

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

โปรแกรมคำนวณการเคลื่อนที่ของ 4 bar – linkage และ
slider – crank พร้อมแสดงด้วย graphic
4 bar – linkage and slider – crank movement calculation
program with graphic display



กำพล วิญญูหัตถกิจ

Kumpol Winyuhuttakit

พิทักษ์ พลชัย

Pitak Polachai

สรชัย วุฑฒิกรรมรักษา

Sornchai Vudhikumruksa

ปริญญาานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต
สาขาวิศวกรรมศาสตร์เครื่องกล

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2541

เลขหมู่.....
เลขทะเบียน..... 33903
วัน, เดือน, ปี 20 ก.ย. 2542

การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
เปลี่ยนแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4 bar – linkage and slider – crank movement calculation
program with graphic display



Kumpol Winyuhuttakit
Pitak Polachai
Sornchai Vudhikumruksa

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
BACHELOR OF ENGINEERING IN MECHANICAL ENGINEERING
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

1998

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

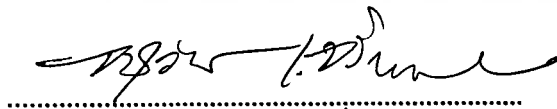
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีเจ้าคุณทหารลาดกระบัง
ใบรับรองปริญญาโท

หัวข้อปริญญาโท โปรแกรมคำนวณการเคลื่อนที่ของ 4 bar-linkage และ slider-crank พร้อม
แสดงด้วย graphic
4 bar-linkage and slider-crank movement calculation program with
graphic display

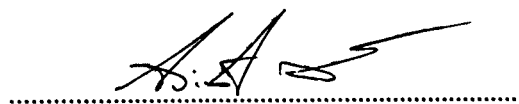
ชื่อนักศึกษา กำพล วิญญูหัตถกิจ รหัสประจำตัว 38014022
 พัทธ์ชัย พลชัย รหัสประจำตัว 38014341
 ศรชัย วุฑฒิกกรรมรักษา รหัสประจำตัว 38014485

หลักสูตร วิศวกรรมศาสตรบัณฑิต
สาขาวิชา วิศวกรรมเครื่องกล

อาจารย์ผู้ควบคุมปริญญาโท



(ดร. จารุวัตร เจริญสุข)



(อ. ประสิทธิ์ คำพันธ์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทคัดย่อ

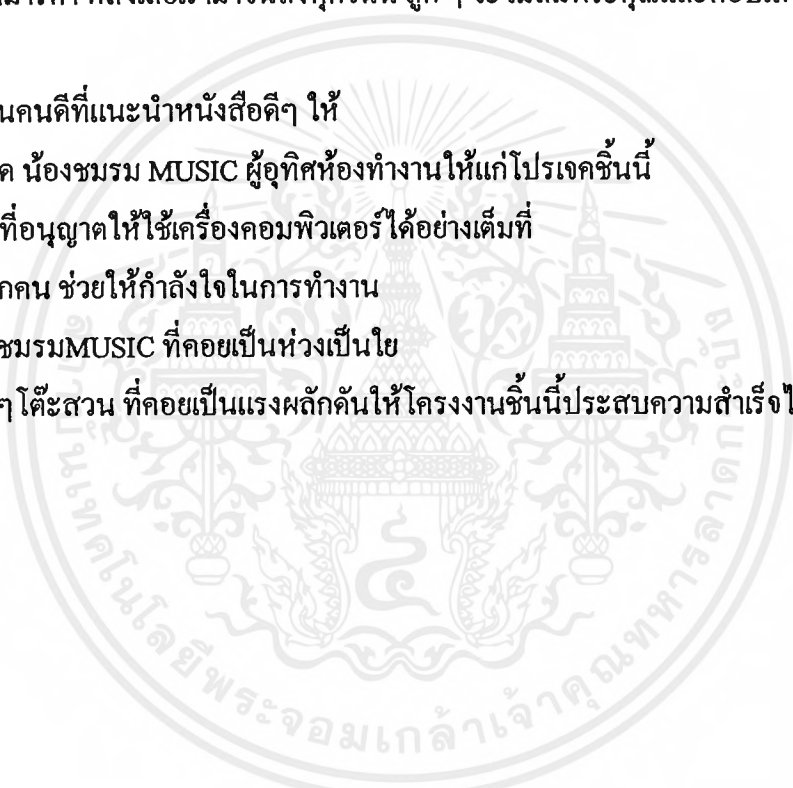
โครงการนี้เป็นกรนำคอมพิวเตอร์มาใช้ช่วยในการคำนวณแก้ไขปัญหาทาง Machinery ซึ่งได้แก่ ลักษณะของ Four-bar linkage และ Slider Crank การแก้ปัญหานี้จะมีการรับค่าข้อมูลต่าง ๆ ซึ่งจะใช้ในการวิเคราะห์หาความเร็ว , ความเร่ง และแรงที่กระทำกับวัตถุ

Abstract

Computer program is invented in order to perform the calculation on problems in mechanics of machinery. This project is scoped within the problems on four-bar linkage and slider-crank mechanisms. In solving the problem , data is entered to the program which enables it to calculate the velocity , acceleration and force magnitudes involved.

กิตติกรรมประกาศ

ขอขอบพระคุณที่ปัญญา ให้ความช่วยเหลือในทุก ๆ ด้าน ถ้าไม่มีพี่พวกเราคงลำบาก
ขอขอบพระคุณ ดร.จาร์วัฒน์ อาจารย์ที่ปรึกษาที่ไม่กดดันพวกเราในการทำงาน และคอยหาทางออก
ที่ดีให้แก่เราเสมอ
ขอขอบพระคุณบิดามารดา ที่ส่งเสียเรามาจนถึงทุกวันนี้ ลูก ๆ จะไม่ลืมพระคุณและตอบแทนบุญ
คุณอย่างดีที่สุด
ขอขอบคุณ พี่ เพื่อนคนดีที่แนะนำหนังสือดีๆ ให้
ขอขอบคุณน้องเบิร์ต น้องชมรม MUSIC ผู้อุทิศห้องทำงานให้แก่โปรเจกต์ชิ้นนี้
ขอขอบคุณน้องตั้ม ที่อนุญาตให้ใช้เครื่องคอมพิวเตอร์ได้อย่างเต็มที่
ขอขอบคุณพี่น้องทุกคน ช่วยให้งำลังใจในการทำงาน
ขอขอบคุณเพื่อนๆ ชมรมMUSIC ที่คอยเป็นห่วงเป็นใย
ขอขอบคุณพี่ๆน้องๆ ใต้สวน ที่คอยเป็นแรงผลักดันให้โครงการชิ้นนี้ประสบความสำเร็จไปได้



สารบัญ

บทที่ 1 ทฤษฎีเกี่ยวกับ Machinery	1
ส่วนที่ 1 Velocity Analysis	1
ส่วนที่ 2 Acceleration Analysis	4
- การหา velocity และ acceleration โดยวิธี Analytical	6
ส่วนที่ 3 Force Analysis	8
1.1 ลักษณะของ Inertia Force ใน linkage	11
1.2 วิธีในการคำนวณหาแรง	12
- วิธีที่ 1 The Superposition Method	12
- วิธีที่ 2 Matrix Method	20
บทที่ 2 หลักวิธีการเขียนโปรแกรมในการคำนวณ	25
2.1 หลักการเขียนโปรแกรมในการคำนวณ four-bar linkage	25
2.2 หลักการเขียนโปรแกรมในการคำนวณ slider-crank	26
บทที่ 3 การใช้งาน โปรแกรมคำนวณการเคลื่อนที่ของ 4 bar-linkage และ slider-crank	27
3.1 ส่วนประกอบของโปรแกรม	27
3.2 การสร้างไฟล์ project.mln	27
3.3 เริ่มใช้งาน โปรแกรม	29
3.4 การใช้งาน โปรแกรม	30
- INPUT	31
- V VECTOR	33
- A VECTOR	34
- F VECTOR	34
- MOVEMENT	35
บทที่ 4 การออกแบบโปรแกรม	36
4.1 การออกแบบโปรแกรม	36
บทที่ 5 คำสั่งในโปรแกรม Autolisp	39
5.1 การใช้คำสั่งในโปรแกรม Autolisp	39
5.2 คำสั่งต่าง ๆ ใน Autolisp	39

บทที่ 6 คำสั่งในโปรแกรม Autocad	44
6.1 การใช้คำสั่งของ Autocad ในโปรแกรม Autolisp	44
6.2 คำสั่งใน Autocad	44
สรุปผลของโครงการ	53
บรรณานุกรม	
ภาคผนวก	



สารบัญญภาพ

บทที่ 1 ทฤษฎีเกี่ยวกับ Machinery

- รูปที่ 1.1 ความเร็วที่ตำแหน่ง A	1
- รูปที่ 1.2 การเปรียบเทียบความเร็วของตำแหน่ง A และ B	2
- รูปที่ 1.3 ลักษณะการวาดเป็น velocity polygon	3
- รูปที่ 1.4 การเปรียบเทียบความเร่งของตำแหน่ง A และ B	5
- รูปที่ 1.5 ลักษณะของ acceleration polygon	6
- รูปที่ 1.6 เวกเตอร์ตำแหน่งของ four-bar linkage	7
- รูปที่ 1.7 วัตถุเป็นไปตามกฎข้อที่ 1	9
- รูปที่ 1.8 วัตถุเป็นไปตามกฎข้อที่ 2	9
- รูปที่ 1.9 วัตถุเป็นไปตามกฎข้อที่ 3	9
- รูปที่ 1.10 ลักษณะ link k ในระนาบใด	10
- รูปที่ 1.11 การหา dynamic equilibrium	11
- รูปที่ 1.12 ผลของแรงเฉื่อยเพียงแรงเดียวที่กระทำกับวัตถุ	13
- รูปที่ 1.13 ผลของแรงเฉื่อยทำให้เกิดแรงระหว่างเครื่องต่อ	14
- รูปที่ 1.14 การคำนวณเพื่อหาทิศทาง F'_{23}	16
- รูปที่ 1.15 การคำนวณเพื่อหาค่า F'_{23} และ F'_{43}	16
- รูปที่ 1.16 การหาค่าผลของแรงที่กระทำต่อเครื่องต่อ 4	17
- รูปที่ 1.17 การหาค่า F''_{34} และ F''_{14}	18
- รูปที่ 1.18 ผลของแรงที่กระทำต่อเครื่องต่อ 4	18
- รูปที่ 1.19 เครื่องต่อ 2 มีแรงเฉื่อยกระทำเพียงแรงเดียว	19
- รูปที่ 1.20 สมดุลย์ของแรงและทอร์กของเครื่องต่อ 2	19
- รูปที่ 1.21 การหาผลลัพธ์ของแรงและทอร์กทั้งหมด	20
- รูปที่ 1.22 (a),(b),(c) ผลของแรงที่กระทำเครื่องต่อต่าง ๆ (d) ลักษณะของเวกเตอร์โมเมนต์.....	21

บทที่ 2 หลักวิธีการเขียนโปรแกรมในการคำนวณ

บทที่ 3 การใช้งาน โปรแกรมคำนวณการเคลื่อนที่ของ 4 bar-linkage และ slider-crank

- รูปที่ 3.1 เมนู mlstyle	27
- รูปที่ 3.2 หน้าต่าง Multiline Properties	28
- รูปที่ 3.3 การเรียกใช้งาน project.mln	29
- รูปที่ 3.4 การเรียกใช้งาน project.lsp	29
- รูปที่ 3.5 โปรแกรมคำนวณการเคลื่อนที่ของ 4 bar-linkage และ slider-crank	30
- รูปที่ 3.6 INPUT ของ 4 bar-linkage	31
- รูปที่ 3.7 INPUT ของ slider-crank	32
- รูปที่ 3.8 คำสั่ง V VECTOR	33
- รูปที่ 3.9 คำสั่ง A VECTOR	34
- รูปที่ 3.10 คำสั่ง F VECTOR	34
- รูปที่ 3.11 คำสั่ง MOVEMENT	35
- รูปที่ 3.12 เลือก EXIT เพื่อออกจาก โปรแกรม	35

บทที่ 4 การออกแบบ โปรแกรม

- รูปที่ 4.1 ลำดับการทำงานของโปรแกรม	38
--	----

บทที่ 5 คำสั่งใน โปรแกรม Autolisp

บทที่ 6 คำสั่งใน โปรแกรม Autocad

- รูปที่ 6.1 หน้าต่างสำหรับเลือกสี โดยมีสีมาตรฐานอยู่แถบบนสุด	45
- รูปที่ 6.2 การเลือกแบบ WINDOW	46
- รูปที่ 6.3 การเลือกแบบ CROSSING	46
- รูปที่ 6.4 Toolbar สำหรับบอกและแก้ไขเลขอร์ชั้นต่าง ๆ	47
- รูปที่ 6.4 คำสั่ง line จากจุด (0,0) ถึง (1,1)	47
- รูปที่ 6.4 คำสั่ง line จากจุด (0,0) ถึง (1,1) ถึง (1,0.5)	47
- รูปที่ 6.7 รูปแบบการวางเส้น mline	48
- รูปที่ 6.8 เส้น mline จากจุด (0,0) ถึง (10,0)	48
- รูปที่ 6.9 เส้น mline จากจุด (0,0) ถึง (0,5)	49
- รูปที่ 6.10 หน้าต่างของคำสั่ง mlstyle	49

- รูปที่ 6.11 – 6.15 ตัวอย่างของรูประบายที่บจากคำสั่ง solid	50
- รูปที่ 6.16 คำสั่ง rectangle	51
- รูปที่ 6.17 คำสั่ง Text	51
- รูปที่ 6.18 คำสั่ง Text โดยกำหนดการวางแบบ Center	52

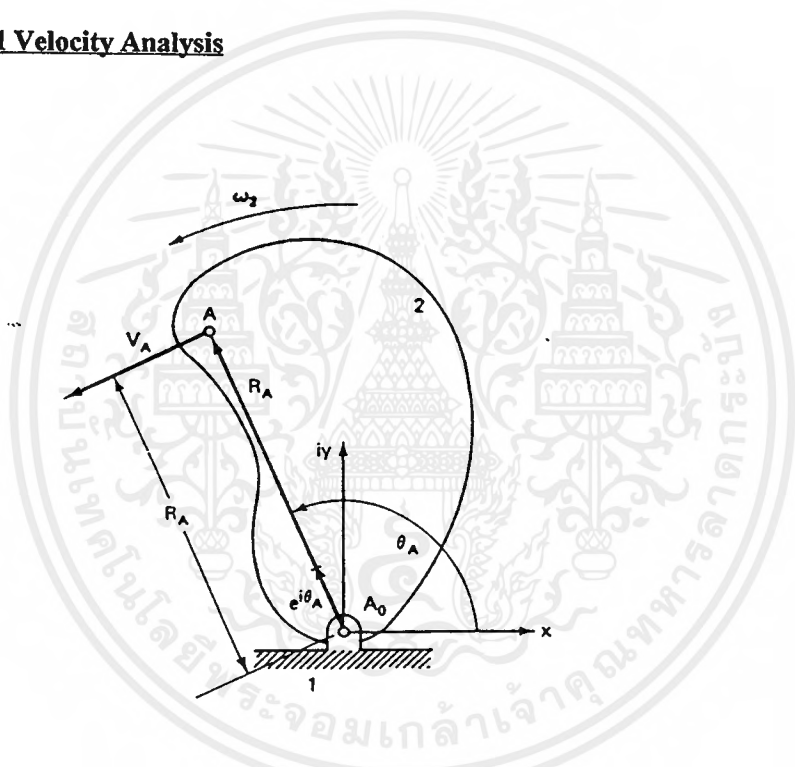


บทที่ 1

ทฤษฎีที่เกี่ยวข้องกับ Machinery

เนื่องจาก project นี้เป็นการนำโปรแกรมคอมพิวเตอร์มาคิดคำนวณเกี่ยวกับ โครงสร้างใน ลักษณะของ Four-bar linkage ดังนั้นทฤษฎีจึงเกี่ยวข้องกับเนื้อหาวิชา Machinery โดยตรง โดยเนื้อหาวิชานั้นจะสามารถแบ่งออกได้เป็น 3 ส่วนคือ

ส่วนที่ 1 Velocity Analysis



รูปที่ 1.1 ความเร็วที่ตำแหน่ง A

จากรูปที่ 1.1 ตำแหน่งของจุด A ที่วัดจากจุด A_0 จะกำหนดให้มีความยาว R_A และ กำหนดให้ θ_A เป็นมุมที่วัดเทียบกับแกน X เราจะสามารถหาความเร็วที่ตำแหน่ง A ได้จาก เวกเตอร์ตำแหน่ง A สามารถเขียนในรูปแบบของ Polar Form ได้เป็น

$$R_A = R_A e^{i\theta_A} \quad (1.1)$$

จาก (1.1) เราสามารถกระจายได้โดยใช้ Euler's equation

$$R_A = R_A \cos \theta_A + i R_A \sin \theta_A \quad (1.2)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถที่จะหา absolute linear velocity ของตำแหน่ง A ได้โดยการหาอัตราการเปลี่ยนแปลงระหว่างเวกเตอร์ตำแหน่งของ A กับ time

$$\mathbf{V}_A = \frac{dR_A}{dt} \quad (1.3)$$

หรืออยู่ในลักษณะ Polar form (โดยกำหนดความเร็วเชิงมุม $\omega_2 = d\theta_A/dt$)

$$\mathbf{V}_A = iR_A \frac{d\theta_A}{dt} e^{i\theta_A} = R_A \omega_2 i e^{i\theta_A} = iR_A \omega_2 \quad (1.4a)$$

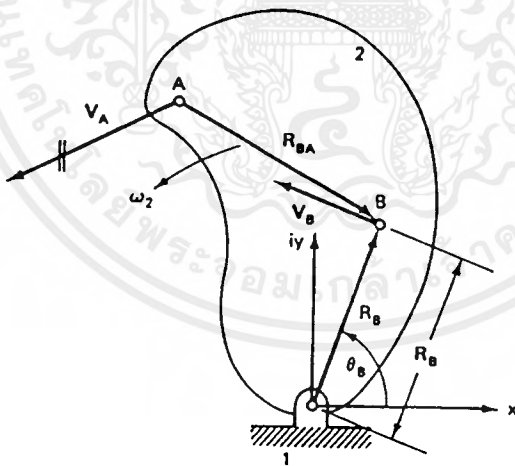
จาก (1.4) เราสามารถกระจายได้

$$\mathbf{V}_A = R_A (-\sin\theta_A) \frac{d\theta_A}{dt} + iR_A (\cos\theta_A) \frac{d\theta_A}{dt} = R_A \omega_2 (-\sin\theta_A + i\cos\theta_A) \quad (1.4b)$$

เนื่องจากว่า

$$|\mathbf{V}_A| = V_A = R_A |\omega_2|$$

โดยที่ทิศทางของ \mathbf{V}_A นั้นมีทิศทางที่ตั้งฉากกับเวกเตอร์ตำแหน่งของ A สำหรับค่าของ ω_2 นั้นถ้ามีทิศทางทวนเข็มนาฬิกาจะกำหนดให้เป็นบวก แต่ถ้ามีทิศทางตามเข็มนาฬิกาจะกำหนดให้เป็นลบ



รูปที่ 1.2 การเปรียบเทียบความเร็วของตำแหน่ง A และ B

ต่อมาพิจารณารูปที่ 1.2 เราสามารถที่จะหา absolute linear velocity ของตำแหน่ง B ได้เช่นเดียวกับตำแหน่งของ A โดยสามารถที่จะเขียนให้อยู่ในรูปของ Polar form ได้เป็น

$$\mathbf{V}_B = R_B \omega_2 i e^{i\theta_B} = R_B i \omega_2 \quad (1.5)$$

หรือสามารถที่จะกระจายได้เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\mathbf{V}_B = R_B \omega_2 (-\sin \theta_B + i \cos \theta_B) \quad (1.6)$$

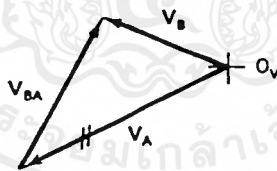
จากตำแหน่ง A และ B เราสามารถที่จะหาความเร็วสัมพัทธ์ระหว่างจุดทั้ง 2 ได้จากสมการ

$$\mathbf{V}_{BA} = \mathbf{V}_B - \mathbf{V}_A = R_B i \omega_2 - R_A i \omega_2 = (\overline{AB}) i \omega_2 = R_{BA} i \omega_2$$

หรือสามารถจัดรูปใหม่ได้เป็น

$$\mathbf{V}_B = \mathbf{V}_A + \mathbf{V}_{BA} = R_B i \omega_2 = R_A i \omega_2 + (\overline{AB}) i \omega_2 \quad (1.7)$$

ขั้นตอนต่อมาเป็นการนำสมการ (1.7) มาเขียนเป็นลักษณะของ velocity polygon ดังรูปที่ 1.3 ซึ่งกำหนดให้ตำแหน่ง O_V เป็นตำแหน่งเริ่มต้นของการวาด โดยเริ่มต้นจากทางซ้ายมือของสมการ (1.7) \mathbf{V}_B และเพื่อความสะดวกให้ลากเส้นโดยมีขนาดและทิศทางเช่นเดียวกับ \mathbf{V}_B ในรูปที่ 1.2 การลากเส้นให้เริ่มลากเส้นจากตำแหน่ง O_V หลังจากนั้นให้ทำการวาดทางขวามือของสมการ (1.7) โดยให้มีขนาดและทิศทางเดียวกับ \mathbf{V}_A ในรูปที่ 1.1 ต่อไปก็ทำการลากเวกเตอร์ระหว่าง \mathbf{V}_B และ \mathbf{V}_A ซึ่งเวกเตอร์ที่อยู่ระหว่าง \mathbf{V}_B และ \mathbf{V}_A ก็คือ เวกเตอร์ \mathbf{V}_{BA} นั้นเอง จากสมการ (2.7) จะเห็นว่ารูป velocity polygon นั้นจะเป็นลักษณะรูปเหลี่ยมปิดพอดีจากนั้นก็ทำการวัดขนาดและทิศทางของ \mathbf{V}_{BA} รูปที่ 1.3 ได้ตามต้องการ จะเห็นได้ว่า \mathbf{V}_{BA} นั้นจะมีทิศทางที่ตั้งฉากกับ R_{BA} ในรูปที่ 1.2 ดังนั้นจะเห็นได้ว่า



$$\mathbf{V}_B = \mathbf{V}_A + \mathbf{V}_{BA}$$

รูปที่ 1.3 ลักษณะการวาดเป็น velocity polygon

$$\mathbf{V}_{BA} = -\mathbf{V}_{AB} \quad \text{หรือ} \quad R_{BA} i \omega_2 = -R_{AB} i \omega_2 \quad (1.8)$$

และ

$$\mathbf{V}_{BA} = i \omega_2 R_{BA} \quad (1.9)$$

จากสมการ (1.9) สามารถที่จะจัดรูปใหม่ได้เป็น

$$|\omega_2| = \frac{V_{BA}}{R_{BA}} \quad (1.10)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากหลักเกณฑ์ข้างต้นดังกล่าวมาแล้วทั้งหมดนี้จะเป็นพื้นฐานในการที่จะสามารถหา velocity ในวิชา mechnery ได้ ซึ่งนับได้ว่าเป็นพื้นฐานที่สำคัญในการทำ project ชิ้นนี้

ส่วนที่ 2 Acceleration Analysis

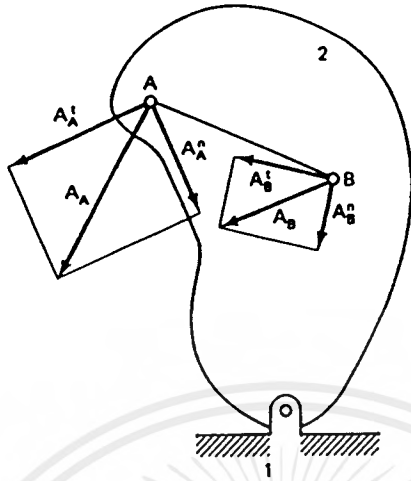
จากทฤษฎีเกี่ยวกับการเคลื่อนที่เบื้องต้นนั้น การที่จะหาความเร่งของระบบได้นั้นจะต้องหาความเร็วของระบบได้เสียก่อน สำหรับในวิชา mechnery นี้ก็เช่นเดียวกัน ซึ่งในส่วนของความเร็วนั้นเราสามารถที่จะหาได้จากเนื้อหาในที่แล้ว และในส่วนนี้จะเป็นการหาความเร่ง โดยจากสมการความเร็วของตำแหน่ง A ในสมการ (1.5)

$$V_B = R_B \omega_2 i e^{i\theta_B} = R_B i \omega_2 \quad (1.5)$$

จากสมการดังกล่าวทำการ derivative ความเร็วในสมการดังกล่าวเทียบกับ time จะได้ absolute linear acceleration ที่ตำแหน่ง A ดังนี้ (โดยความเร่งเชิงมุม $\alpha_2 = d\omega_2/dt$)

$$\begin{aligned} A_A &= R_A \alpha_2 i e^{i\theta_A} - R_A \omega_2^2 e^{i\theta_A} \\ A_A &= R_A (-\omega_2^2 + \alpha_2 i) e^{i\theta_A} \\ &= (-\omega_2^2 + i\alpha_2) R_A \end{aligned} \quad (1.11)$$

จากสมการความเร่งดังกล่าวจะเห็นได้ว่าในสมการจะประกอบด้วยความเร่ง 2 ส่วน โดยใน ส่วนแรกเป็นส่วนที่มีขนาด $R_A \alpha_2$ และมีทิศทางที่สัมผัสกับการเคลื่อนที่ของตำแหน่ง A เราจะเรียก ความเร่งในส่วนนี้ว่า ความเร่งเชิงเส้น (Tangential Acceleration) ความเร่งในส่วนที่ 2 จะมีขนาด $R_A \omega_2^2$ จากสมการจะเห็นว่าความเร่งในส่วนนี้จะมีเครื่องหมายติดลบ ทั้งนี้เป็นเพราะว่า ความเร่งนี้ จะมีทิศทางที่ตรงข้ามกับเวกเตอร์ตำแหน่ง A จากเครื่องหมายดังกล่าวทำให้เราทราบว่า ความเร่งนี้ จะมีทิศทางที่เข้าสู่ตำแหน่งศูนย์กลางของการหมุนทำให้เราเรียกความเร่งในส่วนนี้ว่า ความเร่งเข้าสู่ ศูนย์กลาง (Normal or Centripetal Acceleration)



รูปที่ 1.4 การเปรียบเทียบความเร่งของตำแหน่ง A และ B

จากรูปที่ 1.4 เราสามารถที่จะหาความเร่งในตำแหน่ง B ได้เช่นเดียวกับตำแหน่ง A ดังนั้นความเร่งที่ตำแหน่ง B จะมีค่าเท่ากับ

$$A_B = R_B(-\omega_2^2 + i\alpha_2)e^{i\theta_B} = (-\omega_2^2 + i\alpha_2)R_B \quad (1.12)$$

เช่นเดียวกันกับในส่วนของความเร่ง ความเร่งระหว่างตำแหน่ง A และตำแหน่ง B หรือความเร่งสัมพัทธ์เราสามารถหาได้จาก

$$A_B = A_A + A_{BA}$$

หรือ

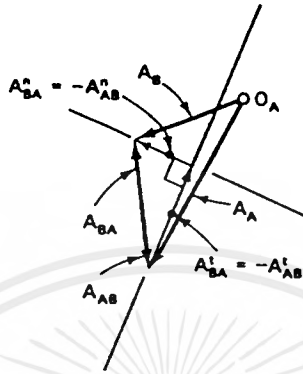
$$A_{BA} = A_B - A_A \quad (1.13)$$

แต่เนื่องจากความเร่งในแต่ละเทอมจะต้องประกอบด้วยความเร่ง 2 ส่วน โดยกำหนดให้ความเร่งเข้าสู่ศูนย์กลางและความเร่งเชิงเส้น แทนด้วยสัญลักษณ์ด้วย n และ t ตามลำดับ จะได้เป็น

$$A_{BA}'' + A_{BA}' = A_B'' + A_B' - A_A'' - A_A' \quad (1.14)$$

จากสมการ (1.14) เราจะเห็นได้ทางขวามือของสมการ (1.14) เป็นตัวแปรที่เราทราบค่าหมดแล้ว ไม่ว่าจะเป็นขนาดหรือทิศทาง แต่ทางซ้ายมือของสมการนั้นเราทราบแต่เพียงทิศทางเท่านั้น โดยทิศทางของ A_{BA}'' จะมีทิศทางพุ่งตรงจากตำแหน่ง B มายังตำแหน่ง A ส่วนทิศทางของ A_{BA}' นั้นจะมีทิศทางที่ตั้งฉากกับ A_{BA}'' ซึ่งเราสามารถที่จะหาขนาดความเร่งที่เหลือได้ โดยการนำสมการ (1.14) มาเขียนให้อยู่ในรูปแบบ acceleration polygon ซึ่งรูปเหลี่ยมที่เราได้จะเป็นรูปเหลี่ยมปิดพอ

ดี หลังจากนั้นก็ทำการวัดขนาดจากรูป เราจะได้ขนาดของความเร่งที่ต้องการ ดังที่ได้แสดงได้ในรูปที่ 1.5



รูปที่ 1.5 ลักษณะของ acceleration polygon

และจากรูปที่ 1.5 จะได้ว่า

$$A'_{BA} = -A'_{AB} \text{ และ } A''_{BA} = -A''_{AB} \quad (1.15)$$

และเช่นเดียวกับสมการ (1.11) เราจะได้ว่า

$$A'_{BA} = i\alpha_j R_{BA} \quad (1.16)$$

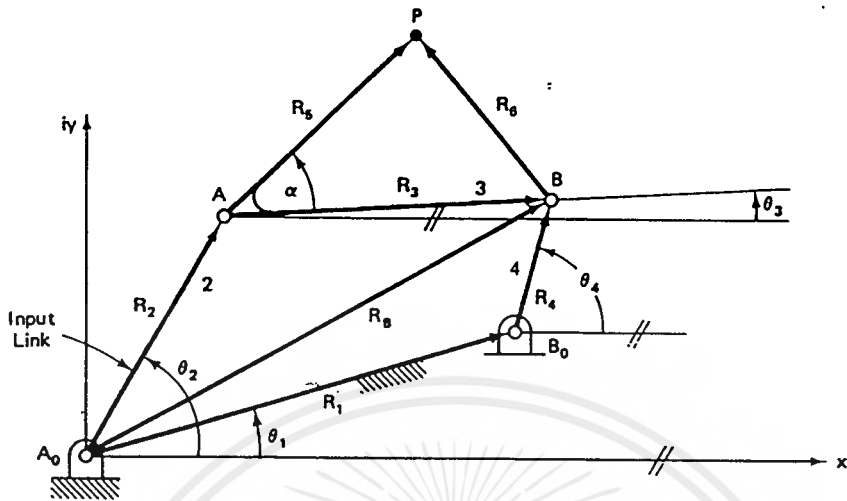
หรือสามารถที่จะจัดรูปใหม่ได้

$$\alpha_j = \frac{A'_{BA}}{R_{BA}} \quad (1.17)$$

ลักษณะการกำหนดเครื่องหมายของความเร่งเชิงมุม จะกำหนดให้ทิศทางการเคลื่อนที่ทวนเข็มนาฬิกาให้เป็นบวก ส่วนการเคลื่อนที่ตามเข็มนาฬิกาจะให้เป็นลบ

การหา velocity และ acceleration โดยวิธี Analytical

เนื่องจากการใช้โปรแกรมคอมพิวเตอร์เข้ามามีส่วนช่วยในการคำนวณบางครั้งจะประสบปัญหาความยุ่งยากซับซ้อนในการจะหาความเร็วและความเร่งโดยวิธีปกติ (จะเรียกวิธีนี้ว่า Graphical Method) จึงได้มีการคิดค้นวิธีที่จะสามารถหามาให้ได้ซึ่งความเร็วและความเร่งให้ง่ายขึ้น ซึ่งวิธีหนึ่งก็คือการใช้วิธี Analytical



รูปที่ 1.6 เวกเตอร์ตำแหน่งของ four-bar linkage

พิจารณาลักษณะของ four-bar linkage ในรูปที่ 1.6 เราสามารถที่จะหาเวกเตอร์ตำแหน่ง B

จาก

$$\mathbf{R}_B = \mathbf{R}_2 + \mathbf{R}_3 \quad (1.18)$$

$$\mathbf{R}_B = \mathbf{R}_1 + \mathbf{R}_4 \quad (1.19)$$

จากสมการ (1.18), (1.19) สามารถที่เขียนให้อยู่ในรูป Polar form

$$\mathbf{R}_B = r_2 e^{i\theta_2} + r_3 e^{i\theta_3} = r_1 e^{i\theta_1} + r_4 e^{i\theta_4} \quad (1.20)$$

ทำการ derivative สมการ (1.20) เทียบกับ time เพื่อหาความเร็วได้เป็น

$$\mathbf{V}_B = r_2 \omega_2 i e^{i\theta_2} + r_3 \omega_3 i e^{i\theta_3} = r_4 \omega_4 i e^{i\theta_4} \quad (1.21)$$

ทำการกระจายสมการ (1.21) แล้วแยกส่วนแยกและส่วนจินตภาพ เราจะได้

$$r_2 \omega_2 \sin \theta_2 + r_3 \omega_3 \sin \theta_3 = r_4 \omega_4 \sin \theta_4 \quad (1.22)$$

$$r_2 \omega_2 \cos \theta_2 + r_3 \omega_3 \cos \theta_3 = r_4 \omega_4 \cos \theta_4 \quad (1.23)$$

จากสมการ (1.22) และ (1.23) เราสามารถที่จะหา ω_3 และ ω_4 โดยการจัดรูปแบบสมการแล้วใช้ Cramer's laws จะได้

$$\omega_3 = \frac{r_2 \omega_2 \sin \theta_2 r_4 \cos \theta_4 - r_2 \omega_2 \cos \theta_2 r_4 \sin \theta_4}{r_3 \cos \theta_3 r_4 \sin \theta_4 - r_3 \sin \theta_3 r_4 \cos \theta_4} \quad (1.24)$$

และ

$$\omega_4 = \frac{r_2 \omega_2 \sin \theta_2 r_3 \cos \theta_3 - r_2 \omega_2 \cos \theta_2 r_3 \sin \theta_3}{r_3 \cos \theta_3 r_4 \sin \theta_4 - r_3 \sin \theta_3 r_4 \cos \theta_4} \quad (1.25)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเราได้ ω_3 และ ω_4 แล้วขั้นต่อไปเป็นการวิเคราะห์หาความเร่งต่อไป โดยการ derivative สมการ (1.21) เทียบกับ time จะได้ว่า

$$A_B = (r_2\alpha_2 e^{i\theta_2} - r_2\omega_2^2 e^{i\theta_2}) + (r_3\alpha_3 e^{i\theta_3} - r_3\omega_3^2 e^{i\theta_3}) = (r_4\alpha_4 e^{i\theta_4} - r_4\omega_4^2 e^{i\theta_4}) \quad (1.26)$$

จากนั้นทำการกระจายเพื่อแยกส่วนจริงและจินตภาพออกจากกันเป็น

$$r_2\alpha_2 \sin \theta_2 + r_2\omega_2^2 \cos \theta_2 + r_3\alpha_3 \sin \theta_3 + r_3\omega_3^2 \cos \theta_3 = r_4\alpha_4 \sin \theta_4 + r_4\omega_4^2 \cos \theta_4 \quad (1.27)$$

$$r_2\alpha_2 \cos \theta_2 - r_2\omega_2^2 \sin \theta_2 + r_3\alpha_3 \cos \theta_3 - r_3\omega_3^2 \sin \theta_3 = r_4\alpha_4 \cos \theta_4 - r_4\omega_4^2 \sin \theta_4 \quad (1.28)$$

ต่อมานำ ((1.27) * $\cos \theta_4$) - ((1.28) * $\sin \theta_4$) เราจะสามารถจัดรูปสมการเพื่อหาค่า α_3 ได้เป็น

$$\alpha_3 = \frac{-r_2\alpha_2 \sin(\theta_4 - \theta_2) + r_2\omega_2^2 \cos(\theta_4 - \theta_2) + r_3\omega_3^2 \cos(\theta_4 - \theta_3) - r_4\omega_4^2}{r_3 \sin(\theta_4 - \theta_3)} \quad (1.29)$$

หลังจากนั้นนำ ((1.27) * $\cos \theta_3$) - ((1.28) * $\sin \theta_3$) เราก็จะสามารถที่จะหาค่า α_4 ได้เช่นเดียวกัน

$$\alpha_4 = \frac{r_2\alpha_2 \sin(\theta_3 - \theta_2) - r_2\omega_2^2 \cos(\theta_3 - \theta_2) + r_4\omega_4^2 \cos(\theta_3 - \theta_4) - r_3\omega_3^2}{r_4 \sin(\theta_3 - \theta_4)} \quad (1.30)$$

ส่วนที่ 3 Force Analysis

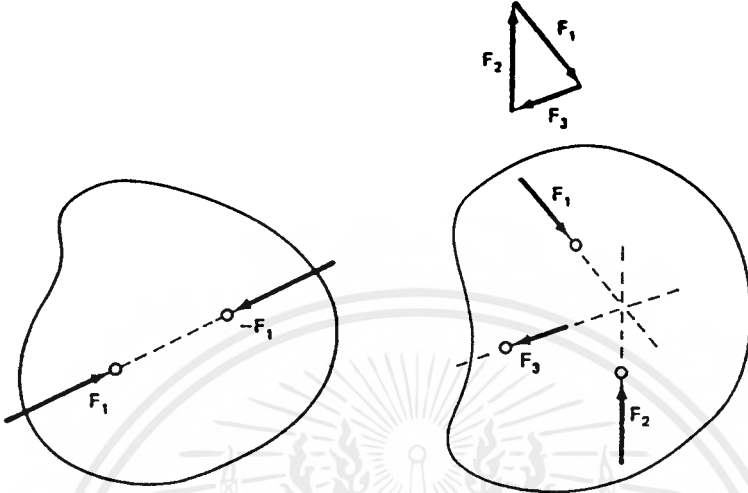
แรงที่กระทำต่อ four-bar linkage นั้นเราสามารถที่จะแบ่งออกได้เป็น 2 ลักษณะ คือ

