
void Plot_TransferH(void)
{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    int _k;
    FTP _a1,_stp;
    FTP _miny, _maxy;

    // กำหนดค่าเริ่มต้น ————————————————
    _stp = 1.0/MAXOUTPUT;
    _miny = _maxy = 0.0;
    mainwin.SetViewport(MINX,MINY,WIDTH,HEIGHT,TRUE);
    gpx.Moveto(0,aOutput[0]=Function_H(0));

    setcolor(BLUE);
    for(_a1=_k=0; _k<MAXOUTPUT; _k++,_a1+=_stp)
    {
      gpx.Lineto(_a1,aOutput[_k]=Function_H(_a1));
      if (_miny > aOutput[_k]) _miny = aOutput[_k];
      if (_maxy < aOutput[_k]) _maxy = aOutput[_k];
    }
    mainwin.UseWindowScope(TRUE);
}

//————————————————————————————
void Save2File(void)
{
    WORD _k,_i;
    FTP _a1;
    FILE *_fhandle,*_hhandle;

    if ((_hhandle = fopen(DATAFILENAME ".Hdr","wt")) != NULL)
    {
      // เขียนจำนวนสูงที่สุด และจำนวนกราฟที่ส่วนหัว .Hdr ————
      fprintf(_hhandle,"%u %d\n", MAXOUTPUT, graphicx);
      fclose(_hhandle);

      // เขียนข้อมูลจริงๆไว้ที่ .Dat ————————————
      if ((_fhandle = fopen(DATAFILENAME ".Dat","wt")) != NULL)
      {
        for(_k=0,_a1=0.0;_k<MAXOUTPUT;_k++,_a1+=1.0/MAXOUTPUT)
        {
          fprintf(_fhandle,"\n%" FTP_PREFIX "g ",_a1);
          for(_i=0;_i<graphicx;_i++)
            fprintf(_fhandle,"%" FTP_PREFIX "g ",
              ((FTP huge *)aaOutput[_i])[_k]);
        }
        fclose(_fhandle);
      }
    }
}

/*————————————————————————————*/
void TransferHvsA1(void)
{
    currentobj = OBJ_IW0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    graphidx = 0;
    try {
      aOutput = new FTP [MAXOUTPUT];
      aaOutput[graphidx++]=aOutput;
    }
    catch (xalloc) {
      return;
    }

    initAllObjects();
    ShowGetParametersDialog();

    while(ActivateInput())
      WaitInput();

    while(graphidx)
      delete[] aaOutput[--graphidx];

    mainwin.UseWindowScope(FALSE);
}


DspExp3.Cpp
// โปรแกรม plot กราฟทดสอบว่าจะใช้ alpha 0 แบบ adaptive ได้หรือไม่
//

#pragma hdrfile "DspExp3.Sym"
#include "dsps.h"

/*-------------------- define Macros --------------------*/
#define DATAFILENAME "DspExp4"

#define MAXGRAPH 20
#define MAXOUTPUT 1000

#if MAXOUTPUT*sizeof(FTP) > 65535
#define FAR_ALLOC
#endif

#define BTHICK 2
#define MINX  50
#define MINY  50
#define WIDTH  (MWIDTH-100)
#define HEIGHT (MHEIGHT-70)
#define MAXGRAPHX 1.0

#define SBTHICK  1
#define SWIDTH   (textw*60+(SBTHICK<<1))
#define SHEIGHT  (texth*3+(SBTHICK<<1))
#define SMINX    ((getmaxx()-SWIDTH)>>1)
#define SMINY    ((getmaxy()-SHEIGHT)>>1)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
static enum {
    OBJ_TOP=1,
    OBJ_PLOTGRAPH=1,OBJ_SAVE,OBJ_EXIT,
    OBJ_IA1,OBJ_IW0,
    OBJ_NULL
    } currentobj;

/*^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^ Global variables ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^*/
static FTP vA0,vA1,vW0;
static FTP *aaOutput[MAXGRAPH],*aOutput;
static int graphidx;

static FTP MAXGRAPHY;
static cTextInput iA1,iW0;
static TextButton bExit,bPlotGraph,bSave;
static mBaseGraphic gpx;

extern int textw,texth;
extern cWindowBasic mainwin;
extern int KeyboardCmd;
extern MOUSE_PARAM MStat;

/*---------------------------------- Function  Prototypes ----------------------------------*/
/* First part ---------------------------------------------*/
void ADTesting(void);

static void ProcessingGraph(void);
static void Save2File(void);
static void InitAllObjects(void);
static void ShowGetParametersDialog(void);
static int  ActivateInput(void);

extern void WaitInput(void);
extern int TextInputFunction(void);
#pragma hdrstop

/*^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^ Functions body ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^*/
//---------------------------------------------------------
void InitAllObjects(void)
{
  MAXGRAPHY=1.1;  /* Set ให้เป็น default */

  iA1.AdjustString("0",11);
  iW0.AdjustString("0",11);

  // ให้รับ input ตั้งแต่เริ่มเข้าโปรแกรม ----------
  KeyboardCmd = (int)CMD_MOUSE;
  MStat.x = textw*100;  MStat.y = 5;  /* ให้รับ alpha 0 */
  MStat.butt = MOUSELBIT;

  iA1.AdjustPosition(textw*100, 5);  iA1.AdjustThick(3);
  iW0.AdjustPosition(textw*100,25);  iW0.AdjustThick(3);
  iW0.inputfn=iA1.inputfn=TextInputFunction;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  bPlotGraph.AdjustPosition(MRIGHT-300, 5,80,texth+10);
  bPlotGraph.AssignText("Plot graph");
  bPlotGraph.AdjustJustify(CHJUSTIFY|CVJUSTIFY);
  bPlotGraph.ReleaseButton();

  bSave.AdjustPosition(MRIGHT-200, 5,80,texth+10);
  bSave.AssignText("Save");
  bSave.AdjustJustify(CHJUSTIFY|CVJUSTIFY);
  bSave.ReleaseButton();

  bExit.AdjustPosition(MRIGHT-100, 5,80,texth+10);
  bExit.AssignText("Exit");
  bExit.AdjustJustify(CHJUSTIFY|CVJUSTIFY);
  bExit.ReleaseButton();

  // ทำการ scale ให้พอเหมาะและเปลี่ยนจุด Origin ————————*/
  gpx.AdjustGraphSize(0,0,WIDTH,HEIGHT);
  gpx.xzoomfactor = WIDTH/MAXGRAPHX;
  gpx.yzoomfactor = HEIGHT/MAXGRAPHY;
  gpx.MoveOrigin(0.0,30/gpx.yzoomfactor);
}

//————————————————————————————
void ShowGetParametersDialog(void)
{
  EDGECOLORPACK _ec;

  _ec.Up = _ec.Left = DARKCOLOR;  _ec.Down = _ec.Right = HIGHCOLOR;

  // clear main window ————————————————
  setfillstyle(SOLID_FILL,BLUE);
  bar(0,0,MWIDTH,MHEIGHT);

  // เขียน Legend ————————————————
  setcolor(YELLOW);
  outtextxy(getmaxx()/2,MINY+HEIGHT+textheight("I0"),"I0"); /* พิมพ์ชื่อตัวแปรที่แกน X */
  outtextxy(MINX-textwidth("H(I0)"),getmaxy()/2,"H(I0)"); /* พิมพ์ชื่อตัวแปรที่แกน Y */

  mainwin.OutTextat(textw, 5,"I1 [-1, 1]");
  mainwin.OutTextat(textw,25,"w0");

  W0.Paint();  iA1.Paint();
  bPlotGraph.Paint();  bSave.Paint();  bExit.Paint();

  // วาดพื้นที่สำหรับวาดกราฟ ————————————
  LineStrip(MINX,MINY,WIDTH,HEIGHT,_ec.BLACK,WHITE,LIGHTGRAY,BTHICK);

  mainwin.SetViewport(MINX,MINY,WIDTH,HEIGHT,TRUE);
  setcolor(GREEN);
  gpx.DrawGridxy(MAXGRAPHX/20,0.1);  /* ความละเอียดของจอหารด้วยจำนวนกริดที่ต้องการ */
  setcolor(BLACK);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    gpx.DisplayAxis();

    setcolor(RED);
    gpx.DrawNumberxy(MAXGRAPHX/10,0.1,2,2); /* เขียนตัวเลขที่กริด */

    mainwin.UseWindowScope(TRUE);
}

/*-------------------------------------*/
int ActivateInput(void)
{
    int _x,_y;

    if (KeyboardCmd == (int)CMD_MOUSE)
    {
      if (MStat.butt & MOUSELBIT)  /* ถ้า mouse กดปุ่มซ้าย */
      {
        _x = MStat.x;  _y = MStat.y;
      }
      else
      {
        _x=_y=-1;
      }
    }
    else
    {
ChkKeyboardCmd:;
      _x=_y=-1;
      switch(KeyboardCmd)
      {
      case 0x010F: /* Shift-TAB */
      case 0x09:  /* TAB */
        // ลบ cursor ปัจจุบัน ---------------------
        setcolor(BLUE);  /* ใช้สีพื้นของ mainwin */
        switch(currentobj)
        {
        case OBJ_PLOTGRAPH:
          rectangle(bPlotGraph.Left()-3,bPlotGraph.Top()-3,
              bPlotGraph.Right()+3,bPlotGraph.Bottom()+3);
          break;
        case OBJ_SAVE:
          rectangle(bSave.Left()-3,bSave.Top()-3,
              bSave.Right()+3,bSave.Bottom()+3);
          break;
        case OBJ_EXIT:
          rectangle(bExit.Left()-3,bExit.Top()-3,
              bExit.Right()+3,bExit.Bottom()+3);
          break;
        }
NextField:;
        if (KeyboardCmd == 0x09)
        {
          currentobj++;  /* เลื่อน cursor ไปตำแหน่งถัดไป */
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    if (currentobj == OBJ_NULL)  currentobj = OBJ_TOP;
  }
  else
  {
    if (currentobj==OBJ_TOP)  currentobj = OBJ_NULL;
    currentobj--;
  }

  setcolor(YELLOW);
  switch(currentobj)
  {
  case OBJ_PLOTGRAPH:
     rectangle(bPlotGraph.Left()-3,bPlotGraph.Top()-3,
           bPlotGraph.Right()+3,bPlotGraph.Bottom()+3);
     break;
  case OBJ_SAVE:
     rectangle(bSave.Left()-3,bSave.Top()-3,
           bSave.Right()+3,bSave.Bottom()+3);
     break;
  case OBJ_EXIT:
     rectangle(bExit.Left()-3,bExit.Top()-3,
           bExit.Right()+3,bExit.Bottom()+3);
     break;
  case OBJ_IA1:
     _x = textw+100;  _y = 5;  break;
  case OBJ_IW0:
     _x = textw+100;  _y = 25;  break;
  }
  break;

case 0x0D:  /* ENTER */
  if (currentobj == OBJ_IW0 || currentobj == OBJ_IA1)
  {
    KeyboardCmd = 0x09;  /* ทำหน้าที่เป็น TAB แทน */

    goto NextField;
  }
  switch(currentobj)
  {
  case OBJ_PLOTGRAPH:
     _x = bPlotGraph.Left();  _y = bPlotGraph.Top();
     break;

  case OBJ_SAVE:
     _x = bSave.Left();  _y = bSave.Top();
     break;

  case OBJ_EXIT:
     _x = bExit.Left();  _y = bExit.Top();
     break;
  }
  break;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    }

    if (iW0.Touch(_x,_y))
    {
      iW0.Getstring();
      sscanf((char *)iW0, "%" FTP_PREFIX "f", &vW0);

      goto ChkKeyboardCmd;
    }
    else
    if (iA1.Touch(_x,_y))
    {
      iA1.Getstring();
      sscanf((char *)iA1, "%" FTP_PREFIX "f", &vA1);

      goto ChkKeyboardCmd;
    }
    else
    if (bPlotGraph.Touch(_x,_y))
    {
      bPlotGraph.PushButton();  bPlotGraph.Paint();
      ProcessingGraph();
      bPlotGraph.ReleaseButton();  bPlotGraph.Paint();
    }
    else
    if (bSave.Touch(_x,_y))
    {
      bSave.PushButton();  bSave.Paint();
      Save2File();
      bSave.ReleaseButton();  bSave.Paint();
    }
    else
    if (bExit.Touch(_x,_y))
    {
      bExit.PushButton();
      bExit.Paint();
      return FALSE;
    }
    return TRUE;
}

//-------------------------------------
void ProcessingGraph(void)
{
  int _k;
  FTP _stp;
  FTP _a,_b,_c,_d;
  FTP _miny, _maxy;

  // กำหนดค่าเริ่มต้น ---------------------
  _stp = 1.0/MAXOUTPUT;
  _b = cos(vW0)+vA1;  _miny = _maxy = 0.0;
  mainwin.SetViewport(MINX,MINY,WIDTH,HEIGHT,TRUE);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    setcolor(BLUE);
    _a = 1/sin(vW0);
    _a *= _a;
    _c = _a * _b;
    _d = _c * _b;
    aOutput[0] = 1/(_d+1);  aOutput[0] = 2*_c*SQR(aOutput[0]);
    gpx.Moveto(0,aOutput[0]);
    for(_k=1,vA0=_stp; _k<MAXOUTPUT; _k++,vA0+=_stp)
    {
      _a = (1+vA0)/((1-vA0)*sin(vW0));
      _a *= _a;
      _c = _a * _b;
      _d = _c * _b;
      aOutput[_k] = 1/(_d+1);  aOutput[_k] = 2*_c*SQR(aOutput[_k]);

      gpx.Lineto(vA0,aOutput[_k]);
      if (_miny > aOutput[_k]) _miny = aOutput[_k];
      if (_maxy < aOutput[_k]) _maxy = aOutput[_k];
    }

    mainwin.UseWindowScope(TRUE);
// if (_maxy > 0 && (_maxy > MAXGRAPHY || _maxy < MAXGRAPHY/2))
    if (_maxy > MAXGRAPHY)
    {
      MAXGRAPHY = _maxy*1.1;
      gpx.yzoomfactor = HEIGHT/MAXGRAPHY;   // คำนวณความสูงใหม่ //
      gpx.MoveOrigin(0.0,20/gpx.yzoomfactor);

      WaitVertRetrace();
      ShowGetParametersDialog();
      ProcessingGraph();
    }
}

//-------------------------------------------
void Save2File(void)
{
  WORD _k,_i;
  FTP _a;
  FILE *_fhandle,*_hhandle;

  if ((_hhandle = fopen(DATAFILENAME ".Hdr","wt")) != NULL)
  {
    // เขียนจำนวนสูงที่สุด และจำนวนภาพที่สวนหัว .Hdr ------
    fprintf(_hhandle,"%u %d\n", MAXOUTPUT, graphidx);
    fclose(_hhandle);

    // เขียนข้อมูลจริงๆไว้ที่ .Dat ------------------
    if ((_fhandle = fopen(DATAFILENAME ".Dat","wt")) != NULL)
    {
      for(_k=0,_a=0.0;_k<MAXOUTPUT;_k++,_a+=1.0/MAXOUTPUT)
      {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            fprintf(_fhandle,"\n%" FTP_PREFIX "g ",_a);
            for(_i=0;_i<graphidx;_i++)
              fprintf(_fhandle,"%" FTP_PREFIX "g ",
                ((FTP huge *)aaOutput[_i])[_k]);
          }
          fclose(_fhandle);
        }
      }
    }

/*-------------------------------------------*/
void A0Testing(void)
{
  currentobj = OBJ_IA1;

  graphidx = 0;
#if FAR_ALLOC
  aOutput = farcalloc(MAXOUTPUT, sizeof(FTP));
#else
  try {
    aOutput = new FTP [MAXOUTPUT];
  }
  catch (xalloc) {
    return;
  }
#endif
  aaOutput[graphidx++]=aOutput;

  InitAllObjects();
  ShowGetParametersDialog();

  while(ActivateInput())
    WaitInput();

  while(graphidx)
#if FAR_ALLOC
    farfree(aaOutput[--graphidx]);
#else
    delete[] aaOutput[--graphidx];
#endif

  mainwin.UseWindowScope(FALSE);
}


DspExp4.Cpp

/*-------------------------------------------*/
/*
**  LL      aA    RRRR  GGG   EEEE
**  LL     AA AA  RRRR G     EE
**  LL     AAAAAA RR   G GG  EEE
**  LLLLL  AA  AA R RR  GGG   EEEE  Model only!!
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
**
**  โปรแกรม plot Alpha 0 กับ Alpha 1 เมื่อใช้สมการ adaptive กับทั้งคู่
*/
/*-------------------------------------------*/
#pragma hdrfile "DspExp4.Sym"

#include "dsps.h"
#undef MAXOUTPUT

/*-=-=-=-=-=-=-=-=-=-=-=-=-= define  Macros =-=-=-=-=-=-=-=-=-=-=-=-=-=*/
#define DATAFILENAME "DspExp5"
#define MAXGRAPH  5

#define A0MAX     0.65
#define A0MIN     0

#define BTHICK 2
#define MINX   50
#define MINY   100
#define WIDTH  (MWIDTH-100)
#define HEIGHT ((MHEIGHT-140)>>1)
#define MAXGRAPHX (FTP)MAXOUTPUT
#define MAXGRAPHY 1.1

#define MINX2  MINX
#define MINY2  (MINY+HEIGHT+20)
#define WIDE2  WIDTH
#define HIGH2  HEIGHT
#define MAXGX2 MAXGRAPHX
#define MAXGY2 1.1

static enum {
    OBJ_TOP=1,
    OBJ_PLOTGRAPH=1,OBJ_SAVE,OBJ_EXIT,
    OBJ_IMU,OBJ_IMU0,OBJ_IW0,OBJ_IMAG,
    OBJ_NULL
    } currentobj;

/*=-=-=-=-=-=-=-=-=-=-=-=-= Global variables =-=-=-=-=-=-=-=-=-=-=-=-=-=*/
static int magnitude;
static WORD MAXOUTPUT;
static FTP vMU,vMU0,vW0;
static FTP aA0[3],aA1[3],aPhi1[3],aPhi2[3],aY[3],aX[3],aPHI[3];
static FTP *aaOutput[MAXGRAPH],huge *aOutput,huge *aOutput1,huge *aOutput2;

static int graphidx;

extern int textw,texth;
extern int KeyboardCmd;
extern MOUSE_PARAM MStat;
extern cWindowBasic mainwin;

static pTextInput iMU[3],iA1[3],iMU0[3],iW0[3],iMag[3];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น ทั้งนี้ห้ามมิให้ทำซ้ำ ดัดแปลง และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
static cTextInput iMAXOUTPUT("0",6);
static TextButton bExit,bPlotGraph,bSave;
static mBaseGraphic gpx,gpx1;

/*-------------------- Function Prototypes --------------------*/
/* First part ----------------------------------*/
void AdaptivePlot(void);

static void Function_Y(void);
static void Function_A0(void);
static void Function_A1(void);
static void Function_PHI(void);
static void Function_Phi1(void);
static void Function_Phi2(void);

static void ProcessingGraph(void);
static void Save2File(void);
static void InitAllObjects(void);
static void ShowGetParametersDialog(void);
static int  ActivateInput(void);

extern void WaitInput(void);
extern int TextInputFunction(void);
extern void ShiftArray(FTP *array);

#pragma hdrstop

/*-------------------- Functions body --------------------*/
/*----------------------------------*/
void Function_Y(void)
{
  aY[0] = (1-aA0[0])/2*(aX[0]-aX[2])+ aA1[0]*(1-aA0[0])*aY[1] -aA0[0]*aY[2];
}

/*----------------------------------*/
void Function_A0(void)
{
  aA0[0]=aA0[1]+vMU0*aY[1]*aPHI[1]*(aPHI[1]*aPhi1[1] + aPhi2[1]*aY[1]);

  /* เนื่องจาก Alpha 0 ถูกคำนวณโดยสมการของ Alpha 1 จึงต้อง */
  /* จำกัดค่า Alpha 0 ไม่ให้เกิน range ของมัน (0,1) ----*/
  if (aA0[0] > A0MAX) aA0[0] = A0MAX;
  else
  if (aA0[0] < A0MIN) aA0[0] = A0MIN;
}

/*----------------------------------*/
void Function_A1(void)
{
  aA1[0] = aA1[1] +vMU*aY[1]*aPHI[1];
}

/*----------------------------------*/
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void Function_PHI(void)
{
  aPHI[0] = aA1[0]*(1+aA0[0])*aPHI[1] -aA0[0]*aPHI[2] +(1+aA0[0])*aY[1];
}

/*-------------------------------------*/
void Function_Phi1(void)
{
  aPhi1[0]=(aX[2]-aX[0])/2 +aA1[0]*aY[1] -aY[2] +aA1[0]*(1+aA0[0])*aPhi1[1]
    -aA0[0]*aPhi1[2];
}

/*-------------------------------------*/
void Function_Phi2(void)
{
  aPhi2[0]=aA1[0]*(1+aA0[0])*aPhi2[1] +aA1[0]*aPHI[1] -aA0[0]*aPhi2[2]
    -aPHI[2] +(1+aA0[0])*aPhi1[1] -aY[1];
}

