

บทความเรื่อง

**โปรแกรมวิเคราะห์สเปกตรัมสำหรับการทดสอบโมเดล
(SPECTRUM ANALYSIS PROGRAM FOR MODAL TESTING)**

โดย

ดร. วิบูลย์ แสงวีระพันธุ์ศิริ
นาย กิติ ผดุงชีวิต
ภาควิชาวิศวกรรมเครื่องกล คณะวิศวกรรมศาสตร์
จุฬาลงกรณ์มหาวิทยาลัย

โปรแกรมวิเคราะห์สเปกตรัมสำหรับการทดสอบโมเดล
(SPECTRUM ANALYSIS PROGRAM FOR MODAL TESTING)

บทคัดย่อ ปัจจุบันนี้อุปกรณ์ที่ใช้วิเคราะห์สเปกตรัมนั้นราคาแพงมากและยากต่อการบำรุงรักษา อุปกรณ์เหล่านี้เป็นอุปกรณ์ที่ใช้วิเคราะห์ในลักษณะ real time ซึ่งการคำนวณหาค่าต่าง ๆ ใช้ Hardware เป็นตัวทำการคำนวณเกือบทั้งหมด โปรแกรมวิเคราะห์สเปกตรัมที่จะพัฒนาขึ้นนี้แทนที่จะใช้ Hardware ทำการคำนวณ ก็จะใช้โปรแกรมทำการคำนวณแทนโดยอาศัย routines ต่าง ๆ ที่พัฒนาขึ้นจากภาษา C โปรแกรมนี้พัฒนาบนเครื่องตระกูล IBM PC โดยจะมีตัว Hardware A/D converter ซึ่งทำหน้าที่รับสัญญาณจากตัว Transducers หรือ Sensors ต่าง ๆ เพื่อแปลงให้เป็นสัญญาณดิจิทัลเพื่อนำไปคำนวณหาสเปกตรัมอีกทีหนึ่ง ซึ่งผลที่ได้นั้นจะแสดงได้บนจออยู่ในรูปของ Frequency response function แล้วนำผลที่ได้นี้ไปวิเคราะห์หาค่าตัวพารามิเตอร์ของระบบที่กำลังสนใจได้

คำนำ

ปัจจุบันนี้อุปกรณ์ที่ใช้วิเคราะห์สเปกตรัมนั้นราคาแพงมากและยากต่อการบำรุงรักษา อุปกรณ์เหล่านี้เป็นอุปกรณ์ที่ใช้วิเคราะห์ในลักษณะ Real Time ซึ่งการคำนวณค่าต่าง ๆ ที่เกี่ยวข้องกับค่าสเปกตรัมนั้นใช้ hardware เป็นตัวทำการคำนวณเกือบทั้งหมด เครื่องวิเคราะห์สเปกตรัมที่มีขายอยู่ในตลาดเมืองไทยราคาไม่ต่ำกว่าล้านบาทขึ้นไปทั้งนั้น ที่สำคัญก็คือการบำรุงรักษาและการซ่อมแซมเมื่อเกิดปัญหานั้นไม่เป็นที่น่าพอใจ เพราะขาดผู้รู้ที่มีความเข้าใจใน hardware นั้น ๆ

โปรแกรมวิเคราะห์สเปกตรัมที่พัฒนานี้ แทนที่จะใช้ hardware ทำการคำนวณ ก็จะใช้โปรแกรมทำการคำนวณแทนและ routines ต่าง ๆ ที่เกี่ยวข้องกับค่าสเปกตรัมเพื่อหาสเปกตรัมก็สามารถพัฒนาและเขียนขึ้นเองทั้งหมด

โปรแกรมนี้อาจพัฒนาบนเครื่องระดับบุคคลในตระกูล IBM PC/PC Compatible โดยจะมีตัว Hardware A/D converter ซึ่งอาจจะสร้างขึ้นเอง แต่ตัว IC ที่ทำหน้าที่ในการเปลี่ยนสัญญาณ analog ที่ได้รับจาก transducer ไปเป็นสัญญาณ digital ที่คอมพิวเตอร์รู้จัก เป็นส่วนที่ต้องจัดหาซื้อมาเพิ่มเข้าไปกับระบบคอมพิวเตอร์ที่มีอยู่ หรือจะซื้อเป็นแบบแผงวงจรสำเร็จรูปโดยจะต้องวางเข้ากับ BUS ของเครื่องตระกูล IBM PC/PC Compatible ตัว A/D converter นี้จะเป็นตัวอ่านข้อมูลจาก transducer ซึ่งปกติเป็นสัญญาณไฟฟ้าอยู่ในรูปของความต่างศักย์ เมื่อตัว A/D converter รับสัญญาณที่อยู่ในรูปของความต่างศักย์แล้วก็จะแปลงสัญญาณนั้นให้อยู่ในรูปของดิจิทัลแล้วจะนำข้อมูลที่เป็นดิจิทัลนั้นไปเก็บชั่วคราวไว้ในหน่วยความจำก่อน หลังจากเสร็จสิ้นการอ่านข้อมูลจาก transducer แล้ว ก็จะนำข้อมูลดิจิทัลที่อยู่ในหน่วยความจำนั้นไปเก็บไว้ในไฟล์ในแผ่นดิสเกตอีกทีหนึ่ง หลังจากนั้นโปรแกรมวิเคราะห์สเปกตรัมก็จะนำข้อมูลที่เก็บไว้ในไฟล์นั้นนำมาวิเคราะห์หาสเปกตรัม ซึ่งผลที่ได้นั้นจะแสดงไว้บนจออยู่ในรูปของ Frequency Response Function แล้วนำผลที่ได้ไปวิเคราะห์หาค่าตัวพารามิเตอร์ของระบบที่กำลังสนใจได้

จะเห็นว่าราคาของระบบนี้จะถูกกว่าเครื่องวิเคราะห์สเปกตรัมอย่างมากมาย และสามารถดัดแปลงเพิ่มเติมได้ ปัจจุบันนี้เครื่องตระกูล IBM PC/PC Compatible มีราคาไม่แพงมากนักและซื้อได้ไม่ยาก ส่วนแผงวงจรเปลี่ยนสัญญาณ A/D Converter นั้นก็พอหาได้ส่วนราคาของแผงวงจรชนิดนี้ก็ขึ้นอยู่กับความเร็วในการอ่านและบันทึกข้อมูลนอกจากนี้ความละเอียดที่สามารถอ่านได้ก็มีส่วนสำคัญไม่ใช่น้อย อุปกรณ์ A/D Converter ที่หาซื้อได้ในขณะนี้ความละเอียดในการอ่านข้อมูลจะเป็นตัวหนึ่งที่จำกัดความเร็วในการอ่านและแปลงข้อมูล

ส่วนสำคัญของโปรแกรม

การหาเพอร์เวอร์สเปกตรัมเด้นซิตี (Power Spectral Density) ซึ่งอธิบายง่าย ๆ ก็คือการหาค่าสเปกตรัมของข้อมูลที่สุ่มมาจากขบวนการทาง deterministic และ stochastic นั้นสามารถหาได้โดยอาศัยการหาฟูเรียร์ทรานสฟอร์ม (Fourier Transform) ของข้อมูลที่สุ่มมาได้นั้น การหาฟูเรียร์ทรานสฟอร์มโดยตรงนั้นกินเวลาคำนวณค่อนข้างนาน เพราะฉะนั้นจึงจำเป็นต้องอาศัยเทคนิค Fast Fourier Transform (FFT) เข้ามาช่วย การใช้วิธีนี้ทำให้การคำนวณเร็วขึ้นผลที่ได้รับเมื่อข้อมูลที่ให้มีจำนวนมากก็ได้ผลดีพอสมควร อย่างไรก็ตามวิธี FFT นี้ก็มีขอบเขตจำกัดอยู่หลายอย่างเหมือนกัน ข้อจำกัดที่สำคัญที่สุดก็คือ ความละเอียดของการอ่านค่าความถี่ ซึ่งจะขึ้นอยู่กับส่วนกลับของเวลาที่ใช้ในการสุ่มข้อมูล ถ้าต้องการความละเอียดทางด้านความถี่สูง ก็จำเป็นต้องใช้ช่วงเวลาในการวัดข้อมูลยาวด้วย ข้อจำกัดต่อไปก็คือการทำ FFT นั้น ข้อมูลที่ใส่เข้าไปนั้นเป็นข้อมูลที่สุ่มได้ภายในช่วงเวลาหนึ่งเท่านั้น ซึ่งถ้าเป็นการทำฟูเรียร์ทรานสฟอร์มตามทฤษฎีแล้ว ช่วงเวลาที่วัดข้อมูลนั้นควรจะต้องนานมากจนถึงค่าอนันต์ (Infinity) โดยทางปฏิบัติแล้วไม่สามารถทำเช่นนั้นได้ การจำกัดช่วงเวลาในการวัดข้อมูลก็เหมือนกับการทำ windowing นั่นเอง โดยชนิดของ window เป็นชนิดสี่เหลี่ยมผืนผ้า (Rectangular Window) การทำ windowing นี้ทำให้เกิดการรั่วไหลของสเปกตรัมจาก mainlobe ไปทางด้านข้างหรือ sidelobe เมื่อเป็นเช่นนี้จึงอาจจะจำเป็นต้องเลือก window ชนิดอื่นที่เหมาะสมเช่น แบบสามเหลี่ยม (Triangular window) แบบ Hanning window แบบ Hamming window เป็นต้น แต่อย่างไรก็ตามก็จำเป็นที่จะต้องคำนึงถึงผลกระทบที่มีต่อค่าความละเอียดที่สามารถอ่านได้ด้วย อัตราการสุ่มข้อมูลก็มีความสำคัญมาก ซึ่งให้ถือกฎเกณฑ์ของ Nyquist Frequency ถ้าถือเอา Nyquist Frequency เป็นหลักแล้วก็จำเป็นจะต้องสามารถประมาณค่าความถี่สูงสุดของสัญญาณที่จะทำการวัด แล้วให้อัตราความถี่ของการสุ่มข้อมูลอย่างน้อยต้องไม่ต่ำกว่าสองเท่าของความถี่สูงสุดของสัญญาณนี้ เพื่อป้องกันการเกิด Aliasing ขึ้นซึ่งจะทำให้สัญญาณที่วัดนั้นผิดเพี้ยนไป ในบางครั้งอาจจะต้องมีวงจร Lowpass filter ที่ต่อระหว่างสัญญาณจาก transducer กับแผงวงจร A/D converter

เนื่องจากสัญญาณทางด้านขาเข้าระบบอาจจะเป็น random signals ซึ่งจะทำให้สัญญาณด้านขาออกจากระบบเป็น random signals ด้วย โดยมีค่าความเฉลี่ยเป็นศูนย์ แต่ในทางปฏิบัติแล้วอาจจะไม่เป็นเช่นนั้นซึ่งเป็นผลให้เกิด bias ขึ้นกับการหาค่าสเปกตรัม ดังนั้นจึงอาจจะต้องมีการแก้ bias นี้ด้วย