1. **Static Force** เป็นแรงที่เกิดขึ้นอันเนื่องมาจาก น้ำหนักชิ้นส่วน, แรงจากส่วนประกอบของชิ้นส่วนต่าง ๆ , แรงจากการรับภาระและแรงที่เกิดจากการถ่ายเทพลังงาน ซึ่ง static force นี้จะต้องเป็นไปตามกฎพื้นฐานทาง mechanism ดังนี้

1. ถ้ามีแรง 2 แรงกระทำกับ rigid body จะต้องเป็นไปตาม static equilibrium โดยแรง 2 แรงนั้นจะต้องมีทิศทางตรงข้ามกันกระทำที่ตำแหน่งเดียวกัน รวมทั้งต้องมีขนาดที่เท่ากันด้วย

(เป็นไปตามลักษณะรูปที่ 1.7)

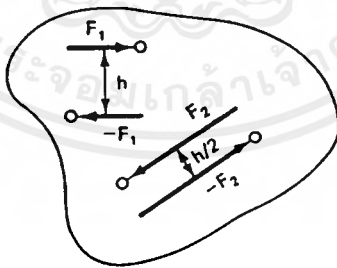
2. ถ้าเป็นแรง 3 แรงกระทำกับ rigid body จะต้องเป็นไปตาม static equilibrium โดยแรงทั้ง 3 แรงนี้จะต้องกระทำผ่านตำแหน่งเดียวกัน และผลรวมของแรงลัพธ์จะต้องเท่ากับศูนย์ คือเมื่อรวมแรงแล้วจะต้องเป็นลักษณะของสามเหลี่ยมปิด (ตามรูปที่ 1.8)



รูปที่ 1.7 วัตถุเป็นไปตามกฎข้อที่ 1

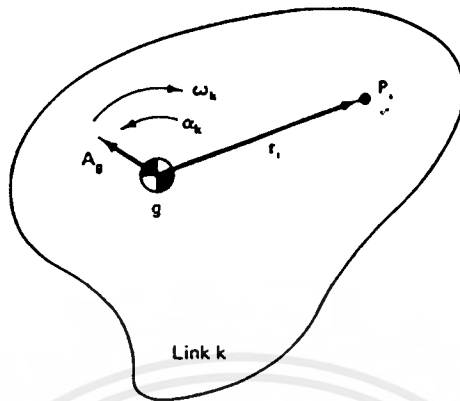
รูปที่ 1.8 วัตถุเป็นไปตามกฎข้อที่ 2

3. ถ้ามีแรงคู่ควบมากระทำต่อ rigid body จะต้องเป็นไปตาม static equilibrium โดยต้องแรงคู่ควบอื่นมากระทำด้วยและให้มีขนาดเท่ากับแรงคู่ควบคู่แรงแต่ต้องทิศทางที่ตรงข้ามกัน (ตามรูปที่ 1.9)



รูปที่ 1.9 วัตถุเป็นไปตามกฎข้อที่ 3

2. **Dynamic Force** เป็นการนำแรงอันเนื่องมาจากความเร่งหรือที่เรียกว่า แรงเฉื่อย (Inertia Force) มาทำการวิเคราะห์ โดยหลักการวิเคราะห์ยังคงเป็นไปตามกฎของนิวตัน



รูปที่ 1.10 ลักษณะ link k ในระนาบใด

พิจารณารูปที่ 1.10 กำหนดให้ link k มีความเร็วเชิงมุม ω_k และมีความเร่งเชิงมุม α_k และให้ตำแหน่ง g แทนศูนย์กลางมวลซึ่งมีความเร่ง A_g จากสมการความเร่งสัมพัทธ์จะได้ว่า

$$A_{P_i}'' + A_{P_i}' = A_g + A_{(P_i)g}'' + A_{(P_i)g}' \quad (1.31)$$

เราทราบว่า $A_{(P_i)g}'' = -r_i \omega_k^2$ และมีทิศทางจาก P_i ไปยัง g, ส่วน $A_{(P_i)g}' = r_i \alpha_k e^{i\pi/2}$ และมีทิศทางที่ตั้งฉากกับความเร่งเข้าสู่ศูนย์กลาง

จากกฎข้อที่ 2 ของนิวตันเราสามารถที่จะพิจารณาพื้นที่ย่อย P_i ได้จาก

$$\frac{dM_{P_i}}{dt} = m_i \frac{dV_{P_i}}{dt} = m_i A_{P_i} = m_i A_g - m_i r_i \omega_k^2 + m_i r_i \alpha_k e^{i\pi/2} = F_i \quad (1.32)$$

เมื่อ m_i คือ มวลย่อย ๆ และ M คือ โมเมนต์จัม จากนั้นเราสามารถที่จะหาผลรวมของแรงย่อยได้เป็น

$$F = \sum_i F_i = A_g \sum_i m_i - \omega_k^2 \sum_i m_i r_i + \alpha_k e^{i\pi/2} \sum_i m_i r_i \quad (1.33)$$

จากสมการ (1.33) จะสามารถสังเกตเห็นว่า

$$\sum_i m_i = M \quad \text{ซึ่งก็คือมวลทั้งหมดของ link k} \quad (1.34)$$

และ

$$\sum_i m_i r_i = 0 \quad (1.35)$$

ดังนั้นสมการ (1.33) จึงสามารถลดรูปได้เป็น

$$F = M A_g \quad (1.36)$$

จากนั้นทำการรวม โมเมนต์ของแรงรวมตำแหน่งศูนย์กลางถ่วง (CG) จะได้เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับวิชาการเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$T = \sum_i m_i r_i \alpha_k r_i = \alpha_k \sum_i m_i r_i^2 \quad (1.37)$$

จากสมการ (1.37) ทางขวามือของสมการจะเรียกว่า mass moment of inertia รอบจุดศูนย์กลางถ่วง (I_g) ทำให้สมการ (1.37) สามารถที่จะเขียนใหม่ได้เป็น

$$T = I_g \alpha_k \quad (1.38)$$

ดังนั้นเราสามารถที่จะหา dynamic equilibrium ได้เป็น

$$\sum F_x = MA_{gx} \quad (1.39)$$

$$\sum F_y = MA_{gy} \quad (1.40)$$

$$\sum T = I_g \alpha_k \quad (1.41)$$

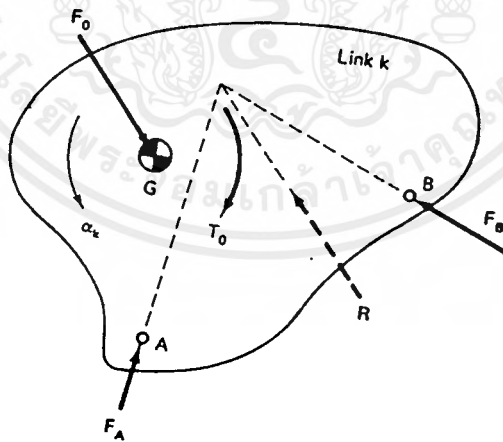
เราสามารถที่จะรวม inertia forces, inertia torques, external forces, external torques เข้าด้วยกันจะกลายเป็นสมการ

$$\sum F_x + (-MA_{gx}) = 0 \quad (1.42)$$

$$\sum F_y + (-MA_{gy}) = 0 \quad (1.43)$$

$$\sum T + (-I_g \alpha_k) = 0 \quad (1.44)$$

1.1 ลักษณะของ Inertia Force ใน linkage



รูปที่ 1.11 การหา dynamic equilibrium

พิจารณารูปที่ 1.11 ถ้ากำหนดให้ link k มีความเร่ง A_g และกำหนดให้มีความเร่งเชิงมุม α_k นอกจากนั้นมีสลักข้อต่อ A และ B ซึ่งมีแรงที่ไม่ทราบค่า F_A และ F_B ซึ่งมีแรงลัพธ์เท่ากับ R และเพื่อความสะดวกกำหนดให้แกน y มีทิศทางเช่นเดียวกับ A_g จากสมการที่ (1.42) และ (1.43) จะ

ได้ว่า
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$R_y - mA_{gy} = 0 \quad (1.45)$$

$$R_x = 0 \quad (1.46)$$

Inertia force F_o เป็นแรงเฉื่อยของ link k ซึ่งมีขนาดเท่ากับ

$$F_o = -mA_g \quad (1.47)$$

ดังนั้นสมการจะกลายเป็น

$$R + F_o = 0 \quad (1.48)$$

ขั้นตอนต่อไปเป็นการหาสมการ dynamic equilibrium ให้ครบถูกต้องก็ขาดเพียงแต่สมการ moment ซึ่งเราจะกำหนดให้ F_o กระทำห่างจากตำแหน่ง g เป็นระยะ ε_{kx} จากนั้นก็ทำการหา moment รอบตำแหน่ง g ได้เป็น

$$R_y \varepsilon_{kx} - I_g \alpha_k = 0 \quad (1.49)$$

จัดรูปสมการใหม่ได้

$$\varepsilon_k = \varepsilon_{kx} = \frac{I_g \alpha_k}{R_y} = \frac{I_g \alpha_k}{mA_{gy}} \quad (1.50)$$

1.2 วิธีในการคำนวณหาแรง

สำหรับวิธีในการคำนวณหาแรงที่กระทำกับเครื่องค้อนนั้น ในโครงการชิ้นนี้จะมีวิธีอยู่ 2 วิธี คือ

1. The Superposition Method
2. The Matrix Method

โดยทั้ง 2 วิธีนี้จะมีการกำหนดสมมติฐานให้ไม่มีแรงเสียดทานกระทำที่จุดเชื่อมต่อ (joints) และกำหนดให้ความเร่ง ณ ตำแหน่งจุดศูนย์กลางมวลมีลักษณะเป็นเชิงเส้น

วิธีที่ 1 The Superposition Method

หลักสำคัญในการพิจารณาลักษณะการคำนวณลักษณะนี้ก็คือเป็นการวิเคราะห์ในลักษณะที่ว่า ผลของ Inertia Force ในแต่ละเครื่องค้อนนั้นจะเป็นเอกเทศ (individual) สามารถที่จะแยกทำการวิเคราะห์ได้ และเมื่อเราแยกทำการวิเคราะห์แล้วก็สามารถที่จะนำมาวิเคราะห์ผลของ Inertia Force ของแต่ละเครื่องค้อนมาทำการวิเคราะห์รวมกันได้ ซึ่งก็จะสามารถคำนวณได้แรงที่ถูกดึงออกไปได้

จะเห็นได้จากรูปที่ 1.12 แสดงให้เห็น Four-bar Linkage ที่ถูกกระทำด้วย Inertia Force F_{o3} เพียงแรงเดียว ทำให้เราสามารถคำนวณหาแรงที่กระทำระหว่างเครื่องค้อนตลอดจนสามารถคำนวณหาทอร์คซึ่งเกิดจาก F_{o3} เพียงแรงเดียว ซึ่งในลักษณะเดียวกันทำให้เราสามารถคำนวณหาผลของ Inertia Force อันเกิดมาจาก F_{o2} และ F_{o4} ได้ ดังนั้นตอนที่แสดงให้เห็นดังต่อไปนี้

เนื้อหาทั้งหมดเอกสารทั้งหมดนี้จัดทำขึ้นเพื่อการศึกษาเท่านั้น เมื่อผู้ใดนำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 1 การคำนวณหาผลของแรงจาก F_{03}

จะเห็นว่าในเครื่องต่อ 3 (รูปที่ 1.13b) มี F_{03} กระทำเพียงแรงเดียว ดังนั้นหลังจากที่เราสามารถคำนวณหาความเร่งที่ตำแหน่งศูนย์กลางมวล $A_{g3} = (A_{g3}e^{i\beta_3})$ พร้อมทั้งสามารถหาค่าของความเร่งเชิงมุม α_3 ได้แล้ว ทำให้เราสามารถที่จะคำนวณหาเวกเตอร์ของ Inertia Force ได้จาก

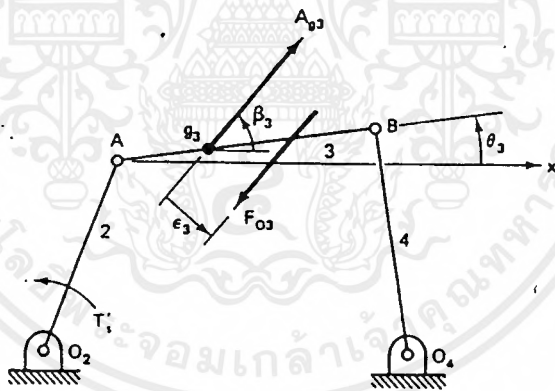
$$F_{03} = (m_3 A_{g3})e^{i(\beta_3+\pi)} \quad (1.51)$$

โดยที่ $(\beta_3 + \pi)$ แสดงทิศทางของ F_{03} ซึ่งมีทิศทางตรงกันข้ามกับทิศทางของ A_{g3} ซึ่งมีทิศทาง β_3 และเนื่องจากเราสามารถหาค่าของความเร่งเชิงมุม และจากสมการสมดุลทำให้แนวแรง F_{03} เกิดระยะ offset ($\epsilon_3 = |I_3 \alpha_3 / F_{03}|$) ซึ่งมีสมการในรูปเชิงซ้อนคือ

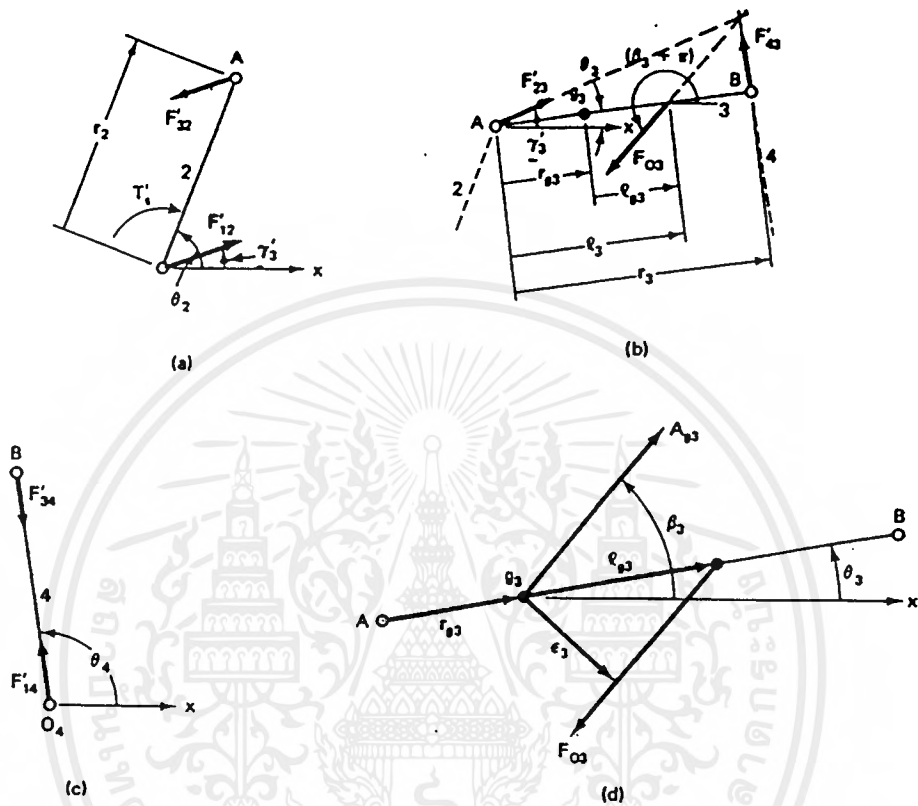
$$\epsilon_3 = -\frac{\alpha_3}{|\alpha_3|} \epsilon_3 i e^{i\beta_3}$$

ซึ่ง $i e^{i\beta_3}$ จะแสดงทิศทางของ ϵ_3 โดยที่จะตั้งฉากกับ A_{g3} และ $-\alpha_3 / |\alpha_3|$ จะแสดงเครื่องหมาย ดังนั้นจากหัวข้อที่แล้วทำให้เราได้สมการคือ

$$\epsilon_3 = -\frac{\alpha_3}{|\alpha_3|} \left| \frac{I_3 \alpha_3}{F_{03}} \right| i e^{i\beta_3} = -\frac{I_3 \alpha_3}{F_{03}} i e^{i\beta_3}$$



รูปที่ 1.12 ผลของแรงเฉื่อยเพียงแรงเดียวที่กระทำกับวัตถุ



รูปที่ 1.13 ผลของแรงเฉื่อยทำให้เกิดแรงระหว่างเครื่องต่อ

เพื่อความสะดวกเราสามารถที่จะหาตำแหน่งที่แนวแรง F_{03} กระทำที่เครื่องต่อ 3 ได้จากรูปที่ 1.13b ซึ่งจะเห็นได้ว่า

$$l_{g3} = r_{g3} + l_{g3}$$

ดังนั้นเราจะได้ว่า

$$\epsilon_3 = l_{g3} \cos(\theta_3 + \frac{\pi}{2} - \beta_3) e^{-i[\theta_3 + (\pi/2) - \beta_3]}$$

$$\epsilon_3 = -l_{g3} \sin(\theta_3 - \beta_3) e^{-i[\theta_3 + (\pi/2) - \beta_3]}$$

$$l_{g3} = \frac{-\epsilon_3}{\sin(\theta_3 - \beta_3)} e^{i[\theta_3 + (\pi/2) - \beta_3]} = + \frac{I_3 \alpha_3 i e^{i\beta_3} e^{i[\theta_3 + (\pi/2) - \beta_3]}}{F_{03} \sin(\theta_3 - \beta_3)}$$

$$l_{g3} = \frac{-I_3 \alpha_3 e^{i\theta_3}}{F_{03} \sin(\theta_3 - \beta_3)}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\text{หรือ} \quad l_3 = \left[r_{g3} - \frac{I_3 \alpha_3 / F_{03}}{\sin(\theta_3 - \beta_3)} \right] e^{i\theta_3} \quad (1.52)$$

หลังจากนั้นจะเห็นได้ว่า ณ เครื่องต่อ 3 มีแรง 3 แรงกระทำ ได้แก่ F_{03} ซึ่งเป็นแรงที่เราทราบค่า ส่วนแรงที่เหลืออีก 2 แรง ได้แก่ F'_{23} และ F'_{43} ซึ่งเป็นแรงกระทำระหว่างเครื่องต่อที่เราไม่ทราบค่า ดังนั้นจากรูปที่ 1.13c พิจารณาเครื่องต่อ 4 กำหนดให้ไม่มีผลของมวล จะเห็นได้ว่ามีแรง 2 แรงกระทำทำให้เราสามารถที่จะพิจารณาได้ว่าเป็นลักษณะของ two-force member และทำให้สามารถที่จะกำหนดได้ว่าให้ F'_{43} กระทำที่มุม θ_4 ดังนั้นจากสมการสมดุลย์ทำให้เราได้ว่า

$$F'_{23} + F'_{43} + F_{03} = 0 \quad (1.53)$$

ซึ่งเราสามารถที่จะแสดงให้เห็นจากรูปที่ 1.14 และ รูปที่ 1.15 หลังจากนั้นเราก็เขียนให้อยู่ในรูปสมการเอกซ์โปเนนเชียลได้

$$F'_{23} (e^{i\gamma_3}) + F'_{43} (e^{i\theta_4}) + F_{03} [e^{i(\beta_3 + \pi)}] = 0 \quad (1.54)$$

ทำการแยกส่วนจริงและส่วนจินตภาพจาก (1.54) เราจะได้ว่า

$$F'_{23} \cos \gamma_3 + F'_{43} \cos \theta_4 + F_{03} \cos(\beta_3 + \pi) = 0 \quad (1.55)$$

$$F'_{23} \sin \gamma_3 + F'_{43} \sin \theta_4 + F_{03} \sin(\beta_3 + \pi) = 0 \quad (1.56)$$

ในการคำนวณสมการ 3 ตัวแปรเราจำเป็นต้องมีอีก 1 สมการมาช่วยก็คือสมการสมดุลย์ของโมเมนต์ โดยกำหนดให้จุดหมุนอยู่ที่จุด A ดังนั้นจะได้

$$F'_{43} r_3 \sin(\theta_4 - \theta_3) + F_{03} l_3 \sin(\beta_3 + \pi - \theta_3) = 0 \quad (1.57)$$

$$\text{หรือ} \quad F'_{43} = -F_{03} \frac{l_3 \sin(\beta_3 + \pi - \theta_3)}{\sin(\theta_4 - \theta_3)}$$

ทำให้เราสามารถที่จะคำนวณหาค่าของ F'_{23} (ซึ่งมีส่วนประกอบเป็น F'_{23x} และ F'_{23y}) ได้เป็น

$$F'_{23x} = F'_{23} \cos \gamma_3 = -F'_{43} \cos \theta_4 - F_{03} \cos(\beta_3 + \pi) \quad (1.58)$$

$$F'_{23y} = F'_{23} \sin \gamma_3 = -F'_{43} \sin \theta_4 - F_{03} \sin(\beta_3 + \pi) \quad (1.59)$$

โดยที่ F'_{23x} และ F'_{23y} อาจจะเป็นเครื่องหมายบวกหรือ ลบก็ได้ ดังนั้นเราจึงหาค่าของ F'_{23} ได้จาก

$$F'_{23} = \left| \sqrt{(F'_{23x})^2 + (F'_{23y})^2} \right| \quad (1.60)$$

$$\gamma_3 = \arg(F'_{23x} + iF'_{23y}) \quad (1.61)$$

และเราสามารถที่จะหาค่าทอร์คได้โดย

$$\begin{aligned} T'_S &= -F'_{12} r_2 \sin(\gamma_3 - \theta_2 + \pi) \\ &= -F'_{23} r_2 \sin(\theta_2 - \gamma_3) \end{aligned} \quad (1.62)$$

ขั้นตอนที่ 2 การคำนวณหาผลของแรง F_{04}

ลักษณะการคำนวณจะเหมือนกับขั้นตอนแรก โดยเราสามารถที่จะหา Inertia Force (F_{04}) ของเครื่องต่อ 4 ได้จาก (รูปที่ 1.16)

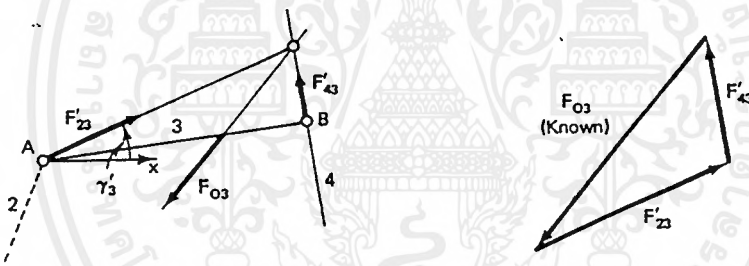
$$F_{04} = (m_4 A_{g4}) e^{i(\beta_4 + \pi)} \quad (1.63)$$

และระยะ offset ของ F_{04} จาก

$$\varepsilon_4 = -\frac{I_4 \alpha_4}{F_{04}} i e^{i\beta_4} \quad (1.64)$$

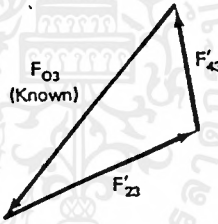
และตำแหน่งที่แนวแรง F_{04} กระทำที่เครื่องต่อ 4 ได้จาก

$$l_4 = r_{g4} + \frac{\varepsilon_4 (\alpha_4 / |\alpha_4|) e^{i\theta_4}}{\sin(\beta_4 - \theta_4)} = r_{g4} + \frac{I_4 \alpha_4 / F_{04}}{\sin(\beta_4 - \theta_4)} e^{i\theta_4} \quad (1.65)$$



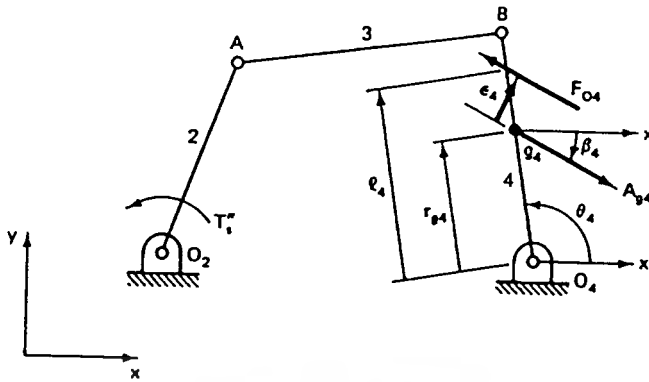
รูปที่ 1.14 การคำนวณเพื่อหาทิศทาง F'_{23}

รูปที่ 1.15 การคำนวณเพื่อหาค่า F'_{23} และ F'_{43}



จากสมการสมดุลย์

$$F'_{34} + F'_{14} + F_{04} = 0 \quad (1.66)$$



รูปที่ 1.16 การหาค่าผลของแรงที่กระทำต่อเครื่องต่อ 4

จัดรูปสมการให้อยู่ในรูปสมการเอกซ์โปเนนเชียลจะได้

$$F''_{34}e^{i\theta_3} + F''_{14}e^{i\gamma_4} + F_{04}e^{i(\beta_4+\pi)} = 0 \quad (1.67)$$

จาก (1.67) ทำการแยกส่วนจริงและส่วนจินตภาพ โดยแสดงดังรูปที่ 1.17

$$F''_{34} \cos \theta_3 + F''_{14} \cos \gamma_4 + F_{04} \cos(\beta_4 + \pi) = 0 \quad (1.68)$$

$$F''_{34} \sin \theta_3 + F''_{14} \sin \gamma_4 + F_{04} \sin(\beta_4 + \pi) = 0 \quad (1.69)$$

และจากสมการสมดุลย์โมเมนต์รอบจุด O_4 เราจะได้

$$F''_{34}r_4 \sin(\theta_3 - \theta_4) + F_{04}l_4 \sin(\theta_4 - \beta_4) = 0 \quad (1.70)$$

หรือว่าสามารถที่จะจัดรูปเป็น

$$F''_{34} = -F_{04} \frac{l_4 \sin(\theta_4 - \beta_4)}{r_4 \sin(\theta_3 - \theta_4)} \quad (1.71)$$

ทำให้เราสามารถแทนค่า F''_{34} เพื่อที่จะหาค่าตอบของ F''_{14} สมการ (1.68) และ (1.69)

$$F''_{14x} = F''_{14} \cos \gamma_4 = -F_{04} \cos(\beta_4 + \pi) - F''_{34} \cos \theta_3 \quad (1.72)$$

$$F''_{14y} = F''_{14} \sin \gamma_4 = -F_{04} \sin(\beta_4 + \pi) - F''_{34} \sin \theta_3 \quad (1.73)$$

ดังนั้นจะได้ว่า

$$F''_{14} = \left| \sqrt{(F''_{14x})^2 + (F''_{14y})^2} \right| \quad (1.74)$$

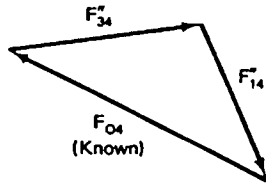
และ

$$\gamma_4 = \arg(F''_{14x} + iF''_{14y}) \quad (1.75)$$

จากรูปที่ 1.18b เราสามารถจะเห็นได้ว่า

$$\mathbf{F}''_{43} = -\mathbf{F}''_{34} \quad (1.76)$$

$$\mathbf{F}''_{32} = -\mathbf{F}''_{23} = -\mathbf{F}''_{34} \quad (1.77)$$

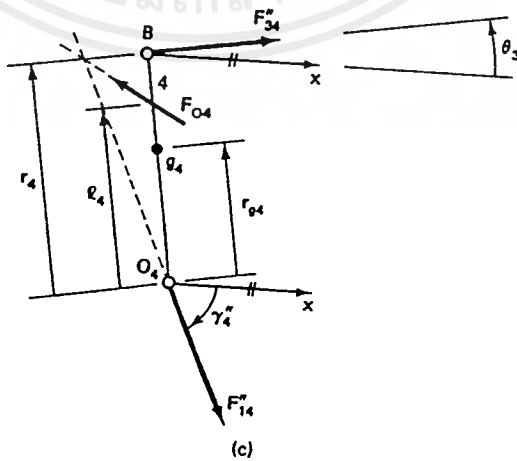
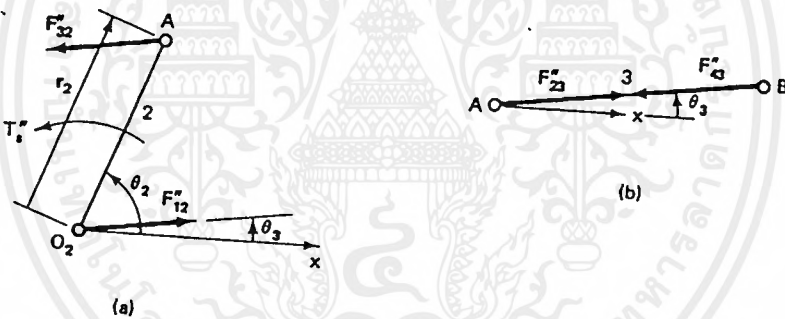


รูปที่ 1.17 การหาค่า F_{34} และ F_{14}

หาค่าของโมเมนต์รอบจุด O_2 ของเครื่องต่อ 2 (รูปที่ 1.18a)

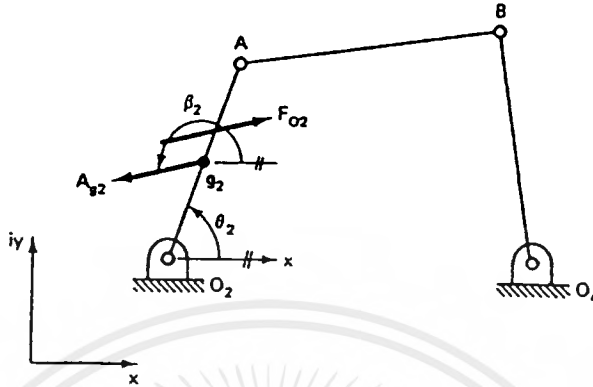
$$T''_s = -F''_{32} r_2 \sin(\theta_3 - \theta_2) \quad (1.78)$$

ขั้นตอนที่ 3 การคำนวณหาผลของแรง F_{02}



รูปที่ 1.18 ผลของแรงที่กระทำต่อเครื่องต่อ 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 1.19 เครื่องต่อ 2 มีแรงเฉื่อยกระทำเพียงแรงเดียว

จะเหมือนกับ 2 ขั้นตอนแรกคือ สามารถคำนวณหา F_{O2} ได้จาก (รูปที่ 1.19 และ 1.20)

$$F_{O2} = (m_2 A_{g2}) e^{i(\beta_2 + \pi)} \quad (1.79)$$

และตำแหน่งที่แนวแรงกระทำบนเครื่องต่อ 2 ได้เป็น

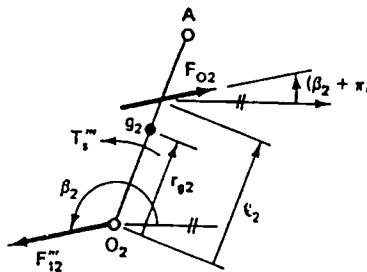
$$l_2 = r_{g2} + \frac{\epsilon_3 (\alpha_2 / |\alpha_2|) e^{i\theta_2}}{\sin(\beta_2 - \theta_2)} = r_{g2} + \frac{I_2 \alpha_2 / F_{O2}}{\sin(\beta_2 - \theta_2)} e^{i\theta_2} \quad (1.80)$$

ในการวิเคราะห์นี้จะเป็นการง่ายถ้าเราสามารถกำหนดให้แรงที่กระทำกันระหว่างเครื่องต่อเท่ากับศูนย์ ทั้งเครื่องต่อ 3 และเครื่องต่อ 4 และเราก็พิจารณาถึงว่าผลของมวลไม่มีผลต่อการคำนวณ ซึ่งจะทำให้เราได้

$$F''_{12} = - F_{O2} \quad (1.81)$$

และทำให้เราสามารถหาค่าทอร์คที่กระทำที่เครื่องต่อ 2 ได้เป็น (ซึ่ง ได้แสดงไว้ในรูปที่ 1.20)

$$T''_s = F_{O2} l_2 \sin(\beta_2 - \theta_2) \quad (1.82)$$



รูปที่ 1.20 สมดุลย์ของแรงและทอร์คของเครื่องต่อ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 4 การคำนวณหาผลของแรงทั้งหมด

เราสามารถที่จะหาผลของแรงที่กระทำทั้งหมดของเครื่องต่อทั้ง 3 เครื่องต่อได้จากการนำ 3 ขั้นตอนข้างต้นมาทำการรวมกัน ซึ่งก็จะ ได้เป็น (รูปที่ 1.21)

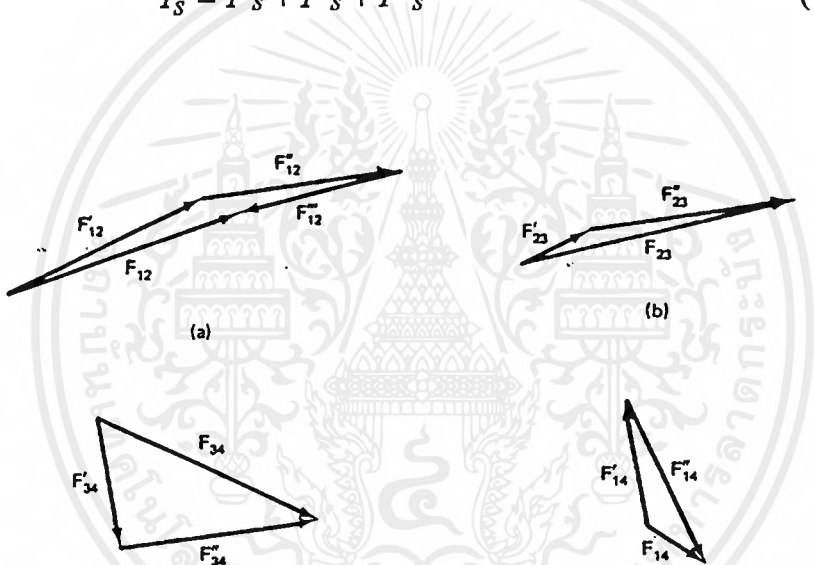
$$\mathbf{F}_{14} = \mathbf{F}'_{14} + \mathbf{F}''_{14} \quad (1.83)$$

$$\mathbf{F}_{34} = -\mathbf{F}'_{43} + \mathbf{F}''_{34} \quad (1.84)$$

$$\mathbf{F}_{23} = \mathbf{F}'_{23} + \mathbf{F}''_{23} \quad (1.85)$$

$$\mathbf{F}_{12} = \mathbf{F}'_{12} + \mathbf{F}''_{12} + \mathbf{F}'''_{12} \quad (1.86)$$

$$\mathbf{T}_S = \mathbf{T}'_S + \mathbf{T}''_S + \mathbf{T}'''_S \quad (1.87)$$



รูปที่ 1.21 การหาผลลัพธ์ของแรงและทอร์คทั้งหมด

จากสมการทั้งหมดดังกล่าว ((1.83) ถึง (1.87)) จะทำให้เราสามารถที่จะหาแรงที่กระทำกันระหว่างเครื่องต่อ และทอร์คได้ในที่สุดนั่นเอง

วิธีที่ 2 Matrix Method

การคำนวณหาแรงในลักษณะนี้จะใช้เวลาที่น้อยกว่าการใช้วิธี The Superposition Method ทำให้เกิดความเหมาะสมที่จะใช้ช่วยในการหาแรงที่กระทำสำหรับ four-bar linkage เนื่องจากการพิสูจน์จะใช้ในลักษณะของ four-bar linkage เป็นสำคัญ ดังนั้นจากสมการสมมูลย์ที่เครื่องต่อ 2 เราจะได้ว่า (รูปที่ 1.22a)

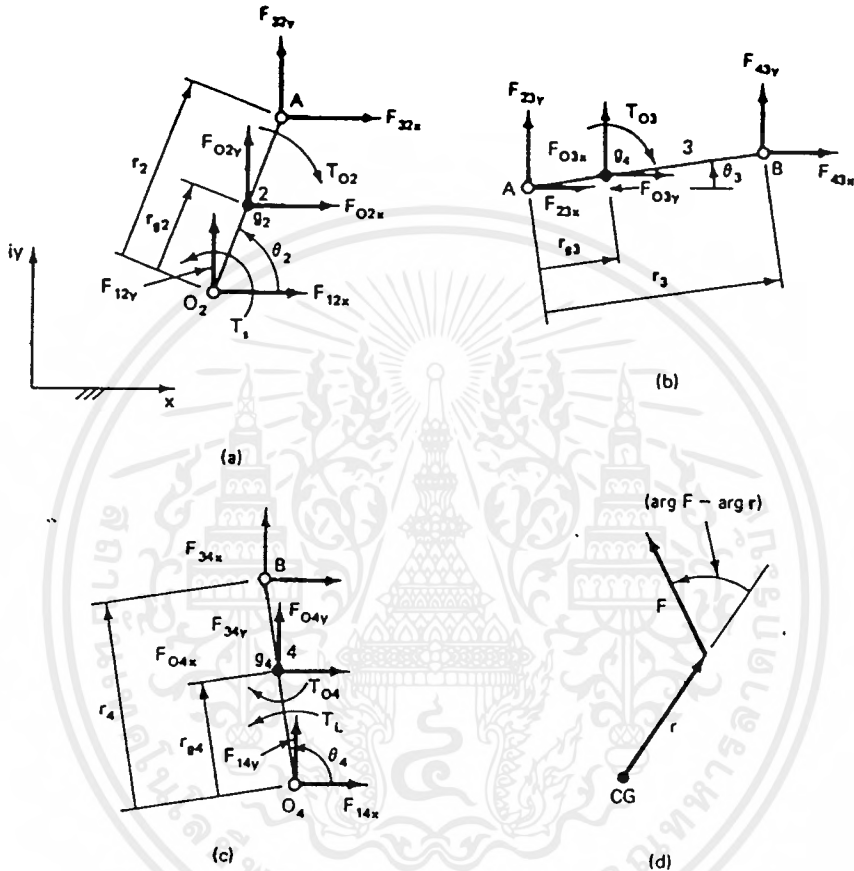
$$F_{12x} + F_{32x} + F_{02x} = 0 \quad (1.88)$$

$$F_{12y} + F_{32y} + F_{02y} = 0 \quad (1.89)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และจากสมการสมมูลย์โมเมนต์ ซึ่งเราสามารถที่จะเขียนให้อยู่ในรูปใด ๆ (โดยที่ไม่ต้องทำการวิเคราะห์ว่าแรงใดทำให้เกิดโมเมนต์ใด) รอบจุดหมุนใด ๆ ได้เป็น

$$M_F = r_x F_y - r_y F_x \quad (1.90)$$



รูปที่ 1.22 (a),(b),(c) ผลของแรงที่กระทำเครื่องต่อต่าง ๆ (d) ลักษณะของเวกเตอร์โมเมนต์