//-------------------------------------
void ProcessingGraph(void)
{
  int _k;
  FTP _sinpcosw,  /* เก็บค่า sin(w)+cos(w) ไว้ใช้ในการหา alpha 0 opt และ */
      _sinscosw;  /* เก็บค่า sin(w)-cos(w) ไว้ใช้ในการหา apha 0 opt ด้วย */
  FTP _cosw;      /* เก็บค่า cos(w) เพื่อเลือกใช้ aona 0 opt  */

  // clear array ที่ใช้สำหรับการคำนวณ ----------
  for(_k=0;_k<3;_k++)
  {
    aX[_k]=aY[_k]=aA0[_k]=aA1[_k]=aPHI[_k]=aPhi1[_k]=aPhi2[_k]=(FTP)0;
  }

  // กำหนดค่าเริ่มต้น ----------
  _cosw = cos(vW0);
  _sinpcosw = sin(vW0)+_cosw;  _sinscosw = sin(vW0)-_cosw;
  aX[0] = magnitude;
  aPhi1[0] = -magnitude/2;
  Function_Y();
// aOutput[0] = 0;     /* output ของ alpha 1 เริ่มต้น */
  aOutput1[0] = 0;                  /* output ของ alpha 0  */
  aOutput2[0] = _sinpcosw/_sinscosw;  /* output ของ alpha 0 opt  */

  setcolor(BLUE);
  for(_k=1; _k<MAXOUTPUT; _k++)
  {
    /* เลื่อนแต่ละ array เพื่อคำนวณค่าต่อมา ----------*/
    ShiftArray(aX);  ShiftArray(aY);  ShiftArray(aA1);  ShiftArray(aA0);
    ShiftArray(aPHI);  ShiftArray(aPhi1);  ShiftArray(aPhi2);

    /* คำนวณค่า parameters ต่อมา ----------*/
    aX[0] = magnitude*cos(_k*vW0);
    Function_A0();   /* เลือกคำนวณ Alpha 0 และ 1 ก่อน เพราะมันใช้ค่า param*/
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Function_A1();   /* ที่เป็นอดีตเท่านั้น                    */
    Function_PHI();  /* fn เหล่านี้ใช้ Alpha 0,1 ที่เป็นปัจจุบันเท่านั้น*/
    Function_Phi1();
    Function_Phi2();
    Function_Y(); /* คำนวณ output Y */

    /* คำนวณค่า Alpha 0 opt ----------------------*/
    if (aA1[0] <= _cosw)
      aOutput2[_k] = (_sinscosw +aA1[0])/(_sinccosw -aA1[0]);
    else
      aOutput2[_k] = (_sinpcosw -aA1[0])/(_sinscosw +aA1[0]);

    // เขียน alpha 1 ----------------------------
    mainwin.SetViewport(MINX,MINY,WIDTH,HEIGHT,TRUE);
//  gpx.Line((FTP)(_k-1),aOutput[_k-1],(FTP)_k,aOutput[_k]=aA1[0]);
    gpx.Line((FTP)(_k-1),aA1[1],(FTP)_k,aA1[0]);

    // เขียน Alpha 0 และ Alpha 0 optimum ----------
    mainwin.SetViewport(MINX2,MINY2,WIDE2,HIGH2,TRUE);
    setcolor(RED);
    gpx1.Line((FTP)(_k-1),aOutput2[_k-1],(FTP)_k,aOutput2[_k]);
    setcolor(BLUE);
    gpx1.Line((FTP)(_k-1),aOutput1[_k-1],(FTP)_k,aOutput1[_k]=aA0[0]);
    gpx1.Line((FTP)(_k-1),aA0[1],(FTP)_k,aA0[0]);

//  if (_miny > aOutput[_k]) _miny = aOutput[_k];
//  if (_maxy < aOutput[_k]) _maxy = aOutput[_k];
  }
  mainwin.UseWindowScope(TRUE);
}

//-------------------------------------------
void InitAllObjects(void)
{
  iMAXOUTPUT.AdjustAttr(LIGHTBLUE,WHITE);

  iMAXOUTPUT.AdjustPosition(textw+100, 5);  iMAXOUTPUT.AdjustThick(3);
  iMU.AdjustPosition(textw+100,25);  iMU.AdjustThick(3);
  iMU0.AdjustPosition(textw+100,45);  iMU0.AdjustThick(3);
  iW0.AdjustPosition(textw+100,65);  iW0.AdjustThick(3);
  iMag.AdjustPosition(textw+100,85);  iMag.AdjustThick(3);
  iMU.inputfn=iMU0.inputfn=iW0.inputfn=iMag.inputfn=TextInputFunction;

  bPlotGraph.AdjustPosition(MRIGHT-300, 5,80,texth+10);
  bPlotGraph.AssignText("Plot graph");
  bPlotGraph.AdjustJustify(CHJUSTIFY|CVJUSTIFY);
  bPlotGraph.ReleaseButton();

  bSave.AdjustPosition(MRIGHT-200, 5,80,texth+10);
  bSave.AssignText("Save");
  bSave.AdjustJustify(CHJUSTIFY|CVJUSTIFY);
  bSave.ReleaseButton();
```

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  bExit.AdjustPosition(MRIGHT-100, 5,80,texth+10);
  bExit.AssignText("Exit");
  bExit.AdjustJustify(CHJUSTIFY|CVJUSTIFY);
  bExit.ReleaseButton();

  gpx.AdjustGraphSize(0,0,WIDTH,HEIGHT);
  gpx1.AdjustGraphSize(0,0,WIDE2,HIGH2);
}

//-----------------------------------------
void ShowGetParametersDialog(void)
{
  EDGECOLORPACK _ep;

  _ep.Up = _ep.Left = DARKCOLOR;  _ep.Down = _ep.Right = HIGHCOLOR;

  // clear Main Window ----------------------
  mainwin.UseWindowScope(FALSE);
  WaitVertRetrace();  mainwin.Paint();
  mainwin.UseWindowScope(TRUE);

  // เขียน Legend ---------------------------
  setcolor(YELLOW);
  outtextxy(getmaxx()/2,MINY+HEIGHT+textheight("x"),"x"); /* พิมพ์ชื่อตัวแปรที่แกน X */
  outtextxy(MINX-textwidth("1:1"),MINY+HEIGHT/2, "1:1"); /* พิมพ์ชื่อตัวแปรที่แกน Y */

  setfillstyle(SOLID_FILL,WHITE);  bar(5,MINY2+5,MINX2-5,MINY2+HIGH2-5);
  setcolor(RED);
  outtextxy(MINX2-textwidth("10,opt"),MINY2+HIGH2/2-10, "10,opt");
  setcolor(BLUE);
  outtextxy(MINX2-textwidth("10"),MINY2+HIGH2/2+10, "10"); /* พิมพ์ชื่อตัวแปรที่แกน Y */

  mainwin.OutTextat(textw, 5,"MAXOUTPUT");
  mainwin.OutTextat(textw,25,"Mu");
  mainwin.OutTextat(textw,45,"Mu0");
  mainwin.OutTextat(textw,65,"W0");
  mainwin.OutTextat(textw,85,"Magnitude");

  iMAXOUTPUT.Paint();
  iMU.Paint();  iMU0.Paint();  iW0.Paint();  iMag.Paint();
  bPlotGraph.Paint();  bSave.Paint();  bExit.Paint();

  // วาดพื้นที่สำหรับวาดกราฟ ----------------
  LineStrip(MINX,MINY,WIDTH,HEIGHT,_ep,BLACK,WHITE,LIGHTGRAY,BTHICK);
  LineStrip(MINX2,MINY2,WIDE2,HIGH2,_ep,BLACK,WHITE,LIGHTGRAY,BTHICK);
}

/*-----------------------------------------*/
int ActivateInput(void)
{
  int _x,_y;

  if (KeyboardCmd == (int)CMD_MOUSE)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    {
      if (MStat.butt & MOUSELBIT)  /* ถ้า mouse กดปุ่มซ้าย */
      {
        _x = MStat.x;  _y = MStat.y;
      }
      else
      {
        _x=_y=-1;
      }
    }
    else
    {
ChkKeyboardCmd:;
      _x=_y=-1;
      switch(KeyboardCmd)
      {
      case 0x010F: /* Shift-TAB */
      case 0x09:  /* TAB */
        // ลบ cursor ปัจจุบัน ————————————————
        setcolor(BLUE);  /* ใช้สีพื้นของ mainwin */
        switch(currentobj)
        {
        case OBJ_PLOTGRAPH:
          rectangle(bPlotGraph.Left()-3,bPlotGraph.Top()-3,
                    bPlotGraph.Right()+3,bPlotGraph.Bottom()+3);
          break;
        case OBJ_SAVE:
          rectangle(bSave.Left()-3,bSave.Top()-3,
                    bSave.Right()+3,bSave.Bottom()+3);
          break;
        case OBJ_EXIT:
          rectangle(bExit.Left()-3,bExit.Top()-3,
                    bExit.Right()+3,bExit.Bottom()+3);
          break;
        }
NextField:;
        if (KeyboardCmd == 0x09)
        {
          currentobj++;  /* เลื่อน cursor ไปตำแหน่งถัดไป */
          if (currentobj == OBJ_NULL)  currentobj = OBJ_TOP;
        }
        else
        {
          if (currentobj==OBJ_TOP)  currentobj = OBJ_NULL;
          currentobj--;
        }

        setcolor(YELLOW);
        switch(currentobj)
        {
        case OBJ_PLOTGRAPH:
          rectangle(bPlotGraph.Left()-3,bPlotGraph.Top()-3,
                    bPlotGraph.Right()+3,bPlotGraph.Bottom()+3);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            break;
        case OBJ_SAVE:
            rectangle(bSave.Left()-3,bSave.Top()-3,
                    bSave.Right()+3,bSave.Bottom()+3);
            break;
        case OBJ_EXIT:
            rectangle(bExit.Left()-3,bExit.Top()-3,
                    bExit.Right()+3,bExit.Bottom()+3);
            break;
        case OBJ_IMU:
            _x = textw+100;   _y = 25;   break;
        case OBJ_IMU0:
            _x = textw+100;   _y = 45;   break;
        case OBJ_IW0:
            _x = textw+100;   _y = 65;   break;
        case OBJ_IMAG:
            _x = textw+100;   _y = 85;   break;
        }
        break;

    case 0x0D:  /* ENTER */
        switch(currentobj)
        {
        case OBJ_PLOTGRAPH:
            _x = bPlotGraph.Left();   _y = bPlotGraph.Top();
            break;

        case OBJ_SAVE:
            _x = bSave.Left();   _y = bSave.Top();
            break;

        case OBJ_EXIT:
            _x = bExit.Left();   _y = bExit.Top();
            break;
        default:
            KeyboardCmd = 0x09;  /* ทำหน้าที่เป็น TAB แทน */
            goto NextField;
        }
        break;
    }
}

if (iMU.Touch(_x,_y))
{
  MU.Getstring();
  sscanf((char *)iMU, "%" FTP_PREFIX "F", &vMU);

  goto ChkKeyboardCmd;
}
else
if (iMU0.Touch(_x,_y))
{
  MU0.Getstring();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    sscanf((char *)iMU0, "%" FTP_PREFIX "f", &vMU0);

    goto ChkKeyboardCmd;
  }
  else
  if (iW0.Touch(_x,_y))
  {
    iW0.Getstring();
    sscanf((char *)iW0, "%" FTP_PREFIX "f", &vW0);

    goto ChkKeyboardCmd;
  }
  else
  if (iMag.Touch(_x,_y))
  {
    iMag.Getstring();
    sscanf((char *)iMag, "%d", &magnitude);

    goto ChkKeyboardCmd;
  }
  else
  if (bPlotGraph.Touch(_x,_y))
  {
    bPlotGraph.PushButton();  bPlotGraph.Paint();
    ProcessingGraph();
    bPlotGraph.ReleaseButton();  bPlotGraph.Paint();
  }
  else
  if (bSave.Touch(_x,_y))
  {
    bSave.PushButton();  bSave.Paint();
    Save2File();
    bSave.ReleaseButton();  bSave.Paint();
  }
  else
  if (bExit.Touch(_x,_y))
  {
    bExit.PushButton();
    bExit.Paint();
    return FALSE;
  }
  return TRUE;
}

//-------------------------------------------------
void Save2File(void)
{
  WORD _k;
  int _i;
  FILE *_fhandle,*_hhandle;

  if ((_hhandle = fopen(DATAFILENAME ".Hdr","wt")) != NULL)
  {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    // เขียนจำนวนสูงที่สุด และจำนวนกราฟที่ส่วนหัว .Hdr ————
    fprintf(_hhandle,"%u %d\n", MAXOUTPUT, graphidx);
    fclose(_hhandle);

    // เขียนข้อมูลจริง ๆไว้ที่ .Dat ————————————
    if ((_fhandle = fopen(DATAFILENAME ".Dat","wt")) != NULL)
    {
      for(_k=0;_k<MAXOUTPUT;_k++)
      {
        fprintf(_fhandle,"\n%u ",_k);
        for(_i=0;_i<graphidx;_i++)
          fprintf(_fhandle,"%" FTP_PREFIX "g ",
            ((FTP huge *)aaOutput[_i])[_k]);
      }
      fclose(_fhandle);
    }
  }
}

/*————————————————————————*/
void AdaptivePlot(void)
{
  int _ef;

  KeyboardCmd = CMD_NOINPUT;
  currentobj = OBJ_IMU;
  graphidx = 0;

  InitAllObjects();
  ShowGetParametersDialog();

  // get MAXOUTPUT เป็นอันดับแรก ————————————
  do {
    _ef = FALSE;
    iMAXOUTPUT.Getstring();
    sscanf((char *)iMAXOUTPUT, "%u", &MAXOUTPUT);

    if (!MAXOUTPUT)  return;  /* ออกจากโปรแกรมเมื่อ MAXOUTPUT = 0 */

    // เราไม่ใช้ new operator เพราะว่ามันจองได้ไม่เกิน 64 KB ——
    aaOutput[graphidx] = (FTP *)farcalloc(MAXOUTPUT,sizeof(FTP));
    if (aaOutput[graphidx] == NULL)
    {
      _ef = TRUE;
    }
    else
    { /* จอง array 2 columns */
      aaOutput[++graphidx] = (FTP *)farcalloc(MAXOUTPUT,sizeof(FTP));
      if (aaOutput[graphidx] == NULL)
      {
        _ef = TRUE;
        farfree(aaOutput[--graphidx]);
      }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
//      else
//      { /* จอง array สำหรับ 3 columns */
//        aaOutput[++graphidx] = (FTP *)farcalloc(MAXOUTPUT,sizeof(FTP));
//        if (aaOutput[graphidx] == NULL)
//        {
//          _ef = TRUE;
//          farfree(aaOutput[--graphidx]);
//          farfree(aaOutput[--graphidx]);
//        }
//      }
    }
  } while(_ef);
// aOutput = aaOutput[graphidx-2];
  aOutput1 = aaOutput[graphidx-1];
  aOutput2 = aaOutput[graphidx];
  graphidx++;  /* เลื่อน index ไปเตรียมไว้สำหรับจองครั้งต่อไป */
  iMAXOUTPUT.AdjustAttr(DARKGRAY,WHITE);
  iMAXOUTPUT.Paint();

  // ทำการ scale ให้พอเหมาะและเปลี่ยนจุด Origin ---------*/
  gpx.xzoomfactor = WIDTH/MAXGRAPHX;   gpx1.xzoomfactor = WIDE2/MAXGX2;
  gpx.yzoomfactor = HEIGHT/MAXGRAPHY;  gpx1.yzoomfactor = HIGH2/MAXGY2;
  gpx.MoveOrigin(0,16/gpx.yzoomfactor);
  gpx1.MoveOrigin(0,16/gpx1.yzoomfactor);

  // เขียนหน้าจอสำหรับ alpha 1 ----------------------
  mainwin.SetViewport(MINX,MINY,WIDTH,HEIGHT,TRUE);
  setcolor(GREEN);
  gpx.DrawGridxy(MAXOUTPUT/20,0.1);  /* ความละเอียดของจอหารด้วยจำนวนกริดที่ต้องการ */

  setcolor(BLACK);
  gpx.DisplayAxis();

  setcolor(RED);
  gpx.DrawNumberxy(MAXOUTPUT/10,0.1,0,1);  /* เขียนตัวเลขที่กริด */
  mainwin.UseWindowScope(TRUE);

  // เขียนหน้าจอสำหรับ alpha 0 ----------------------
  mainwin.SetViewport(MINX2,MINY2,WIDE2,HIGH2,TRUE);
  setcolor(GREEN);
  gpx1.DrawGridxy(MAXOUTPUT/20,0.1);  /* ความละเอียดของจอหารด้วยจำนวนกริดที่ต้องการ */

  setcolor(BLACK);
  gpx1.DisplayAxis();

  setcolor(RED);
  gpx1.DrawNumberxy(MAXOUTPUT/10,0.1,0,1);  /* เขียนตัวเลขที่กริด */
  mainwin.UseWindowScope(TRUE);

  // run program loop ----------------------
  do {
    WaitInput();
  } while(ActivateInput());
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  while(graphidx)
    farfree(aaOutput[--graphidx]);

  mainwin.UseWindowScope(FALSE);
}
```