โครงสร้างของโปรแกรม

ขั้นตอนของการหาสเปกตรัมนั้นสามารถแบ่งออกเป็นขั้นตอนได้ดังต่อไปนี้

1. ในขั้นแรกจำเป็นต้องกะประมาณค่า bandwidth ของสัญญาณที่จะทำการหาสเปกตรัมเสียก่อน เพื่อจะได้ทราบว่าควรจะใช้ช่วงเวลาในการสุ่ม (Sampling Period) เท่าไร โดยอาศัย Nyquist Frequency เป็นหลัก
2. ต้องกำหนดค่าความละเอียดที่ต้องการอ่านความถี่ (Frequency Resolution) เพื่อที่จะสามารถประมาณช่วงเวลาในการวัดสัญญาณ ในขั้นตอนนี้อาจจะกำหนดจำนวนข้อมูลแน่นอนลงไปเลยก็ได้ แล้วช่วงเวลาในการวัดก็หาได้จากผลคูณของช่วงเวลาในการสุ่มข้อมูลแต่ละครั้งกับจำนวนข้อมูลที่สุ่ม และค่าความละเอียดของการอ่านค่าความถี่ก็ขึ้นอยู่กับส่วนกลับของช่วงเวลาในการวัดนี้ วิธีนี้เป็นวิธีที่เครื่องสเปกตรัมใช้กันอยู่
3. การทำ Windowing เพื่อป้องกันไม่ให้สเปกตรัมรั่วไหลออกจาก mainlobe มากเกินไป ซึ่งอาจจะเลือกชนิดของ window คือ Triangular, Hanning, Hamming หรือ Blackman and Tucky เป็นต้น
4. การแก้ไขปัญหา bias โดยการหาค่าเฉลี่ยของสัญญาณ แล้วนำค่าเฉลี่ยนี้มาปรับสัญญาณเพื่อให้สัญญาณมีค่าความเฉลี่ยเป็นศูนย์
5. หาฟูเรียร์ทรานสฟอร์มของสัญญาณทั้งขาเข้าและขาออกจากระบบ โดยใช้วิธี FFT
6. Transfer Function ก็สามารถหาได้จากค่าฟูเรียร์ทรานสฟอร์มของสัญญาณทั้งขาเข้าและขาออกจากระบบที่กำลังสนใจอยู่
7. ผลที่ได้จากข้อ 6 อาจจะต้องมีการทำ smoothing เพื่อให้ได้ผลดีขึ้น
8. ผลที่ได้สามารถแสดงออกได้ทั้งจอภาพหรือส่งออกไปยังเครื่องพิมพ์ที่ต่ออยู่ก็ได้

Flowchart ของโปรแกรมของการหาสเปกตรัมแสดงไว้ในรูปที่ 1 โดยที่

$$f(t) = \text{input signal}$$

$$y(t) = \text{output signal}$$

$$F(w) = \text{fourier transform of } f(t)$$

$$Y(w) = \text{fourier transform of } y(t)$$

$$G_{y,y} = \text{fourier transform of autocorelation of } y(t)$$

$$G_{f,f} = \text{fourier transform of autocorelation of } f(t)$$

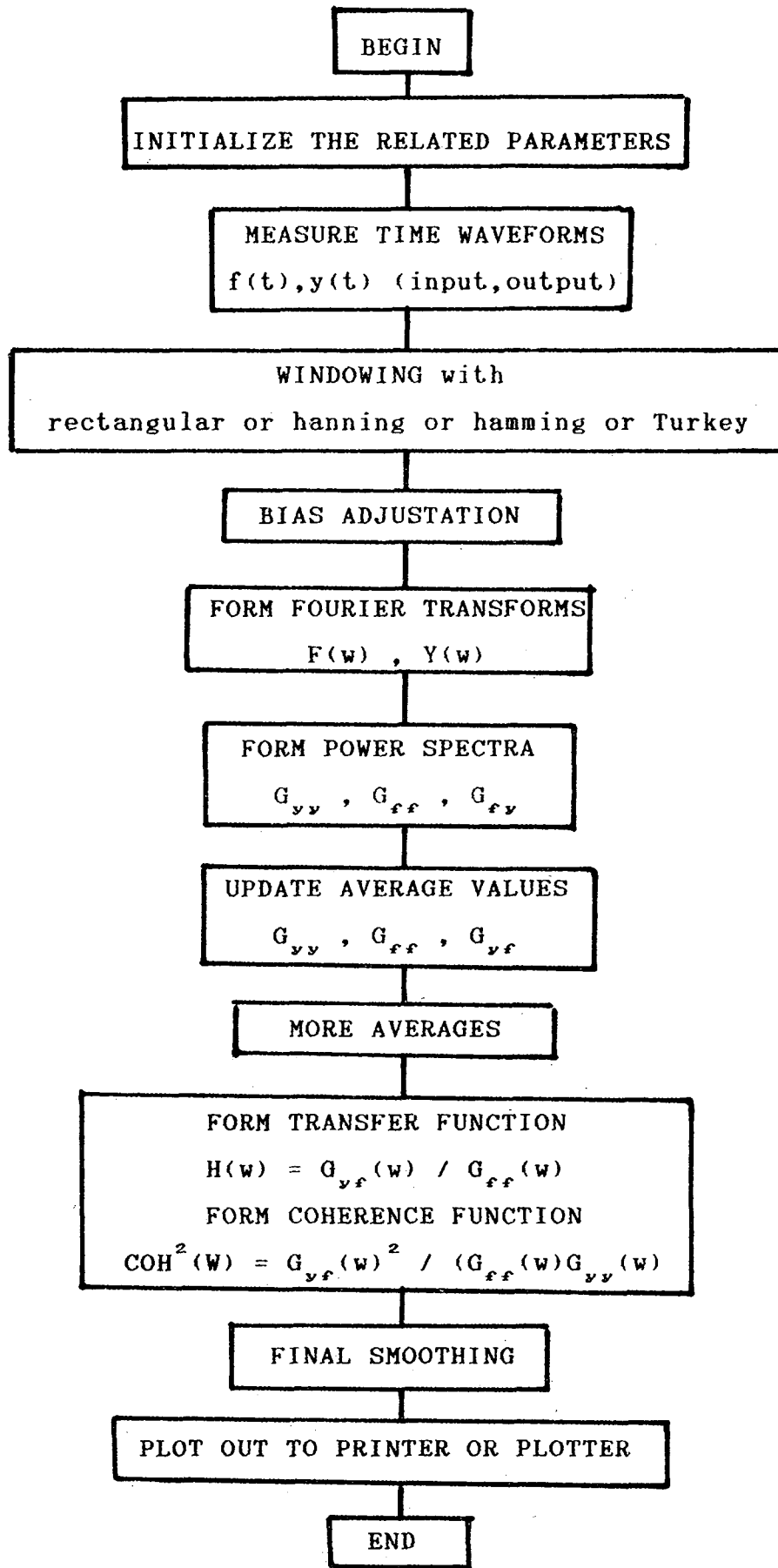
$$G_{f,y} = \text{fourier transform of crosscorelation of } y(t) \text{ and } f(t)$$

$$H(w) = \text{transfer function}$$

$$\text{COH} = \text{coherence function}$$

การทำงานของโปรแกรมวิเคราะห์หาสเปกตรัมมีดังนี้คือ

- 1) เก็บข้อมูลที่วัดได้จาก Transducer โดยแปลงสัญญาณที่วัดได้ซึ่งเป็นสัญญาณ analog ให้เป็นสัญญาณ digital เสียก่อนแล้วจึงนำข้อมูล



- นี้เก็บเข้าไปเก็บไว้ในแฟ้มข้อมูลในดิสเกจ เนื่องจากคาบเวลาในการอ่านหรือสุ่ม (sampling) ค่อนข้างเร็ว ฉะนั้นจึงต้องการความเร็วในการอ่านข้อมูลค่อนข้างสูง การจะเก็บข้อมูลโดยส่งไปเก็บไว้ที่ดิสเกจโดยตรงนั้นอาจจะไม่เร็วพอ จึงจำเป็นต้องเก็บค่าสัญญาณที่อ่านได้ไว้ในหน่วยความจำก่อน แล้วจึงนำลงไปเก็บไว้ในแฟ้มข้อมูลในแผ่นดิสเกจอีกทีหนึ่ง หลังจากช่วงการเก็บข้อมูลแล้วนั้น ก็สามารถแสดงผลที่อ่านได้เป็นกราฟโดยแสดงได้ทั้งแอมพลิจูดเพื่อตรวจสอบข้อมูลขั้นต้นเสียก่อน
- 2) นอกจากการอ่านสัญญาณจากตัว Transducer แล้ว เรายังสามารถสร้างสัญญาณขึ้นมาก็ได้ สัญญาณที่สร้างนั้นก็มาจากแบบจำลองทางคณิตศาสตร์ที่อยู่ในรูปของสมการดิฟเฟอเรนเชียลเชิงเส้น แล้วก็ใช้วิธีการทาง Numerical Method มาทำการแก้สมการเหล่านี้
 - 3) สัญญาณที่วัดเก็บไว้นั้นอาจจะเป็นเฉพาะสัญญาณขาออกจากระบบอย่างเดี่ยว (Single Channel) หรืออาจจะเป็นสัญญาณทั้งขาเข้าและขาออกจากระบบ (Two Channel) ที่กำลังสนใจก็ได้ ขึ้นอยู่กับจุดประสงค์ของการทำการทดลอง สัญญาณเหล่านี้จะอยู่ในรูปของ time domain ฉะนั้นจึงจำเป็นต้องเปลี่ยนให้อยู่ในรูปของ frequency domain โดยแปลงให้เป็นอนุกรมฟูริเยร์ (Fourier Series) โดยอาศัยวิธีการ Fast Fourier Transform (FFT) ซึ่งผลลัพธ์ที่ได้นั้นมีลักษณะเป็นค่า Complex numbers โดยอาจจะแสดงผลอยู่ในรูปของ magnitue กับ phase หรือ real part กับ imaginary part ก็ได้
 - 4) ข้อมูลสัญญาณที่อยู่ในรูปของ frequency domain ซึ่งได้จากการทำ Fast Fourier Transform นั้นสามารถนำมาแสดงผลเป็นกราฟบนจอภาพได้
 - 5) การวิเคราะห์หาสมการแบบจำลองทางคณิตศาสตร์ที่แทนระบบที่กำลังสนใจและต้องการทราบนั้นสามารถหาได้จากค่า real part กับ imaginary part ของ spectrum ที่ได้จากการทำ fourier transform ของสัญญาณขาเข้าและขาออกที่วัดได้จากระบบที่กำลังพิจารณาอยู่ การวิเคราะห์หาสมการทางคณิตศาสตร์ที่แทนระบบ หรือที่เรียกว่า System Identification นั้นใช้วิธีการที่เรียกว่า Instrumental Variable (IV)
 - 6) นอกจากนั้นแล้วค่า real part กับ imaginary part ของ spectrum นั้นก็สามารถนำมาทำการวิเคราะห์ เพื่อหาความน่า

เชื่อถือของการวัดข้อมูลได้โดยใช้วิธีการที่เรียกว่า Circular Curvefit

รายละเอียดเกี่ยวกับโปรแกรม

รายละเอียดเกี่ยวกับโปรแกรมที่เขียนขึ้นตั้งแต่เริ่มโครงการมาจนถึงปัจจุบันนี้ก็พอสรุปได้ดังนี้คือ