ซึ่งเวกเตอร์ r เป็นเวกเตอร์ระยะห่างจากจุดหมุนใด ๆ ไปยังแนวแรง ดังนั้นจากรูปที่ ทำให้เราสามารถที่จะ โมเมนต์ของ F_{12} รอบจุด g_2 ได้เป็น (รูปที่ 1.22d)

$$\begin{aligned} M_{F_{12}/g_2} &= (-r_{g_2}) \times F_{12} = (-r_{g_2x})F_{12y} - (-r_{g_2y})F_{12x} \\ &= F_{12x}r_{g_2} \sin \theta_2 - F_{12y}r_{g_2} \cos \theta_2 \end{aligned}$$

และจากหลักการเดียวกันทำให้เราสามารถที่จะหาสมการ โมเมนต์บนเครื่องต่อ 2 ได้

$$T_{02} + T_S + F_{12x}r_{g_2} \sin \theta_2 - F_{12y}r_{g_2} \cos \theta_2 - F_{32x}(r_2 - r_{g_2}) \sin \theta_2 + F_{32y}(r_2 - r_{g_2}) \cos \theta_2 = 0 \quad (1.91)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่ T_S ได้แก่ทอร์คแรงจับ

และจากสมการสมดุลบนเครื่องต่อ 3 เราจะได้ว่า (รูปที่ 1.22b)

$$F_{23x} + F_{43x} + F_{03x} = 0 \quad (1.92)$$

$$F_{23y} + F_{43y} + F_{03y} = 0 \quad (1.93)$$

$$T_{03} + F_{23x}r_{g3} \sin \theta_3 - F_{23y}r_{g3} \cos \theta_3 - F_{43x}(r_3 - r_{g3}) \sin \theta_3 + F_{43y}(r_3 - r_{g3}) \cos \theta_3 = 0 \quad (1.94)$$

ณ เครื่องต่อ 4 เราก็สามารถที่จะหาสมการสมดุลได้เป็น (รูปที่ 1.22c)

$$F_{34x} + F_{14x} + F_{04x} = 0 \quad (1.95)$$

$$F_{34y} + F_{14y} + F_{04y} = 0 \quad (1.96)$$

$$T_{04} - F_{34x}(r_4 - r_{g4}) \sin \theta_4 + F_{34y}(r_4 - r_{g4}) \cos \theta_4 + T_L + F_{14x}r_{g4} \sin \theta - F_{14y}r_{g4} \cos \theta_4 = 0 \quad (1.97)$$

ซึ่งจากสมการ T_L เราจะแทนด้วยทอร์คซึ่งเกิดจากโหลดภายใน และจากสมการข้างบนดังกล่าวเราสามารถที่จะเห็นได้ว่า ระบบสมการนี้ประกอบด้วยแปดที่เราไม่ทราบค่าอยู่ทั้งหมด 9 ตัวแปรร่วมกัน ซึ่งก็ได้แก่ $F_{12x}, F_{12y}, F_{23x}, F_{23y}, F_{34x}, F_{34y}, F_{14x}, F_{14y}$ และ T_S โดยที่เราจะต้องทราบว่า $F_{jlx} = -F_{ljx}$ และ $F_{jly} = -F_{lly}$ ดังนั้นเพื่อความสะดวกจากสมการที่ (1.88) ถึง (1.97) เราสามารถที่จะจัดรูปใหม่ได้เป็น

$$F_{02x} = -F_{12x} + F_{23x} \quad (1.98)$$

$$F_{02y} = -F_{12y} + F_{23y} \quad (1.99)$$

$$T_{02} = -F_{12x}r_{g2} \sin \theta_2 + F_{12y}r_{g2} \cos \theta_2 - T_S - F_{23x}(r_2 - r_{g2}) \sin \theta_2 + F_{23y}(r_2 - r_{g2}) \cos \theta_2 \quad (1.100)$$

$$F_{03x} = -F_{23x} + F_{34x} \quad (1.101)$$

$$F_{03y} = -F_{23y} + F_{34y} \quad (1.102)$$

$$T_{03} = -F_{23x}r_{g3} \sin \theta_3 + F_{23y}r_{g3} \cos \theta_3 - F_{34x}(r_3 - r_{g3}) \sin \theta_3 + F_{34y}(r_3 - r_{g3}) \cos \theta_3 \quad (1.103)$$

$$F_{04x} = -F_{34x} - F_{14x} \quad (1.104)$$

$$F_{04y} = -F_{34y} - F_{14y} \quad (1.105)$$

$$T_{04} = F_{34x}(r_4 - r_{g4}) \sin \theta_4 - F_{34y}(r_4 - r_{g4}) \cos \theta_4 - F_{14x}r_{g4} \sin \theta_4 + F_{14y}r_{g4} \cos \theta_4 - T_L \quad (1.106)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และเพื่อความสะดวกต่อการจัดรูป เราจะกำหนดให้

$$R_2 \equiv r_2 - r_{g2}$$

$$R_3 \equiv r_3 - r_{g3}$$

$$R_4 \equiv r_4 - r_{g4}$$

เนื่องจากระบบสมการข้างต้นนี้เป็นระบบสมการเชิงเส้น ซึ่งมี 9 ตัวแปร 9 สมการ ทำให้เราสามารถที่จะหาค่าของตัวแปรที่เหลือได้ ในที่นี้เราจะจัดรูประบบสมการดังกล่าวให้อยู่ในรูปของ Matrix ได้โดยเราจะกำหนดให้

$$[F_1] = \begin{bmatrix} F_{02x} \\ F_{02y} \\ T_{02} \\ F_{03x} \\ F_{03y} \\ T_{03} \\ F_{04x} \\ F_{04y} \\ T_{04} + T_L \end{bmatrix}$$

$$[L] = \begin{bmatrix} -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ -r_{g2} \sin \theta_2 & r_{g2} \cos \theta_2 & -1 & -R_2 \sin \theta_2 & R_2 \cos \theta_2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -r_{g3} \sin \theta_3 & r_{g3} \cos \theta_3 & -R_3 \sin \theta_3 & R_3 \cos \theta_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & R_4 \sin \theta_4 & -R_4 \cos \theta_4 & -r_{g4} \sin \theta_4 & r_{g4} \cos \theta_4 & 0 \end{bmatrix}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$[F_B] = \begin{bmatrix} F_{12x} \\ F_{12y} \\ T_S \\ F_{23x} \\ F_{23y} \\ F_{34x} \\ F_{34y} \\ F_{14x} \\ F_{14y} \end{bmatrix}$$

ดังนั้นจะได้ว่า

$$[F_I] = [L][F_B] \quad (1.108)$$

โดยที่จะกำหนดให้

$[F_I]$ = เป็นเมตริกซ์หลัก (column matrix) ซึ่งเป็นการใส่ค่าที่เราทราบค่าเข้าไปจะประกอบด้วย inertia force และทอร์ค

$[L]$ = เป็นเมตริกซ์จัตุรัส (square matrix) ซึ่งจะประกอบด้วยค่าความยาวของเครื่องต่อต่าง ๆ และมุมของเครื่องต่อ

$[F_B]$ = เป็นเมตริกซ์เหมือนกับ $[F_I]$ แต่เป็นค่าที่เราต้องการทราบ ซึ่งก็คือแรงระหว่างเครื่องต่อ และทอร์ค (ซึ่งก็คือคำตอบของระบบสมการนั่นเอง)

และหลังจากที่เราจัดได้ระบบสมการที่อยู่ในรูปเมตริกซ์แล้ว เราจะสามารถหาค่าคำตอบของระบบสมการ ($[F_B]$) ได้โดยการคูณ $[L]^{-1}$ (ซึ่งก็คืออินเวอร์สของ $[L]$ นั่นเอง)

$$[L]^{-1}[F_I] = [L]^{-1}[L][F_B] \quad (1.109)$$

ซึ่งเราจะได้ว่า

$$[F_B] = [L]^{-1}[F_I] \quad (1.110)$$

ซึ่งจะทำให้เราสามารถที่จะคำนวณหาคำตอบของระบบสมการได้ต่อไป

บทที่ 2

หลักวิธีการเขียนโปรแกรมในการคำนวณ

สำหรับหลักวิธีการเขียนโปรแกรมการคำนวณในโครงการชิ้นนี้ เราจะสามารถแบ่งออกเป็น 2 ลักษณะ คือ หลักวิธีการเขียนโปรแกรมในการคำนวณลักษณะของ four-bar linkage และ หลักวิธีการเขียนโปรแกรมการคำนวณลักษณะของ slider crank ซึ่งในแต่ละลักษณะจะประกอบด้วยการคำนวณเพื่อที่จะหา ความเร็ว ความเร่ง และแรงที่กระทำกับวัตถุ ซึ่งก็จะมีลักษณะในการคำนวณที่แตกต่างกันไป

2.1 หลักการเขียนโปรแกรมในการคำนวณ four-bar linkage

ในการเขียนโปรแกรมการคำนวณลักษณะนี้จะประกอบด้วย

2.1.1 การเขียนโปรแกรมเพื่อที่จะคำนวณหาความเร็ว

สำหรับในการเขียนโปรแกรมเพื่อที่จะคำนวณหาความเร็วของ four-bar linkage นั้น เพื่อที่จะสามารถคำนวณหาความเร็วได้อย่างรวดเร็วและง่ายต่อการลงมือปฏิบัติ ดังนั้น ในโครงการชิ้นนี้ จะใช้วิธีคำนวณแบบ Analytic เพราะเนื่องจากการคำนวณโดยวิธีนี้เราสามารถที่จะคำนวณหาได้จากค่า input ที่เราใส่เข้าไป โดยที่ เป็นค่า input ที่ผู้ป้อนข้อมูลจะเป็นผู้กำหนดค่าไว้เองอยู่แล้ว และหลังจากที่เราทราบค่า แล้ว จะทำให้เราสามารถหาค่า ได้ โดยใช้สมการตามที่กล่าวมาแล้วข้างต้น คือ

และหลังจากที่เราสามารถหาค่าขนาดของความเร็วของเครื่องต่อได้แล้ว เราก็สามารถที่จะหาทิศทางของเวกเตอร์ความเร็วโดยการใช้วิธีความเร็วสัมพัทธ์ ซึ่งก็จะทำให้เราสามารถที่จะนำเวกเตอร์ที่ได้นำไปเขียนเป็น velocity polygon ได้ต่อไป

2.1.2 การเขียนโปรแกรมเพื่อที่จะคำนวณหาความเร่ง

ในการเขียนโปรแกรมเพื่อที่จะหาค่าความเร่งของ four-bar linkage นั้น เพื่อความสะดวกและรวดเร็วเราจึงสามารถใช้วิธีได้เช่นเดียวกับการหาความเร็วก็คือการใช้วิธี Analytic เพราะเนื่องจากค่า input ที่ผู้ป้อนข้อมูลเข้ามานั้นสามารถที่จะใช้ข้อมูลเหล่านี้ในการคำนวณหาค่าของ ได้ ซึ่งเราสามารถนำค่าที่คำนวณได้เหล่านี้ใช้ในการคำนวณหาค่าของ ได้ต่อไป

และเช่นกันเมื่อเราคำนวณได้ขนาดของความเร่งต่าง ๆ ได้แล้วก็สามารถที่จะหาทิศทางของเวกเตอร์เหล่านี้ได้จากการใช้วิธีความเร่งสัมพัทธ์ก็จะทำให้เราได้เวกเตอร์เพื่อที่จะนำไปเขียนเป็น acceleration polygon ได้

2.1.3 การเขียนโปรแกรมเพื่อที่จะคำนวณหาแรงที่กระทำกับเครื่องต่อต่าง ๆ

จากที่กล่าวมาในบทที่แล้ว วิธีในการคำนวณเพื่อที่จะหาค่าของแรงนั้นจะมีอยู่ 2 วิธี คือ The Superposition Method และ The Matrix Method แต่ในการคำนวณสำหรับหาค่าของแรงโดยการใช้วิธี The Matrix Method จะเป็นวิธีที่มีความสะดวกกว่า เพราะวิธีนี้เพียงแต่เราป้อนค่า input เข้าไป ซึ่งค่า input ที่เราป้อนเข้าไปในนั้นก็จะเป็นค่าที่เราสามารถที่จะคำนวณได้อยู่แล้ว และเมื่อเราป้อนค่าเข้าไปแล้วเราก็สามารถที่จะหาค่าของแรงต่าง ๆ ได้ทันที

สำหรับการแก้ปัญหาเพื่อที่จะหาค่าคำตอบที่จัดอยู่ในรูป matrix นั้น ในโครงงานนี้ได้ใช้วิธี Gaussian Elimination

2.2 หลักการเขียนโปรแกรมในการคำนวณ Slider Crank

การคำนวณสำหรับ Slider Crank นั้นเรามีความต้องการที่จะคำนวณหาความเร็ว ความเร่ง และแรงที่กระทำกับวัตถุเช่นเดียวกับ Four-bar Linkage แต่วิธีและขั้นตอนในการเขียนโปรแกรมจะมีลักษณะที่แตกต่างกันไป ดังที่จะกล่าวต่อไปนี้

2.2.1 การเขียนโปรแกรมเพื่อที่จะคำนวณหาความเร็ว

สำหรับในโครงงานชิ้นนี้นั้น วิธีการเขียนโปรแกรมเพื่อที่จะทำการคำนวณหาค่าความเร็ว นั้นเราจะใช้วิธีความเร็วสัมพัทธ์ โดยลักษณะการหาความเร็วนั้นจะมีขั้นตอนการคำนวณเหมือนกับวิธีความเร็วสัมพัทธ์ตามที่กล่าวมาแล้วในข้างต้น ซึ่งผลลัพธ์ที่เราได้จากการคำนวณในลักษณะนี้ เราจะนำมาเขียนเป็นลักษณะของ velocity polygon ได้นั่นเอง

2.2.2 การเขียนโปรแกรมเพื่อที่จะคำนวณหาความเร่ง

โดยวิธีการเขียนโปรแกรมเพื่อที่จะคำนวณหาความเร่งนั้นจะใช้วิธีความเร่งสัมพัทธ์ ซึ่งขั้นตอนการคำนวณต่าง ๆ ก็เหมือนกับการที่ได้กล่าวมาในบทที่แล้ว และเช่นเดียวกันผลลัพธ์ของการคำนวณ เราสามารถนำมาเขียนเป็น acceleration polygon ได้

2.2.3 การเขียนโปรแกรมเพื่อที่จะคำนวณหาแรงที่กระทำกับวัตถุ

สำหรับในขั้นตอนนี้ลักษณะจะใช้วิธีการวิเคราะห์ลักษณะของแรงสถิตย์ร่วมกับแรงเฉื่อยในกลไกรวมกัน โดยจะใช้วิธี The Superposition Method ตามที่ได้กล่าวมาแล้วในข้างต้น ซึ่งผลจากการคำนวณนี้เราจะได้เป็น แรงที่กระทำกันระหว่างเครื่องต่อ และทอร์ค

บทที่ 3

การใช้งานโปรแกรมคำนวณการเคลื่อนที่ของ 4 bar – linkage และ Slider – crank

3.1 ส่วนประกอบของโปรแกรม

โปรแกรมคำนวณการเคลื่อนที่ของ 4 bar-linkage และ slider crank ประกอบด้วย 2 ไฟล์ คือ

1.project.lsp

2.project.mln

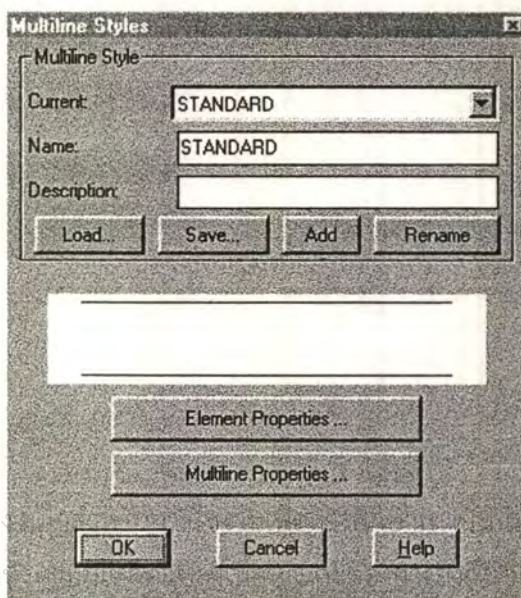
project.lsp เป็นไฟล์ Autolisp ใช้เป็นโปรแกรมหลักในการคำนวณและแสดงผลการเคลื่อนที่ของ 4 bar-linkage และ slider crank

project.mln เป็นไฟล์เก็บรูปแบบของเส้น multiline สำหรับใช้ในโปรแกรม Autocad ใช้ในการวาดรูปก้านต่อแต่ละก้าน ในการแสดงผลของโปรแกรมคำนวณการเคลื่อนที่ของ 4 bar-linkage และ slider crank

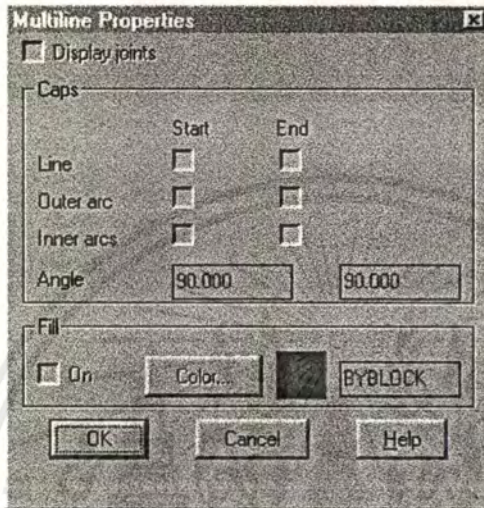
3.2 การสร้างไฟล์ project.mln

ในการวาดก้านต่อแต่ละก้านนั้น โปรแกรมคำนวณการเคลื่อนที่ของ 4 bar-linkage และ slider crank ใช้คำสั่ง mline ในการวาดรูป โดยรูปแบบที่วาดจะถูกเก็บข้อมูลไว้ในไฟล์ project.mln ขั้นตอนในการสร้างไฟล์ project.mln มีดังนี้

- เปิดโปรแกรม Autocad
- เรียกใช้คำสั่ง mstyle บน command line ของ Autocad จะปรากฏเมนู ดังรูปที่ 3.1



- พิมพ์ชื่อชนิดของเส้น mline ที่ต้องการจะสร้างลงในช่อง NAME สำหรับโปรแกรมนี้ พิมพ์ halfbar แล้วคลิกที่ปุ่ม Add
- คลิกที่ปุ่ม Multiline Properties ... จะปรากฏเมนูใหม่ขึ้นมา ดังรูปที่ 3.2

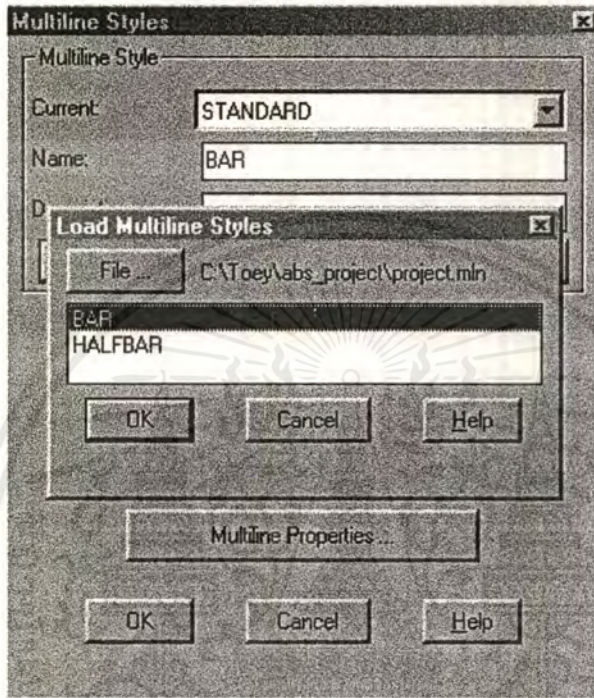


รูปที่ 3.2 หน้าต่าง Multiline Properties

- เลือกคลิกเครื่องหมาย ✓ ที่ช่อง Line-End และ Outer arc-Start ส่วนมุมทั้ง 2 ด้าน กำหนดให้เป็น 90.000 เหมือนเดิม แล้วกดปุ่ม OK ก็จะได้รูปแบบของ bar ที่มีส่วนโค้ง 1 ด้าน และ หน้าตัดอีก 1 ด้าน
- จากนั้นสร้างเส้น mline อีกเส้นหนึ่งให้ชื่อว่า bar และคลิก ✓ เลือกที่ช่อง Outer arc-Start , End โดยทำตามขั้นตอนเหมือนที่สร้างเส้น halfbar จะได้รูปแบบของเส้น mline 2 รูปแบบ ที่จะนำไปใช้ในโปรแกรมคำนวณการเคลื่อนที่ของ 4 bar-linkage และ slider crank
- ทำการบันทึกข้อมูลโดยกดปุ่ม Save ... และกำหนดให้ชื่อ project.mln

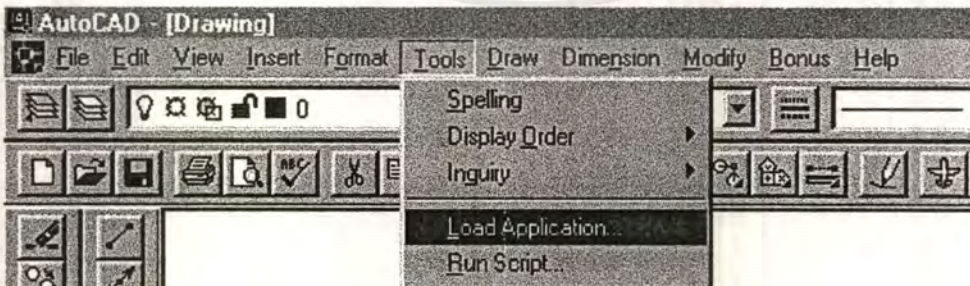
3.3 เริ่มใช้งานโปรแกรม

ก่อนที่จะใช้งานโปรแกรมคำนวณการเคลื่อนที่ของ 4 bar-linkage และ slider crank นั้น จำเป็นต้องกำหนดรูปแบบของ mline จากไฟล์ project.mln เสียก่อนโดยพิมพ์ mstyle ที่ command line จากนั้นกดปุ่ม Load ... แล้วเลือกไฟล์ project.mln จากรูปที่ 3.3



รูปที่ 3.3 การเรียกใช้งาน project.mln

จากนั้นจึงเรียกใช้งานโปรแกรม project.lsp โดยเลือกที่เมนู Tools ของ Autocad แล้วเลือกไปยังที่ Load Application... แล้วเลือกไฟล์ project.lsp จากรูปที่ 3.4



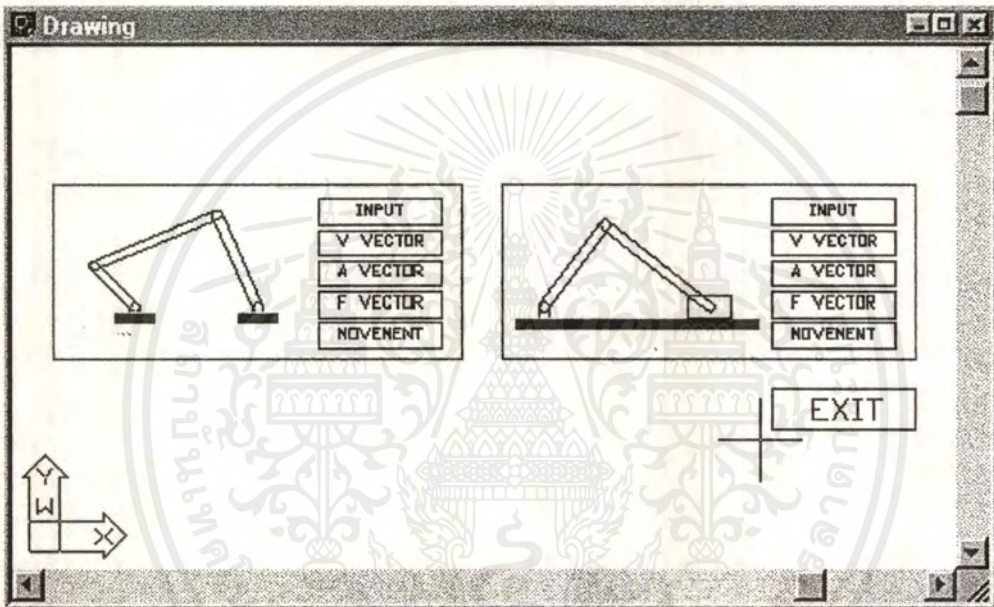
รูปที่ 3.4 การเรียกใช้งาน project.lsp

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อทำการโหลดไฟล์ project.isp เรียบร้อยแล้ว ให้พิมพ์ project ที่ command line เพื่อเริ่มโปรแกรม

3.4การใช้งานโปรแกรม

หลังจากเรียกโปรแกรมโดยการพิมพ์คำสั่ง project ที่ command line แล้ว โปรแกรมจะทำการเรียกเมนูขึ้นมา ดังรูปที่ 3.5



รูปที่ 3.5 โปรแกรมคำนวณการเคลื่อนที่ ของ 4bar-linkage และ slider-crank

เมนูจะแบ่งออกเป็น 2 ส่วน คือ ส่วนของ 4 bar-linkage และ ส่วนของ slider crank โดยที่ทั้ง 2 ส่วนจะประกอบไปด้วยเมนูย่อย คือ

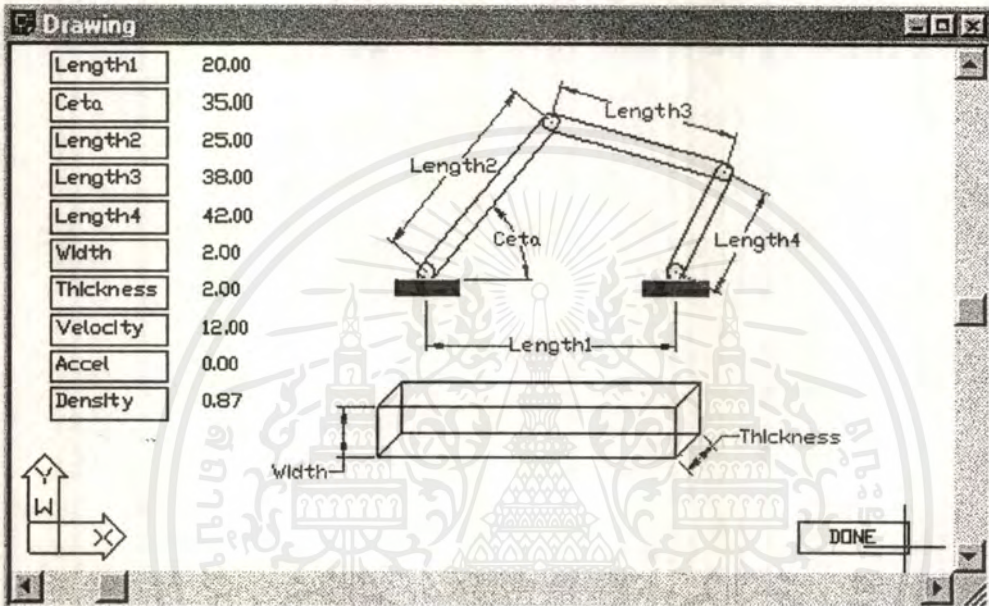
- | | |
|-------------|--|
| 1. INPUT | ใช้สำหรับใส่ข้อมูลที่ต้องการนำไปคำนวณการเคลื่อนที่ |
| 2. V VECTOR | แสดงแผนภาพความเร็ว |
| 3. A VECTOR | แสดงแผนภาพความเร่ง |
| 4. F VECTOR | แสดงแผนภาพแรง |
| 5. MOVEMENT | แสดงการเคลื่อนที่ของ 4 bar-linkage หรือ slider crank |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. INPUT

ทำหน้าที่รับข้อมูลจากผู้ใช้ผ่านทางคีย์บอร์ด ข้อมูลที่รับมานี้จะถูกนำไปประมวลผล และนำเสนอในรูปแบบของ V vector , A vector , F vector และ Movement ต่อไป

ข้อมูลที่รับมานี้จะถูกตรวจสอบว่ามีความถูกต้องหรือไม่ หากไม่ถูกต้องจะเตือนให้ทำการแก้ไขก่อน จึงจะอนุญาตให้ทำการประมวลผลต่อไปได้



รูปที่ 3.6 INPUT ของ 4 bar-linkage

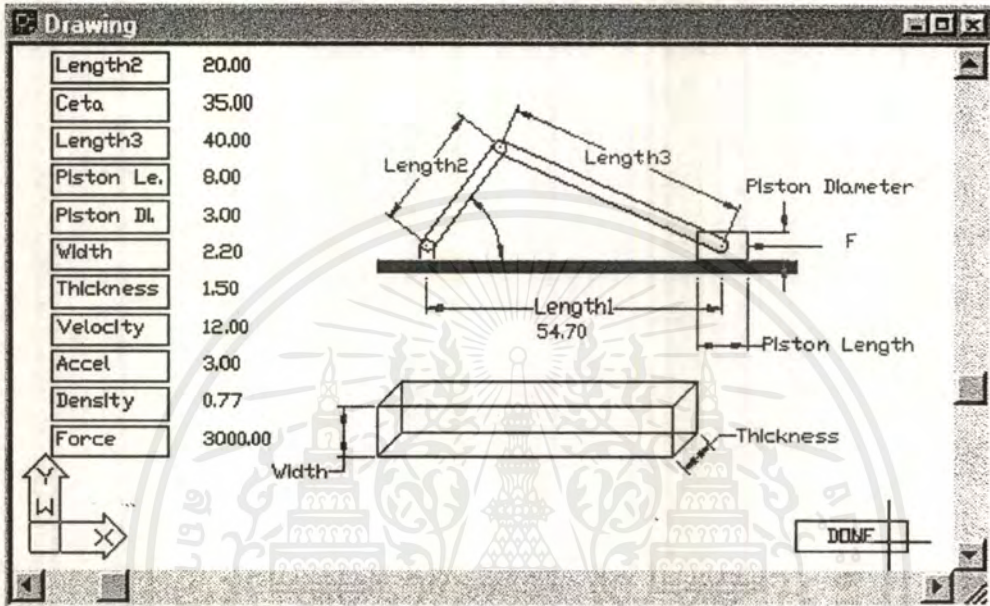
สำหรับ INPUT ของ 4 bar-linkage จะประกอบไปด้วย

- Length1 ความยาวของ ก้านต่อที่ 1 ในที่นี้กำหนดให้เป็นระยะห่างบนพื้น หน่วยเป็น เมตร
- Length2 ความยาวของ ก้านต่อที่ 2 หน่วยเป็น เมตร
- Ceta มุมที่ก้านต่อที่ 2 กระทำต่อ ก้านต่อที่ 1 หรือกระทำต่อพื้น หน่วยเป็น องศา
- Length3 ความยาวของ ก้านต่อที่ 3 หน่วยเป็น เมตร
- Length4 ความยาวของ ก้านต่อที่ 4 หน่วยเป็น เมตร
- Width ความกว้างของก้านต่อแต่ละก้าน หน่วยเป็น เมตร
- Thickness ความหนาของก้านต่อแต่ละก้าน หน่วยเป็น เมตร
- Velocity ความเร็วของการหมุนของก้านต่อที่ 2 หน่วยเป็น เรเดียน / วินาที

ทิศทางกำหนดให้ + ทวนเข็มนาฬิกา / - ตามเข็มนาฬิกา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Acceleration ความเร่งของการหมุนของก้านต่อที่ 2 หน่วยเป็น เรเดียน / วินาที²
ทิศทางการกำหนดให้ + ทวนเข็มนาฬิกา / - ตามเข็มนาฬิกา
- Density ความหนาแน่นของก้านต่อแต่ละอัน หน่วยเป็น กิโลกรัม / เมตร³
- DONE กดปุ่มนี้ เมื่อป้อนข้อมูลเสร็จแล้ว



รูปที่ 3.7 INPUT ของ slider-crank

สำหรับ INPUT ของ Slider crank ประกอบด้วย

- Length2 ความยาวของ ก้านต่อที่ 2 หน่วยเป็น เมตร
- Ceta มุมที่ก้านต่อที่ 2 กระทำต่อก้านต่อที่ 1 หรือกระทำต่อพื้น หน่วยเป็น องศา
- Length3 ความยาวของ ก้านต่อที่ 3 หน่วยเป็น เมตร
- Piston Le. ความยาวของลูกสูบ หน่วยเป็น เมตร
- Piston Di. เส้นผ่านศูนย์กลางกลางของลูกสูบ หน่วยเป็น เมตร
- Width ความกว้างของก้านต่อแต่ละก้าน หน่วยเป็น เมตร
- Thickness ความหนาของก้านต่อแต่ละก้าน หน่วยเป็น เมตร
- Velocity ความเร็วของการหมุนของก้านต่อที่ 2 หน่วยเป็น เรเดียน / วินาที
ทิศทางการกำหนดให้ + ทวนเข็มนาฬิกา / - ตามเข็มนาฬิกา
- Acceleration ความเร่งของการหมุนของก้านต่อที่ 2 หน่วยเป็น เรเดียน / วินาที²
ทิศทางการกำหนดให้ + ทวนเข็มนาฬิกา / - ตามเข็มนาฬิกา

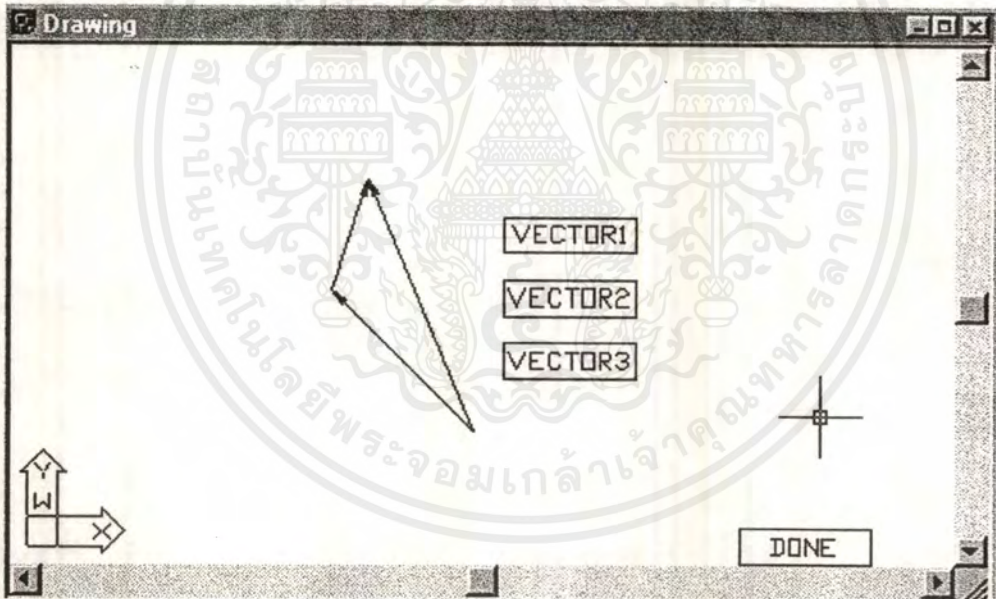
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Density ความหนาแน่นของก้านต่อแต่ละอัน หน่วยเป็น กิโลกรัม / เมตร³
- Force แรงที่กระทำต่อลูกสูบ หน่วยเป็น นิวตัน
- DONE กดปุ่มนี้ เมื่อป้อนข้อมูลเสร็จแล้ว

หากข้อมูลที่ทำการป้อนให้แก่โปรแกรมนั้นไม่ถูกต้อง ไม่ว่าจะเป็นส่วนของ 4bar-linkage หรือ Slider crank โปรแกรมจะทำการแจ้งให้ผู้ใช้ได้ทราบ และให้ทำการป้อนข้อมูลใหม่ และจะไม่ยอมให้ผู้ใช้โปรแกรมสั่งการประมวลผล จนกว่าจะป้อนข้อมูลที่ถูกต้องจนเสร็จ

2. V VECTOR

ใช้แสดงผล แผนภาพความเร็ว (velocity polygon) และต้องใส่ข้อมูลในส่วน INPUT ให้เรียบร้อยเสียก่อน โปรแกรมจึงจะยอมให้ทำงานในส่วนนี้ได้