DspExp5.Cpp

```
/*-------------------------------------------------*/
/*
** LL     aA   RRRR  GGG   EEEE
** LL    AA AA RRRR G     EE
** LL    AAAAAA RR   G GG EEE
** LLLLL AA  AA R RR  GGG  EEEE Model only!!
**
** โปรแกรม plot alpha 1 เมื่อ fix alpha 0 ค่าที่
**
/*-------------------------------------------------*/
#pragma hdrfile "DspExp5.Sym"

#include "dsps.h"
#undef MAXOUTPUT

/*-------------------- define Macros ----------------------*/
#define DATAFILENAME "DspExp3"
#define MAXGRAPH 5

#define BTHICK 2
#define MINX  50
#define MINY  120
#define WIDTH  (MWIDTH-100)
#define HEIGHT (MHEIGHT-140)
#define MAXGRAPHX (FTP)MAXOUTPUT
#define MAXGRAPHY 1.1

static enum {
   OBJ_TOP=1,OBJ_PLOTGRAPH=1,OBJ_SAVE,OBJ_EXIT,
   OBJ_IA0,OBJ_IMU,OBJ_IW0,OBJ_IMAG,OBJ_NULL
   } currentobj;

/^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^ Global variables ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^/
static int magnitude;
static WORD MAXOUTPUT;
static FTP vA0,vMU,vW0;
static FTP aX[3],aY[3],aA1[3],aPHI[3];
static FTP *aaOutput[MAXGRAPH],huge *aOutput;

static int graphidx;

extern int textw,texth;
extern int KeyboardCmd;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
extern MOUSE_PARAM MStat;
extern cWindowBasic mainwin;

static cTextInput iA0("0",11),iMU("0",11),iW0("0",11),iMag("0",3);
static cTextInput iMAXOUTPUT("0",6);
static TextButton bExit,bPlotGraph,bSave;
static mBaseGraphic gpx;

/*-.-.-.-.-.-.-.-.-.-.-.-.-.- Function  Prototypes -.-.-.-.-.-.-.-.-.-.-.-.-.-*/
/* First part ----------------------------------*/
void MaximumA0(void);
static void Function_Y(void);
static void Function_A1(void);
static void Function_PHI(void);

static void ProcessingGraph(void);
static void Save2File(void);
static void InitAllObjects(void);
static void ShowGetParametersDialog(void);
static int  ActivateInput(void);

extern void WaitInput(void);
extern int TextInputFunction(void);
extern void ShiftArray(FTP *array);

#pragma hdrstop

/*^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^ Functions body ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^*/
/*--------------------------------------------*/
void Function_Y(void)
{
  aY[0] = (1-vA0)^2*(aX[0]-aX[2])+ aA1[0]*(1+vA0)*aY[1] -vA0*aY[2];
}

/*--------------------------------------------*/
void Function_A1(void)
{
  aA1[0] = aA1[1] +vMU*aY[1]*aPHI[1];
}

/*--------------------------------------------*/
void Function_PHI(void)
{
  aPHI[0] = aA1[0]*(1+vA0)*aPHI[1] -vA0*aPHI[2] +(1+vA0)*aY[1];
}

//--------------------------------------------
void ProcessingGraph(void)
{
  WORD _k;
  FTP _miny, _maxy;
  // clear array ที่ใช้สำหรับการคำนวณ ----------
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  for(_k=0;_k<3;_k++)
  {
    aX[_k]=aY[_k]=aA1[_k]=aPHI[_k]=(FTPX0;
  }

  // กำหนดค่าเริ่มต้น ————————————————
  _miny = _maxy = 0.0;

  aX[0] = magnitude;
  Function_Y();

  mainwin.SetViewport(MINX,MINY,WIDTH,HEIGHT,TRUE);
  gpx.Moveto(0,aOutput[0]=aA1[0]);

  setcolor(BLUE);
  for(_k=1; _k<MAXOUTPUT; _k++)
  {
    ShiftArray(aX); ShiftArray(aY); ShiftArray(aA1); ShiftArray(aPHI);
    aX[0] = magnitude*cos(_k*vW0);
    Function_A1();
    Function_PHI();
    Function_Y();

    gpx.Lineto(_k,aOutput[_k]=aA1[0]);
    if (_miny > aOutput[_k]) _miny = aOutput[_k];
    if (_maxy < aOutput[_k]) _maxy = aOutput[_k];
  }
  mainwin.UseWindowScope(TRUE);
}

//————————————————————————————
void InitAllObjects(void)
{
  int _k;

  // Clear ตัวแปรพวก array ————————————
  for(_k=0;_k<3;_k++)
  {
    aX[_k]=aY[_k]=aA1[_k]=aPHI[_k]=0.0;
  }

  iMAXOUTPUT.AdjustAttr(LIGHTBLUE,WHITE);

  iMAXOUTPUT.AdjustPosition(textw*100, 5);  iMAXOUTPUT.AdjustThick(3);
  iA0.AdjustPosition(textw*100,25);  iA0.AdjustThick(3);
  iMU.AdjustPosition(textw*100,45);  iMU.AdjustThick(3);
  iW0.AdjustPosition(textw*100,65);  iW0.AdjustThick(3);
  iMag.AdjustPosition(textw*100,85);  iMag.AdjustThick(3);
  iA0.inputfn=iMU.inputfn=iW0.inputfn=iMag.inputfn=TextInputFunction;

  bPlotGraph.AdjustPosition(MRIGHT-300, 5,80,texth*10);
  bPlotGraph.AssignText("Plot graph");
  bPlotGraph.AdjustJustify(CHJUSTIFY|CVJUSTIFY);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  bPlotGraph.ReleaseButton();

  bSave.AdjustPosition(MRIGHT-200, 5,80,texth+10);
  bSave.AssignText("Save");
  bSave.AdjustJustify(CHJUSTIFY|CVJUSTIFY);
  bSave.ReleaseButton();

  bExit.AdjustPosition(MRIGHT-100, 5,80,texth+10);
  bExit.AssignText("Exit");
  bExit.AdjustJustify(CHJUSTIFY|CVJUSTIFY);
  bExit.ReleaseButton();

  gpx.AdjustGraphSize(0,0,WIDTH,HEIGHT);
}

//-----------------------------------------
void ShowGetParametersDialog(void)
{
  EDGECOLORPACK _ep;

  _ep.Up = _ep.Left = DARKCOLOR;  _ep.Down = _ep.Right = HIGHCOLOR;

  // clear Main Window ------------------------
  mainwin.UseWindowScope(FALSE);
  WaitVertRetrace();  mainwin.Paint();
  mainwin.UseWindowScope(TRUE);

  // เขียน Legend ------------------------------
  setcolor(YELLOW);
  outtextxy(getmaxx()/2,MINY+HEIGHT+textheight("k"),"k"); /* พิมพ์ชื่อตัวแปรที่แกน X */
  outtextxy(MINX-textwidth("u1(k)"),getmaxy()/2, "u1(k)"); /* พิมพ์ชื่อตัวแปรที่แกน Y */

  mainwin.OutTextat(textw, 5,"MAXOUTPUT");
  mainwin.OutTextat(textw,25,"a0 [ 0, 1]");
  mainwin.OutTextat(textw,45,"Mu");
  mainwin.OutTextat(textw,65,"W0");
  mainwin.OutTextat(textw,85,"Magnitude");

  iMAXOUTPUT.Paint();
  iA0.Paint();  iMU.Paint();  iW0.Paint();  iMag.Paint();
  bPlotGraph.Paint();  bSave.Paint();  bExit.Paint();

  // วาดพื้นที่สำหรับวาดกราฟ ------------------
  LineStrip(MINX,MINY,WIDTH,HEIGHT,_ep,BLACK,WHITE,LIGHTGRAY,BTHICK);
}

/*-----------------------------------------*/
int ActivateInput(void)
{
  int _x,_y;

  if (KeyboardCmd == (int)CMD_MOUSE)
  {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    if (MStat.butt & MOUSELBIT)  /* ถ้า mouse กดปุ่มซ้าย */
    {
      _x = MStat.x;  _y = MStat.y;
    }
    else
    {
      _x=_y=-1;
    }
  }
  else
  {
ChkKeyboardCmd:;
    _x=_y=-1;
    switch(KeyboardCmd)
    {
    case 0x010F: /* Shift-TAB */
    case 0x09:  /* TAB */
      // ลบ cursor ปัจจุบัน ----------
      setcolor(BLUE);  /* ใช้สีพื้นของ mainwin */
      switch(currentobj)
      {
      case OBJ_PLOTGRAPH:
        rectangle(bPlotGraph.Left()-3,bPlotGraph.Top()-3,
                  bPlotGraph.Right()+3,bPlotGraph.Bottom()+3);
        break;
      case OBJ_SAVE:
        rectangle(bSave.Left()-3,bSave.Top()-3,
                  bSave.Right()+3,bSave.Bottom()+3);
        break;
      case OBJ_EXIT:
        rectangle(bExit.Left()-3,bExit.Top()-3,
                  bExit.Right()+3,bExit.Bottom()+3);
        break;
      }
NextField:;
      if (KeyboardCmd == 0x09)
      {
        currentobj++;  /* เลื่อน cursor ไปตำแหน่งถัดไป */
        if (currentobj == OBJ_NULL)  currentobj = OBJ_TOP;
      }
      else
      {
        if (currentobj==OBJ_TOP)  currentobj = OBJ_NULL;
        currentobj--;
      }

      setcolor(YELLOW);
      switch(currentobj)
      {
      case OBJ_PLOTGRAPH:
        rectangle(bPlotGraph.Left()-3,bPlotGraph.Top()-3,
                  bPlotGraph.Right()+3,bPlotGraph.Bottom()+3);
        break;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        case OBJ_SAVE:
            rectangle(bSave.Left()-3,bSave.Top()-3,
                    bSave.Right()+3,bSave.Bottom()+3);
            break;
        case OBJ_EXIT:
            rectangle(bExit.Left()-3,bExit.Top()-3,
                    bExit.Right()+3,bExit.Bottom()+3);
            break;
        case OBJ_IA0:
            _x = textw+100;  _y = 25;  break;
        case OBJ_IMU:
            _x = textw+100;  _y = 45;  break;
        case OBJ_IW0:
            _x = textw+100;  _y = 65;  break;
        case OBJ_IMAG:
            _x = textw+100;  _y = 85;  break;
        }
        break;

    case 0x0D:  /* ENTER */
        switch(currentobj)
        {
        case OBJ_PLOTGRAPH:
            _x = bPlotGraph.Left();  _y = bPlotGraph.Top();
            break;

        case OBJ_SAVE:
            _x = bSave.Left();  _y = bSave.Top();
            break;

        case OBJ_EXIT:
            _x = bExit.Left();  _y = bExit.Top();
            break;
        default:
            KeyboardCmd = 0x09;  /* ทำหน้าที่เป็น TAB แทน */
            goto NextField;
        }
        break;
    }
}

if (iA0.Touch(_x,_y))
{
  iA0.Getstring();
  sscanf((char *)iA0, "%" FTP_PREFIX "f", &vA0);

  goto ChkKeyboardCmd;
}
else
if (iMU.Touch(_x,_y))
{
  iMU.Getstring();
  sscanf((char *)iMU, "%" FTP_PREFIX "f", &vMU);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    goto ChkKeyboardCmd;
  }
  else
  if (iW0.Touch(_x,_y))
  {
    iW0.Getstring();
    sscanf((char *)iW0, "%" FTP_PREFIX "f", &vW0);

    goto ChkKeyboardCmd;
  }
  else
  if (iMag.Touch(_x,_y))
  {
    iMag.Getstring();
    sscanf((char *)iMag, "%d", &magnitude);

    goto ChkKeyboardCmd;
  }
  else
  if (bPlotGraph.Touch(_x,_y))
  {
    bPlotGraph.PushButton();  bPlotGraph.Paint();
    ProcessingGraph();
    bPlotGraph.ReleaseButton();  bPlotGraph.Paint();
  }
  else
  if (bSave.Touch(_x,_y))
  {
    bSave.PushButton();  bSave.Paint();
    Save2File();
    bSave.ReleaseButton();  bSave.Paint();
  }
  else
  if (bExit.Touch(_x,_y))
  {
    bExit.PushButton();
    bExit.Paint();
    return FALSE;
  }
  return TRUE;
}

//---------------------------------------------------------------------------
void Save2File(void)
{
  WORD _k,_i;
  FILE *_fhandle,*_hhandle;

  if ((_hhandle = fopen(DATAFILENAME ".Hdr","wt")) != NULL)
  {
    // เขียนจำนวนสูงที่สุด และจำนวนกราฟที่ส่วนหัว .Hdr ------
    fprintf(_hhandle,"%u %d\n", MAXOUTPUT, graphidx);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    fclose(_hhandle);

    // เขียนข้อมูลจริงๆไว้ที่ .Dat ————————————
    if ((_fhandle = fopen(DATAFILENAME ".Dat","wt")) != NULL)
    {
      for(_k=0;_k<MAXOUTPUT;_k++)
      {
        fprintf(_fhandle,"\n%d ",_k);
        for(_i=0;_i<graphidx;_i++)
          fprintf(_fhandle,"%" FTP_PREFIX "g ",
           ((FTP huge *)aaOutput[_i])[_k]);
      }
      fclose(_fhandle);
    }
  }
}

/*————————————————————————————*/
void MaximumA0(void)
{
  int _ef;

  KeyboardCmd = CMD_NOINPUT;
  currentobj = OBJ_IA0;
  graphidx = 0;

  InitAllObjects();
  ShowGetParametersDialog();

  // get MAXOUTPUT เป็นอันดับแรก ————————————
  do {
    _ef = FALSE;
    iMAXOUTPUT.GetString();
    sscanf((char *)iMAXOUTPUT, "%u", &MAXOUTPUT);

    if (!MAXOUTPUT)  return;  /* ออกจากโปรแกรมเมื่อ MAXOUTPUT = 0 */

    // เราไม่ใช้ new operator เพราะว่ามันจองได้ไม่เกิน 64 KB ——
    aaOutput[graphidx] = (FTP *)farcalloc(MAXOUTPUT,sizeof(FTP));
    _ef = (aaOutput[graphidx] == NULL);
  } while(_ef);
  aOutput = aaOutput[graphidx++];
  iMAXOUTPUT.AdjustAttr(DARKGRAY,WHITE);
  iMAXOUTPUT.Paint();

  // ทำการ scale ให้พอเหมาะและเปลี่ยนจุด Origin ————*/
  gpx.xzoomfactor = WIDTH/MAXGRAPHX;
  gpx.yzoomfactor = HEIGHT/MAXGRAPHY;
  gpx.MoveOrigin(0,30/gpx.yzoomfactor);

  mainwin.SetViewport(MINX,MINY,WIDTH,HEIGHT,TRUE);
  setcolor(GREEN);
  gpx.DrawGridxy(MAXOUTPUT/20,0.1);  /* ความละเอียดของจอภาพด้วยจำนวนกริดที่ต้องการ */
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  setcolor(BLACK);
  gpx.DisplayAxis();

  setcolor(RED);
  gpx.DrawNumberxy(MAXOUTPUT/10,0.1,0,1); /* เขียนตัวเลขที่กริด */
  mainwin.UseWindowScope(TRUE);

  // run program loop ————————————————
  do {
    WaitInput();
  } while(ActivateInput());

  while(graphidx)
    farfree(aaOutput[--graphidx]);

  mainwin.UseWindowScope(FALSE);
}
```