รูปที่ 2 เป็นรูปแสดงรายการหลักหรือ Main Menu ซึ่งมีให้เลือกด้วยกันทั้งหมด 7 หัวข้อ ถ้าเลือกหัวข้อที่ 1 จะเป็นการเก็บหรือวัดข้อมูลของสัญญาณที่วัดจาก transducers ที่ทางด้านขาเข้าและขาออกจากระบบ เมื่อเลือกข้อหนึ่งนี้แล้วจะเห็นว่า หน้าจอจะเปลี่ยนแปลงไปตามรูปที่ 4 ซึ่งจะมีคำสั่งต่าง ๆ มากมายที่เราสามารถเรียกใช้ เนื่องจากในส่วนนี้ได้เขียนไว้เพื่อให้อ่านและส่งข้อมูลโดยผ่าน A/D converter และ D/A converter เพื่อประโยชน์ในการเพิ่มเติมความสามารถของโปรแกรมนี้ต่อไปในภายภาคหน้า โปรแกรมตรงส่วนนี้เขียนขึ้นโดยใช้ภาษา Basic ในที่นี้จะกล่าวเฉพาะคำสั่งต่าง ๆ ที่เกี่ยวข้องในตอนนี้อย่างที่ได้นำมาซึ่งก็ได้แก่คำสั่งเหล่านี้คือ

- CH ใช้เลือก channel ที่จะอ่านซึ่งมีได้ถึง 8 channels ในแบบ differential end
- XSA ใช้สำหรับ reset or setup วงจรที่ใช้อ่านข้อมูลสัญญาณ
- XSCD ใช้สำหรับตั้งความเร็วในการ sampling data เนื่องจากวงจรที่จัดหามานี้มีความเร็วสูงสุดในการสุ่มข้อมูลอยู่ในช่วงไม่เกิน 28500 Hz. ฉะนั้นค่าตัวเลขนี้จึงควรจะน้อยกว่าค่าความถี่สูงสุดของแผงวงจรที่ใช้แปลงข้อมูลจาก analog ไปเป็น digital
- NM ใช้กำหนดจำนวนข้อมูลที่ต้องการอ่าน
- XAS เป็นคำสั่งที่ใช้สำหรับเริ่มการเก็บข้อมูลสัญญาณที่ผ่านเข้ามาที่ A/D converter
- GR เป็นคำสั่งที่ใช้ตรวจสอบข้อมูลที่วัดได้เพียงคร่าว ๆ เสียก่อนโดยข้อมูลจะแสดงออกมาอยู่ในรูปของกราฟ ดังแสดงตัวอย่างในรูปที่ 3

หมายเหตุ คำสั่งอื่น ๆ ยังไม่จำเป็นที่จะต้องใช้ในตอนนี้

ถ้าเลือกหัวข้อที่ 3 ของรายการหลัก จะเป็นการหาอนุกรมฟูเรียร์ของข้อมูลสัญญาณที่วัดได้ หลังจากเลือกหัวข้อที่ 3 นี้แล้ว จะมีรายการใหม่ปรากฏขึ้นบนจอแสดงในรูปที่ 5 ซึ่งจำเป็นที่จะต้องใส่ข้อมูลบางประการเพื่อประกอบการหาอนุกรมฟูเรียร์ เช่น ช่วงความถี่ที่ กำลังสนใจ ช่วงความถี่นี้จะมีผลกับคาบเวลาการวัดข้อมูลด้วย ดังนั้นค่านี้ควรมีค่าเท่ากับคาบเวลาที่ใช้ในการวัดข้อมูลดังที่กล่าวมาแล้วในหัวข้อการเก็บข้อมูล ส่วน ORDER OF FFT นั้นจะมีค่าไม่เกิน 10 ชนิดของ window ที่ใช้ก็มีให้เลือกอยู่หลายชนิดคือ

Hanning, Hamming, Blackman Tucky, Triangular และ Kriser.

หลังจากการทำ FFT แล้วจะได้ข้อมูลที่เป็นสเปกตรัมหรืออยู่ในรูปของ Frequency Domain ซึ่งเป็นค่า Complex Number โดยจะอยู่ในรูปของ real part กับ imaginary part หรือ magnitude กับ phase ก็ได้ ที่ความถี่ต่าง ๆ กัน

ถ้าต้องการแสดงผลหลังจากการทำ FFT ก็สามารถแสดงผลออกเป็นกราฟที่จอภาพ หรือออกที่เครื่องพิมพ์และเครื่องเขียนแบบก็ได้โดยการเลือกหัวข้อที่ 4 ของรายการหลัก หลังจากเลือกหัวข้อที่ 4 แล้ว ก็จะมีรายการตามรูปที่ 6 ขึ้นแสดงที่จอภาพ และรูปที่ 7 เป็นรูปตัวอย่างที่ได้จากการแสดงผลเป็นกราฟบนจอภาพ ซึ่งภาพที่แสดงจะแบ่งออกเป็นสองส่วนคือเป็นการแสดง magnitude กับ phase เทียบกับความถี่ หรือ เป็น real part กับ imaginary part เทียบกับความถี่ และสามารถอ่านค่าพิกัดของตำแหน่งต่าง ๆ ของกราฟบนจอภาพได้ โดยใช้ cursor key เป็นหลัก

แบบจำลองที่กำหนดขึ้นเพื่อทดสอบโปรแกรมแสดงไว้ดังรูปที่ 8 โดยสร้างแบบจำลองขึ้นโดยใช้คอมพิวเตอร์โปรแกรมอยู่ในรูปของ Dynamic equation แล้วก็ทำการจำลอง (Simulation) ค่าที่ได้ออกมาที่ส่งเข้าไปในโปรแกรมเพื่อหาสเปกตรัม ระบบที่จำลองจะมีทั้งแบบ Single degree of freedom และ Two degrees of freedom ดังแสดงในรูปที่ 9 รูปที่ 10 - 15 เป็นผลที่ได้ออกมาจากระบบที่จำลองขึ้น ซึ่งโปรแกรมนี้อาจแสดงผลได้ที่จอภาพ หรือส่งออกไปยังเครื่องพิมพ์หรือเครื่องเขียนแบบ ซึ่งถ้าแสดงที่จอภาพแล้วก็จะสามารถอ่านค่าของตำแหน่งต่าง ๆ ในกราฟได้โดยตรงโดยอาศัย arrow keys บนแป้นคีย์บอร์ด นอกจากนี้แล้วรูปกราฟของค่าสเปกตรัมรวมทั้งเฟส (phase) ค่า real part และ imaginary part เทียบกับความถี่ ก็สามารถแสดงได้เช่นเดียวกัน รายละเอียดข้อเท็จจริงที่จะนำข้อมูลเหล่านี้มาใช้ในการวิเคราะห์โมเดลนั้นจะนำกล่าวในบทความอื่นต่อไป

การทำ system identification นั้นจะทำได้หลังจากการทำ FFT แล้ว เนื่องจากการทำ system identification แบบ instrumental variable นั้นจะใช้ค่า real part และ imaginary part ของ spectrum ที่หาได้เป็นหลัก จากรายการหลักถ้าเลือกหัวข้อที่ 5 แล้วก็จะมีรายการดังรูปที่ 8 ขึ้นมาแสดงบนจอภาพ ในรายการนี้ก็จำเป็นที่จะต้องใส่ค่าประมาณของลำดับขนาดของระบบตามรูปแบบดังต่อไปนี้คือ

$$G(S) = \frac{a_0 + a_1 S + a_2 S^2 + \dots + a_m S^m}{b_0 + b_1 S + b_2 S^2 + \dots + b_n S^n}$$

โดยที่ $n > m$

$G(S)$ = Transfer function ของระบบที่กำลังพิจารณา

a_0, a_1, \dots, a_n = เป็นค่าสัมประสิทธิ์ Numerator ที่จะต้องหา

b_0, b_1, \dots, b_m = เป็นค่าสัมประสิทธิ์ Denominator ที่จะต้องหา

ผลลัพธ์ที่ได้จะแสดงอยู่ด้านล่างของรายการนี้ดังแสดงในรูปที่ 8

รูปที่ 5 เป็นโครงสร้างของส่วนของโปรแกรมที่ใช้หา Transfer Function ของระบบที่ทำการวัดสัญญาณ โดยใช้เทคนิคที่เรียกว่า Instrumental Variable รูปที่ 16 และ รูปที่ 17 เป็นการเปรียบเทียบผลที่ได้จากการโปรแกรมกับระบบจริงซึ่งให้ผลอยู่ที่ขั้นที่น่าพอใจ รูปที่ 18 แสดงให้เห็นถึงความเร็วในการ converge เข้าหาค่าจริง ซึ่งค่อนข้างจะเร็วมาก

รูปที่ 7 เป็นตัวอย่างของการทำ circular curvefit โดยเลือกหัวข้อ 6 จากรายการหลัก การทำ circular curvefit นี้ก็อาศัยข้อมูล real part และ imaginary part ของ spectrum ที่หาได้จากการทำ FFT นั้นเอง

สรุป

การนำเอาการวิเคราะห์สเปกตรัมเพื่อประโยชน์ในการทดสอบโมเดลนั้น ใช้ทฤษฎีพื้นฐานเกี่ยวกับ Fourier Analysis ซึ่งเป็นวิธีการเปลี่ยนสัญญาณที่อยู่ในรูปของ time domain ให้ไปอยู่ในรูปของ Frequency Domain โดยอาศัย Digital Fourier Transform และ Fast Fourier Transform Technique นอกจากนี้แล้วเพื่อให้ได้ข้อมูลดังที่ต้องการ และเนื่องจากความถี่ที่อยู่ในสัญญาณนั้นมีช่วงความถี่กว้างมาก ซึ่งอาจจะก่อให้เกิด aliasing ขึ้นได้ในระหว่างทำการสุ่มอ่านสัญญาณ เพราะฉะนั้นจึงอาจจะต้องมี Lowpass filter หรือ bandpass filter ต่ออยู่ที่ตัว transducer นอกจากนี้แล้วยังมี routines ที่ทำ windowing อาจจะทำทางด้าน time domain หรือ frequency domain ก็ได้ ซึ่งทฤษฎีเหล่านี้อาจจะหาได้จากหนังสืออ้างอิง [1],[2],[3] เป็นต้น อนึ่งภาษาที่ใช้ในการเขียนโปรแกรมนี้ใช้ภาษา C ซึ่งทำงานอยู่บนโปรแกรมระบบที่เรียกว่า PC DOS แต่ก็มีส่วนของโปรแกรมที่ใช้ภาษา BASIC ซึ่งเรียกใช้จากภาษา C ในส่วนของภาษา BASIC นั้นมีอยู่ไม่มากนัก

PROGRAM SPECTRUM ANALYSIS AND SYSTEM IDENTIFICATION
Copyright by M.E. Dept. Chulalongkorn University

- 1) CAPTURE DATA FROM A/D CONVERTER
- 2) CREATE SAMPLE MODEL BY COMPUTER
- 3) FFT ANALYSIS
- 4) PLOT SPECTRUM AND PHASE ON SCREEN OR PLOTTER
- 5) SYSTEM IDENTIFICATION
- 6) PLOT CIRCULAR CURFIT
- 7) EXIT TO DOS

Select option number :

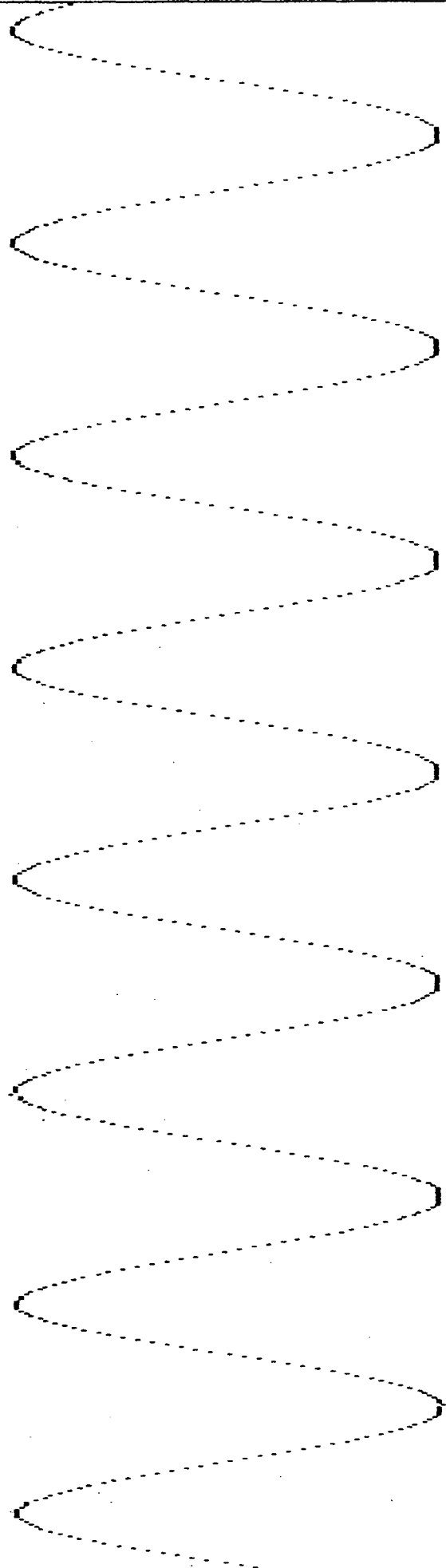
รูปที่ 2 MAIN MENU

TPCLAB: The PCLAB exerciser
 Board code: 5 Microcode Rev 2 DT2801-A Fast PGH
 Number of channels is: 8
 Command (or just enter for info) : ?