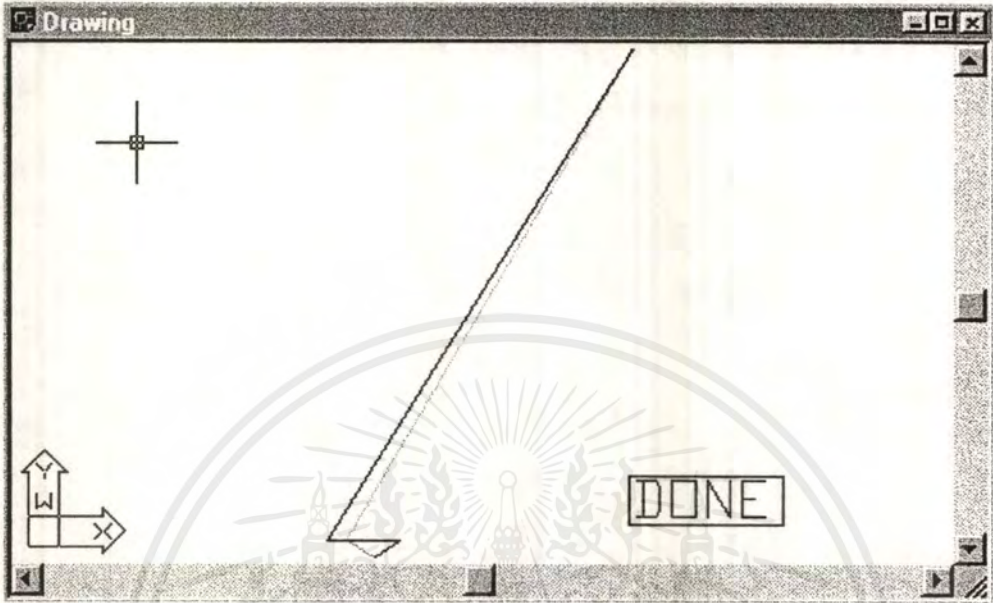


รูปที่ 3.8 คำสั่ง V VECTOR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. A VECTOR

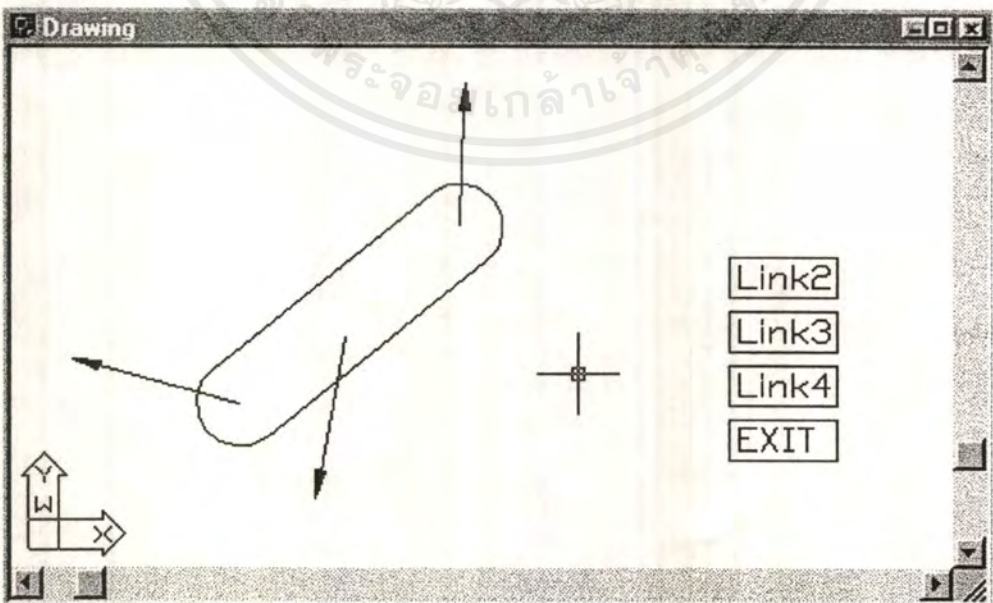
ใช้แสดงผล แผนภาพความเร่ง (acceleration polygon)



รูปที่ 3.9 คำสั่ง A VECTOR

4. F VECTOR

ใช้แสดงผล แรงที่กระทำต่อ เครื่องต่อแต่ละชิ้น

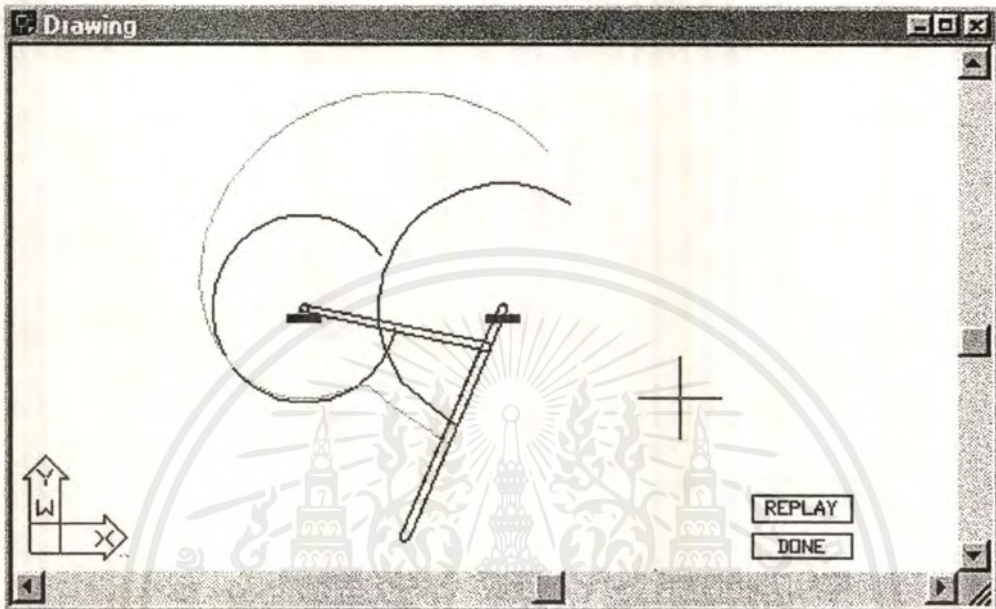


รูปที่ 3.10 คำสั่ง F VECTOR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

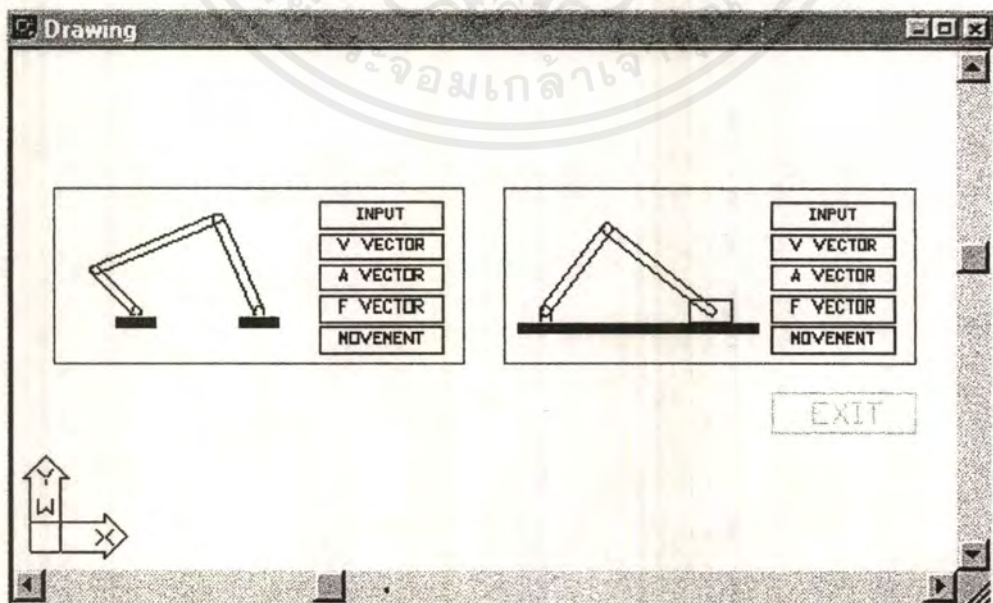
5. MOVEMENT

ใช้แสดงผล การเคลื่อนที่ของ 4 bar-linkage หรือ slider-crank ในการเคลื่อนที่ 1 รอบ หรือจนพบจุดตาย (Deathpoint)



รูปที่ 3.11 คำสั่ง MOVEMENT

หลังจากจบการทำงานของแต่ละส่วนแล้ว โปรแกรมจะกลับมาที่เมนูหลัก เพื่อเลือกการทำงานของส่วนอื่นต่อไป หากต้องการจะออกจากโปรแกรมก็เลือกไปที่ปุ่ม EXIT



รูปที่ 3.12 เลือก EXIT เพื่อออกจากโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การออกแบบโปรแกรม

4.1 การออกแบบโปรแกรม

การออกแบบโปรแกรมคำนวณการเคลื่อนที่ของ 4bar-linkage และ slider-crank แบ่งออกเป็น 2 ส่วน คือ

1. ส่วนของ 4bar-linkage
2. ส่วนของ slider-crank

และแต่ละส่วนจะประกอบด้วยการทำงาน 3 ขั้นตอน คือ

1. INPUT
2. PROCESS
3. OUTPUT

โดยทั้ง 3 ขั้นตอน จะต้องทำงานเรียงลำดับกันมา จะข้ามขั้นตอนไปไม่ได้

1.INPUT

ส่วนของ INPUT จะรับข้อมูลมาจากผู้ใช้โปรแกรม โดยข้อมูลส่วนนี้จะถูกทำการตรวจสอบความผิดพลาดก่อน หากไม่เกิดการผิดพลาดจึงจะส่งข้อมูลต่อไปยังส่วน PROCESS เพื่อทำงานต่อไป

ความผิดพลาดที่อาจเกิดขึ้นได้ในการใส่ข้อมูล

- ความยาวของก้านต่ออันใดอันหนึ่งยาวไม่พอ
- ความยาวของก้านต่ออันใดอันหนึ่งยาวเกินไป
- ใส่ค่าของความยาวก้านต่อเป็นลบหรือศูนย์

โดยโปรแกรมจะบังคับให้ผู้ใช้ใส่ข้อมูลให้ครบทุกค่า ก่อนที่จะไปทำงานในส่วนขอ

PROCESS และ OUTPUT ต่อไป

2.PROCESS

ขั้นตอน PROCESS จะนำข้อมูลที่ผู้ใช้ใส่เข้ามา ไปทำการประมวลผล เพื่อให้ได้ OUTPUT ตามที่ต้องการ หาก INPUT ที่ได้มามีค่าผิดพลาดจะทำให้ส่วน PROCESS มีค่าผิดพลาดไปด้วย และส่งผลให้โปรแกรมเกิดปัญหาขึ้นทันที เพราะฉะนั้นส่วนนี้จึงเป็นส่วนที่มีความสำคัญมาก

ขั้นตอน PROCESS จะแบ่งการทำงานเป็น 4 แบบ ตามลักษณะของ OUTPUT ได้แก่

2.1 การคำนวณของ V VECTOR

2.2 การคำนวณของ A VECTOR

2.3 การคำนวณของ F VECTOR

2.4 การคำนวณการเคลื่อนที่ของก้านต่อ

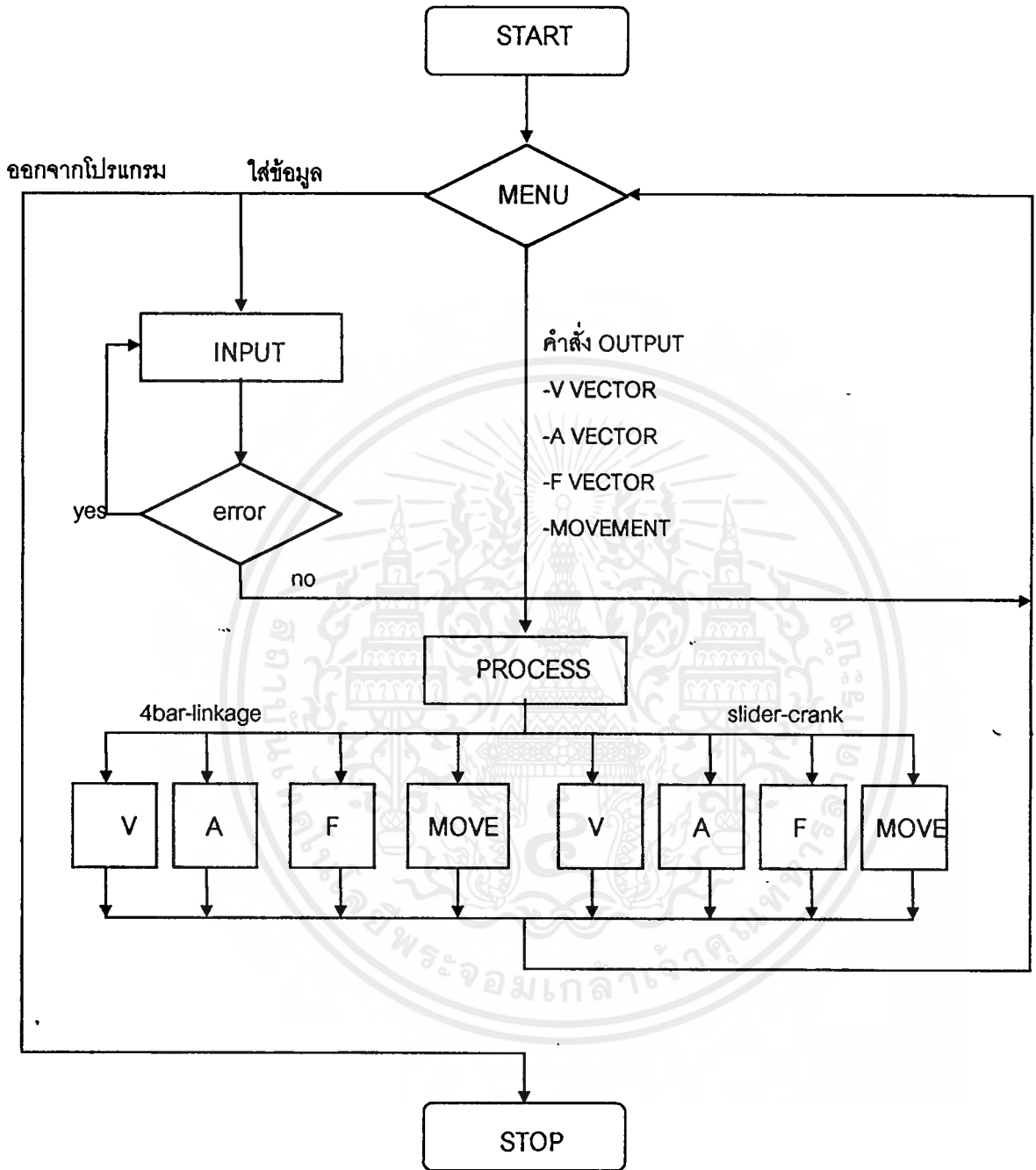
เมื่อรวมรูปแบบของ PROCESS ของทั้ง 2 ส่วน คือ ส่วน 4bar-linkage และ ส่วน slider-crank แล้ว จะได้รูปแบบของ PROCESS ทั้งหมด 8 รูปแบบด้วยกัน

3.OUTPUT

ส่วนของ OUTPUT จะนำข้อมูลที่ได้รับ ทั้งข้อมูลที่ผู้ใช้ใส่เข้ามาในส่วนของ INPUT และ ข้อมูลที่โปรแกรมทำการคำนวณขึ้นมาเองจากส่วนของ PROCESS มาทำการแสดงผลให้ผู้ใช้ได้ดู โดยข้อมูลที่ใส่ใน INPUT 1 ชุด สามารถแสดงผลออกมาได้ 4 แบบ คือ

- | | |
|--------------|--|
| 3.1 V VECTOR | แสดงแผนภาพ v polygon |
| 3.2 A VECTOR | แสดงแผนภาพ a polygon |
| 3.3 F VECTOR | แสดงทิศทางของแรงที่กระทำต่อก้านต่อแต่ละอัน |
| 3.4 MOVEMENT | แสดงการเคลื่อนที่ของก้านต่อชุดนั้น |

จากนั้นจึงนำโปรแกรมทั้ง 3 ส่วนมารวมกัน โดยใช้เมนูเข้ามาช่วยในการสลับการทำงาน สามารถเขียนเป็นอัลกอริทึม ได้ดังนี้



รูปที่ 4.1 ลำดับการทำงานของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

คำสั่งในโปรแกรม Autolisp

5.1 การให้คำสั่งในโปรแกรม Autolisp

โปรแกรม Autolisp เป็นส่วนหนึ่งของ ภาษา commonlisp ซึ่งเป็นภาษาที่โครงสร้างแบบง่าย ทำให้ง่ายต่อการเข้าใจ และโปรแกรม Autolisp เป็นโปรแกรมที่ทำงานอยู่บนโปรแกรม Autocad (ไม่สามารถทำงานได้ด้วยตัวเอง) ทำให้มีข้อได้เปรียบตรงที่มีคุณสมบัติที่จะดึงความสามารถของ Autocad มาใช้ให้เป็นประโยชน์ได้ ทำให้โปรแกรม Autolisp เหมาะกับงานที่ต้องการความละเอียดในการคำนวณ และในการประมวลผล

ลักษณะการทำงานของคำสั่งในโปรแกรม Autolisp จะประมวลผลนิพจน์ที่อยู่ภายในวงเล็บตามลำดับจากวงเล็บใน มาวงเล็บนอก มีลักษณะคล้ายลำดับการคำนวณทางคณิตศาสตร์ ตัวอย่าง เช่น $(+ 3 1)$ หมายความว่า ให้นำ 3 มาบวกด้วย 1

และคำสั่งต่างๆ ก็จะถูกแบ่งด้วยวงเล็บเหมือนกัน

ตัวอย่าง เช่น $(setq le1 15.00)$ หมายความว่า ให้ตัวแปร $le1 = 15.00$

5.2 คำสั่งต่างๆ ใน Autolisp

+, -, *, /, ... การคำนวณทางคณิตมีรูปแบบในการใช้งานดังนี้

(+ ข้อมูลที่1 ข้อมูลที่2 ข้อมูลที่3 ...) – ข้อมูลที่1 เป็นตัวตั้ง ถูกบวกด้วยข้อมูลที่2,3 และข้อมูลต่อไป

(- ข้อมูลที่1 ข้อมูลที่2 ข้อมูลที่3 ...) – ข้อมูลที่1 เป็นตัวตั้ง ถูกลบด้วยข้อมูลที่2,3 และข้อมูลต่อไป

(* ข้อมูลที่1 ข้อมูลที่2 ข้อมูลที่3 ...) – ข้อมูลที่1 เป็นตัวตั้ง ถูกคูณด้วยข้อมูลที่2,3 และข้อมูลต่อไป

(/ ข้อมูลที่1 ข้อมูลที่2 ข้อมูลที่3 ...) – ข้อมูลที่1 เป็นตัวตั้ง ถูกหารด้วยข้อมูลที่2,3 และข้อมูลต่อไป

and ใช้คำนวณทางตรรกศาสตร์

รูปแบบ (and เงื่อนไขที่1 เงื่อนไขที่2 ..)

หากเงื่อนไขที่1 และ เงื่อนไขที่ 2 เป็นจริงแล้ว คำสั่ง and จะคืนค่า T กลับมา

angle ใช้หาค่ามุมเทียบกับแกน x มีหน่วยเป็นเรเดียน

รูปแบบ (angle จุดโคออดิเนตที่1 จุดโคออดิเนตที่2)

จุดโคออดิเนตที่1 – เป็นจุดเริ่มต้น

จุดโคออดิเนตที่2 – เป็นจุดปลายที่จะทำการหาค่าของมุม

cadr ใช้เลือกข้อมูลบางส่วนจากข้อมูลชนิด list โดยคำสั่งcadr จะเลือกข้อมูลที่เหลือจากการตัดข้อมูลตัวแรกทิ้งไป ใช้คู่กับคำสั่ง car

รูปแบบ (cadr ข้อมูลชนิด list)

...

car ใช้เลือกข้อมูลตัวแรกเพียงตัวเดียวจากข้อมูลชนิด list

รูปแบบ (car ข้อมูลชนิด list)

command เป็นคำสั่งเรียกใช้งานคำสั่งของ Autocad

รูปแบบ (command "คำสั่งในAutocad" ..)

คำสั่งของAutocad ที่เรียกใช้งาน จะอยู่ในเครื่องหมาย " " และเรียงลำดับการทำงาน เหมือนกับ บรรทัด command line ใน Autocad หากต้องการนำข้อมูลภายในตัวแปรมาใช้งาน ก็ไม่ต้องใส่เครื่องหมาย " "

cos ใช้หาค่าcosine ของมุมที่กำหนด

รูปแบบ (cos มุมหน่วยเรเดียน)

defun ใช้สร้างโปรแกรมหรือคำสั่งใน Autocad เป็นการเริ่มต้นของการเขียนโปรแกรม Autolisp ทุกครั้ง

รูปแบบ (defun ชื่อโปรแกรม (ตัวแปรภายนอก .. / ตัวแปรภายใน ..)

(คำสั่ง) (คำสั่ง) ..)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อโปรแกรม – ชื่อที่เราตั้งขึ้น เพื่อเรียกใช้โปรแกรมนี้

ตัวแปรภายนอก – ตัวแปรที่ส่งค่ามายังโปรแกรมที่สร้างขึ้น เพื่อใช้งานภายในโปรแกรม

ตัวแปรภายใน – ตัวแปรที่สร้างขึ้นมาเพื่อใช้งานภายในโปรแกรมนั้นๆ เมื่อโปรแกรม

ทำงานเสร็จ จะคืนทรัพยากรให้กับเครื่องคอมพิวเตอร์ ไม่สามารถนำค่าที่เก็บไว้ในตัวแปรมาใช้ได้อีก

คำสั่ง – คำสั่งใช้งานภายในโปรแกรม

distance ใช้หาระยะทางของจุด 2 จุด

รูปแบบ (distance จุดโคออดิเนตที่1 จุดโคออดิเนตที่2)

จุดโคออดิเนตที่1,2 – จุดที่ต้องการจะหาระยะห่าง จุดทั้ง2 ต้องเป็นข้อมูลชนิด list

getpoint รองรับข้อมูลจุดโคออดิเนตจากผู้ที่ใช้โปรแกรม สามารถใส่ข้อมูลได้ทาง mouse หรือ keyboard

รูปแบบ (getpoint "ข้อความบรรยาย")

ข้อความบรรยาย – เพื่อบอกให้ผู้ที่ใช้โปรแกรมทราบว่าต้องใส่ข้อมูลลงไป

getreal รองรับข้อมูลที่เป็นเลขจำนวนจริงจากผู้ที่ใช้โปรแกรม สามารถใส่ข้อมูลได้ทาง keyboard เท่านั้น

รูปแบบ (getreal "ข้อความบรรยาย")

if ใช้กำหนดเงื่อนไขให้กับโปรแกรม

รูปแบบ (if (เงื่อนไข) (คำสั่งที่1) (คำสั่งที่2))

เงื่อนไข – เป็นสมการหรืออสมการที่ใช้กำหนดทางเลือกได้ 2 ทาง คือ T หรือ nil

คำสั่งที่1 – จะทำงานเมื่อเงื่อนไข มีความถูกต้อง เป็น T

คำสั่งที่2 – จะทำงานเมื่อเงื่อนไข มีความถูกต้อง เป็น nil

initget ใช้จำกัดชนิดของข้อมูลที่จะรับมาจากผู้ที่ใช้โปรแกรม ใช้คู่กับคำสั่ง getreal

รูปแบบ (initget (+ รหัสควบคุม ..))

รหัสควบคุม 1 – ไม่ยอมรับการป้อนค่า null หรือ การไม่ป้อนค่าใดๆ

รหัสควบคุม 2 – ไม่ยอมรับการป้อนค่า 0

รหัสควบคุม 4 – ไม่ยอมรับการป้อนค่าเป็นลบ

list ใช้เก็บข้อมูลเป็นชุดที่มีมากกว่า 1 ค่า เช่น คู่อันดับ (x,y) , คู่อันดับ (x,y,z) เป็นต้น
รูปแบบ (list ข้อมูลตัวที่1 ข้อมูลตัวที่2 ..)

ข้อมูลตัวที่1 , ข้อมูลตัวที่2 , .. – เป็นข้อมูลที่ต้องการจัดให้เป็นหมวดหมู่เดียวกัน

or ใช้คำนวณทางตรรกศาสตร์

รูปแบบ (or เงื่อนไขที่1 เงื่อนไขที่2 ..)

หากเงื่อนไขที่1 หรือ เงื่อนไขที่ 2 เป็นจริงแล้ว คำสั่ง or จะคืนค่า T กลับมา

polar ใช้หาจุดโคออดิเนตที่อยู่ห่างจากจุดเริ่ม โดยทราบมุม และระยะห่าง

รูปแบบ (polar จุดโคออดิเนตเริ่มต้น มุมในหน่วยเรเดียน ระยะห่างจากจุดเริ่มต้น)

progn ใช้รวบรวมคำสั่งหลายๆคำสั่งให้เสมือนเป็นคำสั่งเดียวกัน คำสั่งprognนี้ จะใช้ในคำสั่ง if
อีกทีหนึ่ง ในกรณีที่ต้องการให้คำสั่ง if ทำงานได้หลายๆ คำสั่งในเงื่อนไขเดียวกัน

รูปแบบ (if (เงื่อนไข)

(progn (คำสั่ง) (คำสั่ง) ..)

(progn (คำสั่ง) (คำสั่ง) ..)

)

rtos ใช้กำหนดรูปแบบการแสดงผลของตัวเลข

รูปแบบ (rtos ตัวเลข รหัสรูปแบบ จำนวนตำแหน่งทศนิยม)

รหัสรูปแบบ 1 – วิทยาศาสตร์

รหัสรูปแบบ 2 – เลขทศนิยม

รหัสรูปแบบ 3 – ฟุต และ นิ้ว แบบ ทศนิยม

รหัสรูปแบบ 4 – ฟุต และ นิ้ว แบบ เศษส่วน

`setq` ใช้กำหนดข้อมูลของตัวแปร

รูปแบบ (`setq` ชื่อตัวแปร ข้อมูล)

ชื่อตัวแปร – เป็นชื่อของตัวแปรที่ตั้งขึ้น โดยต้องไม่ซ้ำกับคำสั่งที่มีอยู่ในโปรแกรม Autolisp

`sin` ใช้หาค่า sine ของมุมที่กำหนด

รูปแบบ (`sin` มุมหน่วยเรเดียน)

`strcat` ใช้รวมข้อมูลชนิดตัวอักษรหลายๆ ข้อมูลเข้าด้วยกัน

รูปแบบ (`strcat` ข้อมูลที่1 ข้อมูลที่2...)

ข้อมูลที่1,2,... ที่อยู่ภายในคำสั่ง `strcat` จะถูกรวมเป็นข้อมูลเดียว

`while` ใช้ทำซ้ำคำสั่ง ภายใต้เงื่อนไขที่กำหนดไว้

รูปแบบ (`while` (เงื่อนไข)

(คำสั่ง) (คำสั่ง) ..

)

หากเงื่อนไขเป็นจริงแล้ว คำสั่งต่างๆ ที่อยู่ภายในคำสั่ง `while` จะถูกทำงานตามลำดับซ้ำๆ ไปเรื่อยๆ จนกว่าเงื่อนไขจะเป็นเท็จ หรือ `nil`

บทที่ 6

คำสั่งในโปรแกรม Autocad

6.1 การใช้คำสั่งของ Autocad ในโปรแกรม Autolisp

โปรแกรม Autolisp สามารถเรียกใช้งานคำสั่งของ Autocad เหมือนการใช้โปรแกรม Autocad ตามปกติ โดยเรียกใช้งานผ่านคำสั่ง command ของ Autolisp คำสั่ง command จะจำลองการทำงานที่ command line ของ Autocad มาใช้ ยกตัวอย่างเช่น

```
(command "line" "0,0" "34,114" "")
```

โปรแกรม Autolisp จะเรียกใช้คำสั่ง line ในโปรแกรม Autocad จากนั้นโปรแกรม Autocad จะถามหาจุดเริ่มต้นที่จะวาดเส้น <line From point> ก็จะได้รับค่า 0,0 แล้วโปรแกรม Autocad จะถามหาจุดต่อไป <To point> ก็จะได้รับค่า 34,114 สำหรับ "" ตัวสุดท้ายนั้นจะแทนการกด ENTER 1 ครั้ง เพื่อเป็นการสิ้นสุดการลากเส้นตรง เพราะฉะนั้น คำสั่งใดๆ ใน Autocad ที่สามารถสั่งงานผ่าน command line ได้แล้วก็จะสามารถเรียกใช้งานผ่าน Autolisp ได้ด้วย

6.2 คำสั่งใน Autocad

color ใช้เปลี่ยนสีของวัตถุที่เลือก หรือวัตถุที่ต้องการจะวาดต่อไป

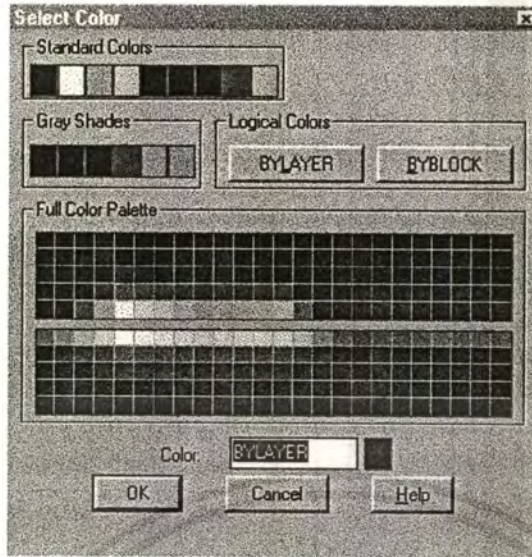
รูปแบบ (command "color" "สีที่เลือก")

สีที่เลือกสามารถกำหนดเป็นตัวเลข หรือบอกเป็นชื่อสีเลยก็ได้ ชื่อสีที่เป็นมาตรฐานของ

Autocad ได้แก่

- RED สีแดง
- YELLOW สีเหลือง
- GREEN สีเขียว
- CYAN สีฟ้า
- BLUE สีน้ำเงิน
- MAGENTA สีม่วงอ่อน
- WHITE สีขาว
- OTHER ... ใช้สำหรับเลือกสีอื่นๆ ที่ไม่ใช่สีมาตรฐาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.1 หน้าต่างสำหรับเลือกสี โดยมีสีมาตรฐานอยู่แถบบนสุด

ตัวอย่าง

(command "color" "7")

(command "color" "red")

erase ใช้ลบวัตถุที่เลือก

รูปแบบ (command "erase" เลือกวัตถุ "")

สำหรับการเลือกวัตถุ สามารถกำหนดได้หลายแบบ

"all" เป็นการเลือกวัตถุทั้งหมด ยกเว้นวัตถุที่ถูกล็อกเอาไว้

"w" "จุดเริ่มต้น" "จุดสิ้นสุด" เป็นการเลือกวัตถุแบบ window วัตถุที่มีชิ้นส่วนทั้ง

หมดอยู่ในขอบเขตการเลือก จะถูกเลือกเอาไว้

"c" "จุดเริ่มต้น" "จุดสิ้นสุด" เป็นการเลือกวัตถุแบบ crossing วัตถุที่มีบางส่วนอยู่

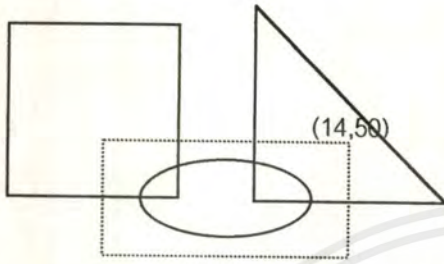
ในขอบเขตการเลือก จะถูกเลือกเอาไว้

ตัวอย่าง

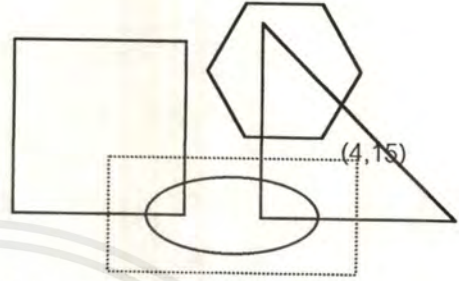
(command "erase" "all" "") วัตถุทั้งหมดที่ไม่ได้ล็อกไว้ จะถูกลบออกทั้งหมด

(command "erase" "w" (list 0 0) (list 14 50) "") วัตถุทุกชิ้น ที่มีองค์ประกอบทั้งหมด
อยู่ในขอบเขต (0,0) – (14,50) จะถูกลบออก ดูรูปที่ 6.2 ประกอบ วัตถุที่ถูกลบออก จะเป็นรูปวงรี
รูปเดียว

(command "erase" "c" "1,0" "4,15" "") วัตถุทุกชิ้น ที่มีองค์ประกอบบางส่วนอยู่ในขอบเขต (1,0) – (4,15) จะถูกลบออก จากรูปที่ 6.3 ประกอบ วัตถุที่ถูกลบออกมี 3 วัตถุ คือ รูปสี่เหลี่ยม รูปสามเหลี่ยม และรูปวงรี



(0,0)
รูปที่ 6.2 การเลือกแบบ window



(1,0)
รูปที่ 6.3 การเลือกแบบ crossing

layer ใช้สร้างชั้นของการวาดภาพ เพื่อความสะดวกในการแบ่งประเภทของภาพต่างๆ

รูปแบบ (command "layer" "คำสั่งจัดการ" "ชื่อของชั้นlayer" "")

คำสั่งจัดการของ layer มีหลายคำสั่งมาก ที่ใช้โดยทั่วไปได้แก่

"make" ใช้สร้าง layer ใหม่ และไปทำงาน ณ layer นั้น

"new" ใช้สร้าง layer ใหม่ แต่ยังทำงานอยู่ ณ layer เดิม

"set" ใช้ในการย้ายไปทำงาน ณ layer ที่กำหนดไว้

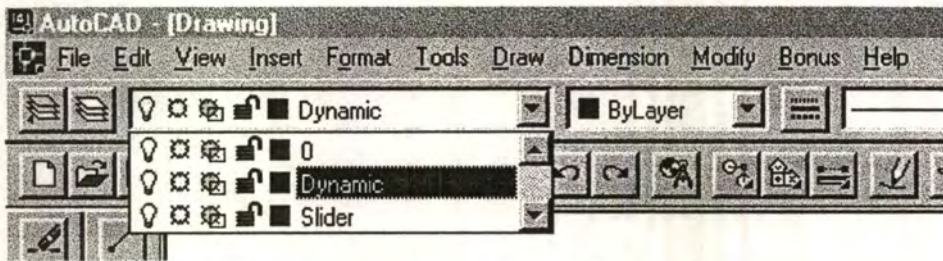
"on"/"off" ใช้เปิด/ปิด การแสดงผลของ layer ที่กำหนด

"lock"/"unlock" ใช้ล็อก/ยกเลิกการล็อก ของ layer ที่กำหนด layer ที่ถูกล็อกไว้จะไม่สามารถทำการแก้ไข หรือลบวัตถุได้

ตัวอย่าง

(command "layer" "make" "SLIDER" "") สร้างเลเยอร์ชื่อ SLIDER และไปทำงานยังเลเยอร์ SLIDER รูปหรือวัตถุที่ทำการวาดหลังจากนี้ จะอยู่บนเลเยอร์ SLIDER

(command "layer" "off" "DYNAMIC" "") ปิดการแสดงผลเลเยอร์ DYNAMIC ทำให้มองไม่เห็นรูปหรือวัตถุที่อยู่บน เลเยอร์ DYNAMIC แต่รูปเหล่านั้นยังไม่ได้ถูกลบไป แค่ถูกซ่อนไว้เท่านั้น



รูปที่ 6.4 Toolbar สำหรับบอกและแก้ไขเลเยอร์ชั้นต่าง ๆ

เราสามารถตรวจสอบสถานะของเลเยอร์ชั้นต่างๆ ได้จาก Toolbar ของ Autocad ด้านบนของจอภาพได้บรรทัดเมนูคำสั่ง

line ใช้วาดเส้นตรง

รูปแบบ (command "line" จุดที่1 จุดที่2 จุดที่3 .. "")

เส้นตรงจะถูกลากจากจุดที่1 ไปยังจุดที่2 และ 3 ต่อไปเรื่อยๆ จนกว่าจะเจอ เครื่องหมาย ""

ตัวอย่าง

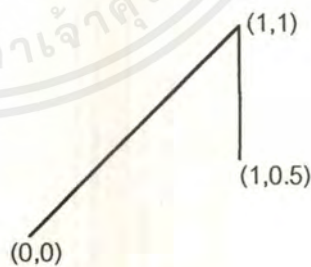
(command "line" "0,0" "1,1" "") ดูผลลัพธ์ได้จาก รูปที่ 6.5

(command "line" (list 0 0) (list 1 1) (list 1 0.5) "") ดูผลลัพธ์ได้จาก รูปที่ 6.6



รูปที่ 6.5

คำสั่ง line จากจุด (0,0) ถึง (1,1)



รูปที่ 6.6

คำสั่ง line จากจุด (0,0) ถึง (1,1) ถึง (1,0.5)

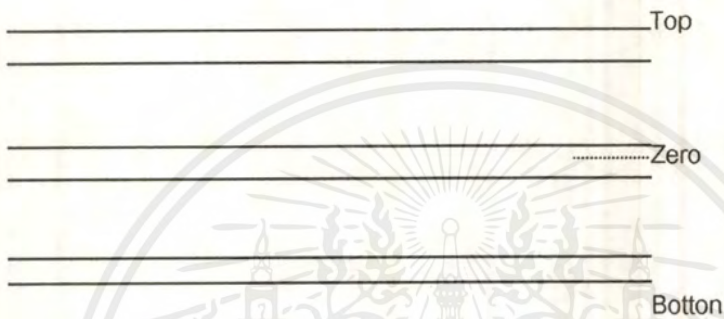
mline ใช้วาดเส้นตรงคู่

รูปแบบ (command "mline" <option> จุดที่1 จุดที่2 จุดที่3 .. "")

<option> จะมีอยู่ 3 ส่วน คือ

1.Justification ให้เลือกกระดัดของการวางเส้น มี Top/Zero/Bottom ดูลักษณะของเส้น

แบบต่างๆ ได้จากรูปที่ 6.7



รูปที่ 6.7 รูปแบบการวางเส้น mline

2.Scale

ใช้กำหนดความกว้างของเส้นตรงคู่

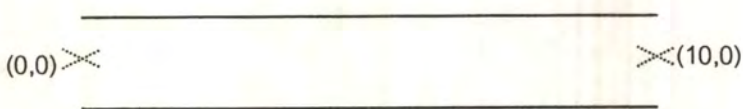
3.Style

ใช้เลือกรูปแบบของเส้นตรงคู่

ตัวอย่าง

(command "mline" "j" "z" "0,0" "10,0" "") วาดเส้นคู่ตั้งแต่ (0,0) – (10,0) โดย

กำหนด Justification มีค่าเป็น zero จากรูปที่ 6.8 ประกอบ



รูปที่ 6.8 เส้น mline จากจุด (0,0) ถึง (10,0)