DspExp6.Cpp

```
/*————————————————————————*/
// เขียน Transfer Function G, G01, G02
//
#pragma hdrfile "DspExp6.Sym"
#include "dsps.h"

/*-=-=-=-=-=-=-=-=-=-=-=-=-=-= define Macros =-=-=-=-=-=-=-=-=-=-=-=-=-=-=*/
#define DATAFILENAME "DspExp7"

#define MAXGRAPH 20
#define MAXOUTPUT 1000

#define BTHICK 2
#define MINX  70
#define MINY  80
#define WIDTH  (MWIDTH-100)
#define HEIGHT (MHEIGHT-110)
#define MAXGRAPHX (FTP)PI

#define SBTHICK 1
#define SWIDTH  (textw*60+(SBTHICK<<1))
#define SHEIGHT (texth*3+(SBTHICK<<1))
#define SMINX   ((getmaxx()-SWIDTH)>>1)
#define SMINY   ((getmaxy()-SHEIGHT)>>1)

static enum {
    OBJ_TOP=1,
    OBJ_PLOTGRAPH=1,OBJ_SAVE,OBJ_EXIT,
    OBJ_IA0,OBJ_IA1,OBJ_PLOTMODE,
    OBJ_NULL
    } currentobj;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/*-=-=-=-=-=-=-=-=-=-=-=-=-=-=^ Global variables =-=-=-=-=-=-=-=-=-=-=-=-=-=^*/
static int plotmode;   /* กำหนดรูปแบบการ plot */
static FTP vA0,vA1;
static FTP *aaOutput[MAXGRAPH],*aOutput,*aG01Output,*aG02Output;
static int graphidx;
static FTP MAXGRAPHY;

static pTextInput iA0,iA1,iPlotMode;  /* input a0, a1 */
static TextButton bExit,bPlotGraph,bSave;
static mBaseGraphic gpx;

extern int textw,texth;
extern pWindowBasic mainwin;
extern int KeyboardCmd;
extern MOUSE_PARAM MStat;

/*-=-=-=-=-=-=-=-=-=-=-=-=-=- Function Prototypes -=-=-=-=-=-=-=-=-=-=-=-=-*/
/* First part ------------------------------------------*/
void TransferGvsz(void);
static inline FTP Function_G(FTP _z);
static inline FTP Function_G(FTP _z);
static inline FTP Function_G(FTP _z);
static void ProcessingGraph(void);
static void Save2File(void);
static void InitAllObjects(void);
static void ShowGetParametersDialog(void);
static int ActivateInput(void);
static inline FTP Pow2(FTP _f);

extern void WaitInput(void);
extern int TextInputFunction(void);
#pragma hdrstop

/*-=-=-=-=-=-=-=-=-=-=-=-=-=-= Functions body -=-=-=-=-=-=-=-=-=-=-=-=-=-*/
FTP Pow2(FTP _f)
{
  return (_f*_f);
}

/*------------------------------------------------*/
FTP Function_G(FTP _w)
{
  return 1/sqrt(
           Pow2(cos(_w)-vA1)
          +Pow2(sin(_w)*(1-vA0)/(1+vA0))
           );
}

//------------------------------------------------
FTP Function_G01(FTP _w)
{
  register FTP _1sA0;  /* 1 - Alpha0 */
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  register FTP _cos_w;
  _cos_w = cos(_w);
  _1sA0 = 1 - vA0;

  return 2*fabs((vA1-_cos_w)/_1sA0)
        sqrt(
          Pow2((1+vA0)*(_cos_w-vA1))+
          Pow2(_1sA0*sin(_w))
          );
}

//-------------------------------------
FTP Function_G02(FTP _w)
{
  return 2/fabs(1-vA0);
}

//-------------------------------------
void InitAllObjects(void)
{
  MAXGRAPHY = 1.0;  /* ค่า Y สูงสุดเริ่มต้น */

  iA0.AdjustString("0",11);
  iA1.AdjustString("0",11);
  iPlotMode.AdjustString("0",2);

  // ให้รับ input ตั้งแต่เริ่มเข้าโปรแกรม ----------
  KeyboardCmd = (int)CMD_MOUSE;
  MStat.x = textw*100;  MStat.y = 5;  /* ให้รับ alpha 0 */
  MStat.butt = MOUSELBIT;

  iA0.AdjustPosition(textw*100, 5);  iA0.AdjustThick(3);
  iA1.AdjustPosition(textw*100,25);  iA1.AdjustThick(3);
  iPlotMode.AdjustPosition(textw*100,45);  iPlotMode.AdjustThick(3);
  iA0.inputfn=iA1.inputfn=iPlotMode.inputfn=TextInputFunction;

  bPlotGraph.AdjustPosition(MRIGHT-300, 5,80,texth+10);
  bPlotGraph.AssignText("Plot graph");
  bPlotGraph.AdjustJustify(CHJUSTIFY|CVJUSTIFY);
  bPlotGraph.ReleaseButton();

  bSave.AdjustPosition(MRIGHT-200, 5,80,texth+10);
  bSave.AssignText("Save");
  bSave.AdjustJustify(CHJUSTIFY|CVJUSTIFY);
  bSave.ReleaseButton();

  bExit.AdjustPosition(MRIGHT-100, 5,80,texth+10);
  bExit.AssignText("Exit");
  bExit.AdjustJustify(CHJUSTIFY|CVJUSTIFY);
  bExit.ReleaseButton();

  // ทำการ scale ให้พอเหมาะและเปลี่ยนจุด Origin ----------//
  gpx.AdjustGraphSize(0,0,WIDTH,HEIGHT);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  gpx.xzoomfactor = WIDTH/MAXGRAPHX;
  gpx.yzoomfactor = HEIGHT/MAXGRAPHY;
  gpx.MoveOrigin(0.0,20/gpx.yzoomfactor);
}

//-----------------------------------------
void ShowGetParametersDialog(void)
{
  EDGECOLORPACK _ep;

  _ep.Up = _ep.Left = DARKCOLOR;  _ep.Down = _ep.Right = HIGHCOLOR;

  // clear main window ----------------------
  setfillstyle(SOLID_FILL,BLUE);
  bar(0,0,MWIDTH,MHEIGHT);

  // เขียน Legend ---------------------------
  setcolor(YELLOW);
  outtextxy(getmaxx()/2,MINY+HEIGHT+textheight("w"),"w"); /* พิมพ์ชื่อตัวแปรที่แกน X */

  mainwin.OutTextat(textw, 5,"w0 [ 0, 1]");
  mainwin.OutTextat(textw,25,"w1 [-1, 1]");
  mainwin.OutTextat(textw,45,"plot mode:");

  mainwin.OutTextat(textw+200, 5,"Plot Mode:");
  mainwin.OutTextat(textw+200,25,"0 - G  ",(WHITE<<4)|BLUE);
  mainwin.OutTextat(textw+200,45,"1 - G01",(WHITE<<4)|MAGENTA);
  mainwin.OutTextat(textw+200,65,"2 - G02",(WHITE<<4)|BROWN);

  iA0.Paint();  iA1.Paint();  iPlotMode.Paint();
  bPlotGraph.Paint();  bSave.Paint();  bExit.Paint();

  // วาดพื้นที่สำหรับวาดกราฟ ----------------
  LineStrip(MINX,MINY,WIDTH,HEIGHT,_ep,BLACK,WHITE,LIGHTGRAY,BTHICK);

  mainwin.SetViewport(MINX,MINY,WIDTH,HEIGHT,TRUE);
  setcolor(GREEN);
  gpx.DrawGridxy(PI/20,MAXGRAPHY/20);  /* ความละเอียดของจอภาพด้วยจำนวนกริดที่ต้องการ */

  setcolor(BLACK);
  gpx.DisplayAxis();

  setcolor(RED);
  gpx.DrawNumbersxy(PI/10,MAXGRAPHY/10,2,2);  /* เขียนตัวเลขที่กริด */

  mainwin.UseWindowScope(TRUE);
}

//-----------------------------------------*/
int ActivateInput(void)
{
  int _x,_y;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    if (KeyboardCmd == (int)CMD_MOUSE)
    {
      if (MStat.butt & MOUSELBIT)  /* ถ้า mouse กดปุ่มซ้าย */
      {
        _x = MStat.x;  _y = MStat.y;
      }
      else
      {
        _x=_y=-1;
      }
    }
    else
    {
ChkKeyboardCmd:;
      _x=_y=-1;
      switch(KeyboardCmd)
      {
      case 0x010F: /* Shift-TAB */
      case 0x09:  /* TAB */
        // ลบ cursor ปัจจุบัน ----------
        setcolor(BLUE);  /* ไร้สีพื้นของ mainwin */
        switch(currentobj)
        {
        case OBJ_PLOTGRAPH:
            rectangle(bPlotGraph.Left()-3,bPlotGraph.Top()-3,
                      bPlotGraph.Right()+3,bPlotGraph.Bottom()+3);
            break;
        case OBJ_SAVE:
            rectangle(bSave.Left()-3,bSave.Top()-3,
                      bSave.Right()+3,bSave.Bottom()+3);
            break;
        case OBJ_EXIT:
            rectangle(bExit.Left()-3,bExit.Top()-3,
                      bExit.Right()+3,bExit.Bottom()+3);
            break;
        }
NextField:;
        if (KeyboardCmd == 0x09)
        {
          currentobj++;  /* เลื่อน cursor ไปตำแหน่งถัดไป */
          if (currentobj == OBJ_NULL)  currentobj = OBJ_TOP;
        }
        else
        {
          if (currentobj==OBJ_TOP)  currentobj = OBJ_NULL;
          currentobj--;
        }

        setcolor(YELLOW);
        switch(currentobj)
        {
        case OBJ_PLOTGRAPH:
            rectangle(bPlotGraph.Left()-3,bPlotGraph.Top()-3,
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    bPlotGraph.Right()+3,bPlotGraph.Bottom()+3);
                break;
            case OBJ_SAVE:
                rectangle(bSave.Left()-3,bSave.Top()-3,
                    bSave.Right()+3,bSave.Bottom()+3);
                break;
            case OBJ_EXIT:
                rectangle(bExit.Left()-3,bExit.Top()-3,
                    bExit.Right()+3,bExit.Bottom()+3);
                break;
            case OBJ_IA0:
                _x = textw+100;  _y = 5;  break;
            case OBJ_IA1:
                _x = textw+100;  _y = 25;  break;
            case OBJ_PLOTMODE:
                _x = textw+100;  _y = 45;  break;
            }
            break;

        case 0x0D:  /* ENTER */
            switch(currentobj)
            {
            case OBJ_PLOTGRAPH:
                _x = bPlotGraph.Left();  _y = bPlotGraph.Top();
                break;

            case OBJ_SAVE:
                _x = bSave.Left();  _y = bSave.Top();
                break;

            case OBJ_EXIT:
                _x = bExit.Left();  _y = bExit.Top();
                break;
            default:
                KeyboardCmd = 0x09;  /* ทำหน้าที่เป็น TAB แทน */
                goto NextField;
            }
            break;
    }
}

if (iA0.Touch(_x,_y))
{
  iA0.Getstring();
  sscanf((char *)iA0, "%" FTP_PREFIX "f", &vA0);

  goto ChkKeyboardCmd;
}
else
if (iA1.Touch(_x,_y))
{
  iA1.Getstring();
  sscanf((char *)iA1, "%" FTP_PREFIX "f", &vA1);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    goto ChkKeyboardCmd;
  }
  else
  if (iPlotMode.Touch(_x,_y))
  {
    iPlotMode.Getstring();
    sscanf((char *)iPlotMode, "%d", &plotmode);

    goto ChkKeyboardCmd;
  }
  else
  if (bPlotGraph.Touch(_x,_y))
  {
    bPlotGraph.PushButton();  bPlotGraph.Paint();
    ProcessingGraph();
    bPlotGraph.ReleaseButton();  bPlotGraph.Paint();
  }
  else
  if (bSave.Touch(_x,_y))
  {
    bSave.PushButton();  bSave.Paint();
    Save2File();
    bSave.ReleaseButton();  bSave.Paint();
  }
  else
  if (bExit.Touch(_x,_y))
  {
    bExit.PushButton();
    bExit.Paint();
    return FALSE;
  }
  return TRUE;
}

//-------------------------------------------
void ProcessingGraph(void)
{
  WORD _k;
  FTP _w,_stpw;
  FTP _miny, _maxy;

  // กำหนดค่าเริ่มต้น ----------------------
  _stpw = PI/MAXOUTPUT;
  _miny = _maxy = 0.0;

  aOutput[0]=Function_G(0);  /* aOutput เก็บค่าของ G1 ไว้ */
  aG01Output[0]=Function_G01(0);
  aG02Output[0]=Function_G02(0);

  // เริ่มเขียนกราฟ ------------------------
  mainwin.SetViewport(MINX,MINY,WIDTH,HEIGHT,TRUE);
  setcolor(BLUE);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
for(_w=_stpw,_k=1; _k<MAXOUTPUT; _k++,_w+=_stpw)
{
  aOutput[_k]=Function_G(_w);  /* aOutput เก็บค่าของ G1 ไว้ */
  aG01Output[_k]=Function_G01(_w);
  aG02Output[_k]=Function_G02(_w);

  switch(plotmode)
  {
  case 0:  /* plot G1 */
        setcolor(BLUE);
        gpx.Line((FTP)_w-_stpw,aOutput[_k-1],(FTP)_w,aOutput[_k]);
        break;
  case 1:  /* plot G01 */
        setcolor(MAGENTA);
        gpx.Line((FTP)_w-_stpw,aG01Output[_k-1],(FTP)_w,aG01Output[_k]);
        break;
  case 2:  /* plot G02 */
        setcolor(BROWN);
        gpx.Line((FTP)_w-_stpw,aG02Output[_k-1],(FTP)_w,aG02Output[_k]);
        break;
  }

  // เก็บค่าสูงสุดต่ำสุดไว้ ————————————
  switch(plotmode)
  {
  case 0:  /* เลือก G1 */
        if (_miny > aOutput[_k]) _miny = aOutput[_k];
        if (_maxy < aOutput[_k]) _maxy = aOutput[_k];
        break;
  case 1:  /* เลือก G01 */
        if (_miny > aG01Output[_k]) _miny = aG01Output[_k];
        if (_maxy < aG01Output[_k]) _maxy = aG01Output[_k];
        break;
  case 2:  /* เลือก G02 */
        if (_miny > aG02Output[_k]) _miny = aG02Output[_k];
        if (_maxy < aG02Output[_k]) _maxy = aG02Output[_k];
        break;
  }
}
mainwin.UseWindowScope(TRUE);

// ปรับ scale หน้าจอและ update ถ้ากราฟถูกเขียนเกินหน้าจอ ——
if (_maxy > MAXGRAPHY)
{
  MAXGRAPHY = _maxy*1.1;
  gpx.yzoomfactor = HEIGHT/MAXGRAPHY;  /* คำนวณความสูงใหม่ */
  gpx.MoveOrigin(0,0,20/gpx.yzoomfactor);

  WaitVertRetrace();
  ShowGetParametersDialog();
  ProcessingGraph();
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    // เขียนตัวแปรกำกับให้สอดคล้องกับ plotmode ————————
    setfillstyle(SOLID_FILL,BLUE);
    bar(0,getmaxy()/2,MINX-1,getmaxy()/2+textheight("M")); /* ลบตัวอักษรเก่า */
    setcolor(YELLOW); /* กำหนดสีตัวอักษรปกติ */
    switch(plotmode)
    {
    case 0:  /* plot G1 */
            outtextxy(MINX-textwidth("|G(w)|"),getmaxy()/2, "|G(w)|");
            break;
    case 1:  /* plot G01 */
            outtextxy(MINX-textwidth("|G01(w)|"),getmaxy()/2, "|G01(w)|");
            break;
    case 2:  /* plot G02 */
            outtextxy(MINX-textwidth("|G02(w)|"),getmaxy()/2, "|G02(w)|");
            break;
    }
}

//————————————————————————————————
void Save2File(void)
{
  WORD _k,_i;
  FTP _w;
  FILE *_fhandle,*_hhandle;

  if ((_hhandle = fopen(DATAFILENAME ".Hdr","wt")) != NULL)
  {
    // เขียนจำนวนสูงที่สุด และจำนวนกราฟที่ส่วนหัว .Hdr ————————
    fprintf(_hhandle,"%u %d\n", MAXOUTPUT, graphidx);
    fclose(_hhandle);

    // เขียนข้อมูลจริงๆไว้ที่ .Dat ————————————————
    if ((_fhandle = fopen(DATAFILENAME ".Dat","wt")) != NULL)
    {
      for(_k=0,_w=0.0;_k<MAXOUTPUT;_k++,_w+=PI/MAXOUTPUT)
      {
        fprintf(_fhandle,"\n%" FTP_PREFIX "g ",_w);
        for(_i=0;_i<graphidx;_i++)
          fprintf(_fhandle,"%" FTP_PREFIX "g ",
            ((FTP huge *)aaOutput[_i])[_k]);
      }
      fclose(_fhandle);
    }
  }
}

/*————————————————————————————————*/
void TransferGvsz(void)
{
  currentobj = OBJ_IA0;

  graphidx = 0;
  try {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    aOutput = new FTP [MAXOUTPUT];
    aG01Output = new FTP [MAXOUTPUT];
    aG02Output = new FTP [MAXOUTPUT];
    aaOutput[graphidx++]=aOutput;
    aaOutput[graphidx++]=aG01Output;
    aaOutput[graphidx++]=aG02Output;
  }
  catch (xalloc) {
    return;
  }

  InitAllObjects();
  ShowGetParametersDialog();

  while(ActivateInput())
    WaitInput();

  while(graphidx)
    delete[] aaOutput[--graphidx];

  mainwin.UseWindowScope(FALSE);
}


DspExp7.Cpp
/*-------------------------------------*/
/*
**  LL      aA    RRRR   GGG   EEEE
**  LL     AA AA  RRRR  G     EE
**  LL     AAAAAA RR   G  GG  EEE
**  LLLLL  AA  AA R RR   GGG   EEEE  Model only!!
**
**  โปรแกรม plot alpha 1 และ input X ,output Y เมื่อ fix alpha 0 คงที่
**
/*-------------------------------------*/
#pragma hdrfile "DspExp7.Sym"

#include "dsps.h"
#undef MAXOUTPUT

// define NOISE macro เพื่อ gen input แบบ noise ----------
#define NOISE

/*-=-=-=-=-=-=-=-=-=-=-=-=-=-=-= define Macros -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=*/
#define DATAFILENAME "DspExp6"
#define MAXGRAPH  5
#define SINENUM 3  /* จำนวนคลื่น sine ที่ต้องการให้ปรากฏในหนึ่งจอภาพ */

#define BTHICK 2
#define MINX   50
#define MINY   (MINYX+HIGHX+20)
#define WIDTH  (MWIDTH-100)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#define HEIGHT ((MHEIGHT-140)>>1)
#define MAXGRAPHX (FTP)MAXOUTPUT
#define MAXGRAPHY 1.1

#define MINXX  50
#define MINYX  110
#define WIDEX  ((WIDTH>>1)-20)
#define HIGHX  HEIGHT
#define MAXGXX (FTP)max_x
#define MAXGYX 2.0  /*  ค่าความสูงสูงสุดของ input กำหนดให้เป็น 1  */

#define MINXY  (MINXX+WIDEX+50)
#define MINYY  MINYX
#define WIDEY  WIDEX
#define HIGHY  HIGHX
#define MAXGXY (FTP)max_x
#define MAXGYY 2.0  /*  ค่าความสูงสูงสุดของ output กำหนดให้เป็น 1  */

static enum {
    OBJ_TOP=1,
    OBJ_PLOTGRAPH=1 OBJ_SAVE,OBJ_EXIT,
    OBJ_IA0,OBJ_IMU,OBJ_IW0,OBJ_INOISEDB,OBJ_DELAY,
    OBJ_NULL
    } currentobj;

/*^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^ Global variables ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^*/
static WORD MAXOUTPUT;
static FTP vA0,vMU,vW0;
static FTP aX[3],aY[3],aA1[3],aPHI[3];
static FTP *aaOutput[MAXGRAPH],huge *aOutput;
static WORD random_seed;  /*  ใช้เก็บค่าเริ่มต้นในการสุ่ม เพียงไม่ต้องใช้ array  */
                          /*  ในการเก็บค่าของ noise  */
static FTP noisefactor;  /*  noisefactor, the component makes S/N ratio 10db  */
static FTP noise_db;  /*  ค่า decibel ของ noise ที่กำหนดให้  */

static int max_x;  /*  ความกว้างสูงสุดของ input และ output window  */
                   /*  จะเปลี่ยนแปลงไปตามความยาว (vW0)          */
static int delayindex;
static int graphidx;

extern int textw,texth;
extern int KeyboardCmd;
extern MOUSE_PARAM MStat;
extern pWindowBasic mainwin;

static pTextInput iA0("0",11),iMU("0",11),iW0("0",11),iNoiseDB("0",4);
static pTextInput iMAXOUTPUT("0",6),idelay("5",2);
static TextButton bExit,bPlotGraph,bSave;
static mBaseGraphic gpx,gpx_x,gpx_y;

/*-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=- Function  Prototypes -=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-*/
/* First part ---------------------------------------------------------------*/
void AdaptiveA1(void);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
static void Function_Y(void);
static void Function_A1(void);
static void Function_PHI(void);

static FTP FindNoiseFactor(void);
static void ProcessingGraph(void);
static void Save2File(void);
static void InitAllObjects(void);
static void ShowGetParametersDialog(void);
static int  ActivateInput(void);

extern void WaitInput(void);
extern int TextInputFunction(void);
extern void ShiftArray(FTP *array);

#pragma hdrstop

/*^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^ Functions body ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^*/
/*-----------------------------------------------------------*/
void Function_Y(void)
{
  aY[0] = (1-vA0)/2*(aX[0]-aX[2])+ aA1[0]*(1+vA0)*aY[1] -vA0*aY[2];
}

/*-----------------------------------------------------------*/
void Function_A1(void)
{
  aA1[0] = aA1[1] +vMU*aY[1]*aPHI[1];
}

/*-----------------------------------------------------------*/
void Function_PHI(void)
{
  aPHI[0] = aA1[0]*(1+vA0)*aPHI[1] -vA0*aPHI[2] +(1+vA0)*aY[1];
}

//-----------------------------------------------------------
//  สร้าง noise และกำหนด noise factor
//
FTP FindNoiseFactor(void)
{
  WORD _i;
  FTP _noise_avr,_actual_noise;

  // คำนวณค่าของ noise  จริงๆ เพื่อนำไปหา noise factor ที่จะนำไปคูณ
  // กับค่า noise ใน noise array แต่ละตัว ---------------
  //
  _actual_noise = cos(vW0)/pow((FTP)10.0,noise_db/10.0);

  // กำหนดค่า random seed เพื่อใช้ในการสุ่ม ---------------
  random_seed = random(32767)+random(32767)+1;

  // หาค่าเฉลี่ยของ noise array ---------------
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  _noise_avr = 0;
  srand(random_seed);
  for(_i=0;_i<MAXOUTPUT;_i++)
    /* ต้องคิด noise เฉพาะค่าบวกเท่านั้น จึงจะหา S/N ratio ได้ถูกต้อง */
    _noise_avr+=fabs((random(2000)/1000.0)-1.0);

  _noise_avr /= MAXOUTPUT;

  return _actual_noise/_noise_avr;  /* ได้อัตราส่วนของ noise factor ที่ต้องการ */
}

//-------------------------------------
void ProcessingGraph(void)
{
  WORD _k,_k1;
  FTP _miny, _maxy;

  // clear array ที่ใช้สำหรับการคำนวณ ----------
  for(_k=0;_k<3;_k++)
  {
    aX[_k]=aY[_k]=aA1[_k]=aPHI[_k]=(FTP)0;
  }

  // กำหนดค่าเริ่มต้น ----------
  _miny = _maxy = 0.0;
  noisefactor = FindNoiseFactor();
  srand(random_seed);

  aX[0] = ((random(2000)/1000.0)-1.0)*noisefactor;  /* เพิ่ม noise เข้าไปด้วย */
  Function_Y();

  aOutput[0]=0.0;
  _k = 1;
  do {
    _k1 = 0;

    // สำหรับ gpx ของ X ----------
    mainwin.SetViewport(MINXX,MINYX,WIDEX,HIGHX,TRUE);
    gpx_x.MoveOrigin(1.0-_k,MAXGYX/2);
    setfillstyle(SOLID_FILL,WHITE);
    bar(0,0,WIDEX,HIGHX);
    setcolor(BLACK);  gpx_x.DisplayAxis();

    // สำหรับ gpx ของ Y ----------
    mainwin.SetViewport(MINXY,MINYY,WIDEY,HIGHY,TRUE);
    gpx_y.MoveOrigin(1.0-_k,MAXGYY/2);
    setfillstyle(SOLID_FILL,WHITE);
    bar(0,0,WIDEY,HIGHY);
    setcolor(BLACK);  gpx_y.DisplayAxis();

    setcolor(BLUE);
    for(; _k<MAXOUTPUT && _k1<MAXGXX; _k++,_k1++)
    {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        ShiftArray(aX); ShiftArray(aY); ShiftArray(aA1); ShiftArray(aPHI);
#ifdef NOISE
        aX[0] = cos(_k*vW0)+((random(2000)/1000.0)-1.0)*noisefactor;
#else
        aX[0] = cos(_k*vW0);
#endif
        Function_A1();
        Function_PHI();
        Function_Y();

        // plot Alpha 1 ----------------------------
        mainwin.SetViewport(MINX,MINY,WIDTH,HEIGHT,TRUE);
        gpx.Line((FTP)_k-1,aOutput[_k-1],(FTP)_k,aOutput[_k]=aA1[0]);

        // plot Input use Output ----------------------
        mainwin.SetViewport(MINXX,MINYX,WIDEX,HIGHX,TRUE);
        setcolor(LIGHTGRAY);  gpx_x.Line((FTP)_k-1,aX[1],(FTP)_k,aX[0]);
        setcolor(BLUE);  gpx_x.Line((FTP)_k,0.0,(FTP)_k,aX[0]);

        mainwin.SetViewport(MINXY,MINYY,WIDEY,HIGHY,TRUE);
        setcolor(LIGHTGRAY);  gpx_y.Line((FTP)_k-1,aY[1],(FTP)_k,aY[0]);
        setcolor(BLUE);  gpx_y.Line((FTP)_k,0.0,(FTP)_k,aY[0]);
        delay(delayindex);

        if (_miny > aOutput[_k]) _miny = aOutput[_k];
        if (_maxy < aOutput[_k]) _maxy = aOutput[_k];
      }
    } while(_k != MAXOUTPUT && !kbhit());
    mainwin.UseWindowScope(TRUE);
}

//-----------------------------------------
void InitAllObjects(void)
{
  iMAXOUTPUT.AdjustAttr(LIGHTBLUE,WHITE);

  iMAXOUTPUT.AdjustPosition(textw+100, 5);  iMAXOUTPUT.AdjustThick(3);
  iA0.AdjustPosition(textw+100,25);  iA0.AdjustThick(3);
  iMU.AdjustPosition(textw+100,45);  iMU.AdjustThick(3);
  iW0.AdjustPosition(textw+100,65);  iW0.AdjustThick(3);
  iNoiseDB.AdjustPosition(textw+100,85);  iNoiseDB.AdjustThick(3);
  idelay.AdjustPosition(textw+300,85);  idelay.AdjustThick(3);
  iA0.inputfn=iMU.inputfn=iW0.inputfn=iNoiseDB.inputfn=idelay.inputfn=TextInputFunction;
  sscanf((char *)idelay, "%d", &delayindex);

  bPlotGraph.AdjustPosition(MRIGHT-300, 5,80,texth+10);
  bPlotGraph.AssignText("Plot graph");
  bPlotGraph.AdjustJustify(CHJUSTIFY|CVJUSTIFY);
  bPlotGraph.ReleaseButton();

  bSave.AdjustPosition(MRIGHT-200, 5,80,texth+10);
  bSave.AssignText("Save");
  bSave.AdjustJustify(CHJUSTIFY|CVJUSTIFY);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  bSave.ReleaseButton();

  bExit.AdjustPosition(MRIGHT-100, 5,80,texth+10);
  bExit.AssignText("Exit");
  bExit.AdjustJustify(CHJUSTIFY|CVJUSTIFY);
  bExit.ReleaseButton();

  gpx.AdjustGraphSize(0,0,WIDTH,HEIGHT);
  gpx_x.AdjustGraphSize(0,0,WIDEX,HIGHX);
  gpx_y.AdjustGraphSize(0,0,WIDEY,HIGHY);

  max_x = 1;  /* ใส่ไว้ชั่วคราว จะต้องถูกเปลี่ยนแปลงโดยการป้อนมุม */
  gpx_x.xzoomfactor = WIDEX/MAXGXX;  gpx_y.xzoomfactor = WIDEY/MAXGXY;
  gpx_x.yzoomfactor = HIGHX/MAXGYX;  gpx_y.yzoomfactor = HIGHY/MAXGYY;
}

//-------------------------------------
void ShowGetParametersDialog(void)
{
  EDGECOLORPACK _ep;

  _ep.Up = _ep.Left = DARKCOLOR;  _ep.Down = _ep.Right = HIGHCOLOR;

  // clear Main Window ----------------
  mainwin.UseWindowScope(FALSE);
  WaitVertRetrace();  mainwin.Paint();
  mainwin.UseWindowScope(TRUE);

  // เขียน Legend ----------------
  setcolor(YELLOW);
  outtextxy(MINXX+WIDEX/2,MINYX+HIGHX+textheight("k"),"k");
  outtextxy(MINXY+WIDEY/2,MINYY+HIGHY+textheight("k"),"k");
  outtextxy(MINX+WIDTH/2,MINY-textheight("k"),"k");
  outtextxy(MINX-textwidth("u1(k)"),MINY+HEIGHT/2, "u1(k)"); /* พิมพ์ชื่อตัวแปรที่แกน Y */
  outtextxy(MINXX-textwidth("Input"),MINYX+HIGHX/2, "Input");
  outtextxy(MINXY-textwidth("Output"),MINYY+HIGHY/2, "Output");

  mainwin.OutTextat(textw, 5,"MAXOUTPUT");
  mainwin.OutTextat(textw,25,"u0 [ 0, 1]");
  mainwin.OutTextat(textw,45,"Mu");
  mainwin.OutTextat(textw,65,"W0");
  mainwin.OutTextat(textw,85,"S/N (db)");
  mainwin.OutTextat(textw+200,85,"Delay");

  iMAXOUTPUT.Paint();
  iA0.Paint();  iMU.Paint();  iW0.Paint();  iNoiseDB.Paint();
  idelay.Paint();
  bPlotGraph.Paint();  bSave.Paint();  bExit.Paint();

  // วาดพื้นที่สำหรับวาดกราฟให้ gpx ทั้งหลาย ----------
  LineStrip(MINX,MINY,WIDTH,HEIGHT,_ep,BLACK,WHITE,LIGHTGRAY,BTHICK);
  LineStrip(MINXX,MINYX,WIDEX,HIGHX,_ep,BLACK,WHITE,LIGHTGRAY,BTHICK);
  LineStrip(MINXY,MINYY,WIDEY,HIGHY,_ep,BLACK,WHITE,LIGHTGRAY,BTHICK);
```