CH	- Channel spec	GA	- Gain Spec		
TS	- Timing Source	DS	- DAC Select		
DD	- Display Data	GR	- Graph data	NM	- Number of values
XAV	- ADC Value	XDV	- DAC Value		
XSA	- Setup ADC	XSD	- Setup DAC		
XAS	- ADC Series	XDS	- DAC Series		
XBAD	- Begin ADC DMA	XBDD	- Begin DAC DMA		
XCAD	- Cont. ADC DMA	XCDD	- Cont. DAC DMA		
XTAD	- Test ADC/DAC DMA	XWAD	- Wait ADC/DAC DMA	XSAD	- Stop ADC/DAC DMA
XEFI	- Enable digital in	XIDV	- Input Dig. Val.	XIDOT	- Triggered Dig I
XEFO	- Enable digital out	XODV	- Output Dig. Val.	XODOT	- Triggered Dig C
XGEC	- Get Error Code	XGDE	- Get DT brd Error		
XST	- Set Timeout	XSCD	- Set Clock Divider		
XWFC	- Command output	XGDS	- Status report		
XWFI	- Data Input	XWFO	- Data Output		

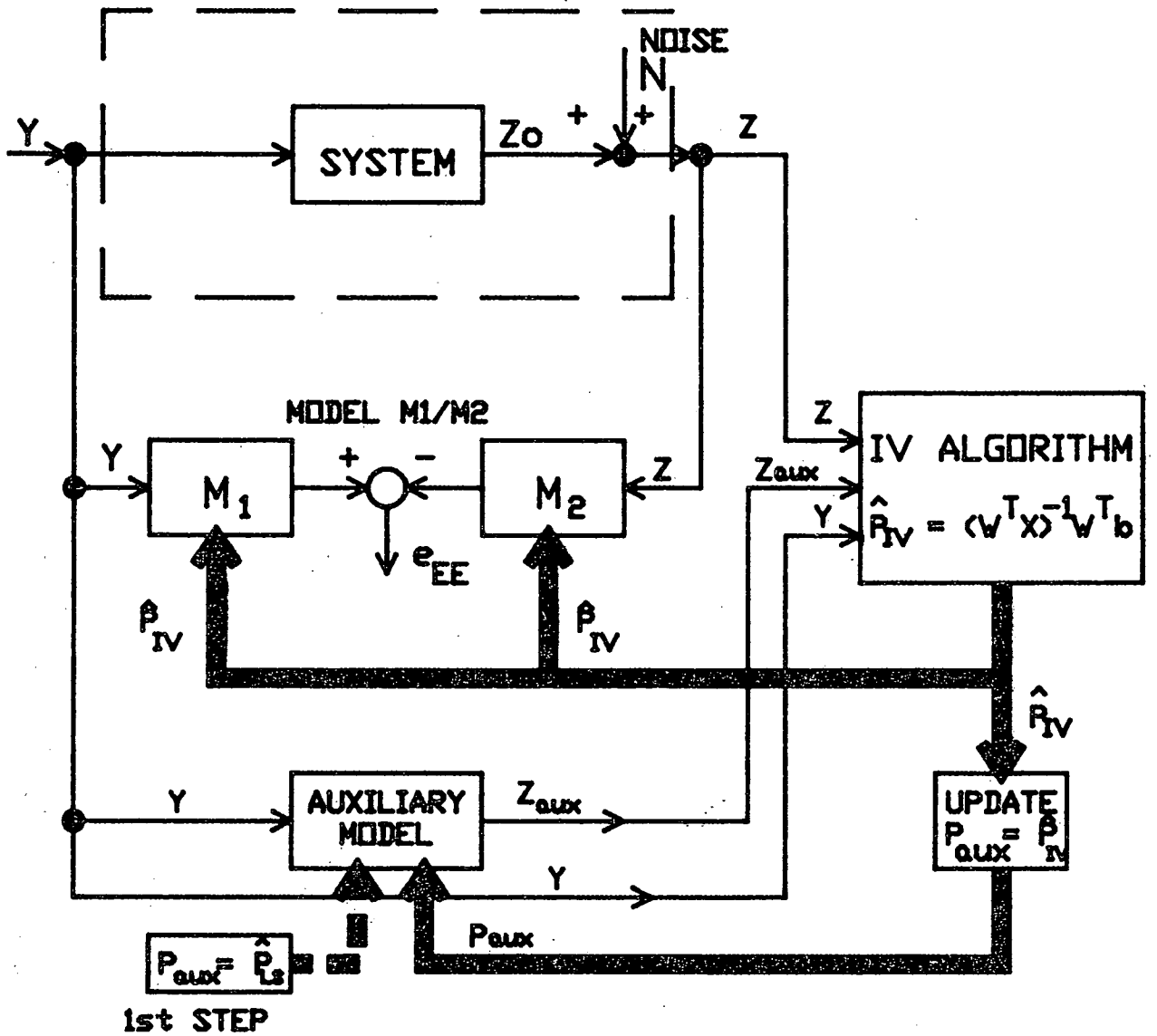
Command (or just enter for info) : ?

รูปที่ 3 CAPTURE DATA FROM A/D CONVERTER MENU



Hit any key to resume

Jun 4 OUTPUT FROM gr COMMAND

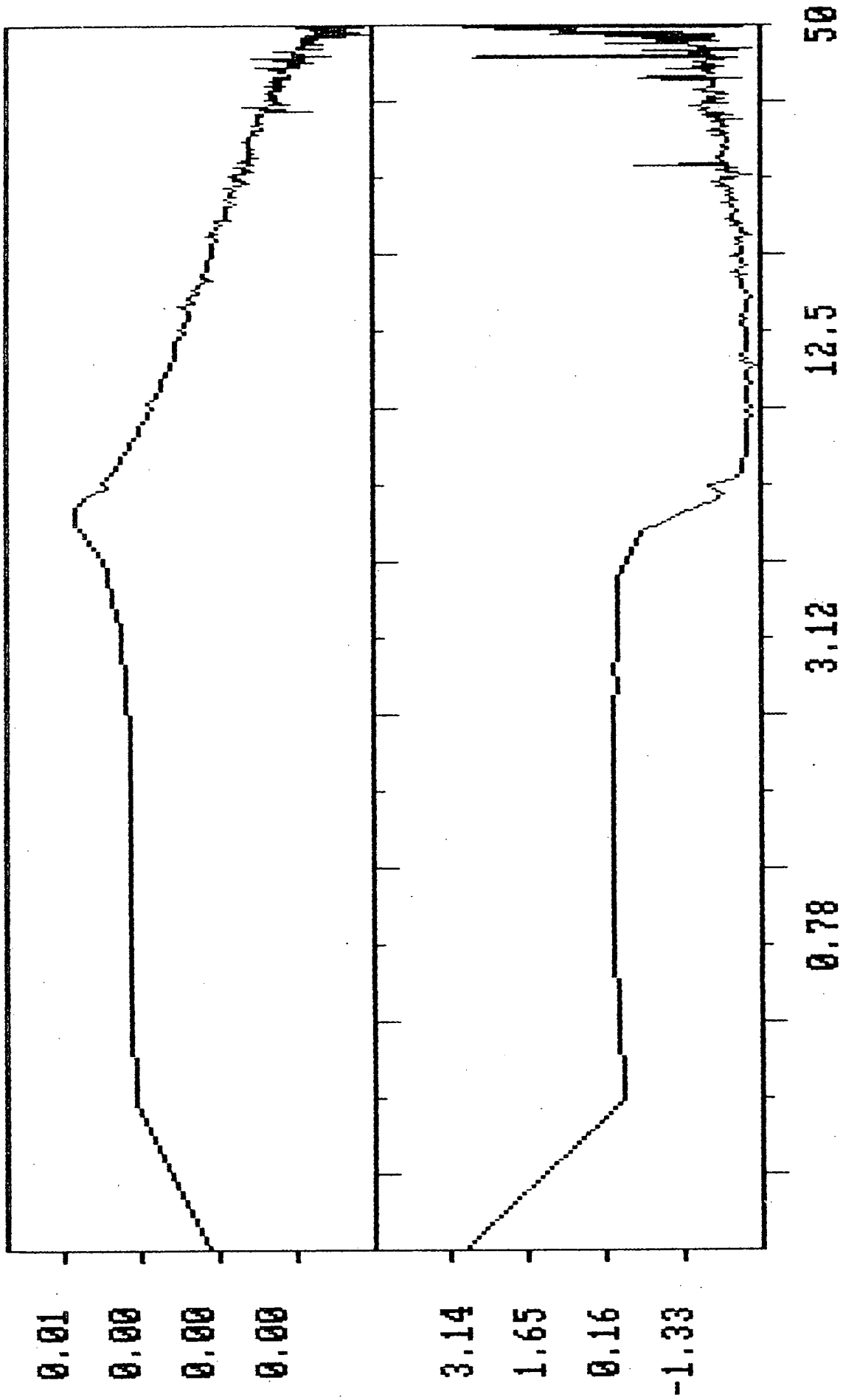


รูปที่ 5 วิธีการ INSTRUMENTAL VARIABLE โดยใช้ Auxiliary Model

FFT MENU

Input frequency range.....: :
Order of FFT.....: :
Window Type.....: :
Number of Ensemble.....: :
Adjusting Bias and Slope.....: :
Smooth output display (1 or 0).....: :
Parameter of Circular Curvefitting (1 or 0).....: :
Plot Spectrum and Phase or Real and Imaginary (1 or 0)..: :
Select the State Variable or Channel.....: :

Window : 0 no 1 Hanning 2 Hamming 3 Black 4 Triang 5 Kriker
Adjusting 1) Bias 2) Bias and Slope 3) No