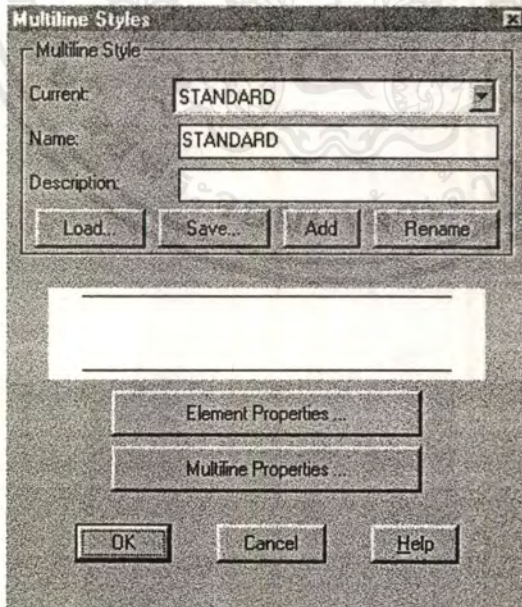
(command "mline" "s" "0.5" (list 0 0) (list 0 5) "") วาดเส้นคู่ตั้งแต่ (0,0) – (0,5) โดยกำหนดให้มีขนาดความกว้าง เท่ากับ 0.5 ดูรูปประกอบที่ 6.9



รูปที่ 6.9 เส้น mline จากจุด (0,0) ถึง (0,5)

mstyle ใช้สร้างรูปแบบของ mline

คำสั่งนี้ไม่สามารถเรียกใช้งานผ่าน Autolisp ได้ เนื่องจากต้องทำการปรับแต่งรูปแบบในกรอบข้อมูลของคำสั่ง mstyle โดยตรง



รูปที่ 6.10 หน้าต่างของ คำสั่ง mstyle

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Load ...	ใช้เรียกแฟ้มข้อมูลที่เก็บข้อมูลของเส้น mline
Save ...	ใช้เก็บแฟ้มข้อมูลที่เก็บข้อมูลของเส้น mline
Add	ใช้สร้างชนิดของเส้น mline ชนิดใหม่
Rename	เปลี่ยนชื่อของชนิดของเส้น mline
Element Properties ...	และ
Multiline Properties ...	ใช้กำหนดรูปแบบของเส้น mline ที่ต้องการสร้าง หรือแก้ไข

solid ใช้สร้างรูประบายทึบ

รูปแบบ (command "solid" จุดที่1 จุดที่2 จุดที่3 .. "")

คำสั่ง solid จะสร้างรูปทึบ โดยพิจารณาทีละ 3 จุด เพราะฉะนั้น จะต้องมีจุดอย่างน้อย 3 จุดอยู่ใน คำสั่งเสมอ โดยคำสั่ง solid จะวาดสามเหลี่ยมทึบต่อกัน โดยดูจากจุด 3 จุดที่เรียงต่อกัน ดูความแตกต่าง ของการสลับตำแหน่งจุด ได้จากรูปที่ 6.11 , 6.12 , 6.13



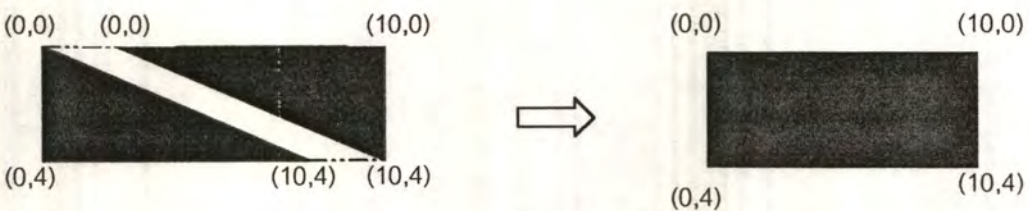
รูปที่ 6.11

รูปที่ 6.12

รูปที่ 6.13

ตัวอย่าง

(command "solid" (list 0 0) (list 10 0) (list 10 4) (list 0 4) "") โปรแกรมจะทำการ วาดรูปสามเหลี่ยมทึบ 2 รูป ประกอบกันเป็นรูปสี่เหลี่ยม คือ สามเหลี่ยม (0,0)-(10,0)-(10,4) และ สามเหลี่ยม (10,0)-(10,4)-(0,4) ดูรูปที่ 6.14 และ 6.15 ประกอบ



รูปที่ 6.14

รูปที่ 6.15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

rectangle วาดรูปสี่เหลี่ยมมุมฉาก

รูปแบบ (command "rectangle" จุดเริ่มต้น จุดสิ้นสุด)

จุดเริ่มต้น , จุดสิ้นสุด คือ มุมทั้งสองของรูปสี่เหลี่ยมมุมฉากที่อยู่ตรงข้ามกัน

ตัวอย่าง

(command "rectangle" (list 0 0) (list 10 5)) วาดรูปสี่เหลี่ยมมุมฉาก จากจุด (0,0) – (10,5) ดูรูปประกอบ

(0,0)



(10,5)

รูปที่ 6.16 คำสั่ง rectangle

text ใช้เขียนข้อความลงบนหน้าจาวาดภาพของAutocad

รูปแบบ (command "text" <option> จุดที่จะเขียน ขนาดตัวอักษร มุมของข้อความหน่วยเรเดียน ข้อความ)

<option> มี 2 ส่วน คือ

1.Justification ใช้บอกแนวระดับในการวางตัวอักษร ปกติจะวางริมซ้าย

2.Style ใช้เลือกรูปแบบของตัวอักษร

ตัวอย่าง

(command "text" (list 0 11) "0.5" "0" "This is our project.") พิมพ์ข้อความ This is our project. ที่ตำแหน่งมุมซ้ายล่างของข้อความ ณ จุด (0,11) ขนาดตัวอักษร เท่ากับ 0.5 มุมเอียงของบรรทัด 0 องศา

(0,11) This is our project.

รูปที่ 6.17 คำสั่ง Text

(command "text" "j" "center" (list 0 5) "1.0" "0" "This is our project.") พิมพ์ข้อความ This is our project. โดยจัดข้อความแบบกึ่งกลาง และมีเส้นบรรทัดอยู่ที่ตำแหน่ง (0,5) ขนาดของตัวอักษร 1.0 มุมเอียงของบรรทัด 0 องศา

This is our project.
(0,5)

รูปที่ 6.18 คำสั่ง Text โดยกำหนดการวางแบบ Center

zoom ใช้ย่อ/ขยาย ส่วนที่ต้องการจะแสดงผลทางจอภาพ

รูปแบบ (command "zoom" จุดเริ่มต้น จุดสิ้นสุด) หรือ (command "zoom" <option>)

<option> ที่ใช้ ได้แก่

- Extents ย่อ / ขยาย วัตถุทุกชิ้นที่วาดไว้ให้เต็มจอภาพ

- 0.3x ย่อเหลือ 0.3 เท่าจากขนาดเดิม ตัวเลข 0.3 สามารถเปลี่ยนได้ตามความต้องการ

ต้องการ

ตัวอย่าง

(command "zoom" "0,0" "11,7") ทำการ ย่อ / ขยาย ขอบเขต (0,0) – (11,7) ให้

เต็มหน้าจอ

(command "zoom" "e") ทำการ ย่อ / ขยาย วัตถุที่วาดไว้ทั้งหมด ให้เต็มหน้าจอ

สรุปผลของโครงการ

เนื่องจากโครงการนี้เป็นโครงการที่นำเนื้อหาทางวิชา Machinery มาเขียนเป็นโปรแกรม เพื่อที่จะคำนวณหาความเร็ว ความเร่ง และแรงที่กระทำของ Four-bar linkage และ Slider-crank

สำหรับ Four-bar linkage นั้นจะมีการป้อนค่าความยาวของเครื่องต่อ 1, 2, 3 และ 4 ค่ามุมที่เครื่องต่อ 2 กระทำต่อแนวแกน X ค่าความเร็วเชิงมุม ความเร่งเชิงมุม ค่าความหนา และความกว้างของเครื่องต่อทั้งหมด รวมทั้งความหนาแน่นของวัสดุที่นำมาสร้างเป็นเครื่องต่อ การที่เราป้อนค่าทั้งหมดนี้จะทำให้เราสามารถที่จะคำนวณหา ความเร็วของแต่ละเครื่องต่อ ความเร่งของแต่ละเครื่องต่อ ตลอดจนแรงที่กระทำต่อเครื่องต่อแต่ละอัน โดยการแสดงผลนั้นจะแสดงออกมาเป็นลักษณะของเวกเตอร์ เพื่อที่จะทำให้เห็นลักษณะและทิศทางของความเร็ว ความเร่ง และแรงที่กระทำได้อย่างชัดเจน

สำหรับ Slider-crank นั้นการป้อนค่าจะมีลักษณะที่ใกล้เคียงกับของ Four-bar linkage คือ ความยาวของเครื่องต่อ 1, 2, 3 และ 4 ค่าของมุมที่เครื่องต่อ 2 กระทำต่อแนวแกน X ค่าความเร็วเชิงมุม และความเร่งเชิงมุมของเครื่องต่อ 2 ค่าความหนา และความกว้างของเครื่องต่อทั้งหมด ค่าความหนาแน่นของเครื่องต่อที่นำมาสร้าง แต่จะมีที่แตกต่างกันอยู่บ้างก็คือ ค่าของเส้นผ่านศูนย์กลางของเครื่องต่อ 4 (เพราะในความเป็นจริงเครื่องต่อ 4 นั้นจะเป็นลักษณะของกระบอกสูบ) โดยค่าที่ป้อนเข้าไปนั้นจะนำไปเป็นค่าประกอบเพื่อที่จะคำนวณหา ความเร็ว ความเร่ง ตลอดจนแรงที่กระทำ โดยจะแสดงผลลัพธ์ออกมาเหมือนกับ Four-bar linkage ก็จะเป็นลักษณะของเวกเตอร์เพื่อที่จะให้เห็นถึง ขนาดและทิศทางของเวกเตอร์ต่าง ๆ

และเมื่อเรากำหนดผลลัพธ์ต่าง ๆ ได้แล้ว เราก็สามารถที่จะแสดงผลของลักษณะของการเคลื่อนที่ได้ ทำให้สามารถที่จะทราบว่า การเคลื่อนที่นั้นจะเป็นไปในลักษณะใด โดยการแสดงผลของการเคลื่อนที่นั้นจะแสดงออกมาให้เห็น ได้อย่างชัดเจน จะมีการเคลื่อนที่ให้เห็นว่ามีทิศทางใด ตลอดจนถ้าเครื่องต่อนั้นมีลักษณะของจุดตาย (deathpoint) ก็จะมีการหยุดการเคลื่อนที่

ดังนั้นสำหรับโครงการชิ้นนี้ค่อนข้างที่จะเป็นที่น่าพอใจ สำหรับผลลัพธ์ที่ได้ตลอดจนการใช้งานของโปรแกรม Autolisp เป็น โปรแกรมที่สามารถเข้าใจได้ง่าย และเอื้ออำนวยต่อการคำนวณได้เป็นอย่างดี

บรรณานุกรม

สมชัย นรเศรษฐ์โสภณ , กลศาสตร์เครื่องจักรกล , โครงการงานตำรา คณะวิศวกรรมศาสตร์
Arthur G.ergman , George N.Sander , Machanism Design , Third Edition , Prentice Hall
George Omura , พื้นฐานการ โปรแกรม AutoLISP



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

(defun c:project (/ l e1 l e2 l e3 l e4 ceta2 ceta3 ceta4 wi th omega2 alfa2 den
  sle1 sle2 sle3 sle4 sdi4 sceta2 sceta3 swi sth somega2 salfa2 sden sfriction
  displaymove pt0 pt1 pt2 pt3
  input_slider_check input_4bars_check menu_check menuchoose
  rad2 rad3 rad4 srad2 srad3
  accen2 accen3 accen4 accet2 accet3 accet4 acceg2 acceg3 acceg4
  cetaaccen2 cetaaccen3 cetaaccen4
  cetaacct2 cetaacct3 cetaacct4
  cetaacceg2 cetaacceg3 cetaacceg4
  saccen2 saccen3 saccen4 saccet2 saccet3 saccet4 sacceg2 sacceg3 sacceg4
  scetaaccen2 scetaaccen3 scetaaccen4
  scetaacct2 scetaacct3 scetaacct4
  scetaacceg2 scetaacceg3 scetaacceg4
  )
  (setq input_4bars_check nil)
  (setq input_slider_check nil)
  (setq menu_check 0)
  (drawmenu)
  (while (/= input_4bars_check 0)
    (if (= menu_check 1) (drawmenu))
    (setq menuchoose (getpoint "MENU :"))
    (cond
      ;INPUT BUTTON
      ((and (>= (car menuchoose) (- 14 3))
        (<= (cadr menuchoose) (+ 17 0.75))
        (<= (car menuchoose) (+ 14 3))
        (>= (cadr menuchoose) (- 17 0.5))
      )
      (command "color" "cyan")
      (menubutton (list 14 17) "INPUT")
    )
  )

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

(command "delay" "1")
(c:input_4bars)
(setq menu_check 1)
(setq input_4bars_check 1)
)

( (and (>= (car menuchoose) (- 14 3)) ;V BUTTON
  (<= (cadr menuchoose) (+ 15.5 0.75))
  (<= (car menuchoose) (+ 14 3))
  (>= (cadr menuchoose) (- 15.5 0.5))
)
(if (= input_4bars_check 1)
  (progn
    ...
    (command "color" "cyan")
    (menubutton (list 14 15.5) "V VECTOR")
    (command "delay" "1")
    (command "color" "7")
    (v_vector)
    (setq menu_check 1)
  )
  (progn
    (command "color" "red")
    (menubutton (list 14 15.5) "V VECTOR")
    (screentxt "You must input data before" (list 4 5))
    (command "delay" "1")
    (command "color" "7")
    (menubutton (list 14 15.5) "V VECTOR")
    (screentxt " " (list 4 5))
  )
)
)
)

```

```

)
( (and (>= (car menuchoose) (- 14 3))           ;A BUTTON
    (<= (cadr menuchoose) (+ 14 0.75))
    (<= (car menuchoose) (+ 14 3))
    (>= (cadr menuchoose) (- 14 0.5))
)
(if (= input_4bars_check 1)
    (progn
      (command "color" "cyan")
      (menubutton (list 14 14) "A VECTOR")
      (command "delay" "1")
      (command "color" "7")
      (a_vector)
      ...
      (setq menu_check 1)
    )
    (progn
      (command "color" "red")
      (menubutton (list 14 14) "A VECTOR")
      (screentxt "You must input data before" (list 4 5))
      (command "delay" "1")
      (command "color" "7")
      (menubutton (list 14 14) "A VECTOR")
      (screentxt " " (list 4 5))
    )
  )
)
)
( (and (>= (car menuchoose) (- 14 3))           ;F BUTTON
    (<= (cadr menuchoose) (+ 12.5 0.75))
    (<= (car menuchoose) (+ 14 3))
    (>= (cadr menuchoose) (- 12.5 0.5))
)

```

```

)
(if (= input_4bars_check 1)
  (progn
    (command "color" "cyan")
    (menubutton (list 14 12.5) "F VECTOR")
    (command "delay" "1")
    (command "color" "7")
    (setq menu_check 1)
    (f_vector)
  )
)
(progn
  (command "color" "red")
  (menubutton (list 14 12.5) "F VECTOR")
  (screentxt "You must input data before" (list 4 5))
  (command "delay" "1")
  (command "color" "7")
  (menubutton (list 14 12.5) "F VECTOR")
  (screentxt " " (list 4 5))
)
)
)
)
((and (>= (car menuchoose) (- 14 3)) ;MOVEMENT BUTTON
  (<= (cadr menuchoose) (+ 11 0.75))
  (<= (car menuchoose) (+ 14 3))
  (>= (cadr menuchoose) (- 11 0.5))
)
)
(if (= input_4bars_check 1)
  (progn
    (command "color" "cyan")
    (menubutton (list 14 11) "MOVEMENT")

```

```

(command "delay" "1")
(command "color" "7")
(setq menu_check 1)
(c:move_4bars)
)
(progn
(command "color" "red")
(menubutton (list 14 11) "MOVEMENT")
(screenstxt "You must input data before" (list 4 5))
(command "delay" "1")
(command "color" "7")
(menubutton (list 14 11) "MOVEMENT")
(screenstxt " " (list 4 5))
)
)
)
)
((and (>= (car menuchoose) (- 36 3)) ;SLIDER INPUT
(<= (cadr menuchoose) (+ 17 0.75))
(<= (car menuchoose) (+ 36 3))
(>= (cadr menuchoose) (- 17 0.5))
)
)
(command "color" "cyan")
(menubutton (list 36 17) "INPUT")
(command "delay" "1")
(c:input_slider)
(setq menu_check 1)
(setq input_slider_check 1)
)
)
((and (>= (car menuchoose) (- 36 3)) ;SLIDER V BUTTON

```

```

(<= (cadr menuchoose) (+ 15.5 0.75))
(<= (car menuchoose) (+ 36 3))
(>= (cadr menuchoose) (- 15.5 0.5))
)
(if (= input_slider_check 1)
  (progn
    (command "color" "cyan")
    (menubutton (list 36 15.5) "V VECTOR")
    (command "delay" "1")
    (command "color" "7")
    (slidevelo)
    (setq menu_check 1)
  )
  ...
  (progn
    (command "color" "red")
    (menubutton (list 36 15.5) "V VECTOR")
    (screentxt "You must input data before" (list 4 5))
    (command "delay" "1")
    (command "color" "7")
    (menubutton (list 36 15.5) "V VECTOR")
    (screentxt " " (list 4 5))
  )
)
)
)
)
( (and (>= (car menuchoose) (- 36 3)) ;SLIDER A BUTTON
  (<= (cadr menuchoose) (+ 14 0.75))
  (<= (car menuchoose) (+ 36 3))
  (>= (cadr menuchoose) (- 14 0.5))
)
)
(if (= input_slider_check 1)

```



```

    (setq menu_check 1)
    (slideforce)
  )
  (progn
    (command "color" "red")
    (menubutton (list 36 12.5) "F VECTOR")
    (screentxt "You must input data before" (list 4 5))
    (command "delay" "1")
    (command "color" "7")
    (menubutton (list 36 12.5) "F VECTOR")
    (screentxt " " (list 4 5))
  )
)
)
)
( (and (>= (car menuchoose) (- 36 3)) ;SLIDER MOVEMENT BUTTON
  (<= (cadr menuchoose) (+ 11 0.75))
  (<= (car menuchoose) (+ 36 3))
  (>= (cadr menuchoose) (- 11 0.5))
)
(if (= input_slider_check 1)
  (progn
    (command "color" "cyan")
    (menubutton (list 36 11) "MOVEMENT")
    (command "delay" "1")
    (command "color" "7")
    (setq menu_check 1)
    (c:move_slider)
  )
)
)
(progn
  (command "color" "red")

```



```

(command "erase" "all" "")
(command "zoom" (list -4 23) (list 42 2))
(command "color" "253")
(command "rectangle" (list -2 18.5) (list 18 10))
(command "rectangle" (list 20 18.5) (list 40 10))
(command "color" "7")
(drawbars 2 12.5 0 14.5 6 17 8 12.5)
(drawslider 22.0 12.5 25.0 16.5 30.0 12.5 2.0 1.0 0.5)
(menubutton (list 14 17) "INPUT")
(if (/= input_4bars _check 1)
  (command "color" "253")
)
(menubutton (list 14 15.5) "V VECTOR")
(menubutton (list 14 14) "A VECTOR")
(menubutton (list 14 12.5) "F VECTOR")
(menubutton (list 14 11) "MOVEMENT")
(command "color" "7")
(menubutton (list 36 17) "INPUT")
(if (/= input_4bars _check 1)
  (command "color" "253")
)
(menubutton (list 36 15.5) "V VECTOR")
(menubutton (list 36 14) "A VECTOR")
(menubutton (list 36 12.5) "F VECTOR")
(menubutton (list 36 11) "MOVEMENT")
(command "color" "7")
(command "rectangle" (list 33 8.5) (list 40 6.5))
(command "text" (list 35 7) "1" "0" "EXIT")
(setq menu_check nil)
(command "erase" "c" (list 4 5) (list 5 5) "")

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

)

(defun menubutton (pnt txt)
  (command "rectangle"
    (list (- (car pnt) 3)
      (+ (cadr pnt) 0.75))
    (list (+ (car pnt) 3)
      (- (cadr pnt) 0.5))
  )
  (command "text" "j" "c" pnt "0.5" "0" txt)
)

(defun drawbars(xx0 yy0 xx1 yy1 xx2 yy2 xx3 yy3
  / pnt0 pnt1 pnt2 pnt3 thic )
  (setq pnt0 (list xx0 yy0))
  (setq pnt1 (list xx1 yy1))
  (setq pnt2 (list xx2 yy2))
  (setq pnt3 (list xx3 yy3))
  (setq thic (* 0.1 (/ (+ (distance pnt0 pnt1) (distance pnt1 pnt2)
    (distance pnt2 pnt3)) 3) ))
  (command "mline" "j" "z" "s" thic "st" "bar" pnt0 pnt1 "")
  (command "mline" pnt1 pnt2 "")
  (command "mline" pnt2 pnt3 "")
  (drawbase pnt0 thic)
  (drawbase pnt3 thic)
)

(defun drawbar(point0 point1 widthz)
  (command "mline" "j" "z" "s" widthz "st" "bar" point0 point1 "")
)

```

```

(defun drawbase (point thick / endpt point1 point2 point3 point4 )
  (setq endpt (list (car point) (- (cadr point) (/ thick 2)) ))
  (command "mline" "j" "z" "s" thick "st" "halfbar" point endpt "")
  (setq point1 (list (- (car endpt) (* thick 2)) (cadr endpt) ))
  (setq point2 (list (+ (car endpt) (* thick 2)) (cadr endpt) ))
  (setq point3 (list (car point1) (- (cadr point1) thick ) ))
  (setq point4 (list (car point2) (- (cadr point2) thick ) ))
  (command "color" "252")
  (command "solid" point1 point2 point3 point4 "")
  (command "color" "254")
  (command "line" point1 point2 "")
  (command "color" "7")
)

(defun zwindow()
  (command "zoom" "e")
  (command "zoom" "0.6x")
)

(defun screennum (input pt)
  (command "erase" "c" pt (list (+ (car pt) 1) (cadr pt)) "")
  (command "text" pt "0.5" "0" (rtos input 2 2))
)

(defun screentxt (input pt)
  (command "erase" "c" pt (list (+ (car pt) 1) (cadr pt)) "")
  (command "text" pt "0.5" "0" input)
)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
(defun input_4bars_length1 ()
  (command "color" "cyan")
  (screentxt "Length1" (list 0 22))
  (entmod (subst (cons 62 4) (assoc 62 dimle1) dimle1))
  (initget (+ 1 2 4))
  (setq le1 (getreal "Input the data of length1 ( m. ) : "))
  (command "color" "7")
  (screentxt "Length1" (list 0 22))
  (setq startpt3 (list (+ 0 le1) 0))
  (screennum le1 (list 6 22))
  (entmod (subst (cons 62 253) (assoc 62 dimle1) dimle1))
  (check_ceta3)
)
```

```
(defun input_4bars_ceta ()
  (command "color" "cyan")
  (screentxt "Ceta" (list 0 20.5 ))
  (entmod (subst (cons 62 4) (assoc 62 dimceta2) dimceta2))
  (initget (+ 1 2 4))
  (setq ceta2 (getreal "Input the data of ceta ( degree ) : "))
  (if (/= le2 nil)
    (setq startpt1 (polar (list 0 0) (* ceta2 0.0174533) le2))
  )
  (command "color" "7")
  (screentxt "Ceta" (list 0 20.5 ))
  (screennum ceta2 (list 6 20.5))
  (entmod (subst (cons 62 253) (assoc 62 dimceta2) dimceta2))
  (check_ceta3)
)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
(defun input_4bars_length2 ()
  (command "color" "cyan")
  (screentxt "Length2" (list 0 19))
  (entmod (subst (cons 62 4) (assoc 62 dimle2) dimle2))
  (initget (+ 1 2 4))
  (setq le2 (getreal "Input the data of length2 ( m.) : "))
  (setq startpt1 (polar (list 0 0) (* ceta2 0.0174533) le2))
  (command "color" "7")
  (screentxt "Length2" (list 0 19))
  (screennum le2 (list 6 19))
  (entmod (subst (cons 62 253) (assoc 62 dimle2) dimle2))
  (check_ceta3)
)
```

```
(defun input_4bars_length3 ()
  (command "color" "cyan")
  (screentxt "Length3" (list 0 17.5))
  (entmod (subst (cons 62 4) (assoc 62 dimle3) dimle3))
  (if (= le4 nil)
    (progn
      (initget (+ 1 2 4))
      (setq le3 (getreal "Input the data of length3 ( m.) : "))
    )
    (progn
      (initget (+ 1 2 4))
      (setq le3 (getreal (strcat "Input the data of length3 ["
        (rtos (abs (- (distance startpt1 startpt3) le4)) 2 2)
        " - "
        (rtos (abs (+ (distance startpt1 startpt3) le4)) 2 2)

```

```

    " m.] : ")))
)
)
(command "color" "7")
(screenxt "Length3" (list 0 17.5))
(screennum le3 (list 6 17.5))
(entmod (subst (cons 62 253) (assoc 62 dimle3) dimle3))
(check_ceta3)
)

(defun input_4bars_length4 ()
  (command "color" "cyan")
  (screenxt "Length4" (list 0 16))
  (entmod (subst (cons 62 4) (assoc 62 dimle4) dimle4))
  (initget (+ 1 2 4))
  (setq le4 (getreal (strcat "Input the data of length4 ["
    (rtos (abs (- (distance startpt1 startpt3) le3)) 2 2)
    " - "
    (rtos (abs (+ (distance startpt1 startpt3) le3)) 2 2)
    " m.] : ")))
  (command "color" "7")
  (screenxt "Length4" (list 0 16))
  (screennum le4 (list 6 16))
  (entmod (subst (cons 62 253) (assoc 62 dimle4) dimle4))
  (check_ceta3)
)

(defun input_4bars_width ()
  (command "color" "cyan")
  (screenxt "Width" (list 0 14.5))

```

```
(entmod (subst (cons 62 4) (assoc 62 dimwi) dimwi))
(initget (+ 1 2 4))
(setq wi (getreal "Input the data of width ( m.) : "))
(command "color" "7")
(screenent "Width" (list 0 14.5))
(screennum wi (list 6 14.5))
(entmod (subst (cons 62 253) (assoc 62 dimwi) dimwi))
)
```

```
(defun input_4bars_thickness ()
  (command "color" "cyan")
  (screenent "Thickness" (list 0 13))
  (entmod (subst (cons 62 4) (assoc 62 dimth) dimth))
  (initget (+ 1 2 4))
  (setq th (getreal "Input the data of thick ( m.) : "))
  (command "color" "7")
  (screenent "Thickness" (list 0 13))
  (screennum th (list 6 13))
  (entmod (subst (cons 62 253) (assoc 62 dimth) dimth))
)
```

```
(defun input_4bars_velocity ()
  (command "color" "cyan")
  (screenent "Velocity" (list 0 11.5))
  (initget (+ 1 2))
  (setq omega2 (getreal "Input the data of velocity (m/s)
                        +/- for the director : "))
  (command "color" "7")
  (screenent "Velocity" (list 0 11.5))
  (screennum omega2 (list 6 11.5))
)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

)

```
(defun input_4bars_acceleration ()
  (command "color" "cyan")
  (screentxt "Accel" (list 0 10))
  (initget (+ 1))
  (setq alfa2 (getreal "Input the data of acceleration (m2/s) : "))
  (command "color" "7")
  (screentxt "Accel" (list 0 10))
  (screennum alfa2 (list 6 10))
)
```

```
(defun input_4bars_density ()
  (command "color" "cyan")
  (screentxt "Density" (list 0 8.5))
  (initget (+ 1 2 4))
  (setq den (getreal "Input the data of density (kg/m3) : "))
  (command "color" "7")
  (screentxt "Density" (list 0 8.5))
  (screennum den (list 6 8.5))
)
```

```
(defun c:input_4bars (/ checkle4 menupoint startpt1 startpt3
  dimle1 dimle2 dimle3 dimle4 dimceta2 dimwi dimth)
  (setq checkle4 nil)
  (setq menupoint nil)
  (command "erase" "all" "")
  (command "zoom" (list -0.5 23) (list 35 2))
```

(command "color" "7")

(drawbars 15 14 20 20 27 18 25 14)

(command "rectangle" (list 14 9.5) (list 26 7.5))

(command "rectangle" (list 13 8.5) (list 25 6.5))

(command "line" (list 14 9.5) (list 13 8.5) "")

(command "line" (list 14 7.5) (list 13 6.5) "")

(command "line" (list 26 7.5) (list 25 6.5) "")

(command "line" (list 26 9.5) (list 25 8.5) "")

(command "color" "253")

(command "dimtih" "on")

(command "dimtoh" "on")

(command "dimtod" "0")

(command "dimasz" "0.6")

(command "dimexe" "0.2")

(command "dimexo" "0.2")

(command "dimtxt" "0.5")

(command "dimaligned" (list 15 14) (list 20 20) "t" "Length2" (list 17.6 20))

(setq dimle2 (entget (ssname (ssget "l") 0)))

(command "dimaligned" (list 20 20) (list 27 18) "t" "Length3" (list 27 19.5))

(setq dimle3 (entget (ssname (ssget "l") 0)))

(command "dimaligned" (list 25 14) (list 27 18) "t" "Length4" (list 27 14))

(setq dimle4 (entget (ssname (ssget "l") 0)))

(command "dimlinear" (list 15 13) (list 25 13) "t" "Length1" (list 25 11))

(setq dimle1 (entget (ssname (ssget "l") 0)))

(command "dimaligned" (list 25 6.5) (list 26 7.5) "t" "Thickness" (list 27 7.5))

(setq dimth (entget (ssname (ssget "l") 0)))

(command "dimlinear" (list 13 8.5) (list 13 6.5) "t" "Width" (list 11.6 8.5))

(setq dimwi (entget (ssname (ssget "l") 0)))

(command "dimangular" (list 16 15) (list 16 13.6) "t" "Ceta" (list 19 14))

```

(setq dimceta2 (entget (ssname (ssget "1") 0)))
(command "color" "7")

(screenxt "Length1" (list 0 22))
(command "rectangle" (list -0.2 22.75) (list 4.5 21.5))
(if (/= le1 nil)
  (screennum le1 (list 6 22))
  (progn
    (command "color" "251")
    (screenxt "No data" (list 6 22))
    (command "color" "7")
  )
)

(screenxt "Ceta" (list 0 20.5))
(command "rectangle" (list -0.2 21.25) (list 4.5 20))
(if (/= ceta2 nil)
  (screennum ceta2 (list 6 20.5))
  (progn
    (command "color" "251")
    (screenxt "No data" (list 6 20.5))
    (command "color" "7")
  )
)

(screenxt "Length2" (list 0 19))
(command "rectangle" (list -0.2 19.75) (list 4.5 18.5))
(if (/= le2 nil)
  (screennum le2 (list 6 19))
  (progn

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
(command "color" "251")
```

```
(screentxt "No data" (list 6 19))
```

```
(command "color" "7")
```

```
)
```

```
)
```

```
(screentxt "Length3" (list 0 17.5))
```

```
(command "rectangle" (list -0.2 18.25) (list 4.5 17))
```

```
(if (/= le3 nil)
```

```
  (screennum le3 (list 6 17.5))
```

```
  (progn
```

```
    (command "color" "251")
```

```
    (screentxt "No data" (list 6 17.5))
```

```
    (command "color" "7")
```

```
  )
```

```
)
```

```
(screentxt "Length4" (list 0 16))
```

```
(command "rectangle" (list -0.2 16.75) (list 4.5 15.5))
```

```
(if (/= le4 nil)
```

```
  (screennum le4 (list 6 16))
```

```
  (progn
```

```
    (command "color" "251")
```

```
    (screentxt "No data" (list 6 16))
```

```
    (command "color" "7")
```

```
  )
```

```
)
```

```
(screentxt "Width" (list 0 14.5))
```

```
(command "rectangle" (list -0.2 15.25) (list 4.5 14))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
(if (/= wi nil)
  (screennum wi (list 6 14.5))
  (progn
    (command "color" "251")
    (screentxt "No data" (list 6 14.5))
    (command "color" "7")
  )
)
```

```
(screentxt "Thickness" (list 0 13))
(command "rectangle" (list -0.2 13.75) (list 4.5 12.5))
(if (/= th nil)
  (screennum th (list 6 13))
  (progn
    (command "color" "251")
    (screentxt "No data" (list 6 13))
    (command "color" "7")
  )
)
```

```
(screentxt "Velocity" (list 0 11.5))
(command "rectangle" (list -0.2 12.25) (list 4.5 11))
(if (/= omega2 nil)
  (screennum omega2 (list 6 11.5))
  (progn
    (command "color" "251")
    (screentxt "No data" (list 6 11.5))
    (command "color" "7")
  )
)
```

```

(screentxt "Accel" (list 0 10))
(command "rectangle" (list -0.2 10.75) (list 4.5 9.5))
(if (/= alfa2 nil)
  (screennum alfa2 (list 6 10))
  (progn
    (command "color" "251")
    (screentxt "No data" (list 6 10))
    (command "color" "7")
  )
)
)

(screentxt "Density" (list 0 8.5))
(command "rectangle" (list -0.2 9.25) (list 4.5 8))
(if (/= den nil)
  (screennum den (list 6 8.5))
  (progn
    (command "color" "251")
    (screentxt "No data" (list 6 8.5))
    (command "color" "7")
  )
)
)

(command "color" "150")
(screentxt "DONE" (list 31.3 3.15))
(command "rectangle" (list 30 4) (list 34.5 2.75))
(command "color" "7")