เอกสารนี้มีลิขสิทธิ์ ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}

/*-------------------------------------*/
int ActivateInput(void)
{
  int _x,_y;

  if (KeyboardCmd == (int)CMD_MOUSE)
  {
    if (MStat.butt & MOUSELBIT)  /* ถ้า mouse กดปุ่มซ้าย */
    {
      _x = MStat.x;  _y = MStat.y;
    }
    else
    {
      _x=_y=-1;
    }
  }
  else
  {
ChkKeyboardCmd:
    _x=_y=-1;
    switch(KeyboardCmd)
    {
    case 0x010F: /* Shift-TAB */
    case 0x09:  /* TAB */
      // ลบ cursor ปัจจุบัน ---------------------
      setcolor(BLUE);  /* ใช้สีพื้นของ mainwin */
      switch(currentobj)
      {
      case OBJ_PLOTGRAPH:
        rectangle(bPlotGraph.Left()-3,bPlotGraph.Top()-3,
              bPlotGraph.Right()+3,bPlotGraph.Bottom()+3);
        break;
      case OBJ_SAVE:
        rectangle(bSave.Left()-3,bSave.Top()-3,
              bSave.Right()+3,bSave.Bottom()+3);
        break;
      case OBJ_EXIT:
        rectangle(bExit.Left()-3,bExit.Top()-3,
              bExit.Right()+3,bExit.Bottom()+3);
        break;
      }
NextField:;
      if (KeyboardCmd == 0x09)
      {
        currentobj++;  /* เลื่อน cursor ไปตำแหน่งถัดไป */
        if (currentobj == OBJ_NULL)  currentobj = OBJ_TOP;
      }
      else
      {
        if (currentobj==OBJ_TOP)  currentobj = OBJ_NULL;
        currentobj--;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  {
    iA0.Getstring();
    sscanf((char *)iA0, "%" FTP_PREFIX "f", &vA0);

    goto ChkKeyboardCmd;
  }
  else
  if (iMU.Touch(_x,_y))
  {
    iMU.Getstring();
    sscanf((char *)iMU, "%" FTP_PREFIX "f", &vMU);

    goto ChkKeyboardCmd;
  }
  else
  if (iW0.Touch(_x,_y))
  {
    iW0.Getstring();
    sscanf((char *)iW0, "%" FTP_PREFIX "f", &vW0);
    max_x = SINENUM*2*PI/vW0;
    gpx_x.xzoomfactor = WIDEX/MAXGXX;  gpx_y.xzoomfactor = WIDEY/MAXGXY;

    goto ChkKeyboardCmd;
  }
  else
  if (iNoiseDB.Touch(_x,_y))
  {
    iNoiseDB.Getstring();
    sscanf((char *)iNoiseDB, "%" FTP_PREFIX "f", &noise_db);

    goto ChkKeyboardCmd;
  }
  else
  if (idelay.Touch(_x,_y))
  {
    idelay.Getstring();
    sscanf((char *)idelay, "%d", &delayindex);

    goto ChkKeyboardCmd;
  }
  else
  if (bPlotGraph.Touch(_x,_y))
  {
    bPlotGraph.PushButton();  bPlotGraph.Paint();
    ProcessingGraph();
    bPlotGraph.ReleaseButton();  bPlotGraph.Paint();
  }
  else
  if (bSave.Touch(_x,_y))
  {
    bSave.PushButton();  bSave.Paint();
    Save2File();
    bSave.ReleaseButton();  bSave.Paint();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        }

        setcolor(YELLOW);
        switch(currentobj)
        {
        case OBJ_PLOTGRAPH:
            rectangle(bPlotGraph.Left()-3,bPlotGraph.Top()-3,
                    bPlotGraph.Right()+3,bPlotGraph.Bottom()+3);
            break;
        case OBJ_SAVE:
            rectangle(bSave.Left()-3,bSave.Top()-3,
                    bSave.Right()+3,bSave.Bottom()+3);
            break;
        case OBJ_EXIT:
            rectangle(bExit.Left()-3,bExit.Top()-3,
                    bExit.Right()+3,bExit.Bottom()+3);
            break;
        case OBJ_IA0:
            _x = textw+100;  _y = 25;  break;
        case OBJ_IMU:
            _x = textw+100;  _y = 45;  break;
        case OBJ_IW0:
            _x = textw+100;  _y = 65;  break;
        case OBJ_INOISEDB:
            _x = textw+100;  _y = 85;  break;
        case OBJ_DELAY:
            _x = textw+300;  _y = 85;  break;
        }
        break;

    case 0x0D:  /* ENTER */
        switch(currentobj)
        {
        case OBJ_PLOTGRAPH:
            _x = bPlotGraph.Left();  _y = bPlotGraph.Top();
            break;

        case OBJ_SAVE:
            _x = bSave.Left();  _y = bSave.Top();
            break;

        case OBJ_EXIT:
            _x = bExit.Left();  _y = bExit.Top();
            break;
        default:
            KeyboardCmd = 0x09;  /* ทำหน้าที่เป็น TAB แทน */
            goto NextField;
        }
        break;
    }
}

if (iA0.Touch(_x,_y))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    }
    else
    if (bExit.Touch(_x,_y))
    {
      bExit.PushButton();
      bExit.Paint();
      return FALSE;
    }
    return TRUE;
}

//-------------------------------------
void Save2File(void)
{
  WORD _k,_i;
  FILE *_fhandle,*_hhandle;

  if ((_hhandle = fopen(DATAFILENAME ".Hdr","wt")) != NULL)
  {
    // เขียนจำนวนสูงที่สุด และจำนวนกราฟที่ส่วนหัว .Hdr ---------
    fprintf(_hhandle,"%u %d\n", MAXOUTPUT, graphidx);
    fclose(_hhandle);

    // เขียนข้อมูลจริงๆไว้ที่ .Dat ---------------------
    if ((_fhandle = fopen(DATAFILENAME ".Dat","wt")) != NULL)
    {
      for(_k=0;_k<MAXOUTPUT;_k++)
      {
        fprintf(_fhandle,"\n%d ",_k);
        for(_i=0;_i<graphidx;_i++)
          fprintf(_fhandle,"%" FTP_PREFIX "g ",
            ((FTP huge *)aaOutput[_i])[_k]);
      }
      fclose(_fhandle);
    }
  }
}

/*-------------------------------------*/
void AdaptiveA1(void)
{
  int _ef;

  KeyboardCmd = CMD_NOINPUT;
  currentobj = OBJ_IA0;
  graphidx = 0;

  randomize();
  InitAllObjects();
  ShowGetParametersDialog();

  // get MAXOUTPUT เป็นอันดับแรก ---------------
  do {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    _ef = FALSE;
    iMAXOUTPUT.Getstring();
    sscanf((char *)iMAXOUTPUT, "%u", &MAXOUTPUT);

    if (!MAXOUTPUT)  return;  /* ออกจากโปรแกรมเมื่อ MAXOUTPUT = 0 */

    // เราไม่ใช้ new operator เพราะว่ามันจองได้ไม่เกิน 64 KB ----
    aaOutput[graphidx] = (FTP *)farcalloc(MAXOUTPUT,sizeof(FTP));
    _ef = (aaOutput[graphidx] == NULL);
  } while(_ef);
  aOutput = aaOutput[graphidx++];
  iMAXOUTPUT.AdjustAttr(DARKGRAY,WHITE);
  iMAXOUTPUT.Paint();

  // ทำการ scale ให้พอเหมาะและเปลี่ยนจุด Origin ----------*/
  gpx.xzoomfactor = WIDTH/MAXGRAPHX;
  gpx.yzoomfactor = HEIGHT/MAXGRAPHY;
  gpx.MoveOrigin(0,20/gpx.yzoomfactor);

  // ตารางสำหรับ gpx ----------
  mainwin.SetViewport(MINX,MINY,WIDTH,HEIGHT,TRUE);
  setcolor(GREEN);
  gpx.DrawGridxy(MAXOUTPUT/20,0.1);  /* ความละเอียดของรูปหารด้วยจำนวนกริดที่ต้องการ */

  setcolor(BLACK);
  gpx.DisplayAxis();

  setcolor(RED);
  gpx.DrawNumberxy(MAXOUTPUT/10,0.1,0,1);  /* เขียนตัวเลขที่กริด */

  mainwin.UseWindowScope(TRUE);

  // run program loop ----------
  do {
    WaitInput();
  } while(ActivateInput());

  while(graphidx)
    farfree(aaOutput[--graphidx]);

  mainwin.UseWindowScope(FALSE);
}


DspExp8.Cpp

/*----------*/
/*
**  LL      aA    RRRR   GGG   EEEE
**  LL     AA AA  RRRR G     EE
**  LL     AAAAAA RR   G GG  EEE
**  LLLLL  AA  AA R RR   GGG  EEEE  Model only!!
**
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
** โปรแกรม plot alpha 1 และ input X ,output Y เมื่อ alpha 0 เปลี่ยนแปลง
**
/*-------------------------------------*/
#pragma hdrfile "DspExp8.Sym"

#include "dsps.h"
#undef MAXOUTPUT

// define NOISE macro เพื่อ gen input แบบ noise ---------
#define NOISE

/*-=-=-=-=-=-=-=-=-=-=-=-=-= define Macros =-=-=-=-=-=-=-=-=-=-=-=-=*/
#define DATAFILENAME "DspExp9"
#define MAXGRAPH 5
#define SINENUM 3  /* จำนวนคลื่น sine ที่ต้องการให้ปรากฏในหนึ่งจอภาพ */

#define BTHICK 2
#define MINX  50
#define MINY  (MINYX+HIGHX+20)
#define WIDTH  (MWIDTH-100)
#define HEIGHT ((MHEIGHT-140)>>1)
#define MAXGRAPHX (FTP)MAXOUTPUT
#define MAXGRAPHY 1.1

#define MINXX  50
#define MINYX  110
#define WIDEX  ((WIDTH>>1)-20)
#define HIGHX  HEIGHT
#define MAXGXX (FTP)max_x
#define MAXGYX 2.0

#define MINXY  (MINXX+WIDEX+50)
#define MINYY  MINYX
#define WIDEY  WIDEX
#define HIGHY  HIGHX
#define MAXGXY (FTP)max_x
#define MAXGYY 2.0

static enum {
   OBJ_TOP=1,
   OBJ_PLOTGRAPH=1,OBJ_SAVE,OBJ_EXIT,
   OBJ_IMU,OBJ_IMU0,OBJ_IW0,OBJ_INOISEDB,OBJ_IA0MIN,OBJ_IA0MAX,OBJ_DELAY,
   OBJ_NULL
   } currentobj;

/*-=-=-=-=-=-=-=-=-=-=-=-=-= Global vanables =-=-=-=-=-=-=-=-=-=-=-=-=*/
static WORD MAXOUTPUT;
static FTP vMU,vMU0,vW0;
static FTP aA0[3],aA1[3],aPhi1[3],aPhi2[3],aY[3],aX[3],aPHI[3];
static FTP *aaOutput[MAXGRAPH],huge *aOutput;
static FTP A0MAX,A0MIN;  /* alpha 0 maximum, minimum */
static WORD random_seed;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
static FTP noisefactor,noise_db;

static int max_x;  /* ความกว้างสูงสุดของ input และ output window */
                   /* จะเปลี่ยนแปลงไปตามค่ามุม (vW0)            */
static int delayindex;
static int graphidx;

extern int textw,texth;
extern int KeyboardCmd;
extern MOUSE_PARAM MStat;
extern pWindowBasic mainwin;

static pTextInput iMU("0",11),iMU0("0",11),iW0("0",11),iNoiseDB("0",4);
static pTextInput iA0MIN("0.1",5),iA0MAX("0.85",5);
static pTextInput iMAXOUTPUT("0",6),idelay("5",2);
static TextButton bExit,bPlotGraph,bSave;
static mBaseGraphic gpx,gpx_x,gpx_y;

/*-------------------- Function  Prototypes --------------------*/
/* First part ----------------------------------------*/
void AdeptiveA1nA0(void);

static void Function_Y(void);
static void Function_A0(void);
static void Function_A1(void);
static void Function_PHI(void);
static void Function_Phi1(void);
static void Function_Phi2(void);

static FTP FindNoiseFactor(void);
static void ProcessingGraph(void);
static void Save2File(void);
static void InitAllObjects(void);
static void ShowGetParametersDialog(void);
static int  ActivateInput(void);

extern void WaitInput(void);
extern int TextInputFunction(void);
extern void ShiftArray(FTP *array);

#pragma hdrstop

/*-------------------- Functions body --------------------*/
/*----------------------------------------*/
void Function_Y(void)
{
  aY[0] = (1-aA0[0])/2*(aX[0]-aX[2])+ aA1[0]*(1+aA0[0])*aY[1] -aA0[0]*aY[2];
}

/*----------------------------------------*/
void Function_A0(void)
{
  aA0[0]=aA0[1]+vMU0*aY[1]*aPHI[1]*(aPHI[1]*aPhi1[1] + aPhi2[1]*aY[1]);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    /* เนื่องจาก Alpha 0 ถูกคำนวณโดยสมการของ Alpha 1 จึงต้อง -*/
    /* จำกัดค่า Alpha 0 ไม่ให้เกิน range ของมัน (0,1) ------*/
    if (aA0[0] > A0MAX) aA0[0] = A0MAX;
    else
    if (aA0[0] < A0MIN) aA0[0] = A0MIN;
}

/*-------------------------------------*/
void Function_A1(void)
{
    aA1[0] = aA1[1] +vMU*aY[1]*aPHI[1];
}

/*-------------------------------------*/
void Function_PHI(void)
{
    aPHI[0] = aA1[0]*(1+aA0[0])*aPHI[1] -aA0[0]*aPHI[2] +(1+aA0[0])*aY[1];
}

/*-------------------------------------*/
void Function_Phi1(void)
{
    aPhi1[0]=(aX[2]-aX[0])/2 +aA1[0]*aY[1] -aY[2] -aA1[0]*(1+aA0[0])*aPhi1[1]
      -aA0[0]*aPhi1[2];
}

/*-------------------------------------*/
void Function_Phi2(void)
{
    aPhi2[0]=aA1[0]*(1+aA0[0])*aPhi2[1] +aA1[0]*aPHI[1] -aA0[0]*aPhi2[2]
      -aPHI[2] +(1+aA0[0])*aPhi1[1] +aY[1];
}

//-------------------------------------
//  สร้าง noise และกำหนด noise factor
//
FTP FindNoiseFactor(void)
{
    WORD _i;
    FTP _noise_avr,_actual_noise;

    // คำนวณค่าของ noise  จริงๆ เพื่อนำไปหา noise factor ที่จะนำไปคูณ
    // กับค่า noise ใน noise array แต่ละตัว ----------
    //
    _actual_noise = cos(vW0)/pow((FTP)10.0,noise_db/10.0);

    // กำหนดค่า random seed เพื่อใช้ในการสุ่ม ----------
    random_seed = random(32767)+random(32767)+1;

    // หาค่าเฉลี่ยของ noise array ----------
    _noise_avr = 0;
    srand(random_seed);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  for(_i=0;_i<MAXOUTPUT;_i++)
    /* ต้องคิด noise เฉพาะค่าบวกเท่านั้น จึงจะหา S/N ratio ได้ถูกต้อง */
    _noise_avr+=fabs((random(2000)/1000.0)-1.0);

  _noise_avr /= MAXOUTPUT;

  return _actual_noise/_noise_avr;  /* ได้อัตราส่วนของ noise factor ที่ต้องการ */
}

//-----------------------------------------
void ProcessingGraph(void)
{
  WORD _k,_k1;
  FTP _miny, _maxy;

  // clear array ที่ใช้สำหรับการคำนวณ ----------
  for(_k=0;_k<3;_k++)
  {
    aX[_k]=aY[_k]=aA0[_k]=aA1[_k]=aPHI[_k]=aPhi1[_k]=aPhi2[_k]=(FTP)0;
  }

  // กำหนดค่าเริ่มต้น ----------
  _miny = _maxy = 0.0;
  noisefactor = FindNoiseFactor();
  srand(random_seed);

#ifdef NOISE
  aX[0] = 1.0+((random(2000)/1000.0)-1.0)*noisefactor;  /* เพิ่ม noise เข้าไปด้วย */
#else
  aX[0] = 1.0;
#endif
  Function_Phi1();
  Function_Y();

  aOutput[0]=0.0;
  _k = 1;
  do {
    _k1 = 0;

    // สำหรับ gpx ของ X ----------
    mainwin.SetViewport(MINXX,MINYX,WIDEX,HIGHX,TRUE);
    gpx_x.MoveOrigin(1.0-_k,MAXGYX/2);
    setfillstyle(SOLID_FILL,WHITE);
    bar(0,0,WIDEX,HIGHX);
    setcolor(BLACK);  gpx_x.DisplayAxis();

    // สำหรับ gpx ของ Y ----------
    mainwin.SetViewport(MINXY,MINYY,WIDEY,HIGHY,TRUE);
    gpx_y.MoveOrigin(1.0-_k,MAXGYY/2);
    setfillstyle(SOLID_FILL,WHITE);
    bar(0,0,WIDEY,HIGHY);
    setcolor(BLACK);  gpx_y.DisplayAxis();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        setcolor(BLUE);
        for(; _k<MAXOUTPUT && _k1<MAXGXX; _k++,_k1++)
        {
          /* เลื่อนแต่ละ array เพื่อคำนวณค่าต่อมา ----------*/
          ShiftArray(aX); ShiftArray(aY); ShiftArray(aA1); ShiftArray(aA0);
          ShiftArray(aPHI); ShiftArray(aPhi1); ShiftArray(aPhi2);
#ifdef NOISE
          aX[0] = cos(_k*vW0)+((random(2000)/1000.0)-1.0)*noisefactor;
#else
          aX[0] = cos(_k*vW0);
#endif
          Function_A0();  /* เลือกคำนวณ Alpha 0 และ 1 ก่อน เพราะมันใช้ค่า param*/
          Function_A1();  /* ที่เป็นอดีตเท่านั้น                          */
          Function_PHI(); /* fn เหล่านี้ใช้ Aplha 0,1 ที่เป็นปัจจุบันเท่านั้น*/
          Function_Phi1();
          Function_Phi2();
          Function_Y(); /* คำนวณ output Y */

          // plot Alpha 1 ----------
          mainwin.SetViewport(MINX,MINY,WIDTH,HEIGHT,TRUE);
          gpx.Line((FTP)_k-1,aOutput[_k-1],(FTP)_k,aOutput[_k]=aA1[0]);

          // plot Alpha 0 หน้าต่างเดียวกับ alpha 1 ----------
//        setcolor(MAGENTA);  gpx.Line((FTP)_k-1,aA0[1],(FTP)_k,aA0[0]);

          // plot Input และ Output ----------
          mainwin.SetViewport(MINXX,MINYX,WIDEX,HIGHX,TRUE);
          setcolor(LIGHTGRAY);  gpx_x.Line((FTP)_k-1,aX[1],(FTP)_k,aX[0]);
          setcolor(BLUE);  gpx_x.Line((FTP)_k,0.0,(FTP)_k,aX[0]);

          mainwin.SetViewport(MINXY,MINYY,WIDEY,HIGHY,TRUE);
          setcolor(LIGHTGRAY);  gpx_y.Line((FTP)_k-1,aY[1],(FTP)_k,aY[0]);
          setcolor(BLUE);  gpx_y.Line((FTP)_k,0.0,(FTP)_k,aY[0]);
          delay(delayindex);

          if (_miny > aOutput[_k]) _miny = aOutput[_k];
          if (_maxy < aOutput[_k]) _maxy = aOutput[_k];
        }
      } while(_k != MAXOUTPUT && !kbhit());
      mainwin.UseWindowScope(TRUE);
}