รูปที่ 7a Magnitude and Phase VS Frequency

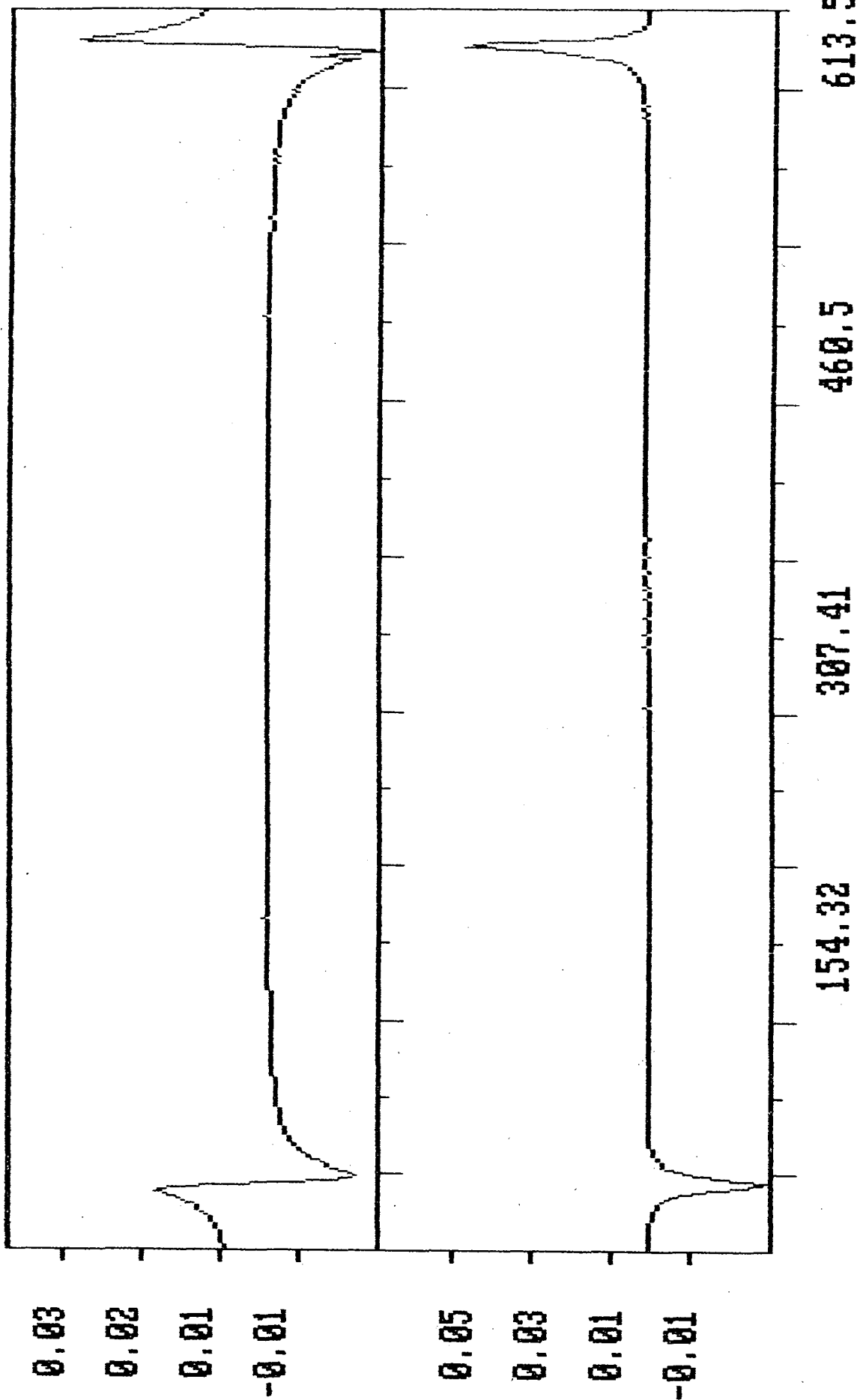
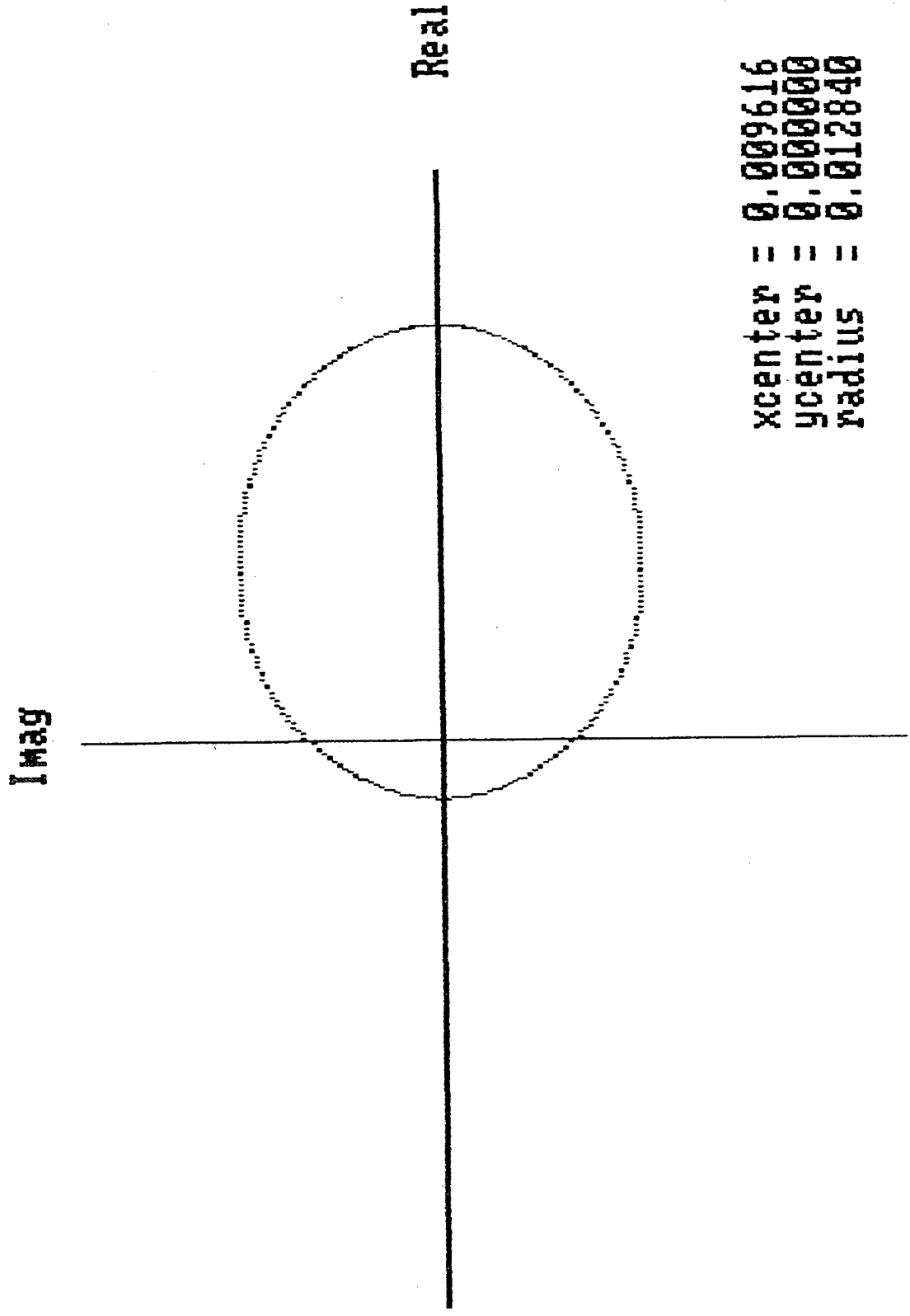


Figure 7b Real part and Imaginary part VS Frequency



307 7 Circular Curvefit

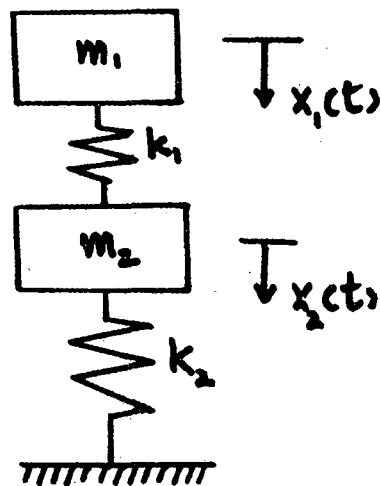
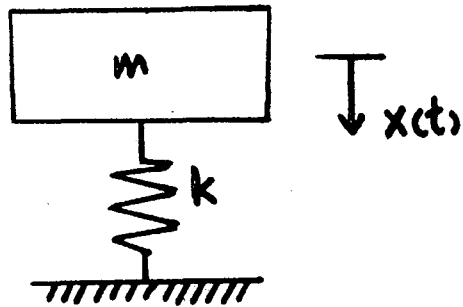
SYSTEM IDENTIFICATION MENU

ORDER OF NOMINATOR..... : 0
ORDER OF DENOMINATOR..... : 2
NUMBER OF DATAPOINT OF FREQUENCY RESPONSE..... : 120
Number OF ITERATIONS..... : 5