```

```
(if (= le1 nil) (input_4bars_length1) )
```

```
(if (= ceta2 nil) (input_4bars_ceta) )
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

(if (= le2 nil) (input_4bars_length2) )
(if (= le3 nil) (input_4bars_length3) )
(if (= le4 nil) (input_4bars_length4) )
(if (= wi nil) (input_4bars_width) )
(if (= th nil) (input_4bars_thickness) )
(if (= omega2 nil) (input_4bars_velocity) )
(if (= alfa2 nil) (input_4bars_acceleration) )
(if (= den nil) (input_4bars_density) )

(while (= checkle4 nil)
  (setq startpt1 (polar (list 0 0) (* ceta2 0.0174533) le2))
  (setq startpt3 (list (+ 0 le1) 0))
  (if (and (>= (car menupoint) 30)
        (<= (cadr menupoint) 4)
        (<= (car menupoint) 34.5)
        (>= (cadr menupoint) 2.75) )
      (progn
        (command "erase" (list 31.3 3.15) (list 31.3 3.15) "")
        (command "color" "150")
        (screentxt "DONE" (list 31.3 3.15))
        (command "color" "7")
      )
  )
)

(setq menupoint (getpoint "Edit data : "))
(cond
  ((and (>= (car menupoint) -0.2)
        (<= (cadr menupoint) 22.75)
        (<= (car menupoint) 4.5)
        (>= (cadr menupoint) 21.5) ) (input_4bars_length1) )

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
( (and (>= (car menupoint) -0.2)
      (<= (cadr menupoint) 21.25)
      (<= (car menupoint) 4.5)
      (>= (cadr menupoint) 20) ) (input_4bars_ceta) )
```

```
( (and (>= (car menupoint) -0.2)
      (<= (cadr menupoint) 19.75)
      (<= (car menupoint) 4.5)
      (>= (cadr menupoint) 18.5) ) (input_4bars_length2) )
```

```
( (and (>= (car menupoint) -0.2)
      (<= (cadr menupoint) 18.25)
      (<= (car menupoint) 4.5)
      (>= (cadr menupoint) 17) ) (input_4bars_length3) )
```

```
( (and (>= (car menupoint) -0.2)
      (<= (cadr menupoint) 16.75)
      (<= (car menupoint) 4.5)
      (>= (cadr menupoint) 15.5) ) (input_4bars_length4) )
```

```
( (and (>= (car menupoint) -0.2)
      (<= (cadr menupoint) 15.25)
      (<= (car menupoint) 4.5)
      (>= (cadr menupoint) 14) ) (input_4bars_width) )
```

```
( (and (>= (car menupoint) -0.2)
      (<= (cadr menupoint) 13.75)
      (<= (car menupoint) 4.5)
      (>= (cadr menupoint) 12.5) ) (input_4bars_thickness) )
```

```
( (and (>= (car menupoint) -0.2)
      (<= (cadr menupoint) 12.25)
      (<= (car menupoint) 4.5)
      (>= (cadr menupoint) 11) ) (input_4bars_velocity) )
```

```
( (and (>= (car menupoint) -0.2)
      (<= (cadr menupoint) 10.75)
      (<= (car menupoint) 4.5)
      (>= (cadr menupoint) 9.5) ) (input_4bars_acceleration) )
```

```
( (and (>= (car menupoint) -0.5)
      (<= (cadr menupoint) 9.25)
      (<= (car menupoint) 4.5)
      (>= (cadr menupoint) 8) ) (input_4bars_density) )
```

```
( (and (>= (car menupoint) 30) ; -----
      (<= (cadr menupoint) 4) ; DONE
      (<= (car menupoint) 34.5) ; -----
      (>= (cadr menupoint) 2.75) )
```

```
(command "erase" "c" (list 31.3 3.15) (list 31.3 3.15) "")
(command "color" "cyan")
(screenxt "DONE" (list 31.3 3.15))
```

```
(if (and (>= le4 (abs (- (distance startpt1 startpt3) le3)))
      (<= le4 (abs (+ (distance startpt1 startpt3) le3))))
    (progn
      (command "erase" "c" (list 5 2.5) (list 13 2.5) "")
      (command "color" "green" )
```



```
(setq pt0 (list 0 0))
```

```
(setq pt3 (list (+ 0 le1) 0))
```

```
(setq pt1 (polar pt0 (* ceta2 0.0174533) le2))
```

```
(setq h1 (car pt1))
```

```
(setq k1 (cadr pt1))
```

```
(setq r1 le3)
```

```
(setq h2 (car pt3))
```

```
(setq k2 (cadr pt3))
```

```
(setq r2 le4)
```

```
(setq za (* -2 h1))
```

```
(setq zb (* -2 k1))
```

```
(setq zc (+ (* h1 h1) (* k1 k1) (* -1 r1 r1)))
```

```
(setq zd (* -2 h2))
```

```
(setq ze (* -2 k2))
```

```
(setq zf (+ (* h2 h2) (* k2 k2) (* -1 r2 r2)))
```

```
(if (<= (abs (- zb ze)) 0.1)
```

```
(progn
```

```
  (setq a (+ (expt (- zb ze) 2) (expt (- zd za) 2))))
```

```
  (setq b (+ (* 2 (- zb ze) (- zc zf)) (* za (- zb ze) (- zd za))
```

```
    (* zb (expt (- zd za) 2))))
```

```
  (setq c (+ (expt (- zc zf) 2) (* za (- zc zf) (- zd za))
```

```
    (* zc (expt (- zd za) 2))))
```

```
(if (>= (- (* b b) (* 4 a c)) 0)
```

```
(progn
```

```
  (setq d (sqrt (- (* b b) (* 4 a c))))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

(setq crossy1 (/ (- d b) (* 2 a)))
(setq crossy2 (/ (* -1 (+ d b)) (* 2 a)))
(setq crossx1 (/ (+ (* (- zb ze) crossy1) (- zc zf)) (- zd za)))
(setq crossx2 (/ (+ (* (- zb ze) crossy2) (- zc zf)) (- zd za)))
))
)
(progn
  (setq a (+ (expt (- ze zb) 2) (expt (- za zd) 2)))
  (setq b (+ (* 2 (- za zd) (- zc zf)) (* zb (- za zd) (- ze zb)
    (* za (expt (- ze zb) 2)))))
  (setq c (+ (expt (- zc zf) 2) (* zb (- zc zf) (- ze zb))
    (* zc (expt (- ze zb) 2)))))
  (if (>= (- (* b b) (* 4 a c)) 0)
    (progn
      (setq d (sqrt (- (* b b) (* 4 a c))))
      (setq crossx1 (/ (- d b) (* 2 a)))
      (setq crossx2 (/ (* -1 (+ d b)) (* 2 a)))
      (setq crossy1 (/ (+ (* crossx1 (- za zd)) (- zc zf)) (- ze zb)))
      (setq crossy2 (/ (+ (* crossx2 (- za zd)) (- zc zf)) (- ze zb)))
    ))
  )
  (if (>= (- (* b b) (* 4 a c)) 0)
    (progn
      (setq crosspt1 (list crossx1 crossy1))
      (setq crosspt2 (list crossx2 crossy2))
      (if (>= (car crosspt1) (car crosspt2))
        (setq pt2 crosspt1)
      )
    )
  )
)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    (setq pt2 crosspt2)
  )
  (setq ceta4 (* (angle pt3 pt2) 57.29575496))
)
(progn
  (setq ceta3 nil)
  (setq ceta4 nil)
)
)
))
)

(defun c:move_4bars (/ circlemove stepmove counter countercheck cetamove
  button button_check end_check
  numercheck numercheck_start numercheck_end numer_ceta
  chpt1 chpt2 pt2s h1 k1 r1 h2 k2 r2 za zb zc zd ze zf a b c d
  mvpnt21 mvpnt31 mvpnt41 mvpnt22 mvpnt32 mvpnt42
  crossx1 crossy1 crossx2 crossy2 crosspt1 crosspt2)

  (setq end_check 1)
  (while (= end_check 1)

    (command "erase" "all" "")
    (command "zoom" (list (- 0 le2 th) (+ 0 (max le2 le4) th))
      (list (+ le1 le4 th (* 0.4 (max le2 le4)))
        (- 0 (max le2 le4) th) )
    (setq button_check nil)

    (while (>= ceta2 360) (setq ceta2 (- ceta2 360)))

```

```
(while (< ceta2 0) (setq ceta2 (+ ceta2 360)))
```

```
(setq move_line 1)
```

```
(setq cetamove ceta2)
```

```
(if (= displaymove nil) (setq displaymove 10))
```

```
(if (> omega2 0) (setq stepmove 0.1) (setq stepmove -0.1))
```

```
(setq pt0 (list 0 0))
```

```
(setq pt3 (list (+ 0 le1) 0))
```

```
(setq pt1 (polar pt0 (* cetamove 0.0174533) le2))
```

```
(setq h1 (car pt1))
```

```
(setq k1 (cadr pt1))
```

```
(setq r1 le3)
```

```
(setq h2 (car pt3))
```

```
(setq k2 (cadr pt3))
```

```
(setq r2 le4)
```

```
(setq za (* -2 h1))
```

```
(setq zb (* -2 k1))
```

```
(setq zc (+ (* h1 h1) (* k1 k1) (* -1 r1 r1)))
```

```
(setq zd (* -2 h2))
```

```
(setq ze (* -2 k2))
```

```
(setq zf (+ (* h2 h2) (* k2 k2) (* -1 r2 r2)))
```

```
(if (<= (abs (- zb ze)) 0.1)
```

```
(progn
```

```
  (setq a (+ (expt (- zb ze) 2) (expt (- zd za) 2)))
```

```
  (setq b (+ (* 2 (- zb ze) (- zc zf)) (* za (- zb ze) (- zd za))
```

```
    (* zb (expt (- zd za) 2))))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

(setq crosspt2 (list crossx2 crossy2))

(if (>= (car crosspt1) (car crosspt2))
    (setq pt2 crosspt1)
    (setq pt2 crosspt2)
)

(setq ceta4 (* (angle pt3 pt2) 57.29575496))

(command "layer" "set" "0" "")
(drawbase pt0 wi)
(drawbase pt3 wi)
(command "layer" "make" "DYNAMIC" "")
(drawbar pt0 pt1 wi)
(drawbar pt1 pt2 wi)
(drawbar pt2 pt3 wi)
(setq chpt2 1)
(setq countercheck (abs (/ displaymove stepmove)))
(setq counter 0)
(setq circlemove 0)
(setq mvpt21 (list (/ (+ (car pt0) (car pt1)) 2) (/ (+ (cadr pt0) (cadr pt1)) 2) ))
(setq mvpt31 (list (/ (+ (car pt1) (car pt2)) 2) (/ (+ (cadr pt1) (cadr pt2)) 2) ))
(setq mvpt41 (list (/ (+ (car pt2) (car pt3)) 2) (/ (+ (cadr pt2) (cadr pt3)) 2) ))

(command "delay" "1")

(while (and (< (abs circlemove) 360) (/= chpt2 nil))
    (setq counter (+ counter 1))
    (setq circlemove (+ circlemove stepmove))
    (setq cetamove (+ cetamove stepmove))
    (while (>= cetamove 360) (setq cetamove (- cetamove 360))))

```

```
(while (< cetamove 0) (setq cetamove (+ cetamove 360)))
```

```
(setq chpt1 (polar pt0 (* cetamove 0.0174533) le2))
```

```
(setq h1 (car chpt1))
```

```
(setq k1 (cadr chpt1))
```

```
(setq r1 le3)
```

```
(setq h2 (car pt3))
```

```
(setq k2 (cadr pt3))
```

```
(setq r2 le4)
```

```
(setq za (* -2 h1))
```

```
(setq zb (* -2 k1))
```

```
(setq zc (+ (* h1 h1) (* k1 k1) (* -1 r1 r1)))
```

```
(setq zd (* -2 h2))
```

```
(setq ze (* -2 k2))
```

```
(setq zf (+ (* h2 h2) (* k2 k2) (* -1 r2 r2)))
```

```
(if (<= (abs (- zb ze)) 0.1)
```

```
(progn
```

```
(setq a (+ (expt (- zb ze) 2) (expt (- zd za) 2)))
```

```
(setq b (+ (* 2 (- zb ze) (- zc zf)) (* za (- zb ze) (- zd za))
```

```
( * zb (expt (- zd za) 2))))
```

```
(setq c (+ (expt (- zc zf) 2) (* za (- zc zf) (- zd za))
```

```
( * zc (expt (- zd za) 2))))
```

```
(if (>= (- (* b b) (* 4 a c)) 0)
```

```
(progn
```

```
(setq d (sqrt (- (* b b) (* 4 a c))))
```

```
(setq crossy1 (/ (- d b) (* 2 a)))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

(setq crossy2 (/ (* -1 (+ d b)) (* 2 a)))
(setq crossx1 (/ (+ (* (- zb ze) crossy1) (- zc zf)) (- zd za)))
(setq crossx2 (/ (+ (* (- zb ze) crossy2) (- zc zf)) (- zd za)))
(setq crosspt1 (list crossx1 crossy1))
(setq crosspt2 (list crossx2 crossy2))
(if (<= (distance crosspt1 pt2) (distance crosspt2 pt2))
    (setq pt2 crosspt1)
    (setq pt2 crosspt2))
)
) (setq chpt2 nil)
)
)
(progn
  (setq a (+ (expt (- ze zb) 2) (expt (- za zd) 2)))
  (setq b (+ (* 2 (- za zd) (- zc zf)) (* zb (- za zd) (- ze zb))
    (* za (expt (- ze zb) 2))))
  (setq c (+ (expt (- zc zf) 2) (* zb (- zc zf) (- ze zb))
    (* zc (expt (- ze zb) 2))))
  (if (>= (- (* b b) (* 4 a c)) 0)
    (progn
      (setq d (sqrt (- (* b b) (* 4 a c))))
      (setq crossx1 (/ (- d b) (* 2 a)))
      (setq crossx2 (/ (* -1 (+ d b)) (* 2 a)))
      (setq crossy1 (/ (+ (* crossx1 (- za zd)) (- zc zf)) (- ze zb)))
      (setq crossy2 (/ (+ (* crossx2 (- za zd)) (- zc zf)) (- ze zb)))
      (setq crosspt1 (list crossx1 crossy1))
      (setq crosspt2 (list crossx2 crossy2))
      (if (<= (distance crosspt1 pt2) (distance crosspt2 pt2))
          (setq pt2 crosspt1)
          (setq pt2 crosspt2))
    )
  )
)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

      (setq pt2 crosspt2)
    )
  ) (setq chpt2 nil)
)
)
)
)
(if (/= chpt2 nil) (setq ceta4 (* (angle pt3 pt2) 57.29575496)) )

(if (or (= chpt2 nil) (= counter countercheck) (>= (abs circlemove) 360))
  (progn
    (if (= chpt2 nil)
      (progn
        (setq numercheck nil)
        (setq numercheck_start (- cetamove stepmove))
        (setq numercheck_end cetamove)
        (while (= numercheck nil)
          (setq numercheck_ceta (/ (+ numercheck_start numercheck_end) 2))
          (setq pt1 (polar pt0 (* numercheck_ceta 0.0174533) le2))

          (setq h1 (car pt1))
          (setq k1 (cadr pt1))
          (setq r1 le3)
          (setq h2 (car pt3))
          (setq k2 (cadr pt3))
          (setq r2 le4)

          (setq za (* -2 h1))
          (setq zb (* -2 k1))
          (setq zc (+ (* h1 h1) (* k1 k1) (* -1 r1 r1)))
          (setq zd (* -2 h2))

```

```

(setq ze (* -2 k2))
(setq zf (+ (* h2 h2) (* k2 k2) (* -1 r2 r2)))

(if (<= (abs (- zb ze)) 0.1)
  (progn
    (setq a (+ (expt (- zb ze) 2) (expt (- zd za) 2)))
    (setq b (+ (* 2 (- zb ze) (- zc zf)) (* za (- zb ze) (- zd za))
              (* zb (expt (- zd za) 2))))
    (setq c (+ (expt (- zc zf) 2) (* za (- zc zf) (- zd za))
              (* zc (expt (- zd za) 2))))

    (if (>= (- (* b b) (* 4 a c)) 0)
      (progn
        (setq d (sqrt (- (* b b) (* 4 a c))))
        (setq crossy1 (/ (- d b) (* 2 a)))
        (setq crossy2 (/ (* -1 (+ d b) (* 2 a)))
        (setq crossx1 (/ (+ (* (- zb ze) crossy1) (- zc zf)) (- zd za)))
        (setq crossx2 (/ (+ (* (- zb ze) crossy2) (- zc zf)) (- zd za)))
        (setq crosspt1 (list crossx1 crossy1))
        (setq crosspt2 (list crossx2 crossy2))
        (if (<= (distance crosspt1 pt2) (distance crosspt2 pt2))
          (setq pt2 crosspt1)
          (setq pt2 crosspt2))
        )
      ) (setq pt2s nil)
    )
  )
  )
  (progn
    (setq a (+ (expt (- ze zb) 2) (expt (- za zd) 2)))
    (setq b (+ (* 2 (- za zd) (- zc zf)) (* zb (- za zd) (- ze zb))
  
```

```

(* za (expt (- ze zb) 2))))
(setq c (+ (expt (- zc zf) 2) (* zb (- zc zf) (- ze zb))
(* zc (expt (- ze zb) 2))))

(if (>= (- (* b b) (* 4 a c)) 0)
  (progn
    (setq d (sqrt (- (* b b) (* 4 a c))))
    (setq crossx1 (/ (- d b) (* 2 a)))
    (setq crossx2 (/ (* -1 (+ d b) (* 2 a)))
    (setq crossy1 (/ (+ (* crossx1 (- za zd)) (- zc zf)) (- ze zb)))
    (setq crossy2 (/ (+ (* crossx2 (- za zd)) (- zc zf)) (- ze zb)))
    (setq crosspt1 (list crossx1 crossy1))
    (setq crosspt2 (list crossx2 crossy2))
    (if (<= (distance crosspt1 pt2) (distance crosspt2 pt2))
      (setq pt2 crosspt1)
      (setq pt2 crosspt2))
    )
  ) (setq pt2s nil)
)
)
)
)
)
(if (/= pt2s nil) (setq ceta4 (* (angle pt3 pt2) 57.29575496)))

(if (= pt2s nil)
  (progn
    (setq numercheck_end numercheck_ceta)
    (setq pt2s 1)
  )
  (progn
    (if (< (abs (- numercheck_start numercheck_ceta)) 0.001)

```

```

    (setq numercheck 1))
  (setq numercheck_start numercheck_ceta)
)
)
)
  (setq cetamove numercheck_ceta)
)
)

(setq counter 0)
(setq pt1 (polar pt0 (* cetamove 0.0174533) le2))
(command "layer" "lock" "0" "")
(command "erase" "all" "")
...
(drawbar pt0 pt1 wi)
(drawbar pt1 pt2 wi)
(drawbar pt2 pt3 wi)

(command "layer" "unlock" "0" "set" "0" "")

(if (= move_line 1)
  (progn
    (setq mvpnt22 (list (/ (+ (car pt0) (car pt1)) 2) (/ (+ (cadr pt0) (cadr pt1)) 2) ))
    (setq mvpnt32 (list (/ (+ (car pt1) (car pt2)) 2) (/ (+ (cadr pt1) (cadr pt2)) 2) ))
    (setq mvpnt42 (list (/ (+ (car pt2) (car pt3)) 2) (/ (+ (cadr pt2) (cadr pt3)) 2) ))

    (command "color" "blue")
    (command "line" mvpnt21 mvpnt22 "" )
    (command "color" "green")
    (command "line" mvpnt31 mvpnt32 "" )
  )
)

```

```

(command "color" "red" )
(command "line" mvpnt41 mvpnt42 "" )

(setq mvpnt21 mvpnt22)
(setq mvpnt31 mvpnt32)
(setq mvpnt41 mvpnt42)
)
)
(command "color" "7")
(command "layer" "set" "DYNAMIC" "")
)
)
)
(command "rectangle" (list (+ le1 le4)
(* (max le2 le4) -0.75) )
(list (+ (+ le1 le4) (* 0.4 (max le2 le4)))
(* (max le2 le4) -0.85) ) )
(command "text" "j" "c"
(list (+ le1 le4 (* 0.2 (max le2 le4)))
(* (max le2 le4) -0.825) )
(* (max le2 le4) 0.05) "0" "REPLAY" )

(command "rectangle" (list (+ le1 le4)
(* (max le2 le4) -0.9))
(list (+ (+ le1 le4) (* 0.4 (max le2 le4)))
(* (max le2 le4) -1) ) )
(command "text" "j" "c"
(list (+ le1 le4 (* 0.2 (max le2 le4)))
(* (max le2 le4) -0.975) )
(* (max le2 le4) 0.05) "0" "DONE" )

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

(while (= button_check nil)
  (setq button (getpoint "Replay or Done : "))
  (cond
    ((and (>= (car button) (+ le1 le4))
      (<= (cadr button) (* (max le2 le4) -0.75))
      (<= (car button) (+ (+ le1 le4) (* 0.4 (max le2 le4))))
      (>= (cadr button) (* (max le2 le4) -0.85))
    )
    (command "color" "cyan")
    (command "rectangle"
      (list (+ le1 le4)
        (* (max le2 le4) -0.75) )
      (list (+ (+ le1 le4) (* 0.4 (max le2 le4)))
        (* (max le2 le4) -0.85) ) )
    (command "text" "j" "c"
      (list (+ le1 le4 (* 0.2 (max le2 le4)))
        (* (max le2 le4) -0.825) )
        (* (max le2 le4) 0.05) "0" "REPLAY" )
    (command "delay" "1")
    (command "color" "7")
    (setq button_check 1)
  )

  ((and (>= (car button) (+ le1 le4))
    (<= (cadr button) (* (max le2 le4) -0.9))
    (<= (car button) (+ (+ le1 le4) (* 0.4 (max le2 le4))))
    (>= (cadr button) (* (max le2 le4) -1))
  )
  (command "color" "cyan")

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
(command "rectangle"
  (list (+ le1 le4)
    (* (max le2 le4) -0.9) )
  (list (+ (+ le1 le4) (* 0.4 (max le2 le4)))
    (* (max le2 le4) -1) ) )
```

```
(command "text" "j" "c"
  (list (+ (+ le1 le4) (* 0.2 (max le2 le4)))
    (* (max le2 le4) -0.975) )
  (* (max le2 le4) 0.05) "0" "DONE" )
```

```
(setq button_check 1)
```

```
(setq end_check nil)
```

```
(command "delay" "1")
```

```
(command "color" "7")
```

```
...
```

```
)
```

```
)
```

```
)
```

```
)
```

```
)
```

```
(defun drawslider(xx0 yy0 xx1 yy1 xx2 yy2 pis_le pis_di thickz
  / rim )
```

```
(command "mline" "j" "z" "s" thickz "st" "halfbar"
```

```
(list xx0 yy0) (list xx0 (- yy0 (/ pis_di 2))) "")
```

```
(setq rim (/ (- xx2 xx0) 6))
```

```
(command "color" "252")
```

```
(command "solid" (list (- xx0 rim) (- yy0 (/ pis_di 2)))
```

```
(list (- xx0 rim) (- yy0 (/ pis_di 2) thickz))
```

```
(list (+ xx2 (/ pis_le 2) rim) (- yy0 (/ pis_di 2)))
```

```
(list (+ xx2 (/ pis_le 2) rim) (- yy0 (/ pis_di 2) thickz))
```

```
"" )
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

(command "color" "254")

(command "line" (list (- xx0 rim) (- yy0 (/ pis_di 2)))
  (list (+ xx2 (/ pis_le 2) rim) (- yy0 (/ pis_di 2)))
  "" )

(command "color" "7")

(command "mline" "j" "z" "s" thickz "st" "bar"
  (list xx0 yy0) (list xx1 yy1) "")

(command "mline" (list xx1 yy1) (list xx2 yy2) "")

(command "rectangle" (list (- xx2 (/ pis_le 2)) (+ yy2 (/ pis_di 2)) )
  (list (+ xx2 (/ pis_le 2)) (- yy2 (/ pis_di 2)) )
)

)

)

(defun input_slider_length2 (/ sceta2temp)
  (if (/= sceta2 nil)
    (progn
      (setq sceta2temp sceta2)
      (while (>= sceta2temp 360) (setq sceta2temp (- sceta2temp 360)))
      (while (< sceta2temp 0) (setq sceta2temp (+ sceta2temp 360)))
    )
  )

  (command "color" "cyan")

  (screentxt "Length2" (list 0 22))

  (entmod (subst (cons 62 4) (assoc 62 dimsle2) dimsle2))

  (if (or (= sle3 nil) (= sceta2 nil) (= sceta2temp 0) (= sceta2temp 180))
    (progn
      (initget (+ 1 2 4))

      (setq sle2 (getreal "Input the data of length2 ( m. ) : "))

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

)
(progn
  (initget (+ 1 2 4))
  (setq sle2 (getreal (strcat "Input the data of length2 [ less than "
    (rtos (abs (/ sle3 (sin (* sceta2 0.0174533)))) 2 2)
    " m.] : ")))
)
)
(command "color" "7")
(screenentxt "Length2" (list 0 22))
(screennum sle2 (list 6 22))
(entmod (subst (cons 62 253) (assoc 62 dimsle2) dimsle2))
(if (/= sceta2 nil)
  (setq startpt1 (polar (list 0 0) (* sceta2 0.0174533) sle2))
)
(check_sceta3)
)
(defun input_slider_ceta ()
  (command "color" "cyan")
  (screenentxt "Ceta" (list 0 20.5))
  (entmod (subst (cons 62 4) (assoc 62 dimsceta2) dimsceta2))
  (initget 1)
  (setq sceta2 (getreal "Input the data of ceta ( degree ) : "))
  (if (/= sle2 nil)
    (setq startpt1 (polar (list 0 0) (* sceta2 0.0174533) sle2))
  )
  (command "color" "7")
  (screenentxt "Ceta" (list 0 20.5))
  (screennum sceta2 (list 6 20.5))

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
(entmod (subst (cons 62 253) (assoc 62 dimsceta2) dimsceta2))
(check_sceta3)
)
```

```
(defun input_slider_length3 ()
  (command "color" "cyan")
  (screentxt "Length3" (list 0 19))
  (entmod (subst (cons 62 4) (assoc 62 dimsle3) dimsle3))
  (if (/= sle2 nil)
    (progn
      (initget (+ 1 2 4))
      (setq sle3 (getreal (strcat "Input the data of length3 [ more than "
        (rtos (abs (cadr startpt1)) 2 2 ) " m.] : ")))
    )
    (progn
      (initget (+ 1 2 4))
      (setq sle3 (getreal "Input the data of length3 ( m.) : "))
    )
  )
  (command "color" "7")
  (screentxt "Length3" (list 0 19))
  (screennum sle3 (list 6 19))
  (entmod (subst (cons 62 253) (assoc 62 dimsle3) dimsle3))
  (check_sceta3)
)
```

```
(defun input_slider_length4 ()
  (command "color" "cyan")
  (screentxt "Piston Le." (list 0 17.5))
  (entmod (subst (cons 62 4) (assoc 62 dimsle4) dimsle4))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
(initget (+ 1 2 4))
(setq sle4 (getreal "Input the data of piston length ( m. ) : "))
(command "color" "7")
(screenent "Piston Le." (list 0 17.5))
(screennum sle4 (list 6 17.5))
(entmod (subst (cons 62 253) (assoc 62 dimsle4) dimsle4))
```

)

```
(defun input_slider_pisdi4 ()
  (command "color" "cyan")
  (screenent "Piston Di." (list 0 16))
  (entmod (subst (cons 62 4) (assoc 62 dimsd1) dimsd1))
  (entmod (subst (cons 62 4) (assoc 62 dimsd2) dimsd2))
  (initget (+ 1 2 4))
  (setq sdi4 (getreal "Input the data of piston diameter ( m. ) : "))
  (command "color" "7")
  (screenent "Piston Di." (list 0 16))
  (screennum sdi4 (list 6 16))
  (entmod (subst (cons 62 253) (assoc 62 dimsd1) dimsd1))
  (entmod (subst (cons 62 253) (assoc 62 dimsd2) dimsd2)))
```

)

```
(defun input_slider_width ()
  (command "color" "cyan")
  (screenent "Width" (list 0 14.5))
  (entmod (subst (cons 62 4) (assoc 62 dimswi) dimswi))
  (initget (+ 1 2 4))
  (setq swi (getreal "Input the data of width ( m. ) : "))
  (command "color" "7")
  (screenent "Width" (list 0 14.5)))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
(screennum swi (list 6 14.5))
(entmod (subst (cons 62 253) (assoc 62 dimswi) dimswi))
```

```
)
(defun input_slider_thickness ()
  (command "color" "cyan")
  (screentxt "Thickness" (list 0 13))
  (entmod (subst (cons 62 4) (assoc 62 dimsth) dimsth))
  (initget (+ 1 2 4))
  (setq sth (getreal "Input the data of thick ( m.) : "))
  (command "color" "7")
  (screentxt "Thickness" (list 0 13))
  (screennum sth (list 6 13))
  (entmod (subst (cons 62 253) (assoc 62 dimsth) dimsth))
)
```

```
(defun input_slider_velocity ()
  (command "color" "cyan")
  (screentxt "Velocity" (list 0 11.5))
  (initget (+ 1 2 ))
  (setq somega2 (getreal "Input the data of velocity (m/s)
                        ;+/- for the director : "))
  (command "color" "7")
  (screentxt "Velocity" (list 0 11.5))
  (screennum somega2 (list 6 11.5))
)
```

```
(defun input_slider_acceleration ()
  (command "color" "cyan")
  (screentxt "Accel" (list 0 10))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
(initget 1 )
(setq salfa2 (getreal "Input the data of acceleration (m/s2) : "))
(command "color" "7")
(screenxt "Accel" (list 0 10))
(screennum salfa2 (list 6 10))
)
```

```
(defun input_slider_density ()
  (command "color" "cyan")
  (screenxt "Density" (list 0 8.5))
  (initget (+ 1 2 4))
  (setq sden (getreal "Input the data of density (kg/m3) : "))
  (command "color" "7")
  (screenxt "Density" (list 0 8.5))
  (screennum sden (list 6 8.5))
)
```

```
(defun input_slider_friction ()
  (command "color" "cyan")
  (screenxt "Force" (list 0 7))
  (entmod (subst (cons 77 4) (assoc 77 dimf1) dimf1))
  (entmod (subst (cons 62 4) (assoc 62 dimf2) dimf2))
  (initget (+ 1 2 4))
  (setq sfriction (getreal "Input the data of friction (N) : "))
  (command "color" "7")
  (screenxt "Force" (list 0 7))
  (screennum sfriction (list 6 7))
  (entmod (subst (cons 77 253) (assoc 77 dimf1) dimf1))
  (entmod (subst (cons 62 253) (assoc 62 dimf2) dimf2))
)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

(defun check_sceta3 ()
  (if (and (/= sle2 nil) (/= sle3 nil) (/= sceta2 nil) (/= startpt1 nil))
      (if (>= (- (expt sle3 2) (expt (- 0 (cadr startpt1)) 2))) 0)
      (progn
        (setq crossx1 (+ (car startpt1)
                        (sqrt (- (expt sle3 2) (expt (- 0 (cadr startpt1)) 2)))))
        (setq crossx2 (- (car startpt1)
                        (sqrt (- (expt sle3 2) (expt (- 0 (cadr startpt1)) 2)))))
        (setq sceta3 (* (angle startpt1 (list (max crossx1 crossx2) 0)) 57.295755))
        (setq sle1 (abs (+ (car startpt1)
                          (sqrt (- (expt sle3 2) (expt (- 0 (cadr startpt1)) 2)))))
        ))
      )
  (progn
    (setq sceta3 nil)
    (setq sle1 nil)
    )
  )
  (command "erase" "c" (list 19.5 11.3) (list 20 12) "")
  (command "color" "253")
  (if (= sle1 nil) (command "text" (list 19.5 11.3) "0.5" "0" "nil")
      (command "text" (list 19.5 11.3) "0.5" "0" (rtos sle1 2 2)))
  )
  (command "color" "7")
  )

```

```
(defun c:input_slider (/ checkle3 menupoint startpt1 starta
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

crossx1 crossx2 sceta31 sceta32

dimsle2 dimsle3 dimsle4 dimsceta2 dimsdil dimsdil2 dimswi dimsth

dimsf1 dimsf2

)

(setq starta 0)

(setq checkle3 nil)

(setq menupoint nil)

(command "erase" "all" "")

(command "zoom" (list -0.5 23) (list 35 2))

(command "color" "7")

(drawslider 15 15 18 19 27 15 2 1.2 0.5)

(command "rectangle" (list 14 9.5) (list 26 7.5))

(command "rectangle" (list 13 8.5) (list 25 6.5))

(command "line" (list 14 9.5) (list 13 8.5) "")

(command "line" (list 14 7.5) (list 13 6.5) "")

(command "line" (list 26 7.5) (list 25 6.5) "")

(command "line" (list 26 9.5) (list 25 8.5) "")

(command "color" "253")

(command "dimtxt" "0.5")

(command "dimtih" "on")

(command "dimtoh" "on")

(command "dimtod" "0")

(command "dimexe" "0.2")

(command "dimexo" "0.2")

(command "dimasz" "0.6")

(command "dimlinear" (list 15 13.7) (list 27 13.7) "t" "Length1" (list 15 12.5))

(if (= sle1 nil) (command "text" (list 19.5 11.3) "0.5" "0" "nil")

(command "text" (list 19.5 11.3) "0.5" "0" (rtos sle1 2 2))

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

)
(command "dimaligned" (list 15 15) (list 18 19) "t" "Length2" (list 15.6 19))
  (setq dimsle2 (entget (ssname (ssget "1") 0)))
(command "dimaligned" (list 18 19) (list 27 15) "t" "Length3" (list 22 19))
  (setq dimsle3 (entget (ssname (ssget "1") 0)))
(command "dimangular" (list 16 16) (list 17 14.5) "t" " " (list 18 15))
  (setq dimsceta2 (entget (ssname (ssget "1") 0)))
(command "dimaligned" (list 28 15.6) (list 28 14.4) "t" " " (list 29.5 16.5))
  (setq dimsdi1 (entget (ssname (ssget "1") 0)))
(command "text" (list 28 17.2) "0.5" "0" "Piston Diameter")
  (setq dimsdi2 (entget (ssname (ssget "1") 0)))
(command "dimaligned" (list 26 13.7) (list 28 13.7) "t" "Piston Length" (list 28.5 11))
  (setq dimsle4 (entget (ssname (ssget "1") 0)))
(command "leader" (list 28 15) (list 31 15) "" "" "n")
  (setq dimf1 (entget (ssname (ssget "1") 0)))
(command "text" (list 32 15) "0.5" "0" "F")
  (setq dimf2 (entget (ssname (ssget "1") 0)))
(command "dimaligned" (list 25 6.5) (list 26 7.5) "t" "Thickness" (list 27 7.5))
  (setq dimsth (entget (ssname (ssget "1") 0)))
(command "dimlinear" (list 13 8.5) (list 13 6.5) "t" "Width" (list 11.6 8.5))
  (setq dimswi (entget (ssname (ssget "1") 0)))
(command "color" "7")

```

```

(screenxt "Length2" (list 0 22))
(command "rectangle" (list -0.2 22.75) (list 4.5 21.5))
(if (/= sle2 nil)
  (screennum sle2 (list 6 22))
  (progn
    (command "color" "251")

```

```
(screentxt "No data" (list 6 22))
```

```
(command "color" "7")
```

```
)
```

```
)
```

```
(screentxt "Ceta" (list 0 20.5))
```

```
(command "rectangle" (list -0.2 21.25) (list 4.5 20))
```

```
(if (/= sceta2 nil)
```

```
(screennum sceta2 (list 6 20.5))
```

```
(progn
```

```
(command "color" "251")
```

```
(screentxt "No data" (list 6 20.5))
```

```
(command "color" "7")
```

```
)
```

```
)
```

```
(screentxt "Length3" (list 0 19))
```

```
(command "rectangle" (list -0.2 19.75) (list 4.5 18.5))
```

```
(if (/= sle3 nil)
```

```
(screennum sle3 (list 6 19))
```

```
(progn
```

```
(command "color" "251")
```

```
(screentxt "No data" (list 6 19))
```

```
(command "color" "7")
```

```
)
```

```
)
```

```
(screentxt "Piston Le." (list 0 17.5))
```

```
(command "rectangle" (list -0.2 18.25) (list 4.5 17))
```

```
(if (/= sle4 nil)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

(screennum sle4 (list 6 17.5))
(progn
  (command "color" "251")
  (screentxt "No data" (list 6 17.5))
  (command "color" "7")
)
)

(screennum sdi4 (list 6 16))
(command "rectangle" (list -0.2 16.75) (list 4.5 15.5))
(if (/= sdi4 nil)
  (screennum sdi4 (list 6 16))
  (progn
    (command "color" "251")
    (screentxt "No data" (list 6 16))
    (command "color" "7")
  )
)

(screennum swi (list 6 14.5))
(command "rectangle" (list -0.2 15.25) (list 4.5 14))
(if (/= swi nil)
  (screennum swi (list 6 14.5))
  (progn
    (command "color" "251")
    (screentxt "No data" (list 6 14.5))
    (command "color" "7")
  )
)

```

```
(screentxt "Thickness" (list 0 13))
```

```
(command "rectangle" (list -0.2 13.75) (list 4.5 12.5))
```

```
(if (/= sth nil)
```

```
  (screennum sth (list 6 13))
```

```
  (progn
```

```
    (command "color" "251")
```

```
    (screentxt "No data" (list 6 13))
```

```
    (command "color" "7")
```

```
  )
```

```
)
```

```
(screentxt "Velocity" (list 0 11.5))
```

```
(command "rectangle" (list -0.2 12.25) (list 4.5 11))
```

```
(if (/= somega2 nil)
```

```
  (screennum somega2 (list 6 11.5))
```

```
  (progn
```

```
    (command "color" "251")
```

```
    (screentxt "No data" (list 6 11.5))
```

```
    (command "color" "7")
```

```
  )
```

```
)
```

```
(screentxt "Accel" (list 0 10))
```

```
(command "rectangle" (list -0.2 10.75) (list 4.5 9.5))
```

```
(if (/= salfa2 nil)
```

```
  (screennum salfa2 (list 6 10))
```

```
  (progn
```

```
    (command "color" "251")
```

```
    (screentxt "No data" (list 6 10))
```

```
    (command "color" "7")
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

)
)

(screentxt "Density" (list 0 8.5))
(command "rectangle" (list -0.2 9.25) (list 4.5 8))
(if (/= sden nil)
  (screennum sden (list 6 8.5))
  (progn
    (command "color" "251")
    (screentxt "No data" (list 6 8.5))
    (command "color" "7")
  )
)
...
(screentxt "Force" (list 0 7))
(command "rectangle" (list -0.2 7.75) (list 4.5 6.5))
(if (/= sfriction nil)
  (screennum sfriction (list 6 7))
  (progn
    (command "color" "251")
    (screentxt "No data" (list 6 7))
    (command "color" "7")
  )
)
)

(command "color" "150")
(screentxt "DONE" (list 31.3 3.15))
(command "rectangle" (list 30 4) (list 34.5 2.75))
(command "color" "7")

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

(if (= sle2 nil) (input_slider_length2) )
(if (= sceta2 nil) (input_slider_ceta) )
(if (= sle3 nil) (input_slider_length3) )
(if (= sle4 nil) (input_slider_length4) )
(if (= sdi4 nil) (input_slider_pisdi4) )
(if (= swi nil) (input_slider_width) )
(if (= sth nil) (input_slider_thickness) )
(if (= somega2 nil) (input_slider_velocity) )
(if (= salfa2 nil) (input_slider_acceleration) )
(if (= sden nil) (input_slider_density) )
(if (= sfriction nil) (input_slider_friction) )

...
(while (= checkle3 nil)
  (setq startpt1 (polar (list 0 0) (* sceta2 0.0174533) sle2))

  (if (and (>= (car menupoint) 30)
        (<= (cadr menupoint) 4)
        (<= (car menupoint) 34.5)
        (>= (cadr menupoint) 2.75) )
    (progn
      (command "erase" (list 31.3 3.15) (list 31.3 3.15) "")
      (command "color" "150")
      (screentxt "DONE" (list 31.3 3.15))
      (command "color" "7")
    )
  )
)
(setq menupoint (getpoint "Edit data : "))
(cond
  ((and (>= (car menupoint) -0.2)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(\leq (cadr menupoint) 22.75)

(\leq (car menupoint) 4.5)

(\geq (cadr menupoint) 21.5)) (input_slider_length2))

((and (\geq (car menupoint) -0.2)

(\leq (cadr menupoint) 21.25)

(\leq (car menupoint) 4.5)

(\geq (cadr menupoint) 20)) (input_slider_ceta))

((and (\geq (car menupoint) -0.2)

(\leq (cadr menupoint) 19.75)

(\leq (car menupoint) 4.5)

(\geq (cadr menupoint) 18.5)) (input_slider_length3))

((and (\geq (car menupoint) -0.2)

(\leq (cadr menupoint) 18.25)

(\leq (car menupoint) 4.5)

(\geq (cadr menupoint) 17)) (input_slider_length4))

((and (\geq (car menupoint) -0.2)

(\leq (cadr menupoint) 16.75)

(\leq (car menupoint) 4.5)

(\geq (cadr menupoint) 15.5)) (input_slider_pisdi4))

((and (\geq (car menupoint) -0.2)

(\leq (cadr menupoint) 15.25)

(\leq (car menupoint) 4.5)

(\geq (cadr menupoint) 14)) (input_slider_width))

((and (\geq (car menupoint) -0.2)

```
(<= (cadr menupoint) 13.75)
(<= (car menupoint) 4.5)
(>= (cadr menupoint) 12.5) ) (input_slider_thickness) )
```

```
((and (>= (car menupoint) -0.2)
 (<= (cadr menupoint) 12.25)
 (<= (car menupoint) 4.5)
 (>= (cadr menupoint) 11) ) (input_slider_velocity) )
```