//---------------------------------------
void InitAllObjects(void)
{
  iMAXOUTPUT.AdjustAttr(LIGHTBLUE,WHITE);

  iMAXOUTPUT.AdjustPosition(textw+100, 5);  iMAXOUTPUT.AdjustThick(3);
  iMU.AdjustPosition(textw+100,25);  iMU.AdjustThick(3);
  iMU0.AdjustPosition(textw+100,45);  iMU0.AdjustThick(3);
  iW0.AdjustPosition(textw+100,65);  iW0.AdjustThick(3);
  iNoiseDB.AdjustPosition(textw+100,85);  iNoiseDB.AdjustThick(3);
  iA0MIN.AdjustPosition(textw+270,25);  iA0MIN.AdjustThick(3);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  iA0MAX.AdjustPosition(textw+270,45);   iA0MAX.AdjustThick(3);
  idelay.AdjustPosition(textw+300,85);   idelay.AdjustThick(3);
  iMU.inputfn=iMU0.inputfn=iW0.inputfn=iNoiseDB.inputfn=idelay.inputfn=TextInputFunction;
  iA0MIN.inputfn=iA0MAX.inputfn=TextInputFunction;
  sscanf((char *)idelay, "%d", &delayindex);
  sscanf((char *)iA0MIN, "%" FTP_PREFIX "f", &A0MIN);
  sscanf((char *)iA0MAX, "%" FTP_PREFIX "f", &A0MAX);

  bPlotGraph.AdjustPosition(MRIGHT-300, 5,80,texth+10);
  bPlotGraph.AssignText("Plot graph");
  bPlotGraph.AdjustJustify(CHJUSTIFY|CVJUSTIFY);
  bPlotGraph.ReleaseButton();

  bSave.AdjustPosition(MRIGHT-200, 5,80,texth+10);
  bSave.AssignText("Save");
  bSave.AdjustJustify(CHJUSTIFY|CVJUSTIFY);
  bSave.ReleaseButton();

  bExit.AdjustPosition(MRIGHT-100, 5,80,texth+10);
  bExit.AssignText("Exit");
  bExit.AdjustJustify(CHJUSTIFY|CVJUSTIFY);
  bExit.ReleaseButton();

  gpx.AdjustGraphSize(0,0,WIDTH,HEIGHT);
  gpx_x.AdjustGraphSize(0,0,WIDEX,HIGHX);
  gpx_y.AdjustGraphSize(0,0,WIDEY,HIGHY);

  max_x = 1;  /* ใส่ไว้ชั่วคราว จะต้องถูกเปลี่ยนแปลงโดยการป้อนมุม */
  gpx_x.xzoomfactor = WIDEX/MAXGXX;  gpx_y.xzoomfactor = WIDEY/MAXGXY;
  gpx_x.yzoomfactor = HIGHX/MAXGYX;  gpx_y.yzoomfactor = HIGHY/MAXGYY;
}

//-------------------------------------
void ShowGetParametersDialog(void)
{
  EDGECOLORPACK _ep;

  _ep.Up = _ep.Left = DARKCOLOR;  _ep.Down = _ep.Right = HIGHCOLOR;

  // clear Main Window ---------------------
  mainwin.UseWindowScope(FALSE);
  WaitVertRetrace();  mainwin.Paint();
  mainwin.UseWindowScope(TRUE);

  // เขียน Legend ---------------------
  setcolor(YELLOW);
  outtextxy(MINXX+WIDEX/2,MINYX+HIGHX+textheight("k"),"k");
  outtextxy(MINXY+WIDEY/2,MINYY+HIGHY+textheight("k"),"k");
  outtextxy(MINX+WIDTH/2,MINY-textheight("k"),"k");
  outtextxy(MINX-textwidth("I1(k)"),MINY+HEIGHT/2, "I1(k)"); /* พิมพ์ชื่อตัวแปรที่แกน Y */
  outtextxy(MINXX-textwidth("Input"),MINYX+HIGHX/2, "Input");
  outtextxy(MINXY-textwidth("Output"),MINYY+HIGHY/2, "Output");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    mainwin.OutTextat(textw, 5,"MAXOUTPUT");
    mainwin.OutTextat(textw,25,"Mu");
    mainwin.OutTextat(textw,45,"Mu0");
    mainwin.OutTextat(textw,65,"W0");
    mainwin.OutTextat(textw,85,"S/N (db)");
    mainwin.OutTextat(textw+200,25,"A0 min");
    mainwin.OutTextat(textw+200,45,"A0 max");
    mainwin.OutTextat(textw+200,85,"Delay");

    iMAXOUTPUT.Paint();
    iMU.Paint();  iMU0.Paint();  iW0.Paint();  iNoiseDB.Paint();
    iA0MIN.Paint();  iA0MAX.Paint();
    idelay.Paint();
    bPlotGraph.Paint();  bSave.Paint();  bExit.Paint();

    // วาดพื้นที่สำหรับวาดกราฟให้ gpx ทั้งหลาย ————————
    LineStrip(MINX,MINY,WIDTH,HEIGHT,_ep,BLACK,WHITE,LIGHTGRAY,BTHICK);
    LineStrip(MINXX,MINYX,WIDEX,HIGHX,_ep,BLACK,WHITE,LIGHTGRAY,BTHICK);
    LineStrip(MINXY,MINYY,WIDEY,HIGHY,_ep,BLACK,WHITE,LIGHTGRAY,BTHICK);
}

/*————————————————————————*/
int ActivateInput(void)
{
  int _x,_y;

  if (KeyboardCmd == (int)CMD_MOUSE)
  {
    if (MStat.butt & MOUSELBIT)  /* ถ้า mouse กดปุ่มซ้าย */
    {
      _x = MStat.x;  _y = MStat.y;
    }
    else
    {
      _x=_y=-1;
    }
  }
  else
  {
ChkKeyboardCmd:;
    _x=_y=-1;
    switch(KeyboardCmd)
    {
    case 0x010F: /* Shift-TAB */
    case 0x09:  /* TAB */
      // ลบ cursor ปัจจุบัน ————————————
      setcolor(BLUE);  /* ใช้สีพื้นของ mainwin */
      switch(currentobj)
      {
      case OBJ_PLOTGRAPH:
        rectangle(bPlotGraph.Left()-3,bPlotGraph.Top()-3,
            bPlotGraph.Right()+3,bPlotGraph.Bottom()+3);
        break;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        case OBJ_SAVE:
            rectangle(bSave.Left()-3,bSave.Top()-3,
                    bSave.Right()+3,bSave.Bottom()+3);
            break;
        case OBJ_EXIT:
            rectangle(bExit.Left()-3,bExit.Top()-3,
                    bExit.Right()+3,bExit.Bottom()+3);
            break;
        }
NextField:;
        if (KeyboardCmd == 0x09)
        {
          currentobj++;  /* เลื่อน cursor ไปตำแหน่งถัดไป */
          if (currentobj == OBJ_NULL)  currentobj = OBJ_TOP;
        }
        else
        {
          if (currentobj==OBJ_TOP)  currentobj = OBJ_NULL;
          currentobj--;
        }

        setcolor(YELLOW);
        switch(currentobj)
        {
        case OBJ_PLOTGRAPH:
            rectangle(bPlotGraph.Left()-3,bPlotGraph.Top()-3,
                    bPlotGraph.Right()+3,bPlotGraph.Bottom()+3);
            break;
        case OBJ_SAVE:
            rectangle(bSave.Left()-3,bSave.Top()-3,
                    bSave.Right()+3,bSave.Bottom()+3);
            break;
        case OBJ_EXIT:
            rectangle(bExit.Left()-3,bExit.Top()-3,
                    bExit.Right()+3,bExit.Bottom()+3);
            break;
        case OBJ_IMU:
            _x = textw+100;  _y = 25;  break;
        case OBJ_IMU0:
            _x = textw+100;  _y = 45;  break;
        case OBJ_IW0:
            _x = textw+100;  _y = 65;  break;
        case OBJ_INOISEDB:
            _x = textw+100;  _y = 85;  break;
        case OBJ_IA0MIN:
            _x = textw+270;  _y = 25;  break;
        case OBJ_IA0MAX:
            _x = textw+270;  _y = 45;  break;
        case OBJ_DELAY:
            _x = textw+300;  _y = 85;  break;
        }
        break;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    case 0x0D:  /* ENTER */
        switch(currentobj)
        {
        case OBJ_PLOTGRAPH:
            _x = bPlotGraph.Left();  _y = bPlotGraph.Top();
            break;

        case OBJ_SAVE:
            _x = bSave.Left();  _y = bSave.Top();
            break;

        case OBJ_EXIT:
            _x = bExit.Left();  _y = bExit.Top();
            break;
        default:
            KeyboardCmd = 0x09;  /* ทำหน้าที่เป็น TAB แทน */
            goto NextField;
        }
        break;
    }
}

if (iMU.Touch(_x,_y))
{
  iMU.Getstring();
  sscanf((char *)iMU, "%" FTP_PREFIX "f", &vMU);

  goto ChkKeyboardCmd;
}
else
if (iMU0.Touch(_x,_y))
{
  iMU0.Getstring();
  sscanf((char *)iMU0, "%" FTP_PREFIX "f", &vMU0);

  goto ChkKeyboardCmd;
}
else
if (iW0.Touch(_x,_y))
{
  iW0.Getstring();
  sscanf((char *)iW0, "%" FTP_PREFIX "f", &vW0);
  max_x = SINENUM*2*PI/vW0;
  gpx_x.xzoomfactor = WIDEX/MAXGXX;  gpx_y.xzoomfactor = WIDEY/MAXGXY;

  goto ChkKeyboardCmd;
}
else
if (iNoiseDB.Touch(_x,_y))
{
  iNoiseDB.Getstring();
  sscanf((char *)iNoiseDB, "%" FTP_PREFIX "f", &noise_db);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    goto ChkKeyboardCmd;
  }
  else
  if (iA0MIN.Touch(_x,_y))
  {
    iA0MIN.Getstring();
    sscanf((char *)iA0MIN, "%" FTP_PREFIX "f", &A0MIN);

    goto ChkKeyboardCmd;
  }
  else
  if (iA0MAX.Touch(_x,_y))
  {
    iA0MAX.Getstring();
    sscanf((char *)iA0MAX, "%" FTP_PREFIX "f", &A0MAX);

    goto ChkKeyboardCmd;
  }
  else
  if (idelay.Touch(_x,_y))
  {
    idelay.Getstring();
    sscanf((char *)idelay, "%d", &delayindex);

    goto ChkKeyboardCmd;
  }
  else
  if (bPlotGraph.Touch(_x,_y))
  {
    bPlotGraph.PushButton();  bPlotGraph.Paint();
    ProcessingGraph();
    bPlotGraph.ReleaseButton();  bPlotGraph.Paint();
  }
  else
  if (bSave.Touch(_x,_y))
  {
    bSave.PushButton();  bSave.Paint();
    Save2File();
    bSave.ReleaseButton();  bSave.Paint();
  }
  else
  if (bExit.Touch(_x,_y))
  {
    bExit.PushButton();
    bExit.Paint();
    return FALSE;
  }
  return TRUE;
}

//---------------------------------------
void Save2File(void)
{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  WORD _k,_i;
  FILE *_fhandle,*_hhandle;

  if ((_hhandle = fopen(DATAFILENAME ".Hdr","wt")) != NULL)
  {
    // เขียนจำนวนสูงที่สุด และจำนวนกราฟที่ส่วนหัว .Hdr ———
    fprintf(_hhandle,"%u %d\n", MAXOUTPUT, graphidx);
    fclose(_hhandle);

    // เขียนข้อมูลจริงๆไว้ที่ .Dat ————————————
    if ((_fhandle = fopen(DATAFILENAME ".Dat","wt")) != NULL)
    {
      for(_k=0;_k<MAXOUTPUT;_k++)
      {
        fprintf(_fhandle,"\n%d ",_k);
        for(_i=0;_i<graphidx;_i++)
          fprintf(_fhandle,"%" FTP_PREFIX "g ",
            ((FTP huge *)aaOutput[_i])[_k]);
      }
      fclose(_fhandle);
    }
  }
}

/*————————————————————*/
void AdaptiveA1nA0(void)
{
  int _ef;

  KeyboardCmd = CMD_NOINPUT;
  currentobj = OBJ_IMU;
  graphidx = 0;

  randomize();
  InitAllObjects();
  ShowGetParametersDialog();

  // get MAXOUTPUT เป็นอันดับแรก ————————————
  do {
    _ef = FALSE;
    MAXOUTPUT.GetString();
    sscanf((char *)MAXOUTPUT, "%u", &MAXOUTPUT);

    if (!MAXOUTPUT)  return;  /* ออกจากโปรแกรมเมื่อ MAXOUTPUT = 0 */

    // เราไม่ใช้ new operator เพราะว่ามันจองได้ไม่เกิน 64 KB ——
    aaOutput[graphidx] = (FTP *)farcalloc(MAXOUTPUT,sizeof(FTP));
    _ef = (aaOutput[graphidx] == NULL);
  } while(_ef);
  aOutput = aaOutput[graphidx++];
  MAXOUTPUT.AdjustAttr(DARKGRAY,WHITE);
  MAXOUTPUT.Paint();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
// ทำการ scale ให้พอเหมาะและเปลี่ยนจุด Ongin ————————*/
gpx.xzoomfactor = WIDTH/MAXGRAPHX;
gpx.yzoomfactor = HEIGHT/MAXGRAPHY;
gpx.MoveOngin(0,20/gpx.yzoomfactor);

// ตีตารางสำหรับ gpx ————————————————
mainwin.SetViewport(MINX,MINY,WIDTH,HEIGHT,TRUE);
setcolor(GREEN);
gpx.DrawGridxy(MAXOUTPUT/20,0.1);  /* ความละเอียดของจอหารด้วยจำนวนกริดที่ต้องการ */

setcolor(BLACK);
gpx.DisplayAxis();

setcolor(RED);
gpx.DrawNumberxy(MAXOUTPUT/10,0.1,0,1); /* เขียนตัวเลขที่กริด */

mainwin.UseWindowScope(TRUE);

// run program loop ————————————————
do {
  WaitInput();
} while(ActivateInput());

while(graphidx)
  farfree(aaOutput[--graphidx]);

mainwin.UseWindowScope(FALSE);
}


DspExp9.Cpp
/*————————————————————————*/
/*
** LL    aA   RRRR  GGG  EEEE
** LL   AA AA RRRR G    EE
** LL   AAAAAA RR   G GG EEE
** LLLLL AA  AA R RR  GGG  EEEE Model only!!
**
** โปรแกรม plot Sine 1, Sine 01 และ Sine 02 เมื่อมีการเปลี่ยนแปลงทั้ง alpha 0.1
**
*————————————————————————*/
#pragma hdrfile "DspExp9.Sym"

#include "dsps.h"
#undef MAXOUTPUT

// define NOISE macro เพื่อ gen input แบบ noise ————————
#define NOISE

/*-=-=-=-=-=-=-=-=-=-=-=-=-= define  Macros -=-=-=-=-=-=-=-=-=-=-=-=-=-=*/
#define DATAFILENAME "DspExp6"
#define MAXGRAPH  5
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#define BTHICK 2
#define MINX  50
#define MINY  110
#define WIDTH  (MWIDTH-100)
#define HEIGHT (MHEIGHT-140)
#define MAXGRAPHX (FTP)MAXOUTPUT
#define MAXGRAPHY 8.0

static enum {
    OBJ_TOP=1,
    OBJ_PLOTGRAPH=1,OBJ_SAVE,OBJ_EXIT,
    OBJ_IMU,OBJ_IMU0,OBJ_IW0,OBJ_INOISEDB,OBJ_IA0MIN,OBJ_IA0MAX,OBJ_PLOTMODE,
    OBJ_NULL
    }  currentobj;

/*-------------------- Global variables --------------------*/
static int plotmode;
static WORD MAXOUTPUT;
static FTP vMU,vMU0,vW0;
static FTP aA0[3],aA1[3],aPhi1[3],aPhi2[3],aY[3],aX[3],aPHI[3];
static FTP *aaOutput[MAXGRAPH],huge *aOutput,huge *aSine01Output,
        huge *aSine02Output;
static FTP A0MIN,A0MAX;

static WORD random_seed;
static FTP noisefactor,noise_db;

static int graphidx;

extern int textw,texth;
extern int KeyboardCmd;
extern MOUSE_PARAM MStat;
extern pWindowBasic mainwin;

static pTextInput iMU("0",11),iMU0("0",11),iW0("0",11),iNoiseDB("0",4);
static pTextInput iA0MIN("0.1",5),iA0MAX("0.85",5);
static pTextInput iMAXOUTPUT("0",6),iPlotMode("0",2);
static TextButton bExit,bPlotGraph,bSave;
static mBaseGraphic gpx;

/*-------------------- Function  Prototypes --------------------*/
/* First part ----------------------------------*/
void AdaptiveSine(void);