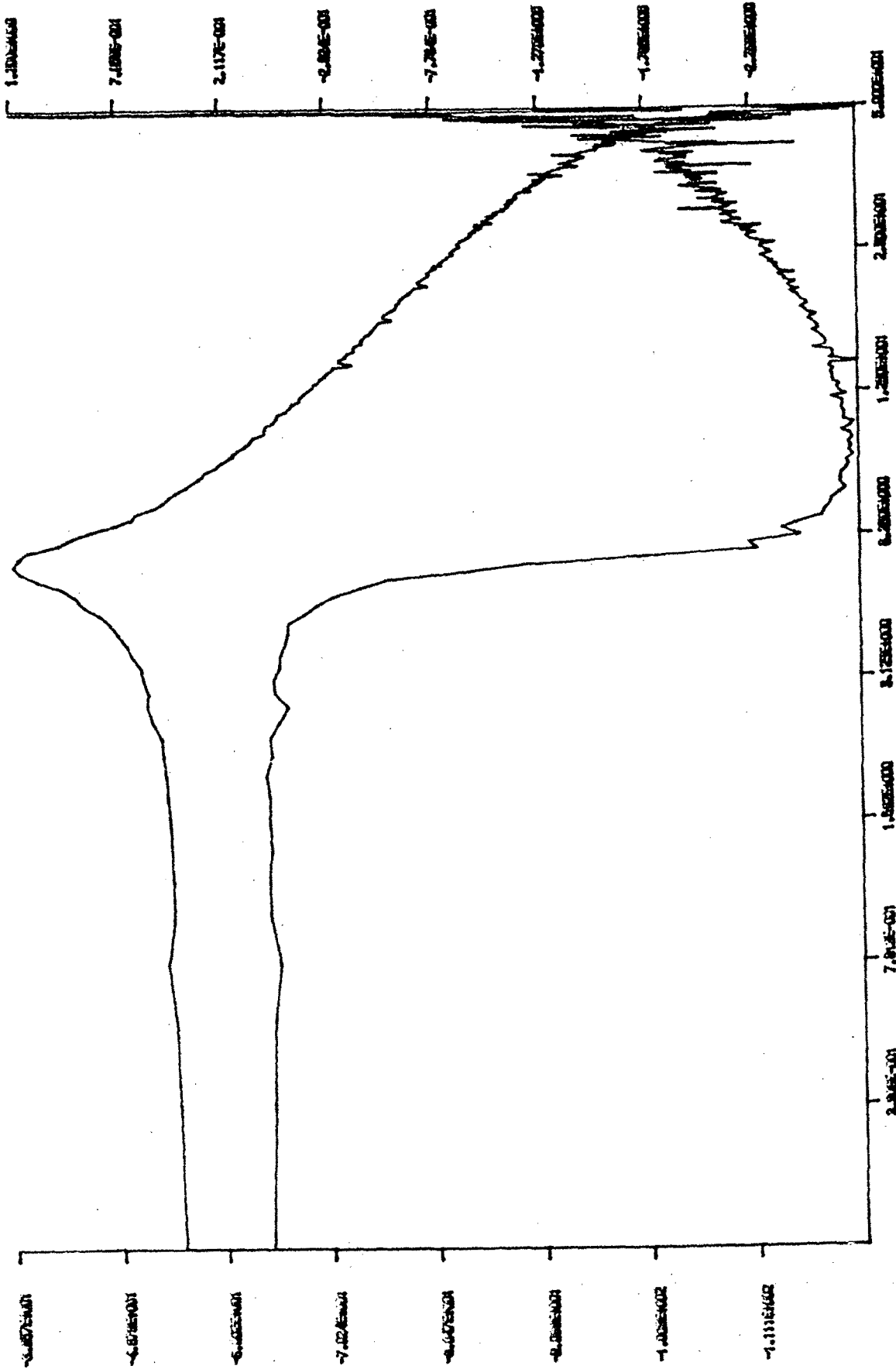
PARAMETERS AFTER 5 ITERATIONS

b[0] = 0.007572
a[1] = 0.006147
a[2] = 0.000805

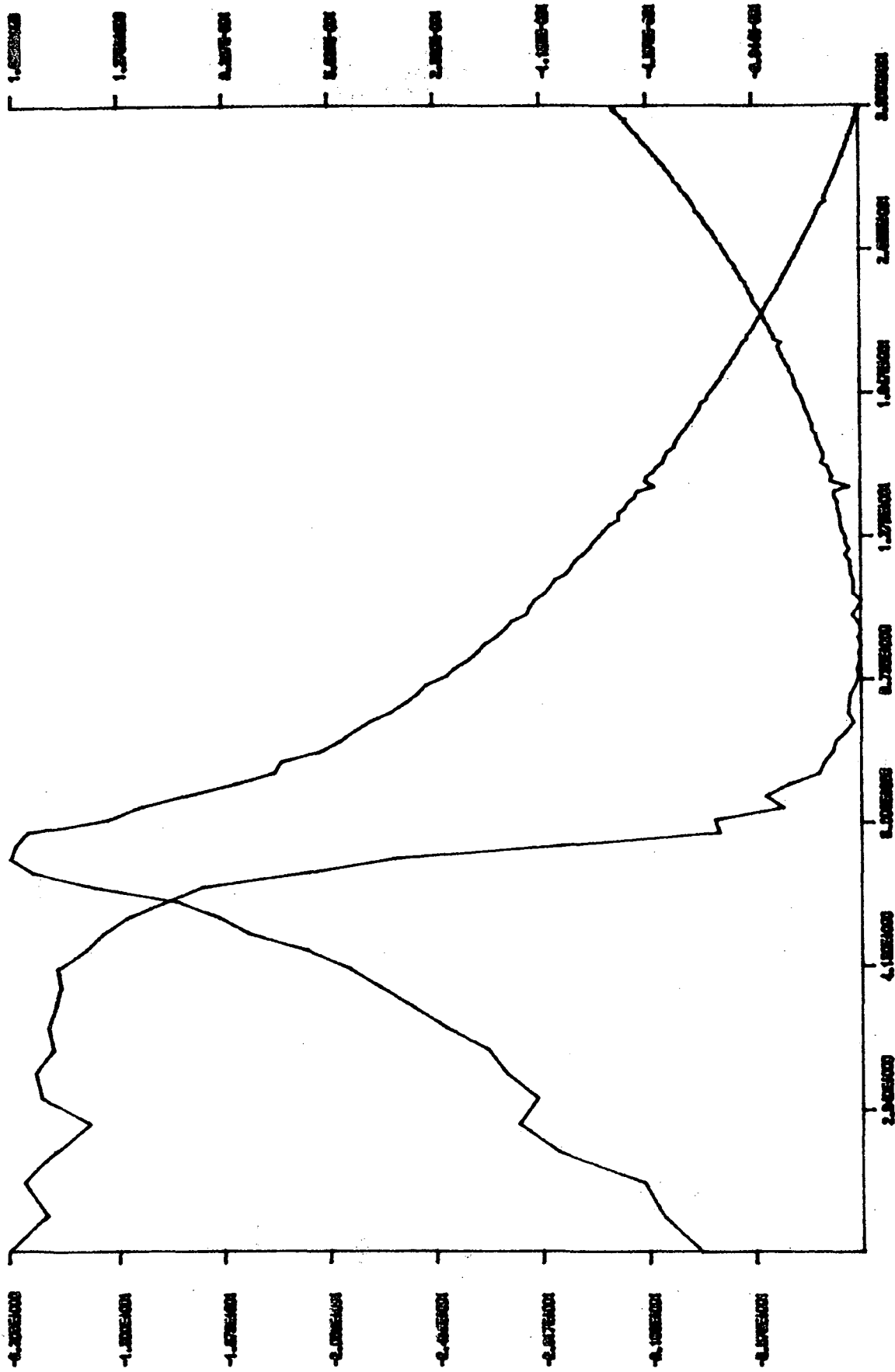
รูปที่ 8 System Identification MENU



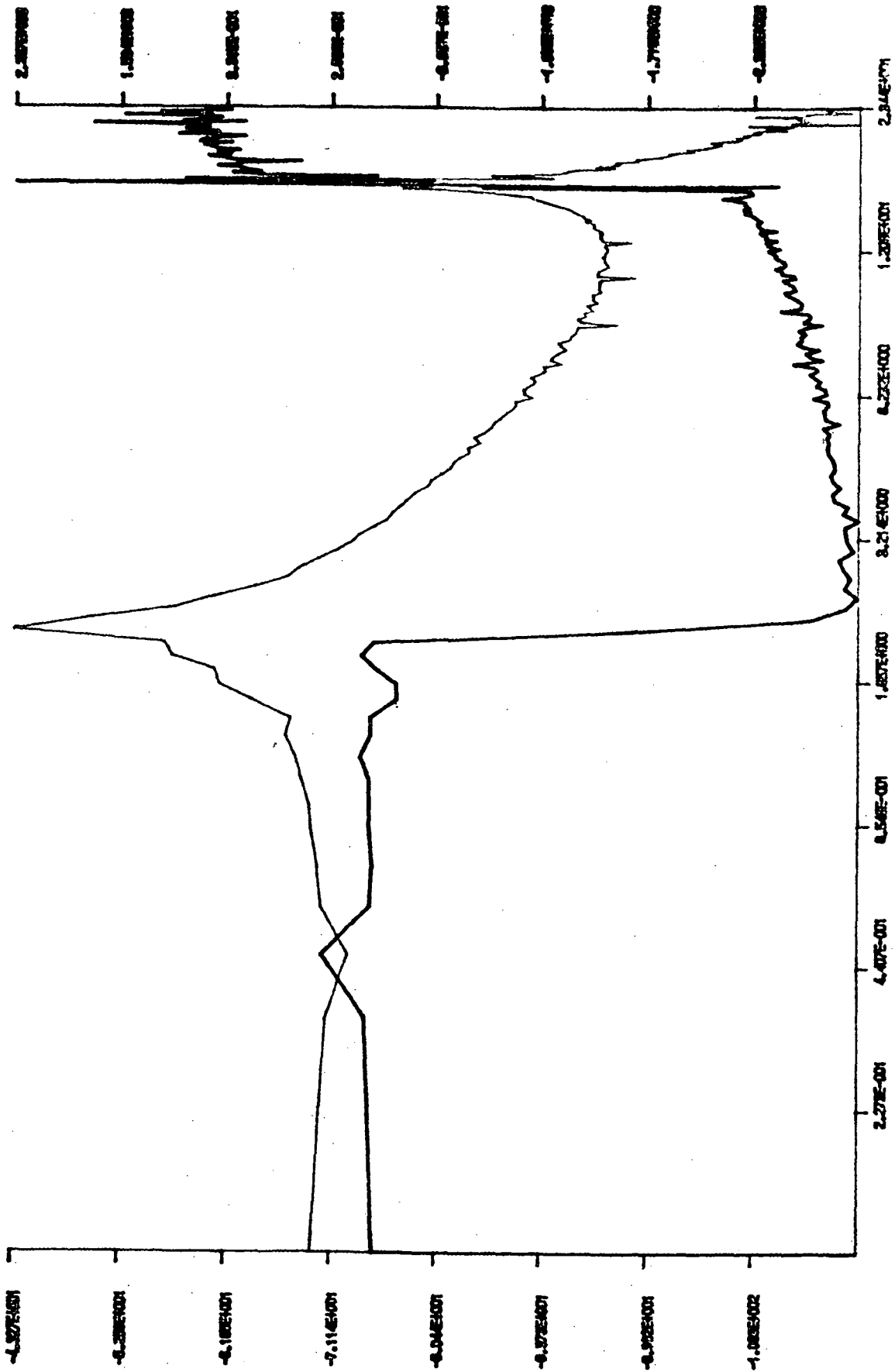
รูปที่ 9 แบบจำลองสองชนิด Single Degree of freedom และ Two Degrees of freedom



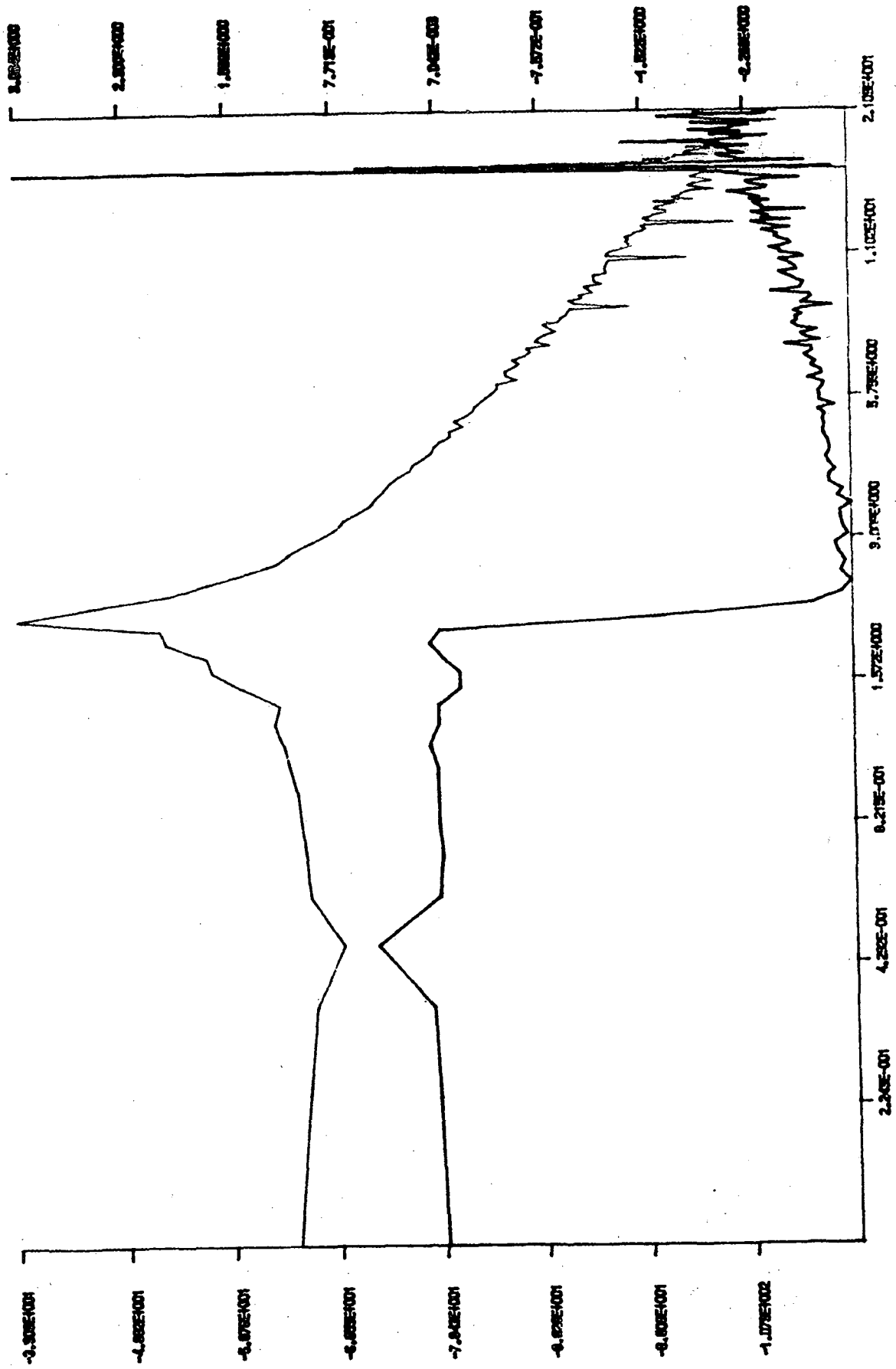
รูปที่ 10 แสดงความสัมพันธ์ Magnitude และ Phase ปรากฏหนึ่ง กับ Frequency
ของระบบที่เป็น Single Degree of Freedom