```
((and (>= (car menupoint) -0.2)
 (<= (cadr menupoint) 10.75)
 (<= (car menupoint) 4.5)
 (>= (cadr menupoint) 9.5) ) (input_slider_acceleration) )
```

```
((and (>= (car menupoint) -0.5)
 (<= (cadr menupoint) 9.25)
 (<= (car menupoint) 4.5)
 (>= (cadr menupoint) 8) ) (input_slider_density) )
```

```
((and (>= (car menupoint) -0.2)
 (<= (cadr menupoint) 7.75)
 (<= (car menupoint) 4.5)
 (>= (cadr menupoint) 6.5) ) (input_slider_friction) )
```

```
((and (>= (car menupoint) 30) ;-----
 (<= (cadr menupoint) 4) ; DONE
 (<= (car menupoint) 34.5) ;-----
 (>= (cadr menupoint) 2.75) )
```

```
(command "erase" "c" (list 31.3 3.15) (list 31.3 3.15) "")
```

```
(command "color" "cyan")
```

```
(screentxt "DONE" (list 31.3 3.15))
```

```
(if (>= sle3 (abs (cadr startpt1)))
```

```
  (progn
```

```
    (command "erase" "c" (list 5 2.5) (list 13 2.5) "")
```

```
    (command "color" "green" )
```

```
    (screentxt "The data is correct." (list 13 2.5))
```

```
    (setq checkle3 1)
```

```
    (command "color" "7" )
```

```
  )
```

```
  (progn
```

```
    (command "erase" "c" (list 5 2.5) (list 13 2.5) "")
```

```
    (command "color" "red" )
```

```
    (screentxt "Data error, choose length2 or length3 to edit."
```

```
      (list 5 2.5))
```

```
    (command "color" "7" )
```

```
  )
```

```
)
```

```
)
```

```
)
```

```
)
```

```
(command "delay" "1")
```

```
)
```

```
(defun sbase (xx0 yy0 lle rle yle thickz )
```

```
  (command "mline" "j" "z" "s" thickz "st" "halfbar"
```

```
    (list xx0 yy0) (list xx0 (- yy0 (/ yle 2))) "")
```

```
  (command "color" "252")
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
(command "solid" (list (- xx0 lle) (- yy0 (/ yle 2)))
  (list (- xx0 lle) (- yy0 (/ yle 2) (/ yle 3)))
  (list (+ xx0 rle) (- yy0 (/ yle 2)))
  (list (+ xx0 rle) (- yy0 (/ yle 2) (/ yle 3)))
  "" )
```

```
(command "color" "254")
```

```
(command "line" (list (- xx0 lle) (- yy0 (/ yle 2)))
  (list (+ xx0 rle) (- yy0 (/ yle 2)))
  "" )
```

```
(command "color" "7")
```

```
)
```

```
(defun c:move_slider(/ circlemove stepmove counter countercheck
  button button_check end_check screenle
  numercheck numercheck_start numercheck_end
  numercheck_ceta pt2s
  cetamove2 cetamove3 cetamove4
  chpt1 chpt2
  mvpnt21 mvpnt31 mvpnt22 mvpnt32
  crossx1 crossx2 )
```

```
(setq move_line 1)
```

```
(setq end_check 1)
```

```
(while (= end_check 1)
```

```
  (while (>= sceta2 360) (setq sceta2 (- sceta2 360)))
```

```
  (while (< sceta2 0) (setq sceta2 (+ sceta2 360)))
```

```
(setq cetamove2 sceta2)
```

```
(setq cetamove3 sceta3)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

(if (= displaymove nil) (setq displaymove 10))
(if (> somega2 0) (setq stepmove 0.1) (setq stepmove -0.1))
(setq pt0 (list 0 0))
(setq pt1 (polar pt0 (* cetamove2 0.0174533) sle2))
(setq pt2 (polar pt1 (* cetamove3 0.0174533) sle3))

(command "erase" "all" "")

(if (or (and (<= sle2 sle3) (< (car pt1) (car pt2)))
      (and (> sle2 sle3) (> (car pt1) 0)))
    (command "zoom" (list (- 0 sle2 sle4) (* 2 (min sle2 sle3)))
              (list (+ sle2 sle3 sle4) (* -2 (min sle2 sle3))))
    (command "zoom" (list (- 0 sle2 sle3 sle4) (* 2 (min sle2 sle3)))
              (list (+ sle2 sle4) (* -2 (min sle2 sle3))))
    )

(command "layer" "set" "0" "")
(if (or (and (<= sle2 sle3) (< (car pt1) (car pt2)))
      (and (> sle2 sle3) (> (car pt1) 0)))
    (sbase (car pt0) (cadr pt0)
            (+ sle2 (/ sle4 2)) (+ sle2 sle3 (/ sle4 2))
            sdi4 swi )
    (sbase (car pt0) (cadr pt0)
            (+ sle2 sle3 (/ sle4 2)) (+ sle2 (/ sle4 2))
            sdi4 swi )
    )

(command "layer" "make" "DYNAMIC" "")
(drawbar pt0 pt1 swi)
(drawbar pt1 pt2 swi)

```

```
(command "rectangle" (list (- (car pt2) (/ sle4 2)) (+ (cadr pt2) (/ sdi4 2)))
  (list (+ (car pt2) (/ sle4 2)) (- (cadr pt2) (/ sdi4 2))))
)
```

```
(setq chpt2 1)
```

```
(setq countercheck (abs (/ displaymove stepmove)))
```

```
(setq counter 0)
```

```
(setq circlemove 0)
```

```
(setq mvpt21 (list (/ (+ (car pt0) (car pt1)) 2) (/ (+ (cadr pt0) (cadr pt1)) 2) ))
```

```
(setq mvpt31 (list (/ (+ (car pt1) (car pt2)) 2) (/ (+ (cadr pt1) (cadr pt2)) 2) ))
```

```
(command "delay" "1")
```

```
(while (and (< (abs circlemove) 360) (/= chpt2 nil))
```

```
  (setq counter (+ counter 1))
```

```
  (setq circlemove (+ circlemove stepmove))
```

```
  (setq cetamove2 (+ cetamove2 stepmove))
```

```
  (while (>= cetamove2 360) (setq cetamove2 (- cetamove2 360)))
```

```
  (while (< cetamove2 0) (setq cetamove2 (+ cetamove2 360)))
```

```
  (setq chpt1 (polar pt0 (* cetamove2 0.0174533) sle2))
```

```
(if (>= (- (expt sle3 2) (expt (- (cadr pt2) (cadr chpt1)) 2)) 0)
```

```
(progn
```

```
  (setq crossx1 (+ (car chpt1)
```

```
    (sqrt (- (expt sle3 2) (expt (- (cadr pt2) (cadr chpt1)) 2)))))
```

```
  (setq crossx2 (- (car chpt1)
```

```
    (sqrt (- (expt sle3 2) (expt (- (cadr pt2) (cadr chpt1)) 2)))))
```

```
  (if (< (abs (- crossx1 (car pt2))) (abs (- crossx2 (car pt2))))
```

```
    (setq pt2 (list crossx1 (cadr pt2)))
```

```
    (setq pt2 (list crossx2 (cadr pt2)))
```

```
)
) (setq chpt2 nil)
)

(if (or (= chpt2 nil) (= counter countercheck) (>= (abs circlemove) 360))
```

```
(progn
```

```
(if (= chpt2 nil)
```

```
(progn
```

```
(setq numercheck nil)
```

```
(setq numercheck_start (- cetamove2 stepmove))
```

```
(setq numercheck_end cetamove2)
```

```
(while (= numercheck nil)
```

```
(setq pt2s 1)
```

```
(setq numercheck_ceta (/ (+ numercheck_start numercheck_end) 2))
```

```
(setq pt1 (polar pt0 (* numercheck_ceta 0.0174533) sle2))
```

```
(if (>= (- (expt sle3 2) (expt (- (cadr pt2) (cadr pt1)) 2)) 0)
```

```
(progn
```

```
(setq crossx1 (+ (car pt1)
```

```
(sqrt (- (expt sle3 2) (expt (- (cadr pt2) (cadr pt1)) 2)))))
```

```
(setq crossx2 (- (car pt1)
```

```
(sqrt (- (expt sle3 2) (expt (- (cadr pt2) (cadr pt1)) 2)))))
```

```
(if (< (abs (- crossx1 (car pt2))) (abs (- crossx2 (car pt2))))
```

```
(setq pt2 (list crossx1 (cadr pt2)))
```

```
(setq pt2 (list crossx2 (cadr pt2)))
```

```
)
```

```
) (setq pt2s nil)
```

```
)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```
(setq mvpnt22 (list (/ (+ (car pt0) (car pt1)) 2) (/ (+ (cadr pt0) (cadr pt1)) 2) ))
```

```
(setq mvpnt32 (list (/ (+ (car pt1) (car pt2)) 2) (/ (+ (cadr pt1) (cadr pt2)) 2) ))
```

```
(command "color" "blue")
```

```
(command "line" mvpnt21 mvpnt22 "" )
```

```
(command "color" "green")
```

```
(command "line" mvpnt31 mvpnt32 "" )
```

```
(setq mvpnt21 mvpnt22)
```

```
(setq mvpnt31 mvpnt32)
```

```
)
```

```
)
```

```
(command "color" "7")
```

```
(command "layer" "set" "DYNAMIC" "")
```

```
)
```

```
)
```

```
)
```

```
(if (or (and (<= sle2 sle3) (< (car pt1) (car pt2)))
```

```
(and (> sle2 sle3) (> (car pt1) 0)))
```

```
(setq screenle (+ sle2 sle3 sle4)) (setq screenle (+ sle2 sle4)) )
```

```
(command "rectangle" (list (+ screenle (* -0.8 (min sle2 sle3)))
```

```
(* (min sle2 sle3) -1.5) )
```

```
(list screenle
```

```
(* (min sle2 sle3) -1.7) ) )
```

```
(command "rectangle" (list (+ screenle (* -0.8 (min sle2 sle3)))
```

```
(* (min sle2 sle3) -1.8) )
```

```
(list screenle
```

```
(* (min sle2 sle3) -2) ) )
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

(command "text" "j" "c"
  (list (+ screenle (* -0.4 (min sle2 sle3)))
    (* (min sle2 sle3) -1.66) )
  (* (min sle2 sle3) 0.1)
  "0" "REPLAY" )

(command "text" "j" "c"
  (list (+ screenle (* -0.4 (min sle2 sle3)))
    (* (min sle2 sle3) -1.96) )
  (* (min sle2 sle3) 0.1)
  "0" "DONE" )

(setq button_check nil)
(while (= button_check nil)
  (setq button (getpoint "Replay or Done : "))
  (cond
    ( (and (>= (car button) (+ screenle (* -0.8 (min sle2 sle3))))
      (<= (cadr button) (* (min sle2 sle3) -1.5) )
      (<= (car button) screenle )
      (>= (cadr button) (* (min sle2 sle3) -1.7) )
    )
    (command "color" "cyan")
    (command "rectangle" (list (+ screenle (* -0.8 (min sle2 sle3)))
      (* (min sle2 sle3) -1.5) )
      (list screenle
        (* (min sle2 sle3) -1.7) ) )
    (command "text" "j" "c"
      (list (+ screenle (* -0.4 (min sle2 sle3)))
        (* (min sle2 sle3) -1.66) )
      (* (min sle2 sle3) 0.1)
      "0" "REPLAY" )
  )
)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```
(setq rad2 (/ (* ceta2 pi) 180))
(setq rad3 (/ (* ceta3 pi) 180))
(setq rad4 (/ (* ceta4 pi) 180))
(setq omega3 (* -1 (/ le2 le3) omega2 (/ (sin (- rad4 rad2)) (sin (- rad4 rad3)))))
(setq omega4 (* (/ le2 le4) omega2 (/ (sin (- rad3 rad2)) (sin (- rad3 rad4)))))
```

```
(setq alfa3 (/ (- (+ (* le2 omega2 omega2 (cos (- rad4 rad2 )))
(* le3 omega3 omega3 (cos (- rad4 rad3 )))
(* le2 alfa2 (sin (- rad4 rad2 )))
(* le4 omega4 omega4 )
(* le3 (sin (- rad4 rad3))))))
)
```

```
(setq alfa4 (/ (- (+ (* le2 alfa2 (sin (- rad3 rad2 )))
(* le4 omega4 omega4 (cos (- rad3 rad4 )))
(* le2 omega2 omega2 (cos (- rad3 rad2)))
(* le3 omega3 omega3 )
(* le4 (sin (- rad3 rad4))))))
)
```

```
(setq velo2 (* (abs omega2) le2))
(if (> omega2 0) (setq cetavelo2 (+ rad2 (/ pi 2))))
(if (< omega2 0) (setq cetavelo2 (- rad2 (/ pi 2))))
(setq velo3 (* (abs omega3) le3))
(if (> omega3 0) (setq cetavelo3 (+ rad3 (/ pi 2))))
(if (< omega3 0) (setq cetavelo3 (- rad3 (/ pi 2))))
(setq velo4 (* (abs omega4) le4))
(if (> omega4 0) (setq cetavelo4 (+ rad4 (/ pi 2))))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

(if (< omega4 0) (setq cetavelo4 (- rad4 (/ pi 2))))

(setq accen2 (/ (* velo2 velo2) le2))
(setq cetaaccen2 (+ rad2 pi))
(setq accet2 (* (abs alfa2) le2))
(if (> alfa2 0)
  (setq cetaaccet2 (+ rad2 (/ pi 2)))
  (setq cetaaccet2 (- rad2 (/ pi 2)))
)
(setq accen3 (/ (* velo3 velo3) le3))
(setq cetaaccen3 (+ rad3 pi))
(setq accet3 (* (abs alfa3) le3))
(if (> alfa3 0)
  (setq cetaaccet3 (+ rad3 (/ pi 2)))
  (setq cetaaccet3 (- rad3 (/ pi 2)))
)
(setq cetaaccet3 1.86)
(setq accen4 (/ (* velo4 velo4) le4))
(setq cetaaccen4 (+ rad4 pi))
(setq accet4 (* (abs alfa4) le4))
(if (> alfa4 0)
  (setq cetaaccet4 (+ rad4 (/ pi 2)))
  (setq cetaaccet4 (- rad4 (/ pi 2)))
)
)
)
(defun v_vector(/ xv1 yv1 xxv1 xxv2 yyv1 yyv2 Xtext Ytext
  temp done)
  (calculate)
  (setq xv1 0)
  (setq yv1 0)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
(setq xxv1 (+ xv1 (* velo2 (cos cetavelo2))))
(setq yyv1 (+ yv1 (* velo2 (sin cetavelo2))))
(setq xxv2 (+ xxv1 (* velo3 (cos cetavelo3))))
(setq yyv2 (+ yyv1 (* velo3 (sin cetavelo3))))
```

```
(command "erase" "all" "")
(command "point" (list 0 0))
(setq xmax (max xv1 xxv1 xxv2))
(setq xmin (min xv1 xxv1 xxv2))
(setq ymax (max yv1 yyv1 yyv2))
(setq ymin (min yv1 yyv1 yyv2))
(setq Htext (/ (- ymax ymin) 15))
(command "dimasz" (* Htext 1))

...

(command "color" "red")
(setq value (rtos velo2 2 2))
(setq V2 (strcat "V2" "=" value))
(command "leader" (list xxv1 yyv1) (list xv1 yv1) "" "" "n")
(command "text" (list (+ xmax (* 3 htext)) (+ ymin (* 6 htext))) (* 1 htext) "0" V2)
```

```
(command "color" "blue")
(setq value (rtos velo3 2 2))
(setq V3 (strcat "V3" "=" value))
(command "leader" (list xxv2 yyv2) (list xxv1 yyv1) "" "" "n")
(command "text" (list (+ xmax (* 3 htext)) (+ ymin (* 4 htext))) (* 1 htext) "0" V3)
```

```
(command "color" "yellow")
(setq value (rtos velo4 2 2))
(setq V4 (strcat "V4" "=" value))
(command "leader" (list xxv2 yyv2) (list xv1 yv1) "" "" "n")
```

```
(command "text" (list (+ xmax (* 3 htext)) (+ ymin (* 2 htext))) (* 1 htext) "0" V4)
```

```
(command "zoom" "e")
```

```
(command "zoom" "0.5x")
```

```
(command "color" "7")
```

```
(command "text" (list (+ xmax (* 3 htext)) ymin) (* 1 htext) "0" "DONE")
```

```
(command "rectangle" (list (+ xmax (* htext 2.5)) (+ ymin (* 1.3 htext)))
```

```
(list (+ xmax (* 7.15 htext)) (- ymin (* 0.25 htext)))
```

```
)
```

```
(setq temp 0)
```

```
(while (= temp 0)
```

```
(setq done (getpoint "done"))
```

```
(if (and (>= (car done) (+ xmax (* htext 2.5)))
```

```
... (<= (car done) (+ xmax (* htext 7.15)))
```

```
(>= (cadr done) (- ymin (* 0.25 htext)))
```

```
(<= (cadr done) (+ ymin (* htext 1.3)))
```

```
)
```

```
(progn
```

```
(setq temp 1)
```

```
)
```

```
)
```

```
)
```

```
)
```

```
(defun a_vector(/ xal ya1 xan2 xat2 xan3 xat3 xan4 yan2 yan3 yan4 yat2 yat3
```

```
xann2 xann3 xann4 xatt2 xatt3 yann2 yann3 yann4 yatt2 yatt3
```

```
a b base xmin xmax ymin ymax check done)
```

```
(calculate)
```

```
(setq xal 0)
```

```
(setq ya1 0)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

(setq xan2 (+ xa1 (* accen2 (cos cetaaccen2))))
(setq yan2 (+ ya1 (* accen2 (sin cetaaccen2))))
(setq xat2 (+ xan2 (* accet2 (cos cetaaccet2))))
(setq yat2 (+ yan2 (* accet2 (sin cetaaccet2))))
(setq xan3 (+ xat2 (* accen3 (cos cetaaccen3))))
(setq yan3 (+ yat2 (* accen3 (sin cetaaccen3))))
(setq xat3 (+ xan3 (* accet3 (cos cetaaccet3))))
(setq yat3 (+ yan3 (* accet3 (sin cetaaccet3))))
(setq xan4 (+ xa1 (* accen4 (cos cetaaccen4))))
(setq yan4 (+ ya1 (* accen4 (sin cetaaccen4))))

(setq acceg2 (/ (distance (list xa1 ya1) (list xat2 yat2)) 2))
(setq acceg4 (/ (distance (list xa1 ya1) (list xat3 yat3)) 2))
(setq a (/ (+ xat2 xat3) 2))
(setq b (/ (+ yat2 yat3) 2))
(setq acceg3 (distance (list xa1 ya1) (list a b)))

(setq xmin (min xan2 xat2 xan3 xat3 xan4 xa1))
(setq xmax (max xan2 xat2 xan3 xat3 xan4 xa1))
(setq ymin (min yan2 yat2 yan3 yat3 yan4 ya1))
(setq ymax (max yan2 yat2 yan3 yat3 yan4 ya1))
(setq htext (/ (- ymax ymin) 20))
(command "dimasz" (* 1 htext))

(setq a (- yat2 ya1))
(setq b (- xat2 xa1))
(if (and (> a 0) (> b 0))
    (setq cetaacce2 (atan (/ a b)))
)
(if (and (> a 0) (< b 0))

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

(setq cetaacceg2 (- pi (abs (atan (/ a b))))))
)
(if (and (< a 0) (< b 0))
  (setq cetaacceg2 (+ pi (atan (/ a b))))
)
(if (and (< a 0) (> b 0))
  (setq cetaacceg2 (- (* 2 pi) (abs (atan (/ a b))))))
)
(setq a (- yat3 ya1))
(setq b (- xat3 xa1))
(if (and (> a 0) (> b 0))
  (setq cetaacceg4 (atan (/ a b)))
)
(if (and (> a 0) (< b 0))
  (setq cetaacceg4 (- pi (abs (atan (/ a b))))))
)
(if (and (< a 0) (< b 0))
  (setq cetaacceg4 (+ pi (atan (/ a b))))
)
(if (and (< a 0) (> b 0))
  (setq cetaacceg4 (- (* 2 pi) (abs (atan (/ a b))))))
)
(setq a (- (+ yat2 yat3) (* 2 ya1)))
(setq b (- (+ xat2 xat3) (* 2 xa1)))
(if (and (> a 0) (> b 0))
  (setq cetaacceg3 (atan (/ a b)))
)
(if (and (> a 0) (< b 0))
  (setq cetaacceg3 (- pi (abs (atan (/ a b))))))
)

```

```

(if (and (< a 0) (< b 0))
  (setq cetaacce3 (+ pi (atan (/ a b))))
)

(if (and (< a 0) (> b 0))
  (setq cetaacce3 (- (* 2 pi) (abs (atan (/ a b)))))
)

(command "erase" "all" "")
(command "erase" "all" "")
(command "dimasz" "2.5")
(command "color" "blue")
(setq value (rtos accen2 2 2))
(setq An2 (strcat "An2" "=" value))
(command "leader" (list xan2 yan2) (list xal ya1) "" "" "n") ;ทิศทางของvector an2
(command "text" (list (+ xmax (* 3 htext)) (+ ymin (* 12 htext))) (* 1 htext) "0" An2)
(command "color" "yellow")
(if (/= alfa2 0)
  (progn
    (setq value (rtos accet2 2 2))
    (setq At2 (strcat "At2" "=" value))
    (command "leader" (list xat2 yat2) (list xan2 yan2) "" "" "n") ;ทิศทางของvector at2
    (command "text" (list (+ xmax (* 3 htext)) (+ ymin (* 10 htext))) (* 1 htext) "0" At2)
  )
)

(command "color" "green")
(setq value (rtos accen3 2 2))
(setq An3 (strcat "An3" "=" value))
(command "leader" (list xan3 yan3) (list xat2 yat2) "" "" "n") ;ทิศทางของvector an3
(command "text" (list (+ xmax (* 3 htext)) (+ ymin (* 8 htext))) (* 1 htext) "0" An3)
(command "color" "cyan")
(setq value (rtos accet3 2 2))

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

(setq At3 (strcat "At3" "=" value))
(command "leader" (list xat3 yat3) (list xan3 yan3) "" "" "n") ;ทิศทางของvector at3
(command "text" (list (+ xmax (* 3 htext)) (+ ymin (* 6 htext))) (* 1 htext) "0" At3)
(command "color" "red")
(setq value (rtos accen4 2 2))
(setq An4 (strcat "An4" "=" value))
(command "leader" (list xan4 yan4) (list xa1 ya1) "" "" "n") ;ทิศทางของvector an4
(command "text" (list (+ xmax (* 3 htext)) (+ ymin (* 4 htext))) (* 1 htext) "0" An4)
(command "color" "magenta")
(setq value (rtos accet4 2 2))
(setq At4 (strcat "At4" "=" value))
(command "leader" (list xat3 yat3) (list xan4 yan4) "" "" "n") ;ทิศทางของvector at4
(command "text" (list (+ xmax (* 3 htext)) (+ ymin (* 2 htext))) (* 1 htext) "0" At4)
(command "color" "7")
(command "text" (list (+ xmax (* 3 htext)) ymin) (* 1 htext) "0" "DONE")
(command "rectangle" (list (+ xmax (* htext 2.5)) (+ ymin (* 1.3 htext)))
(list (+ xmax (* 7.15 htext)) (- ymin (* 0.25 htext)))
)
(command "zoom" "e")
(command "zoom" "0.8x")
(setq check 0)
(while (= check 0)
  (setq done (getpoint "back to menu"))
  (if (and (>= (car done) (+ xmax (* htext 2.5)))
        (<= (car done) (+ xmax (* htext 7.15)))
        (>= (cadr done) (- ymin (* 0.25 htext)))
        (<= (cadr done) (+ ymin (* htext 1.3))))
    )
  (setq check 1)
)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

)
)
(defun f_vector(/ a11 a14 a22 a25 a31 a32 a33 a34 a35 a44 a46 a55 a57
                a64 a65 a66 a67 a76 a78 a87 a89 a96 a97 a98 a99
                m31 aa31 aa32 aa33 aa34 aa35 aa36 aa37 aa38 aa39
                m32 ab31 ab32 ab33 ab34 ab35 ab36 ab37 ab38 ab39
                m64 aa61 aa62 aa63 aa64 aa65 aa66 aa67 aa68 aa69
                m65 ab61 ab62 ab63 ab64 ab65 ab66 ab67 ab68 ab69
                m76 aa71 aa72 aa73 aa74 aa75 aa76 aa77 aa78 aa79
                m78 ab71 ab72 ab73 ab74 ab75 ab76 ab77 ab78 ab79
                m96 aa91 aa92 aa93 aa94 aa95 aa96 aa97 aa98 aa99
                m98 ab91 ab92 ab93 ab94 ab95 ab96 ab97 ab98 ab99
                m97 ac91 ac92 ac93 ac94 ac95 ac96 ac97 ac98 ac99
                mass2 mass3 mass4 s1 s2 s3 s4 s5 s6 s7 s8 s9
                f14y f14x f34x f34y f23x f23y ts f12y f12x temp)

```

```
(setq a11 -1)
```

```
(setq a14 1)
```

```
(setq a22 -1)
```

```
(setq a25 1)
```

```
(setq a31 (* -1 (* (/ le2 2) (sin rad2))))
```

```
(setq a32 (* (/ le2 2) (cos rad2)))
```

```
(setq a33 -1)
```

```
(setq a34 (* -1 (/ le2 2) (sin rad2)))
```

```
(setq a35 (* (/ le2 2) (cos rad2)))
```

```
(setq a44 -1)
```

```
(setq a46 1)
```

```
(setq a55 -1)
```

```
(setq a57 1)
```

```
(setq a64 (* -1 (* (/ le3 2) (sin rad3))))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$(\text{setq } a65 (* (/ le3 2) (\cos \text{rad}3)))$$

$$(\text{setq } a66 (* -1 (/ le3 2) (\sin \text{rad}3)))$$

$$(\text{setq } a67 (* (/ le3 2) (\cos \text{rad}3)))$$

$$(\text{setq } a76 -1)$$

$$(\text{setq } a78 -1)$$

$$(\text{setq } a87 -1)$$

$$(\text{setq } a89 -1)$$

$$(\text{setq } a96 (* (/ le4 2) (\sin \text{rad}4)))$$

$$(\text{setq } a97 (* -1 (/ le4 2) (\cos \text{rad}4)))$$

$$(\text{setq } a98 (* -1 (* (/ le4 2) (\sin \text{rad}4))))$$

$$(\text{setq } a99 (* (/ le4 2) (\cos \text{rad}4)))$$

$$(\text{setq } m31 (/ a31 a11))$$

$$(\text{setq } aa31 (- a31 (* m31 a11)))$$

$$(\text{setq } aa32 (- a32 (* m31 0)))$$

$$(\text{setq } aa33 (- a33 (* m31 0)))$$

$$(\text{setq } aa34 (- a34 (* m31 a14)))$$

$$(\text{setq } aa35 (- a35 (* m31 0)))$$

$$(\text{setq } aa36 (- 0 (* m31 0)))$$

$$(\text{setq } aa37 (- 0 (* m31 0)))$$

$$(\text{setq } aa38 (- 0 (* m31 0)))$$

$$(\text{setq } aa39 (- 0 (* m31 0)))$$

$$(\text{setq } m32 (/ aa32 a22))$$

$$(\text{setq } ab31 (- aa31 (* m32 0)))$$

$$(\text{setq } ab32 (- aa32 (* m32 a22)))$$

$$(\text{setq } ab33 (- aa33 (* m32 0)))$$

$$(\text{setq } ab34 (- aa34 (* m32 0)))$$

$$(\text{setq } ab35 (- aa35 (* m32 a25)))$$

$$(\text{setq } ab36 (- aa36 (* m32 0)))$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(setq ab37 (- aa37 (* m32 0)))

(setq ab38 (- aa38 (* m32 0)))

(setq ab39 (- aa39 (* m32 0)))

(setq m64 (/ a64 a44))

(setq aa61 (- 0 (* m64 0)))

(setq aa62 (- 0 (* m64 0)))

(setq aa63 (- 0 (* m64 0)))

(setq aa64 (- a64 (* m64 a44)))

(setq aa65 (- a65 (* m64 0)))

(setq aa66 (- a66 (* m64 a46)))

(setq aa67 (- a67 (* m64 0)))

(setq aa68 (- 0 (* m64 0)))

(setq aa69 (- 0 (* m64 0)))

(setq m65 (/ aa65 a55))

(setq ab61 (- aa61 (* m65 0)))

(setq ab62 (- aa62 (* m65 0)))

(setq ab63 (- aa63 (* m65 0)))

(setq ab64 (- aa64 (* m65 0)))

(setq ab65 (- aa65 (* m65 a55)))

(setq ab66 (- aa66 (* m65 0)))

(setq ab67 (- aa67 (* m65 a57)))

(setq ab68 (- aa68 (* m65 0)))

(setq ab69 (- aa69 (* m65 0)))

(setq m76 (/ a76 ab66))

(setq aa71 (- 0 (* m76 ab61)))

(setq aa72 (- 0 (* m76 ab62)))

(setq aa73 (- 0 (* m76 ab63)))

(setq aa74 (- 0 (* m76 ab64)))

(setq aa75 (- 0 (* m76 ab65)))

(setq aa76 (- a76 (* m76 ab66)))

(setq aa77 (- 0 (* m76 ab67)))

(setq aa78 (- a78 (* m76 ab68)))

(setq aa79 (- 0 (* m76 ab69)))

(setq m78 (/ aa77 a87))

(setq ab71 (- aa71 (* m78 0)))

(setq ab72 (- aa72 (* m78 0)))

(setq ab73 (- aa73 (* m78 0)))

(setq ab74 (- aa74 (* m78 0)))

(setq ab75 (- aa75 (* m78 0)))

(setq ab76 (- aa76 (* m78 0)))

(setq ab77 (- aa77 (* m78 a87)))

(setq ab78 (- aa78 (* m78 0)))

(setq ab79 (- aa79 (* m78 a89)))

(setq m96 (/ a96 ab66))

(setq aa91 (- 0 (* m96 ab61)))

(setq aa92 (- 0 (* m96 ab62)))

(setq aa93 (- 0 (* m96 ab63)))

(setq aa94 (- 0 (* m96 ab64)))

(setq aa95 (- 0 (* m96 ab65)))

(setq aa96 (- a96 (* m96 ab66)))

(setq aa97 (- a97 (* m96 ab67)))

(setq aa98 (- a98 (* m96 ab68)))

(setq aa99 (- a99 (* m96 ab69)))



(setq m98 (/ aa97 a87))

(setq ab91 (- aa91 (* m98 0)))

(setq ab92 (- aa92 (* m98 0)))

(setq ab93 (- aa93 (* m98 0)))

(setq ab94 (- aa94 (* m98 0)))

(setq ab95 (- aa95 (* m98 0)))

(setq ab96 (- aa96 (* m98 0)))

(setq ab97 (- aa97 (* m98 a87)))

(setq ab98 (- aa98 (* m98 0)))

(setq ab99 (- aa99 (* m98 a89)))

(setq m97 (/ ab98 aa78))

(setq ac91 (- ab91 (* m97 ab71)))

(setq ac92 (- ab92 (* m97 ab72)))

(setq ac93 (- ab93 (* m97 ab73)))

(setq ac94 (- ab94 (* m97 ab74)))

(setq ac95 (- ab95 (* m97 ab75)))

(setq ac96 (- ab96 (* m97 ab76)))

(setq ac97 (- ab97 (* m97 ab77)))

(setq ac98 (- ab98 (* m97 ab78)))

(setq ac99 (- ab99 (* m97 ab79)))

(setq mass2 (* den wi th le2))

(setq mass3 (* den wi th le3))

(setq mass4 (* den wi th le4))

(setq inertia2 (/ (* (+ (* le2 le2) (* wi wi)) mass2) 12))

(setq inertia3 (/ (* (+ (* le3 le3) (* wi wi)) mass3) 12))

(setq inertia4 (/ (* (+ (* le4 le4) (* wi wi)) mass4) 12))

(setq s1 (* mass2 acceg2 (cos (+ cetaacceg2 pi))))

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(setq s2 (* mass2 acceg2 (sin (+ cetaacceg2 pi))))

(setq s3 (* inertia2 (* -1 alfa2)))

(setq s4 (* mass3 acceg3 (cos (+ cetaacceg3 pi))))

(setq s5 (* mass3 acceg3 (sin (+ cetaacceg3 pi))))

(setq s6 (* inertia3 (* -1 alfa3)))

(setq s7 (* mass4 acceg4 (cos (+ cetaacceg4 pi))))

(setq s8 (* mass4 acceg4 (sin (+ cetaacceg4 pi))))

(setq s9 (* inertia4 (* -1 alfa4)))

(setq forceinertia2 (* mass2 acceg2))

(setq forceinertia3 (* mass3 acceg3))

(setq forceinertia4 (* mass4 acceg4))

(setq sa3 (- s3 (* m31 s1)))

(setq sb3 (- sa3 (* m32 s2)))

(setq sa6 (- s6 (* m64 s4)))

(setq sb6 (- sa6 (* m65 s5)))

(setq sa7 (- s7 (* m76 sb6)))

(setq sb7 (- sa7 (* m78 s8)))

(setq sa9 (- s9 (* m96 sb6)))

(setq sb9 (- sa9 (* m98 s8)))

(setq sc9 (- sb9 (* m97 sb7)))

(setq f14y (/ sc9 ac99))

(setq f14x (/ (- sb7 (* ab79 f14y)) ab78))

(setq f34y (/ (- s8 (* 0 f14x) (* a89 f14y)) a87))

(setq f34x (/ (- sb6 (* ab67 f34y) (* ab68 f14y) (* ab69 f14y)) ab66))

(setq f23y (/ (- s5 (* 0 f34x) (* a57 f34y) (* 0 f14x) (* 0 f14y)) a55))

(setq f23x (/ (- s4 (* 0 f23y) (* a46 f34x) (* 0 f34y) (* 0 f14x) (* 0 f14y)) a44))

(setq ts (/ (- sb3 (* ab34 f23x) (* ab35 f23y) (* ab36 f34y) (* ab37 f34y) (* ab38 f14x) (* ab39

f14y)) ab33))

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
(setq f12y (/ (- s2 (* 0 ts) (* 0 f23x) (* a25 f23y) (* 0 f34x) (* 0 f34y) (* 0 f14x) (* 0 f14y))
a22))
(setq f12x (/ (- s2 (* 0 f12y) (* 0 ts) (* a14 f23x) (* 0 f23y) (* 0 f34x) (* 0 f34y) (* 0 f14x) (*
0 f14y)) a11))
```

```
(setq f12 (sqrt (+ (* f12x f12x) (* f12y f12y))))
```

```
(setq cetaf12 (atan (/ f12y f12x)))
```

```
(if (and (> f12y 0) (> f12x 0))
```

```
  (setq cetaf12 cetaf12)
```

```
)
```

```
(if (and (> f12y 0) (< f12x 0))
```

```
  (setq cetaf12 (- pi (abs cetaf12)))
```

```
)
```

```
(if (and (< f12y 0) (< f12x 0))
```

```
  (setq cetaf12 (+ pi (abs cetaf12)))
```

```
)
```

```
(if (and (< f12y 0) (> f12x 0))
```

```
  (setq cetaf12 (- (* 2 pi) (abs cetaf12)))
```

```
)
```

```
(setq f23 (sqrt (+ (* f23x f23x) (* f23y f23y))))
```

```
(setq cetaf23 (atan (/ f23y f23x)))
```

```
(if (and (> f23y 0) (> f23x 0))
```

```
  (setq cetaf23 cetaf23)
```

```
)
```

```
(if (and (> f23y 0) (< f23x 0))
```

```
  (setq cetaf23 (- pi (abs cetaf23)))
```

```
)
```

```
(if (and (< f23y 0) (< f23x 0))
```

```
  (setq cetaf23 (+ pi (abs cetaf23)))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

)
(if (and (< f23y 0) (> f23x 0))
  (setq cetaf23 (- (* 2 pi) (abs ccetaf23))))
)

(setq f14 (sqrt (+ (* f14x f14x) (* f14y f14y))))
(setq ccetaf14 (atan (/ f14y f14x)))
(if (and (> f14y 0) (> f14x 0))
  (setq cetaf14 ccetaf14)
)
(if (and (> f14y 0) (< f14x 0))
  (setq cetaf14 (- pi (abs ccetaf14)))
)
(if (and (< f14y 0) (< f14x 0))
  (setq cetaf14 (+ pi (abs ccetaf14)))
)
(if (and (< f14y 0) (> f14x 0))
  (setq cetaf14 (- (* 2 pi) (abs ccetaf14)))
)

(setq f32 f23)
(setq cetaf32 (+ pi cetaf23))

(setq f34 (sqrt (+ (* f34x f34x) (* f34y f34y))))
(setq ccetaf34 (atan (/ f34y f34x)))
(if (and (> f34y 0) (> f34x 0))
  (setq cetaf34 ccetaf34)
)
(if (and (> f34y 0) (< f34x 0))
  (setq cetaf34 (- pi (abs ccetaf34)))
)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

)
(if (and (< f34y 0) (< f34x 0))
  (setq cetaf34 (+ pi (abs ccetaf34)))
)
(if (and (< f34y 0) (> f34x 0))
  (setq cetaf34 (- (* 2 pi) (abs ccetaf34)))
)
)
(setq f43 f34)
(setq cetaf43 (+ cetaf34 pi))
(setq temp (/ (* inertia3 alfa3) (* forceinertia3 (sin (- rad3 cetaacce3))))))
(setq l3 (- (/ le3 2) temp))
(setq cetal3 rad3)

(setq temp (/ (* inertia4 alfa4) (* forceinertia4 (sin (- cetaacce4 rad4))))))
(setq l4 (+ (/ le4 2) temp))
(setq cetal4 rad4)

(setq temp (/ (* inertia2 alfa2) (* forceinertia2 (sin (- rad2 cetaacce2))))))
(setq l2 (- (/ le2 2) temp))
(setq cetal2 rad2)