static void Function_Y(void);
static void Function_A0(void);
static void Function_A1(void);
static void Function_PHI(void);
static void Function_Phi1(void);
static void Function_Phi2(void);
static FTP FindNoiseFactor(void);
static void DrawGPXgnd(void);
```

เอกสารนี้ไฟล์เอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
static void ProcessingGraph(void);
static void Save2File(void);
static void InitAllObjects(void);
static void ShowGetParametersDialog(void);
static int  ActivateInput(void);

extern void WaitInput(void);
extern int TextInputFunction(void);
extern void ShiftArray(FTP *array);

#pragma hdrstop

/*=-=-=-=-=-=-=-=-=-=-=-=-=-=-=- Functions body =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-*/
/*-----------------------------------------------*/
void Function_Y(void)
{
  aY[0] = (1-aA0[0])/2*(aX[0]-aX[2])+ aA1[0]*(1+aA0[0])*aY[1] -aA0[0]*aY[2];
}

/*-----------------------------------------------*/
void Function_A0(void)
{
  aA0[0]=aA0[1]+vMU0*aY[1]*aPHI[1]*(aPHI[1]*aPhi1[1] + aPhi2[1]*aY[1]);

  /*  เนื่องจาก Alpha 0 ถูกคำนวณโดยสมการของ Alpha 1 จึงต้อง -*/
  /*  จำกัดค่า Alpha 0 ไม่ให้เกิน range ของมัน (0,1) ----*/
  if (aA0[0] > A0MAX) aA0[0] = A0MAX;
  else
  if (aA0[0] < A0MIN) aA0[0] = A0MIN;
}

/*-----------------------------------------------*/
void Function_A1(void)
{
  aA1[0] = aA1[1] +vMU*aY[1]*aPHI[1];
}

/*-----------------------------------------------*/
void Function_PHI(void)
{
  aPHI[0] = aA1[0]*(1+aA0[0])*aPHI[1] -aA0[0]*aPHI[2] +(1+aA0[0])*aY[1];
}

/*-----------------------------------------------*/
void Function_Phi1(void)
{
  aPhi1[0]=(aX[2]-aX[0])/2 +aA1[0]*aY[1] -aY[2] +aA1[0]*(1+aA0[0])*aPhi1[1]
    -aA0[0]*aPhi1[2];
}

/*-----------------------------------------------*/
void Function_Phi2(void)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
  aPhi2[0]=aA1[0]*(1+aA0[0])*aPhi2[1] +aA1[0]*aPHI[1] -aA0[0]*aPhi2[2]
   -aPHI[2] +(1+aA0[0])*aPhi1[1] +aY[1];
}

//-----------------------------------------
// สร้าง noise และกำหนด noise factor
//
FTP FindNoiseFactor(void)
{
  WORD _i;
  FTP _noise_avr,_actual_noise;

  // คำนวณค่าของ noise จริงๆ เพื่อนำไปหา noise factor ที่จะนำไปคูณ
  // กับค่า noise ใน noise array แต่ละตัว ----------
  //
  _actual_noise = cos(vW0)/pow((FTP)10.0,noise_db/10.0);

  // กำหนดค่า random seed เพื่อใช้ในการสุ่ม ----------
  random_seed = random(32767)+random(32767)+1;

  // หาค่าเฉลี่ยของ noise array ----------
  _noise_avr = 0;
  srand(random_seed);
  for(_i=0;_i<MAXOUTPUT;_i++)
    /* ต้องคิด noise เฉพาะค่าบวกเท่านั้น จึงจะหา S/N ratio ได้ถูกต้อง */
    _noise_avr+=fabs((random(2000)/1000.0)-1.0);

  _noise_avr /= MAXOUTPUT;

  return _actual_noise/_noise_avr; /* ได้อัตราส่วนของ noise factor ที่ต้องการ */
}

//-----------------------------------------
void ProcessingGraph(void)
{
  WORD _k;
// FTP _minY, _maxY;

  // clear array ที่ใช้สำหรับการคำนวณ ----------
  for(_k=0;_k<3;_k++)
  {
    aX[_k]=aY[_k]=aA0[_k]=aA1[_k]=aPHI[_k]=aPhi1[_k]=aPhi2[_k]=(FTP)0;
  }

  // กำหนดค่าเริ่มต้น ----------
// _minY = _maxY = 0.0;
  noisefactor = FindNoiseFactor();
  srand(random_seed);

#ifdef NOISE
  aX[0] = 1.0+((random(2000)/1000.0)-1.0)*noisefactor;
#else
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    aX[0] = 1.0;
#endif
    Function_Phi1();
    Function_Y();

    aOutput[0]=aPHI[0];  /* aOutput เก็บข้อมูลของ Sine1 อยู่ */
    aSine01Output[0]=aPhi1[0];
    aSine02Output[0]=0.0;

    ShowGetParametersDialog();  /* clear window graph */
    DrawGPXgrid();
    mainwin.UseWindowScope(TRUE);

    // เขียนตัวแปรกำกับให้สอดคล้องกับ plotmode ————————
    setfillstyle(SOLID_FILL,BLUE);
    bar(0,MINY+HEIGHT/2,MINX-1,MINY+HEIGHT/2+textheight("M"));  /* ลบตัวอักษรเก่า */
    setcolor(YELLOW);  /* กำหนดสีตัวอักษรปกติ */
    switch(plotmode)
    {
    case 0:  /* plot Sine 1 */
            outtextxy(MINX-textwidth("Sine"),MINY+HEIGHT/2, "Sine");
            break;
    case 1:  /* plot Sine 01 */
            outtextxy(MINX-textwidth("Sine01"),MINY+HEIGHT/2, "Sine01");
            break;
    case 2:  /* plot Sine 02 */
            outtextxy(MINX-textwidth("Sine02"),MINY+HEIGHT/2, "Sine02");
            break;
    }

    mainwin.SetViewport(MINX,MINY,WIDTH,HEIGHT,TRUE);
    for(_k=1; _k<MAXOUTPUT; _k++)
    {
      /* เลื่อนแต่ละ array เพื่อคำนวณค่าต่อมา ————————*/
      ShiftArray(aX); ShiftArray(aY); ShiftArray(aA1); ShiftArray(aA0);
      ShiftArray(aPHI); ShiftArray(aPhi1); ShiftArray(aPhi2);
#ifdef NOISE
      aX[0] = cos(_k*vW0)+((random(2000)/1000.0)-1.0)*noisefactor;
#else
      aX[0] = cos(_k*vW0);
#endif
      Function_A0();  /* เลือกคำนวณ Alpha 0 และ 1 ก่อน เพราะมันใช้ค่า param*/
      Function_A1();  /* ที่เป็นอดีตเท่านั้น                          */
      Function_PHI();  /* fn เหล่านี้ใช้ Aplha 0,1 ที่เป็นปัจจุบันเท่านั้น*/
      Function_Phi1();
      Function_Phi2();
      Function_Y();  /* คำนวณ output Y */

      // เก็บค่าของกราฟไว้ save ————————————————
      aOutput[_k]=aPHI[0];
      aSine01Output[_k]=aPhi1[0];
      aSine02Output[_k]=aPhi2[0];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    // plot Sine-Functions ทั้งหลาย ————————————
    switch(plotmode)
    {
    case 0:  /* plot Sine 1 */
            setcolor(BLUE);
            gpx.Line((FTP)_k-1,aOutput[_k-1],(FTP)_k,aOutput[_k]);
            break;
    case 1:  /* plot Sine 01 */
            setcolor(MAGENTA);
            gpx.Line((FTP)_k-1,aSine01Output[_k-1],(FTP)_k,aSine01Output[_k]);
            break;
    case 2:  /* plot Sine 02 */
            setcolor(BROWN);
            gpx.Line((FTP)_k-1,aSine02Output[_k-1],(FTP)_k,aSine02Output[_k]);
            break;
    }

//  if (_miny > aOutput[_k]) _miny = aOutput[_k];
//  if (_maxy < aOutput[_k]) _maxy = aOutput[_k];
  }
  mainwin.UseWindowScope(TRUE);
}

//————————————————————————————————
void InitAllObjects(void)
{
  plotmode = 0;  /* ค่าเริ่มต้นให้เขียนกราฟของ Sine1 */
  KeyboardCmd = (int)CMD_MOUSE;
  currentobj = OBJ_IMU;
  MStat.x = textw*100;  MStat.y = 25;  MStat.butt = MOUSELBIT;
  graphidx = 0;

  iMAXOUTPUT.AdjustAttr(LIGHTBLUE,WHITE);

  iMAXOUTPUT.AdjustPosition(textw*100, 5);  iMAXOUTPUT.AdjustThick(3);
  iMU.AdjustPosition(textw*100,25);  iMU.AdjustThick(3);
  iMU0.AdjustPosition(textw*100,45);  iMU0.AdjustThick(3);
  iW0.AdjustPosition(textw*100,65);  iW0.AdjustThick(3);
  iNoiseDB.AdjustPosition(textw*100,85);  iNoiseDB.AdjustThick(3);
  iA0MIN.AdjustPosition(textw*270,25);  iA0MIN.AdjustThick(3);
  iA0MAX.AdjustPosition(textw*270,45);  iA0MAX.AdjustThick(3);
  iPlotMode.AdjustPosition(textw*240,85);  iPlotMode.AdjustThick(3);
  iMU.inputfn=iMU0.inputfn=iW0.inputfn=iNoiseDB.inputfn=iPlotMode.inputfn=TextInputFunction;
  iA0MIN.inputfn=iA0MAX.inputfn=TextInputFunction;
  sscanf((char *)iA0MIN, "%" FTP_PREFIX "f", &A0MIN);
  sscanf((char *)iA0MAX, "%" FTP_PREFIX "f", &A0MAX);

  bPlotGraph.AdjustPosition(MRIGHT-300, 5,80,texth*10);
  bPlotGraph.AssignText("Plot graph");
  bPlotGraph.AdjustJustify(CHJUSTIFY|CVJUSTIFY);
  bPlotGraph.ReleaseButton();

  bSave.AdjustPosition(MRIGHT-200, 5,80,texth*10);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    bSave.AssignText("Save");
    bSave.AdjustJustify(CHJUSTIFY|CVJUSTIFY);
    bSave.ReleaseButton();

    bExit.AdjustPosition(MRIGHT-100, 5,80,texth+10);
    bExit.AssignText("Exit");
    bExit.AdjustJustify(CHJUSTIFY|CVJUSTIFY);
    bExit.ReleaseButton();

    gpx.AdjustGraphSize(0,0,WIDTH,HEIGHT);
}

//-----------------------------------------
void ShowGetParametersDialog(void)
{
    EDGECOLORPACK _ep;

    _ep.Up = _ep.Left = DARKCOLOR;  _ep.Down = _ep.Right = HIGHCOLOR;

    // clear Main Window ------------------------
    mainwin.UseWindowScope(FALSE);
    WaitVertRetrace();  mainwin.Paint();
    mainwin.UseWindowScope(TRUE);

    // เขียนชื่อตัวแปรที่แกน ------------------------
    setcolor(YELLOW);
    outtextxy(MINX+WIDTH/2,MINY+HEIGHT+5,"x");

    mainwin.OutTextat(textw, 5,"MAXOUTPUT");
    mainwin.OutTextat(textw,25,"Mu");
    mainwin.OutTextat(textw,45,"Mu0");
    mainwin.OutTextat(textw,65,"W0");
    mainwin.OutTextat(textw,85,"S/N (db)");
    mainwin.OutTextat(textw+160,85,"plot mode");
    mainwin.OutTextat(textw+200,25,"A0 min");
    mainwin.OutTextat(textw+200,45,"A0 max");
    mainwin.OutTextat(textw+270,70,"0 - Sine  ",(WHITE<<4)|BLUE);
    mainwin.OutTextat(textw+270,79,"1 - Sine01",(WHITE<<4)|MAGENTA);
    mainwin.OutTextat(textw+270,88,"2 - Sine02",(WHITE<<4)|BROWN);

    iMAXOUTPUT.Paint();
    iMU.Paint();  iMU0.Paint();  iW0.Paint();  iNoiseDB.Paint();
    iA0MIN.Paint();  iA0MAX.Paint();  iPlotMode.Paint();
    bPlotGrapn.Paint();  bSave.Paint();  bExit.Paint();

    // วาดพื้นที่สำหรับวาดกราฟให้ gpx ทั้งหมด ----------
    LineStrip(MINX,MINY,WIDTH,HEIGHT,_ep,BLACK,WHITE,LIGHTGRAY,BTHICK);
}

/*-----------------------------------------*/
int ActivateInput(void)
{
    int _x,_y;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    if (KeyboardCmd == (int)CMD_MOUSE)
    {
      if (MStat.butt & MOUSELBIT)  /* ถ้า mouse กดปุ่มซ้าย */
      {
        _x = MStat.x;  _y = MStat.y;
      }
      else
      {
        _x=_y=-1;
      }
    }
    else
    {
  ChkKeyboardCmd:;
      _x=_y=-1;
      switch(KeyboardCmd)
      {
      case 0x010F: /* Shift-TAB */
      case 0x09:  /* TAB */
        // ลบ cursor ปัจจุบัน ----------
        setcolor(BLUE);  /* ใช้สีพื้นของ mainwin */
        switch(currentobj)
        {
        case OBJ_PLOTGRAPH:
          rectangle(bPlotGraph.Left()-3,bPlotGraph.Top()-3,
                bPlotGraph.Right()+3,bPlotGraph.Bottom()+3);
          break;
        case OBJ_SAVE:
          rectangle(bSave.Left()-3,bSave.Top()-3,
                bSave.Right()+3,bSave.Bottom()+3);
          break;
        case OBJ_EXIT:
          rectangle(bExit.Left()-3,bExit.Top()-3,
                bExit.Right()+3,bExit.Bottom()+3);
          break;
        }
  NextField:;
        if (KeyboardCmd == 0x09)
        {
          currentobj++;  /* เลื่อน cursor ไปตำแหน่งถัดไป */
          if (currentobj == OBJ_NULL)  currentobj = OBJ_TOP;
        }
        else
        {
          if (currentobj==OBJ_TOP)  currentobj = OBJ_NULL;
          currentobj--;
        }

        setcolor(YELLOW);
        switch(currentobj)
        {
        case OBJ_PLOTGRAPH:
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            rectangle(bPlotGraph.Left()-3,bPlotGraph.Top()-3,
                    bPlotGraph.Right()+3,bPlotGraph.Bottom()+3);
            break;
        case OBJ_SAVE:
            rectangle(bSave.Left()-3,bSave.Top()-3,
                    bSave.Right()+3,bSave.Bottom()+3);
            break;
        case OBJ_EXIT:
            rectangle(bExit.Left()-3,bExit.Top()-3,
                    bExit.Right()+3,bExit.Bottom()+3);
            break;
        case OBJ_IMU:
            _x = textw+100;  _y = 25;  break;
        case OBJ_IMU0:
            _x = textw+100;  _y = 45;  break;
        case OBJ_IW0:
            _x = textw+100;  _y = 65;  break;
        case OBJ_INOISEDB:
            _x = textw+100;  _y = 85;  break;
        case OBJ_IA0MIN:
            _x = textw+270;  _y = 25;  break;
        case OBJ_IA0MAX:
            _x = textw+270;  _y = 45;  break;
        case OBJ_PLOTMODE:
            _x = textw+240;  _y = 85;  break;
        }
        break;

    case 0x0D:  /* ENTER */
        switch(currentobj)
        {
        case OBJ_PLOTGRAPH:
            _x = bPlotGraph.Left();  _y = bPlotGraph.Top();
            break;

        case OBJ_SAVE:
            _x = bSave.Left();  _y = bSave.Top();
            break;

        case OBJ_EXIT:
            _x = bExit.Left();  _y = bExit.Top();
            break;
        default:
            KeyboardCmd = 0x09;  /* ทำหน้าที่เป็น TAB แทน */
            goto NextField;
        }
        break;
    }
}

if (iMU.Touch(_x,_y))
{
    iMU.Getstring();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    sscanf((char *)iMU, "%" FTP_PREFIX "f", &vMU);

    goto ChkKeyboardCmd;
  }
  else
  if (iMU0.Touch(_x,_y))
  {
    iMU0.Getstring();
    sscanf((char *)iMU0, "%" FTP_PREFIX "f", &vMU0);

    goto ChkKeyboardCmd;
  }
  else
  if (iW0.Touch(_x,_y))
  {
    iW0.Getstring();
    sscanf((char *)iW0, "%" FTP_PREFIX "f", &vW0);

    goto ChkKeyboardCmd;
  }
  else
  if (iNoiseDB.Touch(_x,_y))
  {
    iNoiseDB.Getstring();
    sscanf((char *)iNoiseDB, "%" FTP_PREFIX "f", &noise_db);

    goto ChkKeyboardCmd;
  }
  else
  if (iA0MIN.Touch(_x,_y))
  {
    iA0MIN.Getstring();
    sscanf((char *)iA0MIN, "%" FTP_PREFIX "f", &A0MIN);

    goto ChkKeyboardCmd;
  }
  else
  if (iA0MAX.Touch(_x,_y))
  {
    iA0MAX.Getstring();
    sscanf((char *)iA0MAX, "%" FTP_PREFIX "f", &A0MAX);

    goto ChkKeyboardCmd;
  }
  else
  if (iPlotMode.Touch(_x,_y))
  {
    iPlotMode.Getstring();
    sscanf((char *)iPlotMode, "%d", &plotmode);
  }
  else
  if (bPlotGraph.Touch(_x,_y))
  {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      bPlotGraph.PushButton();   bPlotGraph.Paint();
      ProcessingGraph();
      bPlotGraph.ReleaseButton();   bPlotGraph.Paint();
    }
    else
    if (bSave.Touch(_x,_y))
    {
      bSave.PushButton();   bSave.Paint();
      Save2File();
      bSave.ReleaseButton();   bSave.Paint();
    }
    else
    if (bExit.Touch(_x,_y))
    {
      bExit.PushButton();
      bExit.Paint();
      return FALSE;
    }
    return TRUE;
}

//-------------------------------------
void Save2File(void)
{
  WORD _k,_i;
  FILE *_fhandle,*_hhandle;

  if ((_hhandle = fopen(DATAFILENAME ".Hdr","wt")) != NULL)
  {
    // :เขียนจำนวนสูงที่สุด และจำนวนกราฟที่ส่วนหัว .Hdr ------
    fprintf(_hhandle,"%u %d\n", MAXOUTPUT, graphidx);
    fclose(_hhandle);

    // :เขียนข้อมูลจริงๆไว้ที่ .Dat ------------------
    if ((_fhandle = fopen(DATAFILENAME ".Dat","wt")) != NULL)
    {
      for(_k=0;_k<MAXOUTPUT;_k++)
      {
        fprintf(_fhandle,"\n%d ",_k);
        for(_i=0;_i<graphidx;_i++)
          fprintf(_fhandle,"%" FTP_PREFIX "g ",
            ((FTP huge *)aaOutput[_i])[_k]);
      }
      fclose(_fhandle);
    }
  }
}

//-------------------------------------
void DrawGPXgnd(void)
{
  // ตั้งตารางสำหรับ gpx ----------------------
  mainwin.SetViewport(MINX,MINY,WIDTH,HEIGHT,TRUE);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  setcolor(GREEN);
  gpx.DrawGridxy(MAXOUTPUT/20,2);  /* ความละเอียดของจอหารด้วยจำนวนกริดที่ต้องการ */

  setcolor(BLACK);
  gpx.DisplayAxis();

  setcolor(RED);
  gpx.DrawNumberxy(MAXOUTPUT/10,2,0,0); /* เขียนตัวเลขที่กริด */
}

/*-------------------------------------*/
void AdaptiveSine(void)
{
  int _ef;

  randomize();
  InitAllObjects();
  ShowGetParametersDialog();

  // get MAXOUTPUT เป็นอันดับแรก ----------
  do {
    _ef = FALSE;
    MAXOUTPUT.GetString();
    sscanf((char *)MAXOUTPUT, "%u", &MAXOUTPUT);

    if (!MAXOUTPUT)  return;  /* ออกจากโปรแกรมเมื่อ MAXOUTPUT = 0 */

    // เราไม่ใช้ new operator เพราะว่ามันจองได้ไม่เกิน 64 KB ----
    aaOutput[graphidx] = (FTP *)farcalloc(MAXOUTPUT,sizeof(FTP));
    if (aaOutput[graphidx] == NULL)
      _ef = TRUE;
    else
    { /* จอง array 2 columns */
      aaOutput[++graphidx] = (FTP *)farcalloc(MAXOUTPUT,sizeof(FTP));
      if (aaOutput[graphidx] == NULL)
      {
        _ef = TRUE;
        farfree(aaOutput[--graphidx]);
      }
      else
      { /* จอง array สำหรับ 3 columns */
        aaOutput[++graphidx] = (FTP *)farcalloc(MAXOUTPUT,sizeof(FTP));
        if (aaOutput[graphidx] == NULL)
        {
          _ef = TRUE;
          farfree(aaOutput[--graphidx]);
          farfree(aaOutput[--graphidx]);
        }
      }
    }
  } while(_ef);
  aOutput = aaOutput[graphidx-2];
  aSine01Output = aaOutput[graphidx-1];
```

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    aSine02Output = saOutput[graphidx];
    graphidx++;  /* Make the true count */
    iMAXOUTPUT.AdjustAttr(DARKGRAY,WHITE);
    iMAXOUTPUT.Paint();

    // ทำการ scale ให้พอเหมาะและเปลี่ยนจุด Origin ---------*/
    gpx.xzoomfactor = WIDTH/MAXGRAPHX;
    gpx.yzoomfactor = HEIGHT/MAXGRAPHY;
  // gpx.MoveOrigin(0,20/gpx.yzoomfactor);
    gpx.MoveOrigin(0,MAXGRAPHY/2.0);

    DrawGPXgrid();

    mainwin.UseWindowScope(TRUE);

    // run program loop ----------------------
    while(ActivateInput())
      WaitInput();

    while(graphidx)
      farfree(saOutput[--graphidx]);

    mainwin.UseWindowScope(FALSE);
}


DspExp10.Cpp
/*-------------------------------------*/
/*
** LL    aA   RRRR  GGG  EEEE
** LL   AA AA RRRR G    EE
** LL   AAAAAA RR  G GG EEE
** LLLLL AA AA R RR  GGG  EEEE Model only!!
**
** โปรแกรม plot alpha 1 และ input X ,output Y เมื่อ alpha 0 เปลี่ยนแปลง
** และใช้ Transfer function G ในการเปลี่ยนค่า
/*-------------------------------------*/
#pragma hdrfile "DspExp8.Sym"

#include "dsps.h"
#undef MAXOUTPUT

// define NOISE macro เพื่อ gen input แบบ noise ---------
#define NOISE

/*-=-=-=-=-=-=-=-=-=-=-=-=-= define Macros -=-=-=-=-=-=-=-=-=-=-=-=-=*/
#define DATAFILENAME "DspExp9"
#define MAXGRAPH 5
#define SINENUM 3  /* จำนวนคลื่น sine ที่ต้องการให้ปรากฏในหนึ่งจอภาพ */

#define BTHICK 2
#define MINX  50
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#define MINY   (MINYX+HIGHX+20)
#define WIDTH  (MWIDTH-100)
#define HEIGHT ((MHEIGHT-140)>>1)
#define MAXGRAPHX (FTP)MAXOUTPUT
#define MAXGRAPHY 1.1

#define MINXX  50
#define MINYX  110
#define WIDEX  ((WIDTH>>1)-20)
#define HIGHX  HEIGHT
#define MAXGXX (FTP)max_x
#define MAXGYX 2.0

#define MINXY  (MINXX+WIDEX+50)
#define MINYY  MINYX
#define WIDEY  WIDEX
#define HIGHY  HIGHX
#define MAXGXY (FTP)max_x
#define MAXGYY 2.0

static enum {
    OBJ_TOP=1,
    OBJ_PLOTGRAPH=1,OBJ_SAVE,OBJ_EXIT,
    OBJ_IMU,OBJ_IMU0,OBJ_IW0,OBJ_INOISEDB,OBJ_IA0MIN,OBJ_IA0MAX,OBJ_DELAY,
    OBJ_NULL
    } currentobj;

/*---------- Global variables ----------*/
static WORD MAXOUTPUT;
static FTP vMU,vMU0,vW0;
static FTP aA0[3],aA1[3],aPhi1[3],aPhi2[3],aY[3],aX[3],aPHI[3];
static FTP *aaOutput[MAXGRAPH],huge *aOutput;
static FTP A0MAX,A0MIN;   /* alpha 0 maximum, minimum */

static WORD random_seed;
static FTP noisefactor,noise_db;

static int max_x;  /* ความกว้างสูงสุดของ input และ output window */
                   /* จะเปลี่ยนแปลงไปตามค่ามุม (vW0)             */
static int delayindex;
static int graphidx;

extern int textw,texth;
extern int KeyboardCmd;
extern MOUSE_PARAM MStat;
extern pWindowBasic mainwin;

static pTextInput iMU("0",11),iMU0("0",11),iW0("0",11),iNoiseDB("0",4);
static pTextInput iA0MIN("0.1",5),iA0MAX("0.85",5);
static pTextInput iMAXOUTPUT("0",6),idelay("5",2);
static TextButton bExit,bPlotGraph,bSave;
static mBaseGraphic gpx,gpx_x,gpx_y;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/*-------------------- Function  Prototypes --------------------*/
/* First part ------------------------------*/
void AdaptiveA1nA0(void);

static void Function_Y(void);
static void Function_A0(void);
static void Function_A1(void);
static void Function_PHI(void);
static void Function_Phi1(void);
static void Function_Phi2(void);

static FTP FindNoiseFactor(void);
static void ProcessingGraph(void);
static void Save2File(void);
static void InitAllObjects(void);
static void ShowGetParametersDialog(void);
static int  ActivateInput(void);

extern void WaitInput(void);
extern int TextInputFunction(void);
extern void ShiftArray(FTP *array);

#pragma hdrstop

/*-------------------- Functions body --------------------*/
/*------------------------------------------*/
void Function_Y(void)
{
  aY[0] = (1-aA0[0])/2*(aX[0]-aX[2])+ aA1[0]*(1+aA0[0])*aY[1] -aA0[0]*aY[2];
}

/*------------------------------------------*/
void Function_A0(void)
{
  aA0[0]=aA0[1]+vMU0*aY[1]*aPHI[1]*(aPHI[1]*aPhi1[1] + aPhi2[1]*aY[1]);