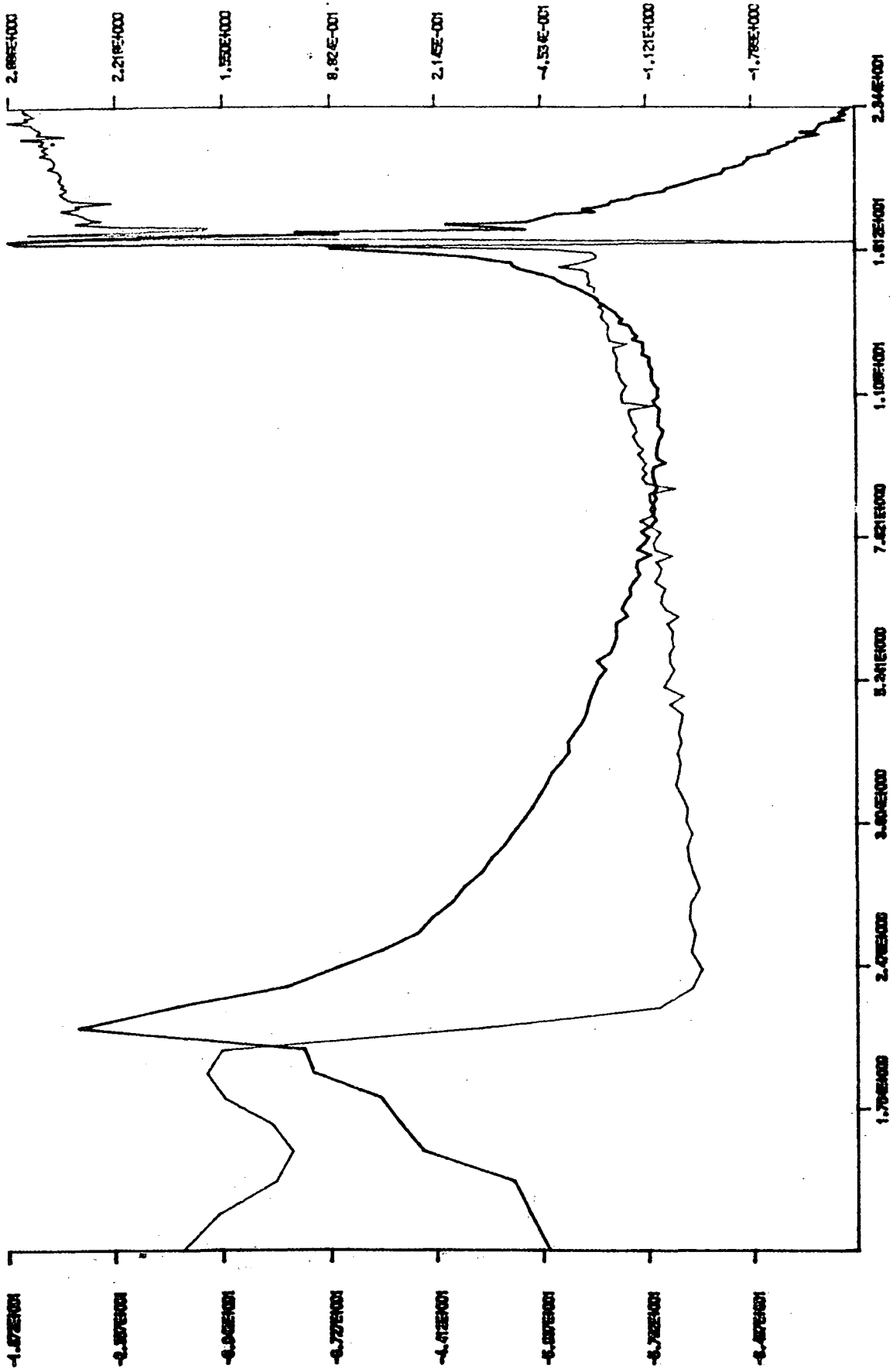
รูปที่ 11 แสดงความสัมพันธ์ Magnitude และ Phase Magnitude ทั่ว กับ Frequency ของระบบที่เป็น Single Degree of Freedom



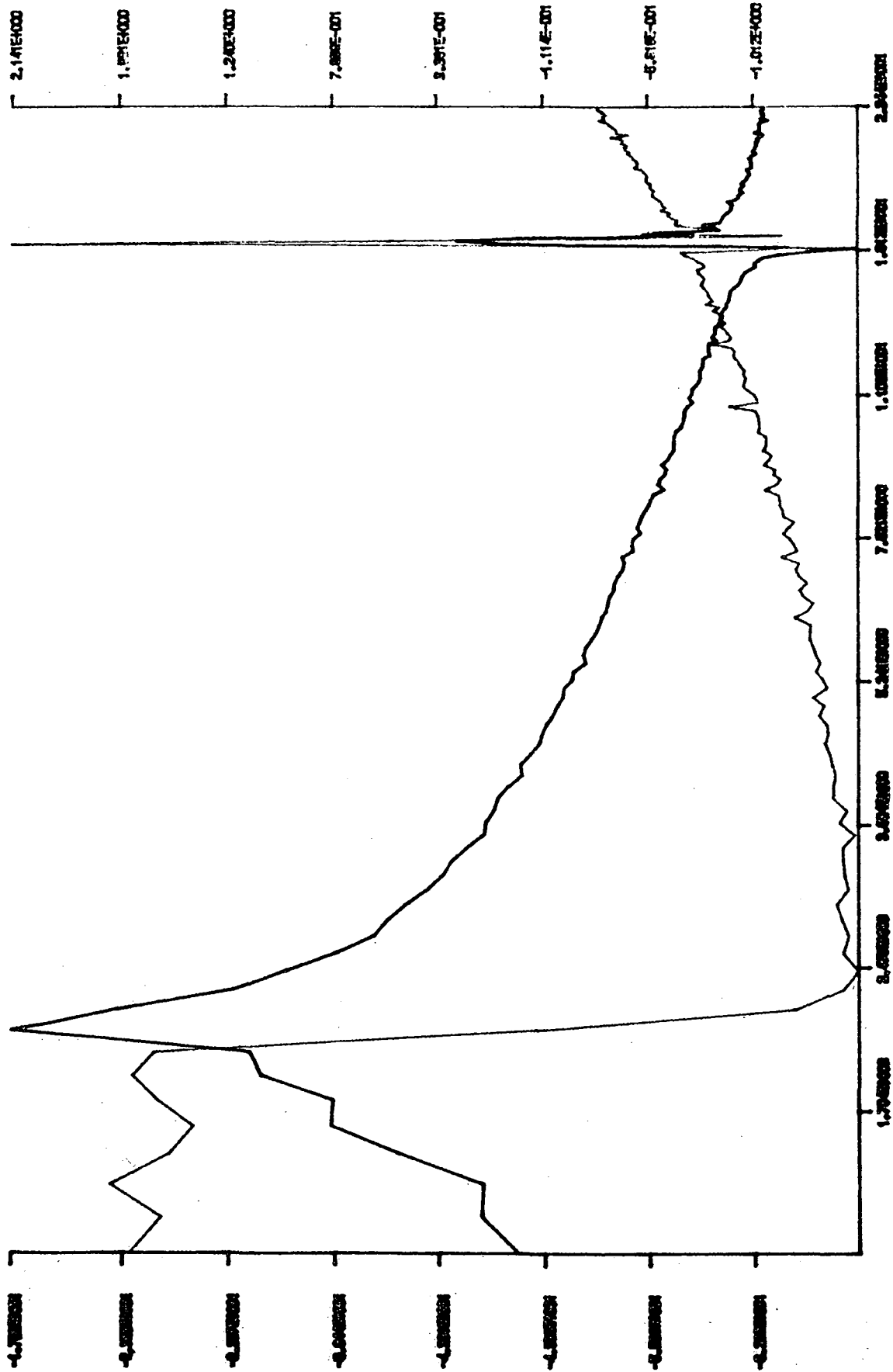
รูปที่ 12 แสดงความสัมพันธ์ Magnitude และ Phase ของค่าหนึ่ง (X_1) กับ
Frequency แสดงว่าเป็น Two Degrees of Freedom



รูปที่ 13 แสดงความสัมพันธ์ Magnitude และ Phase ของความเร็ว (X_1) กับ Frequency ของระบบที่เป็น Two Degrees of Freedom