(setq cetaf02 (+ cetaacce2 pi))
(setq cetaf03 (+ cetaacce3 pi))
(setq cetaf04 (+ cetaacce4 pi))
(setq offset2 (/ (* inertia2 alfa2) forceinertia2))
(if (> alfa2 0)
  (setq cetaoffset2 (- cetaacce2 (/ pi 2)))
  (setq cetaoffset2 (+ cetaacce2 (/ pi 2)))
)
)

```

```

(setq offset3 (/ (* inertia3 alfa3) forceinertia3))
(if (> alfa3 0)
  (setq cetaoffset3 (- cetaacce3 (/ pi 2)))
  (setq cetaoffset3 (+ cetaacce3 (/ pi 2)))
)

(setq offset3 (/ (* inertia3 alfa3) forceinertia3))
(if (> alfa3 0)
  (setq cetaoffset3 (- cetaacce3 (/ pi 2)))
  (setq cetaoffset3 (+ cetaacce3 (/ pi 2)))
)
(force_vector)
)
(defun force_vector(/ f_check f_link vf12 vf23 inertia2 vfinertia2
vf32 vf34 inertia3 vfinertia3
vf43 vf14 inertia4 vfinertia4)
(command "erase" "all" "")
(setq f_check 0)
(setq xmax 7.5)
(f_text)
(command "zoom" "e")
(command "zoom" "0.8x")
(while (= f_check 0)
  (setq f_link (getpoint "CHOOSE F_VECTOR"))
  (cond
    ((and (>= (car f_link) (- xmax 0.25))
      (<= (car f_link) (+ xmax 2.2))
      (>= (cadr f_link) 0.5)
      (<= (cadr f_link) 1.45))
    )
  )
)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

(command "erase" "all" "")
(setq pt2 (polar (list 0 0) rad2 le2))
(command "mline" "j" "z" "s" 1 "st" "bar" (list 0 0) pt2 "")
(setq vf12 (polar (list 0 0) cetaf12 3))
(command "leader" vf12 (list 0 0) "" "" "n")
(command "text" (list (+ (car vf12) 0.5) (cadr vf12)) "0.3" "0" "F12")
(setq vf23 (polar pt2 cetaf23 3))
(command "leader" vf23 (list (car pt2) (cadr pt2)) "" "" "n")
(command "text" (list (+ (car vf23) 0.5) (cadr vf23)) "0.3" "0" "F23")
(setq inertiapt2 (polar (list 0 0) (angle (list 0 0) pt2) l2))
(setq vfinertia2 (polar inertiapt2 cetaf02 3))
(command "leader" vfinertia2 inertiapt2 "" "" "n")
(command "text" (list (+ (car vfinertia2) 0.5) (cadr vfinertia2)) "0.3" "0"
"Finertia2")
(command "point" (polar (list 0 0) (angle (list 0 0) pt2) (/ (distance (list 0 0) pt2) 2)))
(setq xmin (min 0 (car pt2) (car vf12) (car vf23) (car vfinertia2)))
(setq xmax (max 0 (car pt2) (car vf12) (car vf23) (car vfinertia2)))
(setq xmax (+ xmax 5))
(setq ymax (max 0 (cadr pt2) (cadr vf12) (cadr vf23) (cadr vfinertia2)))
(f_text)
(command "zoom" "w" (list (+ xmax 2.2) -4) (list xmin ymax))
(command "zoom" "0.8x")
)

((and (>= (car f_link) (- xmax 0.25))
(<= (car f_link) (+ xmax 2.2))
(>= (cadr f_link) -0.75)
(<= (cadr f_link) 0.2)
)

```

```

(setq pt3 (polar (list 0 3) rad3 le3))
(command "erase" "all" "")
(command "mline" "j" "z" "s" 1 "st" "bar" (list 0 3) pt3 "")
(setq vf32 (polar (list 0 3) cetaf32 3))
(command "leader" vf32 (list 0 3) "" "" "n")
  (command "text" (list (+ (car vf32) 0.5) (cadr vf32)) "0.3" "0" "F32")
(setq vf34 (polar pt3 cetaf34 3))
(command "leader" vf34 (list (car pt3) (cadr pt3)) "" "" "n")
  (command "text" (list (+ (car vf34) 0.5) (cadr vf34)) "0.3" "0" "F34")
(setq inertiapt3 (polar (list 0 3) (angle (list 0 3) pt3) l3))
(setq vfinertia3 (polar inertiapt3 cetaf03 3))
(command "leader" vfinertia3 inertiapt3 "" "" "n")
  (command "text" (list (+ (car vfinertia3) 0.5) (cadr vfinertia3)) "0.3" "0"
"Finertia3")
(command "point" (polar (list 0 3) (angle (list 0 3) pt3) (/ (distance (list 0 3) pt3) 2)))
  (setq xmin (min 0 (car pt3) (car vf32) (car vf34) (car vfinertia3)))
  (setq xmax (max 0 (car pt3) (car vf32) (car vf34) (car vfinertia3)))
(setq xmax (+ xmax 5))
  (setq ymax (max 3 (cadr pt3) (cadr vf32) (cadr vf34) (cadr vfinertia3)))
  (f_text)
(command "zoom" "w" (list (+ xmax 2.2) -4) (list xmin ymax))
  (command "zoom" "0.8x")
)

( (and (>= (car f_link) (- xmax 0.25))
  (<= (car f_link) (+ xmax 2.2))
  (>= (cadr f_link) -2)
  (<= (cadr f_link) -1.05)
)

```

```

(command "erase" "all" "")
(setq pt4 (polar (list 4 0) rad4 le4))
(command "mline" "j" "z" "s" 1 "st" "bar" (list 4 0) pt4 "")
(setq vf43 (polar pt4 cetaf43 3))
(command "leader" vf43 (list (car pt4) (cadr pt4)) "" "" "n")
  (command "text" (list (+ (car vf43) 0.5) (cadr vf43)) "0.3" "0" "F43")
(setq vf14 (polar (list 4 0) cetaf14 3))
(command "leader" vf14 (list 4 0) "" "" "n")
(command "text" (list (+ (car vf14) 0.5) (cadr vf14)) "0.3" "0" "F14")
(setq inertiapt4 (polar (list 4 0) (angle (list 4 0) pt4) l4))
(setq vfinertia4 (polar inertiapt4 cetaf04 3))
(command "leader" vfinertia4 inertiapt4 "" "" "n")
(command "text" (list (+ (car vfinertia4) 0.5) (cadr vfinertia4)) "0.3" "0" "Finertia4")
(command "point" (polar (list 4 0) (angle (list 4 0) pt4) (/ (distance (list 4 0) pt4) 2)))
(setq xmin (min 4 (car pt4) (car vf43) (car vf14) (car vfinertia4)))
(setq xmax (max 4 (car pt4) (car vf43) (car vf14) (car vfinertia4)))
  (setq xmax (+ xmax 5))
  (setq ymax (max 0 (cadr pt4) (cadr vf43) (cadr vf14) (cadr vfinertia4)))
  (f_text)
(command "zoom" "w" (list (+ xmax 2.2) -4) (list xmin ymax))
  (command "zoom" "0.8x")
)

((and (>= (car f_link) (- xmax 0.25))
  (<= (car f_link) (+ xmax 2.2))
  (>= (cadr f_link) -3.25)
  (<= (cadr f_link) -2.3)
)

(setq f_check 1)

```

```

)
)
)
)
(defun f_text()
  (f_menu (list xmax 0.75) "Link2")
  (f_menu (list xmax -0.5) "Link3")
  (f_menu (list xmax -1.75) "Link4")
  (f_menu (list xmax -3) "EXIT")
  (command "dimasz" "0.6")
)
(defun f_menu( fpt ftxt)
  (command "text" fpt "0.5" "0" ftxt)
  (command "rectangle" (list (- xmax 0.25) (+ (cadr fpt) 0.7)) (list (+ xmax 2.2) (- (cadr fpt)
0.25)))
)
(defun slidevelo()
  (setq srad2 (/ (* sceta2 pi) 180))
  (setq srad3 (/ (* sceta3 pi) 180))
  (while (< srad2 0) (setq srad2 (+ srad2 (* 2 pi))) )
  (while (>= srad2 (* 2 pi)) (setq srad2 (- srad2 (* 2 pi)))) )

  (setq svelo2 (* (abs somega2) sle2)) ;Link2
  (if (> somega2 0)
    (setq scetavelo2 (+ srad2 (/ pi 2)))
    (setq scetavelo2 (- srad2 (/ pi 2)))
  )
  (setq xsv1 0)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

(setq ysv1 0)
(setq xsv2 (+ xsv1 (* svelo2 (cos scetavelo2))))
(setq ysv2 (+ ysv1 (* svelo2 (sin scetavelo2))))

(if (or (= sceta2 90) (= sceta2 270)) (setq svelo3 0)) ;Link3
(if (or (= sceta2 0) (= sceta2 180) (= sceta2 360))
    (progn
      (setq svelo3 svelo2)
      (setq scetavelo3 (+ scetavelo2 pi))
    )
)
)
)
(if (> somega2 0)
    (progn
      (if (and (> srad2 0) (< srad2 (/ pi 2)))
          (setq scetavelo3 (- srad3 (/ pi 2)))
        )
      (if (and (> srad2 (/ pi 2)) (< srad2 pi))
          (setq scetavelo3 (+ srad3 (/ pi 2)))
        )
      (if (and (> srad2 pi) (< srad2 (/ (* 3 pi) 2)))
          (setq scetavelo3 (- srad3 (/ pi 2)))
        )
      (if (and (> srad2 (/ (* 3 pi) 2)) (< srad2 (* 2 pi)))
          (setq scetavelo3 (+ srad3 (/ pi 2)))
        )
    )
)
)
)
)
(if (< somega2 0)
    (progn
      (if (and (> srad2 0) (< srad2 (/ pi 2)))

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

(setq scetavelo4 0)
(setq scetavelo4 pi)
)
(if (= draw_v nil)
  (setq draw_v 0)
)
(while (= draw_v 0)
  (setq xmax (max xsv1 xsv2 xsv3))
  (setq xmin (min xsv1 xsv2 xsv3))
  (setq ymax (max ysv1 ysv2 ysv3))
  (setq ymin (min ysv1 ysv2 ysv3))
  (setq htext (/ (- ymax ymin) 15))
  (command "dimasz" (* 1 htext))
  (command "erase" "all" "")
  (command "color" "red")
  (setq value (rtos svelo2 2 2))
  (setq V2 (strcat "V2" "=" value))
  (command "leader" (list xsv2 ysv2) (list xsv1 ysv1) "" "" "n")
  (command "text" (list (+ xmax (* 3 htext)) (+ ymin (* 6 htext))) (* 1 htext) "0" V2)
  (command "color" "green")
  (if (/= svelo3 0)
    (progn
      (setq value (rtos svelo3 2 2))
      (setq V3 (strcat "V3" "=" value))
      (command "leader" (list xsv3 ysv3) (list xsv2 ysv2) "" "" "n")
      (command "text" (list (+ xmax (* 3 htext)) (+ ymin (* 4 htext))) (* 1 htext) "0" V3)
    )
  )
  (command "color" "blue")
  (if (/= svelo4 0)

```

```

(progn
  (setq value (rtos svelo4 2 2))
  (setq V4 (strcat "V4" "=" value))
  (command "leader" (list xsv3 ysv3) (list xsv1 ysv1) "" "" "n")
  (command "text" (list (+ xmax (* 3 htext)) (+ ymin (* 2 htext))) (* 1 htext) "0" V4)
)
)
(command "color" "7")
(command "text" (list (+ xmax (* 3 htext)) ymin) (* 1 htext) "0" "DONE")
(command "rectangle" (list (+ xmax (* htext 2.5)) (+ ymin (* 1.3 htext)))
  (list (+ xmax (* 7.15 htext)) (- ymin (* 0.25 htext)))
)
(command "zoom" "e")
(command "zoom" "0.8x")
(setq v_check 0)
(while (= v_check 0)
  (setq done (getpoint "BACK TO MENU"))
  (if (and (>= (car done) (+ xmax (* htext 2.5)))
    (<= (car done) (+ xmax (* htext 7.15)))
    (>= (cadr done) (- ymin (* 0.25 htext)))
    (<= (cadr done) (+ ymin (* htext 1.3)))
  )
  (progn
    (setq v_check 1)
    (setq draw_v 1)
    (setq draw_a nil)
  )
)
)
)
)
)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

)
(defun slideacce()
  (setq draw_v 1)
  (slidevelo)
  (setq saccen2 (/ (* svelo2 svelo2) sle2))
  (setq scetaaccen2 (+ srad2 pi))
  (setq xsa1 0)
  (setq ysa1 0)
  (setq xsan2 (+ xsa1 (* saccen2 (cos scetaaccen2))))
  (setq ysan2 (+ ysa1 (* saccen2 (sin scetaaccen2))))
  (setq saccet2 (* (abs salfa2) sle2))
  (if (> salfa2 0)
    (setq scetaaccet2 (+ srad2 (/ pi 2)))
    (setq scetaaccet2 (- srad2 (/ pi 2)))
  )
  (setq xsat2 (+ xsan2 (* saccet2 (cos scetaaccet2))))
  (setq ysat2 (+ ysan2 (* saccet2 (sin scetaaccet2))))
  (setq saccen3 (/ (* svelo3 svelo3) sle3))
  (setq scetaaccen3 (+ srad3 pi))
  (setq xsan3 (+ xsat2 (* saccen3 (cos scetaaccen3))))
  (setq ysan3 (+ ysat2 (* saccen3 (sin scetaaccen3))))
  (setq a (* (- ysat2 ysan3) (- ysa1 ysan3)))
  (setq b (- xsan3 xsat2))
  (setq xsat3 (+ (/ a b) xsan3))
  (setq ysat3 ysa1)
  (setq saccet3 (distance (list xsan3 ysan3) (list xsat3 ysat3)))
  (setq scetaaccet3 (atan (- ysat3 ysan3) (- xsat3 xsan3)))
  (if (and (> (- ysat3 ysan3) 0) (> (- xsat3 xsan3) 0)) (setq scetaaccet3 scetaaccet3) )
  (if (and (> (- ysat3 ysan3) 0) (< (- xsat3 xsan3) 0)) (setq scetaaccet3 (- pi (abs scetaaccet3)))) )
  (if (and (< (- ysat3 ysan3) 0) (> (- xsat3 xsan3) 0)) (setq scetaaccet3 (+pi (abs scetaaccet3)))) )

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
(if (and (< (- ysat3 ysan3) 0) (< (- xsat3 xsan3) 0)) (setq scetaacct3 (- (* 2 pi) scetaacct3)))
```

```
(setq saccen4 0)
```

```
(setq saccet4 (distance (list xsa1 ysa1) (list xsat3 ysat3)))
```

```
(if (> (- xsat3 xsa1) 0)
```

```
  (setq scetaacct4 0)
```

```
  (setq scetaacct4 pi)
```

```
)
```

```
(if (= draw_a nil)
```

```
  (setq draw_a 0)
```

```
)
```

```
(while (= draw_a 0)
```

```
  (setq xmax (max xsa1 xsa2 xsat2 xsan3 xsat3))
```

```
  (setq xmin (min xsa1 xsa2 xsat2 xsan3 xsat3))
```

```
  (setq ymax (max ysa1 ysa2 ysat2 ysan3 ysat3))
```

```
  (setq ymin (min ysa1 ysa2 ysat2 ysan3 ysat3))
```

```
  (setq htext (/ (- ymax ymin) 20))
```

```
  (command "dimasz" (* 1 htext))
```

```
  (command "erase" "all" "")
```

```
  (command "color" "red")
```

```
  (setq value (rtos saccen2 2 2))
```

```
  (setq An2 (strcat "An2" "=" value))
```

```
  (command "leader" (list xsan2 ysan2) (list xsa1 ysa1) "" "" "n");ทิศทางของvector an2
```

```
  (command "text" (list (+ xmax (* 3 htext)) (+ ymin (* 10 htext))) (* 1 htext) "0" An2)
```

```
  (command "color" "green")
```

```
(if (/= salfa2 0)
```

```
  (progn
```

```
    (setq value (rtos saccet2 2 2))
```

```
    (setq At2 (strcat "At2" "=" value))
```

```
    (command "leader" (list xsat2 ysat2) (list xsan2 ysan2) "" "" "n");ทิศทางของvector at2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

(command "text" (list (+ xmax (* 3 htext)) (+ ymin (* 8 htext))) (* 1 htext) "0" At2)
)
)
(command "color" "blue")
(if (/= svelo3 0)
  (progn
    (setq value (rtos saccen3 2 2))
    (setq An3 (strcat "An3" "=" value))
    (command "leader" (list xsan3 ysan3) (list xsat2 ysat2) "" "" "n") ;ทิศทางของvector an3
    (command "text" (list (+ xmax (* 3 htext)) (+ ymin (* 6 htext))) (* 1 htext) "0" An3)
  )
)
)
(command "color" "magenta")
...
(setq value (rtos saccet3 2 2))
(setq At3 (strcat "At3" "=" value))
(command "leader" (list xsat3 ysat3) (list xsan3 ysan3) "" "" "n") ;ทิศทางของvector at3
(command "text" (list (+ xmax (* 3 htext)) (+ ymin (* 4 htext))) (* 1 htext) "0" At3)
(command "color" "yellow")
(setq value (rtos saccet4 2 2))
(setq At4 (strcat "At4" "=" value))
(command "leader" (list xsat3 ysat3) (list xsa1 ysa1) "" "" "n") ;ทิศทางของvector at4
(command "text" (list (+ xmax (* 3 htext)) (+ ymin (* 2 htext))) (* 1 htext) "0" At4)
(command "color" "7")
(command "text" (list (+ xmax (* 3 htext)) ymin) (* 1 htext) "0" "DONE")
(command "rectangle" (list (+ xmax (* htext 2.5)) (+ ymin (* 1.3 htext)))
  (list (+ xmax (* 7.15 htext)) (- ymin (* 0.25 htext)))
)
)
(command "zoom" "e" )
(command "zoom" "0.8x")
(setq a_check 0)

```

```

(while (= a_check 0)
  (setq done (getpoint "BACK TO MENU"))
  (if (and (>= (car done) (+ xmax (* htext 2.5)))
        (<= (car done) (+ xmax (* htext 7.15)))
        (>= (cadr done) (- ymin (* 0.25 htext)))
        (<= (cadr done) (+ ymin (* htext 1.3)))
      )
    (progn
      (setq a_check 1)
      (setq draw_a 1)
      (setq draw_v nil)
    )
  )
)
)
)

(setq sacceg2 (/ (distance (list xsa1 ysa1) (list xsat2 ysat2)) 2))
(setq scetaacceg2 (atan (- ysat2 ysa1) (- xsat2 xsa1)))
(setq a (/ (+ xsat2 xsat3) 2))
(setq b (/ (+ ysat2 ysat3) 2))
(setq sacceg3 (distance (list xsa1 ysa1) (list a b)))
(setq scetaacceg3 (atan (+ ysat2 ysat3 (* -2 ysa1)) (+ xsat2 xsat3 (* -2 xsa1))))
(setq sacceg4 (/ saccet4 2))
(setq scetaacceg4 scetaaccet4)
)

(defun slideforce()
  (setq draw_a 1)
  (slideacce)
  (setq smass2 (* sden swi sth sle2))
  (setq smass3 (* sden swi sth sle3))

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

(setq smass4 (* sden (/ (* pi (* sdi4 sdi4)) 4) sle4))
(setq sinertia2 (/ (* (+ (* sle2 sle2) (* swi swi)) smass2) 12))
(setq sinertia3 (/ (* (+ (* sle3 sle3) (* swi swi)) smass3) 12))
(setq sinertia4 (/ (* (* sdi4 sdi4) smass4) 2))

(setq sa (* sle3 (abs (cos srad3))))
(setq sd (* sle3 (abs (sin srad3))))
(setq sforceinertia2 (* smass2 sacceg2))
(setq scetaforceinertia2 (+ scetaacceg2 pi))
(setq sforceinertia3 (* smass3 sacceg3))
(setq scetaforceinertia3 (+ scetaacceg3 pi))
(setq sforceinertia4 (* smass4 sacceg4))
(if (= scetaacceg4 0)
    (setq scetaforceinertia4 pi)
)
(if (= scetaacceg4 pi)
    (setq scetaforceinertia4 0)
)
(setq salfa3 (/ saccet3 sle3))
(if (and (> srad2 0) (< srad2 pi))
    (progn
        (setq saccet3 (+ saccet3 (* 2 pi)))
        (if (and (> saccet3 srad3) (< scaccet3 (+ srad3 pi)))
            (setq salfa3 (* -1 salfa3))
        )
        (if (and (> saccet3 (+ srad3 pi)) (< saccet3 (+ srad3 (* 2 pi))))
            (setq salfa3 salfa3)
        )
    )
)
)
)
)
)

```

```

(if (and (> srad2 pi) (< srad2 (* 2 pi)))
  (progn
    (if (and (> saccet3 srad3) (< saccet3 (+ srad3 pi)))
      (setq salfa3 (* -1 salfa3))
    )
    (if (and (> saccet3 (+ srad3 pi)) (< saccet3 (+ srad3 (* 2 pi))))
      (setq salfa3 salfa3)
    )
  )
)
)
)
(while (>= srad3 (* 2 pi)) (setq srad3 (- srad3 (* 2 pi))))
(while (< srad3 0) (setq srad3 (+ srad3 (* 2 pi))))
(while (>= scetaacceg2 (* 2 pi)) (setq scetaacceg2 (- scetaacceg2 (* 2 pi))))
(while (< scetaacceg2 0) (setq scetaacceg2 (+ scetaacceg2 (* 2 pi))))
(while (>= scetaforceinertia3 (* 2 pi)) (setq scetaforceinertia3 (- scetaforceinertia3 (* 2 pi))))
(while (< scetaforceinertia3 0) (setq scetaforceinertia3 (+ scetaforceinertia3 (* 2 pi))))

(setq a (/ (* sinertia2 salfa2) sforceinertia2))
(setq a (/ a (sin (- srad2 scetaacceg2))))
(setq sl2 (- (/ sle2 2) a))
(setq a (/ (* sinertia3 salfa3) sforceinertia3))
(setq a (/ a (sin (- srad3 scetaacceg3))))
(setq sl3 (- (/ sle3 2) a))
(if (and (> srad2 0) (< srad2 pi))
  (progn
    (if (and (> scetaforceinertia3 srad3) (< scetaforceinertia3 0)) ;A
      (progn
        (setq sb (* sl3 (abs (cos (- (/ pi 2) (- (abs srad3) (abs scetaforceinertia3))))))
        (if (= scetaforceinertia4 0)
          (setq sf14 (/ (- (* sfriktion sd) (* sforceinertia3 sb) (* sforceinertia4 sd)) sa))
        )
      )
    )
  )
)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

(setq sb sl3)
(if (= scetaforceinertia4 0)
  (setq sf14 (/ (- (* sfriction sd) (* sforceinertia3 sb) (* sforceinertia4 sd)) sa))
)
(if (= scetaforceinertia4 pi)
  (setq sf14 (/ (- (* sforceinertia4 sd) (* sforceinertia3 sb) (* -1 sfriction sd)) sa))
)
)
)
)
(if (and (> scetaforceinertia3 (+ srاد3 (/ pi 2))) (< scetaforceinertia3 (- pi (abs srاد3)))) ;E
  (progn
    (setq sb (* sl3 (abs (cos (- (/ pi 2) (- pi (+ scetaforceinertia3 (abs srاد3))))))))
    (if (= scetaforceinertia4 0)
      (setq sf14 (/ (- (* sfriction sd) (* sforceinertia3 sb) (* sforceinertia4 sd)) sa))
    )
    (if (= scetaforceinertia4 pi)
      (setq sf14 (/ (- (* sforceinertia4 sd) (* sforceinertia3 sb) (* -1 sfriction sd)) sa))
    )
  )
)
)
(if (or (= scetaforceinertia3 srاد3) (= scetaforceinertia3 (- pi (abs srاد3))) (=
scetaforceinertia3 (- (* 2 pi) (abs srاد3)))) ;F
  (progn
    (if (= scetaforceinertia4 0)
      (setq sf14 (/ (- (* sfriction sd) (* sforceinertia4 sd)) sa))
    )
    (if (= scetaforceinertia4 pi)
      (setq sf14 (/ (- (* sforceinertia4 sd) (* -1 sfriction sd)) sa))
    )
  )
)
)

```

```

)
(if (and (> scetaforceinertia3 (- pi (abs srad3))) (< scetaforceinertia3 (+ srad3 (/ (* 3 pi
2)))));G
(progn
  (setq sb (* sl3 (abs (cos (- (/ pi 2) (+ scetaforceinertia3 (abs srad3)))))))
  (if (= scetaforceinertia4 0)
    (setq sf14 (/ (- (* sfriiction sd) (* sforceinertia3 sb) (* sforceinertia4 sd)) sa))
  )
  (if (= scetaforceinertia4 pi)
    (setq sf14 (/ (- (* sforceinertia4 sd) (* sforceinertia3 sb) (* -1 sfriiction sd)) sa))
  )
)
)
)
(if (= scetaforceinertia3 (+ (/ (* 3 pi) 2) srad3));H
(progn
  (setq sb sl3)
  (if (= scetaforceinertia4 0)
    (setq sf14 (/ (- (* sfriiction sd) (* sforceinertia3 sb) (* sforceinertia4 sd)) sa))
  )
  (if (= scetaforceinertia4 pi)
    (setq sf14 (/ (- (* sforceinertia4 sd) (* sforceinertia3 sb) (* -1 sfriiction sd)) sa))
  )
)
)
)
(if (and (> scetaforceinertia3 (+ (/ (* 3 pi) 2) srad3)) (< scetaforceinertia3 (- (* 2 pi) (abs
srad3)))));I
(progn
  (setq sb (* sl3 (abs (cos (- (/ pi 2) (* 2 pi) (* -1 (+ scetaforceinertia3 (abs
srad3))))))))
  (if (= scetaforceinertia4 0)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

)
)
)
(if (= scetaforceinertia3 (+ (/ pi 2) srad3)) ;L
  (progn
    (setq sb sl3)
    (if (= scetaforceinertia4 0)
      (setq sf14 (/ (- (* sforceinertia4 sd) (* sforceinertia3 sb) (* -1 sfriction sd))sa))
    )
    (if (= scetaforceinertia4 pi)
      (setq sf14 (/ (- (* sfriction sd) (* sforceinertia3 sb) (* sforceinertia4 sd)) sa))
    )
  )
)
)
)
(if (and (> scetaforceinertia3 (+ (/ pi 2) srad3)) (< scetaforceinertia3 (+ pi srad3))) ;M
  (progn
    (setq sb (* sl3 (abs (cos (- (/ pi 2) pi (- scetaforceinertia3 srad3)))))
    (if (= scetaforceinertia4 0)
      (setq sf14 (/ (- (* sforceinertia4 sd) (* sforceinertia3 sb) (* -1 sfriction sd))sa))
    )
    (if (= scetaforceinertia4 pi)
      (setq sf14 (/ (- (* sfriction sd) (* sforceinertia3 sb) (* sforceinertia4 sd)) sa))
    )
  )
)
)
)
)
(if (and (> scetaforceinertia3 (+ pi srad3)) (< scetaforceinertia3 (+ (/ (* 3 pi) 2) srad3))) ;N
  (progn
    (setq sb (* sl3 (abs (cos (+ (/ pi 2) pi (- scetaforceinertia3 srad3)))))
    (if (= scetaforceinertia4 0)
      (setq sf14 (/ (- (* sforceinertia4 sd) (* sforceinertia3 sb) (* -1 sfriction sd))sa))

```

```

)
(if (= scetaforceinertia4 pi)
  (setq sf14 (/ (- (* sfriction sd) (* sforceinertia3 sb) (* sforceinertia4 sd)) sa))
)
)
)
)
(if (= scetaforceinertia3 (+ (/ (* 3 pi) 2) srad3)) ;O
  (progn
    (setq sb sl3)
    (if (= scetaforceinertia4 0)
      (setq sf14 (/ (- (* sforceinertia4 sd) (* sforceinertia3 sb) (* -1 sfriction sd))sa))
    )
    (if (= scetaforceinertia4 pi)
      (setq sf14 (/ (- (* sfriction sd) (* sforceinertia3 sb) (* sforceinertia4 sd)) sa))
    )
    )
    )
    )
    (if (and (> scetaforceinertia3 (+ (/ (* 3 pi) 2) srad3)) (< scetaforceinertia3 (+ (* 2 pi)
srad3)))) ;P
  (progn
    (setq sb (* sl3 (abs (cos (- (/ pi 2) (* 2 pi) (- scetaforceinertia3 srad3)))))
    (if (= scetaforceinertia4 0)
      (setq sf14 (/ (- (* sforceinertia4 sd) (* sforceinertia3 sb) (* -1 sfriction sd))sa))
    )
    (if (= scetaforceinertia4 pi)
      (setq sf14 (/ (- (* sfriction sd) (* sforceinertia3 sb) (* sforceinertia4 sd)).sa))
    )
    )
    )
    )
    )
  )

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

)
(if (= scetaforceinertia4 0)
(progn
(setq pt1 (list 0 0))
(setq pt2 (list (- (+ 0 sforceinertia4) sfriction) 0))
(setq pt3 (list (- (+ 0 sforceinertia4) sfriction) (+ 0 sf14)))
(setq sf34 (distance pt3 pt1))
(setq scetaf34 (angle pt3 pt1))
)
)
(if (= scetaforceinertia4 pi)
(progn
(setq pt1 (list 0 0))
(setq pt2 (list (- 0 sforceinertia4) sfriction) 0))
(setq pt3 (list (- 0 sforceinertia4) sfriction) (+ 0 sf14)))
(setq sf34 (distance pt3 pt1))
(setq scetaf34 (angle pt3 pt1))
)
)
(setq sf43 sf34)
(setq scetaf43 (+ scetaf34 pi))

(setq pt3 (list 0 0))
(setq sf23 (distance pt3 (list (+ 0 (* sforceinertia3 (cos scetaforceinertia3))
(* sf43 (cos scetaf43))) (+ 0 (* sforceinertia3 (sin scetaforceinertia3))
(* sf43 (sin scetaf43))))))
(setq sy23 (+ (* sforceinertia3 (sin scetaforceinertia3)) (* sf43 (sin scetaf43))))
(setq sx23 (+ (* sforceinertia3 (cos scetaforceinertia3)) (* sf43 (cos scetaf43))))
(setq sscetaf23 (atan (/ sy23 sx23)))

```

```

(if (and (> sy23 0) (> sx23 0)) (setq scetaf23 sscetaf23))
(if (and (> sy23 0) (< sx23 0)) (setq scetaf23 (- pi (abs sscetaf23))))
(if (and (< sy23 0) (< sx23 0)) (setq scetaf23 (+ pi (abs sscetaf23))))
(if (and (< sy23 0) (> sx23 0)) (setq scetaf23 sscetaf23))
(setq sf32 sf23)
(setq scetaf32 (+ scetaf23 pi))

(setq pt2 (list 0 0))
(setq sf12 (distance pt2 (list (+ 0 (* sforceinertia2 (cos scetaforceinertia2))
(* sf32 (cos scetaf32))) (+ 0 (* sforceinertia2 (sin scetaforceinertia2))
(* sf32 (sin scetaf32))))))
(setq sy12 (+ (* sforceinertia2 (sin scetaforceinertia2)) (* sf32 (sin scetaf32))))
(setq sx12 (+ (* sforceinertia2 (cos scetaforceinertia2)) (* sf32 (cos scetaf32))))
(setq sscetaf12 (atan (/ sy12 sx12)))
(if (and (> sy12 0) (> sx12 0)) (setq scetaf12 sscetaf12))
(if (and (> sy12 0) (< sx12 0)) (setq scetaf12 (- pi (abs sscetaf12))))
(if (and (< sy12 0) (< sx12 0)) (setq scetaf12 (+ pi (abs sscetaf12))))
(if (and (< sy12 0) (> sx12 0)) (setq scetaf12 sscetaf12))
(setq sf21 sf12)
(setq scetaf21 (+ scetaf12 pi))

(setq sstl (* sf12 (abs (sin (- scetaf12 srad2))))))
(if (or (and (> scetaf12 srad2) (< scetaf12 (+ srad2 pi)))
(and (> scetaf12 (+ srad2 (* 2 pi))) (< scetaf12 (+ srad2 (* 3 pi))))))
(setq stl sstl))
(if (or (and (> scetaf12 (+ srad2 pi)) (< scetaf12 (+ srad2 (* 2 pi))))
(and (> scetaf12 (- srad2 pi)) (< scetaf12 srad2))))
(setq stl (* -1 sstl)))

(command "erase" "all" "")

```

```

(setq sf_check 0)
(setq xmax 7.5)
(sf_text)
(command "zoom" "e" )
(command "zoom" "0.8x")
(while (= sf_check 0)
  (setq sf_link (getpoint "CHOOSE F_VECTOR"))
  (cond
    ((and (>= (car sf_link) (- xmax 0.25))
          (<= (car sf_link) (+ xmax 2.5))
          (>= (cadr sf_link) 0.5)
          (<= (cadr sf_link) 1.45))
     )
    ...
    (command "erase" "all" ""))
  (setq pt2 (polar (list 0 0) srad2 sle2))
  (command "mline" "j" "z" "s" 1 "st" "bar" (list 0 0) pt2 "")
  (setq vf12 (polar (list 0 0) scetaf12 3))
  (command "leader" vf12 (list 0 0) "" "" "n")
  (command "text" (list (+ (car vf12) 0.5) (cadr vf12)) "0.3" "0" "F12")
  (setq vf23 (polar pt2 scetaf23 3))
  (command "leader" vf23 (list (car pt2) (cadr pt2)) "" "" "n")
  (command "text" (list (+ (car vf23) 0.5) (cadr vf23)) "0.3" "0" "F23")
  (setq sinertiapt2 (polar (list 0 0) (angle (list 0 0) pt2) sl2))
  (setq svfinertia2 (polar sinertiapt2 scetaforceinertia2 3))
  (command "leader" svfinertia2 sinertiapt2 "" "" "n")
  (command "text" (list (+ (car svfinertia2) 0.5) (cadr svfinertia2)) "0.3" "0"
"Finertia2")
  (command "point" (polar (list 0 0) (angle (list 0 0) pt2) (/ sle2 2)))
  (setq xmin (min 0 (car pt2) (car vf12) (car vf23) (car svfinertia2)))
  (setq xmax (max 0 (car pt2) (car vf12) (car vf23) (car svfinertia2)))

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

(setq xmax (+ xamx 5))
  (setq ymax (max 0 (cadr pt2) (cadr vf12) (cadr vf23) (cadr svfinertia2)))
  (sf_text)
(command "zoom" "w" (list (+ xmax 2.5) -4) (list xmin ymax))
  (command "zoom" "0.8x")
)

( (and (>= (car sf_link) (- xmax 0.25))
      (<= (car sf_link) (+ xmax 2.5))
      (>= (cadr sf_link) -0.75)
      (<= (cadr sf_link) 0.2)
  )
  (setq pt3 (polar (list 0 3) srad3 sle3))
  (command "erase" "all" "")
  (command "mline" "j" "z" "s" 1 "st" "bar" (list 0 3) pt3 "")
  (setq vf32 (polar (list 0 3) scetaf32 3))
  (command "leader" vf32 (list 0 3) "" "" "n")
  (command "text" (list (+ (car vf32) 0.5) (cadr vf32)) "0.3" "0" "F32")
  (setq vf34 (polar pt3 scetaf34 3))
  (command "leader" vf34 (list (car pt3) (cadr pt3)) "" "" "n")
  (command "text" (list (+ (car vf34) 0.5) (cadr vf34)) "0.3" "0" "F34")
  (setq sinertiapt3 (polar (list 0 3) (angle (list 0 3) pt3) sl3))
  (setq svfinertia3 (polar sinertiapt3 scetaforceinertia3 3))
  (command "leader" svfinertia3 sinertiapt3 "" "" "n")
  (command "text" (list (+ (car svfinertia3) 0.5) (cadr svfinertia3)) "0.3" "0"
"Finertia3")
  (command "point" (polar (list 0 3) (angle (list 0 3) pt3) (/ sle3 2)))
  (setq xmin (min 0 (car pt3) (car vf32) (car vf34) (car svfinertia3)))
  (setq xmax (max 0 (car pt3) (car vf32) (car vf34) (car svfinertia3)))

```

```

(setq xmax (+ xmax 5))
(setq ymax (max 3 (cadr pt3) (cadr vf32) (cadr vf34) (cadr svfinertia3)))
(setq ymin (min 3 (cadr pt3) (cadr vf32) (cadr vf34) (cadr svfinertia3)))
(sf_text)
(command "zoom" "w" (list (+ xmax 2.5) ymin) (list xmin ymax))
(command "zoom" "0.8x")

)

( (and (>= (car sf_link) (- xmax 0.25))
      (<= (car sf_link) (+ xmax 2.5))
      (>= (cadr sf_link) -2)
      (<= (cadr sf_link) -1.05)
    )
  )
(command "erase" "all" "")
(command "rectangle" (list 0 1.5) (list 4 -1))
(setq pt4 (list 2 0.25))
(setq vf43 (polar pt4 scetaf43 3))
(command "leader" vf43 pt4 "" "" "n")
(command "text" (list (+ (car vf43) 0.5) (cadr vf43)) "0.3" "0" "F43")
(setq vf14 (list 2 -4))
(command "leader" (list 2 -1) vf14 "" "" "n")
(command "text" (list (+ (car vf14) 0.5) (cadr vf14)) "0.3" "0" "F14")
(command "leader" (list 0 0.25) (list -3 0.25) "" "" "n")
(command "text" (list -6 0.25) "0.3" "0" "Finertia4")
(command "leader" (list 4 0.25) (list 7 0.25) "" "" "n")
(command "text" (list 7.5 0.25) "0.3" "0" "F")
(setq xmax 9)
(setq xmin -6)
(setq ymax 3)

```


(command "rectangle" (list (- xmax 0.25) (+ (cadr fpt) 0.7)) (list (+ xmax 2.5) (- (cadr fpt) 0.25)))

)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้