  /* เนื่องจาก Alpha 0 ถูกคำนวณโดยสมการของ Alpha 1 จึงต้อง */
  /* จำกัดค่า Alpha 0 ไม่ให้เกิน range ของมัน (0,1) ------*/
  if (aA0[0] > A0MAX) aA0[0] = A0MAX;
  else
  if (aA0[0] < A0MIN) aA0[0] = A0MIN;
}

/*------------------------------------------*/
void Function_A1(void)
{
  aA1[0] = aA1[1] +vMU*aY[1]*aPHI[1];
}

/*------------------------------------------*/
void Function_PHI(void)
{
  aPHI[0] = aA1[0]*(1+aA0[0])*aPHI[1] -aA0[0]*aPHI[2] +(1+aA0[0])*aY[1];
```

เอกสารนี้ใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}

/*-------------------------------------*/
void Function_Phi1(void)
{
  aPhi1[0]=aA1[0]*(1+aA0[0])*aPhi1[1]-aA0[0]*aPhi1[2]
       +(2*aA1[0]*aY[1]-aY[2]-aY[0])/(1-aA0[0]);
}

/*-------------------------------------*/
void Function_Phi2(void)
{
  aPhi2[0]=aA1[0]*(1+aA0[0])*aPhi2[1]-aA0[0]*aPhi2[2]
       -2*(aA0[0]*aPHI[0]-aA1[0]*(1+aA0[0])*aPHI[1]+aPHI[2])/(1-aA0[0]);
}

//-------------------------------------
//  สร้าง noise และกำหนด noise factor
//
FTP FindNoiseFactor(void)
{
  WORD _i;
  FTP _noise_avr,_actual_noise;

  // คำนวณค่าของ noise จริงๆ เพื่อนำไปหา noise factor ที่จะนำไปคูณ
  // กับค่า noise ใน noise array แต่ละตัว ----------
  //
  _actual_noise = cos(vW0)/pow((FTP)10.0,noise_db/10.0);

  // กำหนดค่า random seed เพื่อใช้ในการสุ่ม ----------
  random_seed = random(32767)+random(32767)+1;

  // หาค่าเฉลี่ยของ noise array ----------
  _noise_avr = 0;
  srand(random_seed);
  for(_i=0;_i<MAXOUTPUT;_i++)
    /* ต้องคิด noise เฉพาะค่าบวกเท่านั้น จึงจะหา S/N ratio ได้ถูกต้อง */
    _noise_avr+=fabs((random(2000)/1000.0)-1.0);

  _noise_avr /= MAXOUTPUT;

  return _actual_noise/_noise_avr;  /* ได้อัตราส่วนของ noise factor ที่ต้องการ */
}

//-------------------------------------
void ProcessingGraph(void)
{
  WORD _k,_k1;
  FTP _miny, _maxy;

  // clear array ที่ใช้สำหรับการคำนวณ ----------
  for(_k=0;_k<3;_k++)
  {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    aX[_k]=aY[_k]=aA0[_k]=aA1[_k]=aPHI[_k]=aPhi1[_k]=aPhi2[_k]=(FTP)0;
  }

  // กำหนดค่าเริ่มต้น ————————————————
  _miny = _maxy = 0.0;
  noisefactor = FindNoiseFactor();
  srand(random_seed);

#ifdef NOISE
  aX[0] = 1.0+((random(2000)/1000.0)-1.0)*noisefactor;  /* เพิ่ม noise เข้าไปด้วย */
#else
  aX[0] = 1.0;
#endif
  Function_Y();
  Function_Phi1();

  aOutput[0]=0.0;
  _k = 1;
  do {
    _k1 = 0;

    // สำหรับ gpx ของ X ————————————————
    mainwin.SetViewport(MINXX,MINYX,WIDEX,HIGHX,TRUE);
    gpx_x.MoveOrigin(1.0-_k,MAXGYX/2);
    setfillstyle(SOLID_FILL,WHITE);
    bar(0,0,WIDEX,HIGHX);
    setcolor(BLACK);  gpx_x.DisplayAxis();

    // สำหรับ gpx ของ Y ————————————————
    mainwin.SetViewport(MINXY,MINYY,WIDEY,HIGHY,TRUE);
    gpx_y.MoveOrigin(1.0-_k,MAXGYY/2);
    setfillstyle(SOLID_FILL,WHITE);
    bar(0,0,WIDEY,HIGHY);
    setcolor(BLACK);  gpx_y.DisplayAxis();

    setcolor(BLUE);
    for(; _k<MAXOUTPUT && _k1<MAXGXX; _k++,_k1++)
    {
      /* เลื่อนแต่ละ array เพื่อคำนวณค่าต่อมา ————————*/
      ShiftArray(aX);  ShiftArray(aY);  ShiftArray(aA1);  ShiftArray(aA0);
      ShiftArray(aPHI);  ShiftArray(aPhi1);  ShiftArray(aPhi2);
#ifdef NOISE
      aX[0] = cos(_k*vW0)+((random(2000)/1000.0)-1.0)*noisefactor;
#else
      aX[0] = cos(_k*vW0);
#endif
      Function_A0();  /* เลือกคำนวณ Alpha 0 และ 1 ก่อน เพราะมันใช้ค่า param*/
      Function_A1();  /* ที่เป็นอดีตเท่านั้น                          */
      Function_PHI();
      Function_Phi2();
      Function_Y();  /* คำนวณ output Y */
      Function_Phi1();  /* Phi1 มีการใช้ค่า y ปัจจุบัน ต้องคำนวณหลัง Y */
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        // plot Alpha 1 ————————————————
        mainwin.SetViewport(MINX,MINY,WIDTH,HEIGHT,TRUE);
        gpx.Line((FTP)_k-1,aOutput[_k-1],(FTP)_k,aOutput[_k]=aA1[0]);

        // plot Alpha 0 หน้าต่างเดียวกับ alpha 1 ————————
//      setcolor(MAGENTA);  gpx.Line((FTP)_k-1,aA0[1],(FTP)_k,aA0[0]);

        // plot Input และ Output ————————————————
        mainwin.SetViewport(MINXX,MINYX,WIDEX,HIGHX,TRUE);
        setcolor(LIGHTGRAY);  gpx_x.Line((FTP)_k-1,aX[1],(FTP)_k,aX[0]);
        setcolor(BLUE);  gpx_x.Line((FTP)_k,0.0,(FTP)_k,aX[0]);

        mainwin.SetViewport(MINXY,MINYY,WIDEY,HIGHY,TRUE);
        setcolor(LIGHTGRAY);  gpx_y.Line((FTP)_k-1,aY[1],(FTP)_k,aY[0]);
        setcolor(BLUE);  gpx_y.Line((FTP)_k,0.0,(FTP)_k,aY[0]);
        delay(delayindex);

        if (_miny > aOutput[_k]) _miny = aOutput[_k];
        if (_maxy < aOutput[_k]) _maxy = aOutput[_k];
    }
  } while(_k != MAXOUTPUT && !kbhit());
  mainwin.UseWindowScope(TRUE);
}

//————————————————————————————
void InitAllObjects(void)
{
  iMAXOUTPUT.AdjustAttr(LIGHTBLUE,WHITE);

  iMAXOUTPUT.AdjustPosition(textw+100, 5);  iMAXOUTPUT.AdjustThick(3);
  iMU.AdjustPosition(textw+100,25);  iMU.AdjustThick(3);
  iMU0.AdjustPosition(textw+100,45);  iMU0.AdjustThick(3);
  iW0.AdjustPosition(textw+100,65);  iW0.AdjustThick(3);
  iNoiseDB.AdjustPosition(textw+100,85);  iNoiseDB.AdjustThick(3);
  iA0MIN.AdjustPosition(textw+270,25);  iA0MIN.AdjustThick(3);
  iA0MAX.AdjustPosition(textw+270,45);  iA0MAX.AdjustThick(3);
  idelay.AdjustPosition(textw+300,85);  idelay.AdjustThick(3);
  iMU.inputfn=iMU0.inputfn=iW0.inputfn=iNoiseDB.inputfn=idelay.inputfn=TextInputFunction;
  iA0MIN.inputfn=iA0MAX.inputfn=TextInputFunction;
  sscanf((char *)idelay, "%d", &delayindex);
  sscanf((char *)iA0MIN, "%" FTP_PREFIX "f", &A0MIN);
  sscanf((char *)iA0MAX, "%" FTP_PREFIX "f", &A0MAX);

  bPlotGraph.AdjustPosition(MRIGHT-300, 5,80,texth+10);
  bPlotGraph.AssignText("Plot graph");
  bPlotGraph.AdjustJustify(CHJUSTIFY|CVJUSTIFY);
  bPlotGraph.ReleaseButton();

  bSave.AdjustPosition(MRIGHT-200, 5,80,texth+10);
  bSave.AssignText("Save");
  bSave.AdjustJustify(CHJUSTIFY|CVJUSTIFY);
  bSave.ReleaseButton();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    bExit.AdjustPosition(MRIGHT-100, 5,80,texth+10);
    bExit.AssignText("Exit");
    bExit.AdjustJustify(CHJUSTIFY|CVJUSTIFY);
    bExit.ReleaseButton();

    gpx.AdjustGraphSize(0,0,WIDTH,HEIGHT);
    gpx_x.AdjustGraphSize(0,0,WIDEX,HIGHX);
    gpx_y.AdjustGraphSize(0,0,WIDEY,HIGHY);

    max_x = 1;   /* ใส่ไว้ชั่วคราว จะต้องถูกเปลี่ยนแปลงโดยการป้อนมุม */
    gpx_x.xzoomfactor = WIDEX/MAXGXX;   gpx_y.xzoomfactor = WIDEY/MAXGXY;
    gpx_x.yzoomfactor = HIGHX/MAXGYX;   gpx_y.yzoomfactor = HIGHY/MAXGYY;
}

//-------------------------------------------
void ShowGetParametersDialog(void)
{
    EDGECOLORPACK _ep;

    _ep.Up = _ep.Left = DARKCOLOR;   _ep.Down = _ep.Right = HIGHCOLOR;

    // clear Main Window ---------------------
    mainwin.UseWindowScope(FALSE);
    WaitVertRetrace();   mainwin.Paint();
    mainwin.UseWindowScope(TRUE);

    // เขียน Legend ---------------------------
    setcolor(YELLOW);
    outtextxy(MINXX+WIDEX/2,MINYX+HIGHX+textheight("k"),"k");
    outtextxy(MINXY+WIDEY/2,MINYY+HIGHY+textheight("k"),"k");
    outtextxy(MINX+WIDTH/2,MINY-textheight("k"),"k");
    outtextxy(MINX-textwidth("u1(k)"),MINY+HEIGHT/2, "u1(k)"); /* พิมพ์ชื่อตัวแปรที่แกน Y */
    outtextxy(MINXX-textwidth("Input"),MINYX+HIGHX/2, "Input");
    outtextxy(MINXY-textwidth("Output"),MINYY+HIGHY/2, "Output");

    mainwin.OutTextat(textw, 5,"MAXOUTPUT");
    mainwin.OutTextat(textw,25,"Mu");
    mainwin.OutTextat(textw,45,"Mu0");
    mainwin.OutTextat(textw,65,"W0");
    mainwin.OutTextat(textw,85,"S/N (db)");
    mainwin.OutTextat(textw+200,25,"A0 min");
    mainwin.OutTextat(textw+200,45,"A0 max");
    mainwin.OutTextat(textw+200,85,"Delay");

    iMAXOUTPUT.Paint();
    iMU.Paint();   iMU0.Paint();   iW0.Paint();   iNoiseDB.Paint();
    iA0MIN.Paint();   iA0MAX.Paint();
    idelay.Paint();
    bPlotGraph.Paint();   bSave.Paint();   bExit.Paint();

    // วาดพื้นที่สำหรับวาดกราฟให้ gpx ทั้งหลาย ----------
    LineStrip(MINX,MINY,WIDTH,HEIGHT,_ep,BLACK,WHITE,LIGHTGRAY,BTHICK);
    LineStrip(MINXX,MINYX,WIDEX,HIGHX,_ep,BLACK,WHITE,LIGHTGRAY,BTHICK);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  LineStrip(MINXY,MINYY,WIDEY,HIGHY,_ep,BLACK,WHITE,LIGHTGRAY,BTHICK);
}

/*-------------------------------------*/
int ActivateInput(void)
{
  int _x,_y;

  if (KeyboardCmd == (int)CMD_MOUSE)
  {
    if (MStat.butt & MOUSELBIT)  /* ถ้า mouse กดปุ่มซ้าย */
    {
      _x = MStat.x;  _y = MStat.y;
    }
    else
    {
      _x=_y=-1;
    }
  }
  else
  {
ChkKeyboardCmd::
    _x=_y=-1;
    switch(KeyboardCmd)
    {
    case 0x010F: /* Shift-TAB */
    case 0x09:  /* TAB */
      // ลบ cursor ปัจจุบัน ----------------------
      setcolor(BLUE);  /* ใช้สีพื้นของ mainwin */
      switch(currentobj)
      {
      case OBJ_PLOTGRAPH:
        rectangle(bPlotGraph.Left()-3,bPlotGraph.Top()-3,
            bPlotGraph.Right()+3,bPlotGraph.Bottom()+3);
        break;
      case OBJ_SAVE:
        rectangle(bSave.Left()-3,bSave.Top()-3,
            bSave.Right()+3,bSave.Bottom()+3);
        break;
      case OBJ_EXIT:
        rectangle(bExit.Left()-3,bExit.Top()-3,
            bExit.Right()+3,bExit.Bottom()+3);
        break;
      }
NextField::
      if (KeyboardCmd == 0x09)
      {
        currentobj++;  /* เลื่อน cursor ไปตำแหน่งถัดไป */
        if (currentobj == OBJ_NULL)  currentobj = OBJ_TOP;
      }
      else
      {
        if (currentobj==OBJ_TOP)  currentobj = OBJ_NULL;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        currentobj--;
    }

    setcolor(YELLOW);
    switch(currentobj)
    {
    case OBJ_PLOTGRAPH:
        rectangle(bPlotGraph.Left()-3,bPlotGraph.Top()-3,
                bPlotGraph.Right()+3,bPlotGraph.Bottom()+3);
        break;
    case OBJ_SAVE:
        rectangle(bSave.Left()-3,bSave.Top()-3,
                bSave.Right()+3,bSave.Bottom()+3);
        break;
    case OBJ_EXIT:
        rectangle(bExit.Left()-3,bExit.Top()-3,
                bExit.Right()+3,bExit.Bottom()+3);
        break;
    case OBJ_IMU:
        _x = textw+100;  _y = 25;  break;
    case OBJ_IMU0:
        _x = textw+100;  _y = 45;  break;
    case OBJ_IW0:
        _x = textw+100;  _y = 65;  break;
    case OBJ_INOISEDB:
        _x = textw+100;  _y = 85;  break;
    case OBJ_IA0MIN:
        _x = textw+270;  _y = 25;  break;
    case OBJ_IA0MAX:
        _x = textw+270;  _y = 45;  break;
    case OBJ_DELAY:
        _x = textw+300;  _y = 85;  break;
    }
    break;

case 0x0D:  /* ENTER */
    switch(currentobj)
    {
    case OBJ_PLOTGRAPH:
        _x = bPlotGraph.Left();  _y = bPlotGraph.Top();
        break;

    case OBJ_SAVE:
        _x = bSave.Left();  _y = bSave.Top();
        break;

    case OBJ_EXIT:
        _x = bExit.Left();  _y = bExit.Top();
        break;
    default:
        KeyboardCmd = 0x09;  /* ทำหน้าที่เป็น TAB แทน */
        goto NextField;
    }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        break;
      }
    }

    if (iMU.Touch(_x,_y))
    {
      iMU.Getstring();
      sscanf((char *)iMU, "%" FTP_PREFIX "f", &vMU);

      goto ChkKeyboardCmd;
    }
    else
    if (iMU0.Touch(_x,_y))
    {
      iMU0.Getstring();
      sscanf((char *)iMU0, "%" FTP_PREFIX "f", &vMU0);

      goto ChkKeyboardCmd;
    }
    else
    if (iW0.Touch(_x,_y))
    {
      iW0.Getstring();
      sscanf((char *)iW0, "%" FTP_PREFIX "f", &vW0);
      max_x = SINENUM*2*PI/vW0;
      gpx_x.xzoomfactor = WIDEX/MAXGXX;  gpx_y.xzoomfactor = WIDEY/MAXGXY;

      goto ChkKeyboardCmd;
    }
    else
    if (iNoiseDB.Touch(_x,_y))
    {
      iNoiseDB.Getstring();
      sscanf((char *)iNoiseDB, "%" FTP_PREFIX "f", &noise_db);

      goto ChkKeyboardCmd;
    }
    else
    if (iA0MIN.Touch(_x,_y))
    {
      iA0MIN.Getstring();
      sscanf((char *)iA0MIN, "%" FTP_PREFIX "f", &A0MIN);

      goto ChkKeyboardCmd;
    }
    else
    if (iA0MAX.Touch(_x,_y))
    {
      iA0MAX.Getstring();
      sscanf((char *)iA0MAX, "%" FTP_PREFIX "f", &A0MAX);

      goto ChkKeyboardCmd;
    }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    else
    if (idelay.Touch(_x,_y))
    {
      idelay.Getstring();
      sscanf((char *)idelay, "%d", &delayindex);

      goto ChkKeyboardCmd;
    }
    else
    if (bPlotGraph.Touch(_x,_y))
    {
      bPlotGraph.PushButton();   bPlotGraph.Paint();
      ProcessingGraph();
      bPlotGraph.ReleaseButton();   bPlotGraph.Paint();
    }
    else
    if (bSave.Touch(_x,_y))
    {
      bSave.PushButton();   bSave.Paint();
      Save2File();
      bSave.ReleaseButton();   bSave.Paint();
    }
    else
    if (bExit.Touch(_x,_y))
    {
      bExit.PushButton();
      bExit.Paint();
      return FALSE;
    }
    return TRUE;
}

//-------------------------------------
void Save2File(void)
{
  WORD _k,_i;
  FILE *_fhandle,*_hhandle;

  if ((_hhandle = fopen(DATAFILENAME ".Hdr","wt")) != NULL)
  {
    // เขียนจำนวนสูงที่สุด และจำนวนภาพที่ส่วนหัว .Hdr ------
    fprintf(_hhandle,"%u %d\n", MAXOUTPUT, graphidx);
    fclose(_hhandle);

    // เขียนข้อมูลจริงๆไว้ที่ .Dat ----------------
    if ((_fhandle = fopen(DATAFILENAME ".Dat","wt")) != NULL)
    {
      for(_k=0;_k<MAXOUTPUT;_k++)
      {
        fprintf(_fhandle,"\n%d ",_k);
        for(_i=0;_i<graphidx;_i++)
          fprintf(_fhandle,"%" FTP_PREFIX "g ",
            ((FTP_huge *)aeOutput[_i])[_k]);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        }
        fclose(_fhandle);
      }
    }
  }

/*-------------------------------------*/
void AdaptiveG(void)
{
  int _ef;

  KeyboardCmd = CMD_NOINPUT;
  currentobj = OBJ_IMU;
  graphidx = 0;

  randomize();
  initAllObjects();
  ShowGetParametersDialog();

  // get MAXOUTPUT เป็นอันดับแรก ----------
  do {
    _ef = FALSE;
    iMAXOUTPUT.Getstring();
    sscanf((char *)iMAXOUTPUT, "%u", &MAXOUTPUT);

    if (!MAXOUTPUT) return;  /* ออกจากโปรแกรมเมื่อ MAXOUTPUT = 0 */

    // เราไม่ใช้ new operator เพราะว่ามันจองได้ไม่เกิน 64 KB ----
    aaOutput[graphidx] = (FTP *)farcalloc(MAXOUTPUT,sizeof(FTP));
    _ef = (aaOutput[graphidx] == NULL);
  } while(_ef);
  aOutput = aaOutput[graphidx++];
  iMAXOUTPUT.AdjustAttr(DARKGRAY,WHITE);
  iMAXOUTPUT.Paint();

  // ทำการ scale ให้พอเหมาะและเปลี่ยนจุด Origin ----------*/
  gpx.xzoomfactor = WIDTH/MAXGRAPHX;
  gpx.yzoomfactor = HEIGHT/MAXGRAPHY;
  gpx.MoveOrigin(0,20/gpx.yzoomfactor);

  // ตารางสำหรับ gpx ----------
  mainwin.SetViewport(MINX,MINY,WIDTH,HEIGHT,TRUE);
  setcolor(GREEN);
  gpx.DrawGridxy(MAXOUTPUT/20,0.1);  /* ความละเอียดของจอภาพด้วยจำนวนกริดที่ต้องการ */

  setcolor(BLACK);
  gpx.DisplayAxis();

  setcolor(RED);
  gpx.DrawNumberxy(MAXOUTPUT/10,0.1,0,1); /* เขียนตัวเลขที่กริด */

  mainwin.UseWindowScope(TRUE);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    // run program loop ----------
    do {
      WaitInput();
    } while(ActivateInput());

    while(graphidx)
      farfree(aaOutput[--graphidx]);

    mainwin.UseWindowScope(FALSE);
}
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้