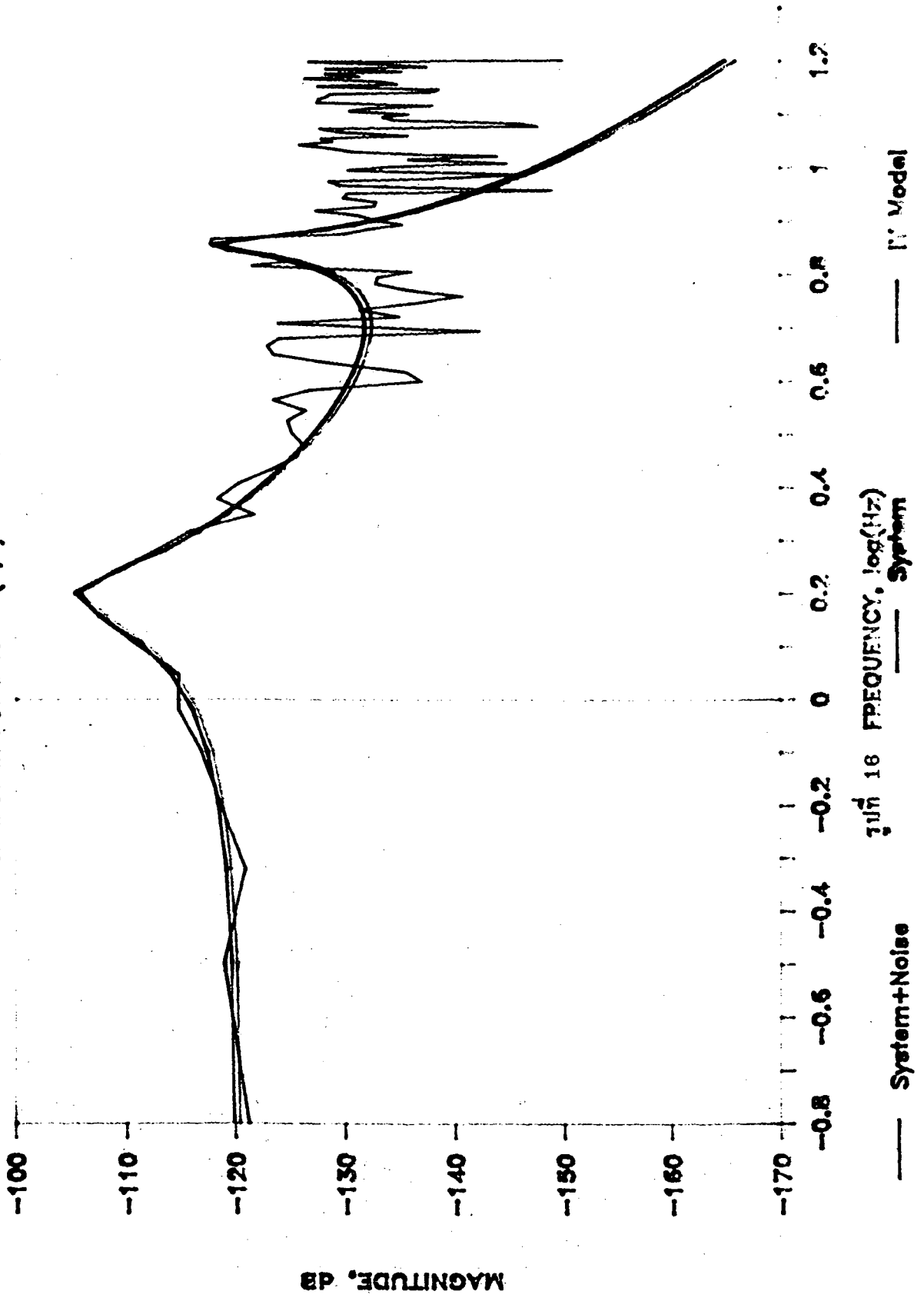


รูปที่ 14 แสดงความสัมพันธ์ Magnitude และ Phase ของค่าหนึ่ง (X_2) กับ Frequency ของระบบที่มี Two Degree of Freedom



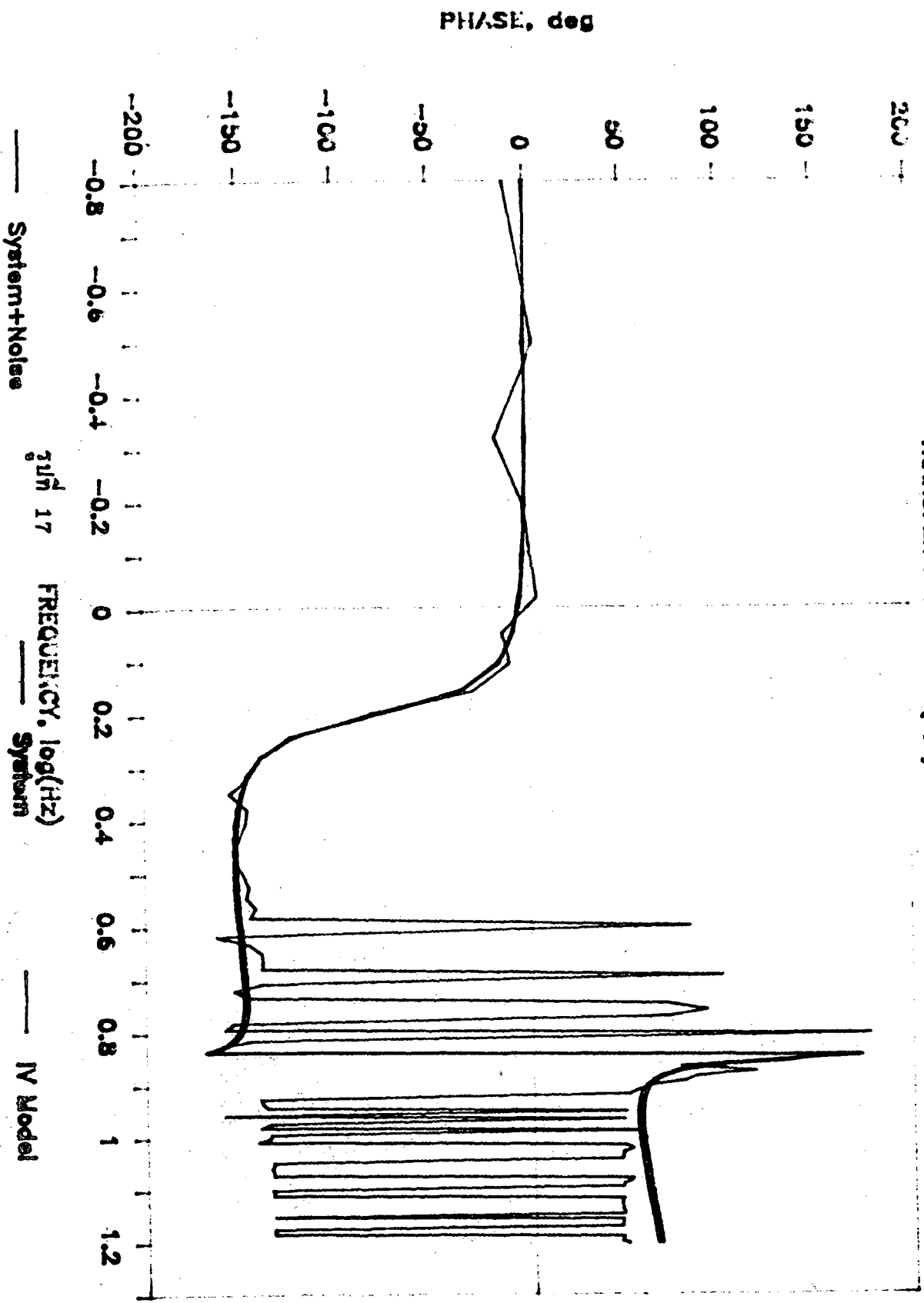
รูปที่ 15 แสดงความสัมพันธ์ Magnitude และ Phase ของค่าหนึ่ง (\dot{X}_2) กับ
Frequency ของระบบที่มี Two Degree of Freedom

SIMULATED SYSTEM TRANSFER FUNCTION $H(1,2)$ IN ESTIMATION



July 16 FREQUENCY, log(Hz)

SIMULATED SYSTEM TRANSFER FUNCTION h(1,2) IV ESTIMATION



SIMULATED SYSTEM, $h(1,2)$

RATE OF CONVERGENCE OF IV ALGORITHM

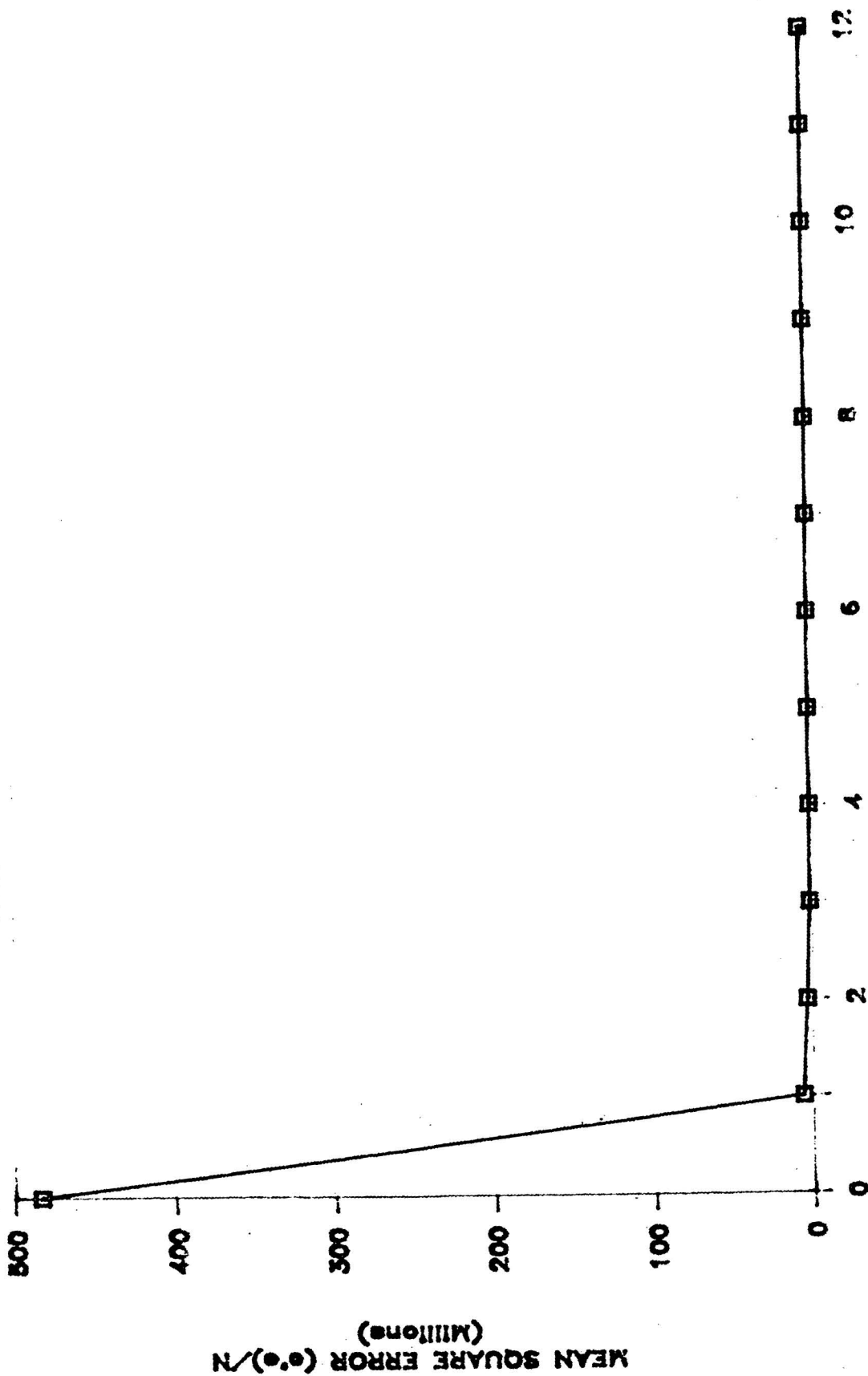


FIG 18. NUMBER OF ITERATIONS

หนังสืออ้างอิง

1. Alan V. Oppenheim, Ronald W. Schaffer, "Digital Signal Processing", Prentice Hall
2. Steven M. Kay, Stanley Lawrence Marple, "Spectrum Analysis", A Modern Perspective, Proceedings of IEEE, Vol 69, No 11, Nov, 1981
3. Eykhoff, P., "System Identification", John Wiley & Sons, N.Y., 1974
4. Lin, P.L. and Y.C. Wu, "Identification of Multi-Input Multi-Output Linear Systems From Frequency Response Data", ASME Trans. on Dynamic Systems, Measurement, and Control, 104, 58-64, 1982