การใช้คอมพิวเตอร์ช่วยในการกำหนดตำแหน่งบุช ในการออกแบบจิกสำหรับเจาะ

นางสาวพรพรรณ ทองแพง

มหาวิทยาลัยเทคโนโลยีสุรนารี

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมอุตสาหการ
มหาวิทยาลัยเทคโนโลยีสุรนารี
ปีการศึกษา 2558

COMPUTER - AIDED DETERMINATION OF BUSH POSITION IN DRILL JIG

Phornpan Thongpang

A Thesis Submitted in Partial Fulfillment of the Requirements for the

Degree of Master of Engineeringe in Industrial Engineering

Suranaree University of Technology

Academic Year 2015

การใช้คอมพิวเตอร์ช่วยในการกำหนดตำแหน่งบุชในการออกแบบจิกสำหรับเจาะ

มหาวิทยาลัยเทคโนโลยีสุรนารี อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

คณะกรรมการสอบวิทยานิพนธ์

_____

(ผศ. ดร.พงษ์ชัย จิตตะมัย)
ประธานกรรมการ

_____

(ผศ. ดร.ยงยุทธ เสริมสุธีอนุวัฒน์)
กรรมการ (อาจารย์ที่ปรึกษาวิทยานิพนธ์)

_____

(รศ. ดร.พรศิริ จงกล)
กรรมการ

_____

(ผศ. ดร.ปภากร พิทยชวาล)
กรรมการ

_____

(ศ. ดร.ชูกิจ ลิมปิจำนงค์)
รองอธิการบดีฝ่ายวิชาการและนวัตกรรม

_____

(รศ. ร.อ. ดร.กนต์ธร ชำนิประศาสน์)
คณบดีสำนักวิชาวิศวกรรมศาสตร์

มหาวิทยาลัยเทคโนโลยีสุรนารี

พรพรรณ ทองแพง : การใช้คอมพิวเตอร์ช่วยในการกำหนดตำแหน่งบุชในการออกแบบจิกสำหรับเจาะ (COMPUTER – AIDED DETERMINATION OF BUSH POSITION IN DRILL JIG) อาจารย์ที่ปรึกษา : ผู้ช่วยศาสตราจารย์ ดร.ยงยุทธ เสริมสุธีอนุวัฒน์, 180 หน้า.

ในกระบวนการตัดโลหะบนเครื่องมือกลต้องมีการใช้อุปกรณ์ยึดชิ้นงานเสมอซึ่งอาจเป็นอุปกรณ์มาตรฐาน เช่น หัวจับ(chucks) และปากกาจับงาน (mechanical vises) และหัวแบ่ง (dividing head) หรืออาจเป็นอุปกรณ์ที่ออกแบบสำหรับใช้เฉพาะงานที่เรียกว่า จิก (jigs) และฟิกซ์เจอร์ (fixtures) เป็นต้น สิ่งที่ต้องคำนึงถึงในการเลือกใช้ หรือออกแบบคือการกำหนดตำแหน่งที่มีความเหมาะสมกับรูปแบบของชิ้นงาน การเลือกรูปแบบตัวกำหนดตำแหน่งที่ดี การวิเคราะห์ความคลาดเคลื่อน และสามารถใช้การได้จริง ดังนั้นการออกแบบอุปกรณ์ยึดชิ้นงานนอกจากจะต้องอาศัยประสบการณ์ของผู้ที่มีความชำนาญในการออกแบบ และสร้างแล้ว ยังเป็นงานที่มีหลายส่วนที่ต้องทำซ้ำ ๆ กันเพราะต้องลองผิดลองถูกเพื่อให้ได้คำตอบที่เหมาะสมที่สุด และด้วยเหตุนี้จึงมีการนำคอมพิวเตอร์เข้ามาช่วยในการออกแบบอุปกรณ์ยึดชิ้นงาน

ปัจจุบันถึงแม้ว่าจะมีรายงานเกี่ยวกับงานวิจัยทางด้านการพัฒนาโปรแกรมคอมพิวเตอร์สำหรับช่วยออกแบบอุปกรณ์ยึดชิ้นงานอย่างต่อเนื่อง แต่เป็นที่น่าสังเกตว่าโปรแกรมที่เกิดขึ้นส่วนใหญ่ไม่ได้มุ่งเน้นไปถึงการกำหนดขนาดและความคลาดเคลื่อนของมิติของอุปกรณ์ยึดชิ้นงานอันเป็นข้อจำกัดที่จำเป็นในการสร้างอุปกรณ์ยึดชิ้นงาน

จิกสำหรับเจาะรูเป็นอุปกรณ์ยึดชิ้นงานเพื่อเจาะรูให้อยู่ในตำแหน่งที่ต้องการตามแบบ เพราะฉะนั้น ตำแหน่งบุชบนตัวจิกจึงเป็นองค์ประกอบที่สำคัญของการสร้างจิก เนื่องจากเป็นส่วนที่ถ่ายทอดความต้องการในแบบไปสู่ชิ้นงาน

งานวิจัยครั้งนี้มีวัตถุประสงค์หลักที่จะพัฒนาเทคนิคในการสร้างความสัมพันธ์ระหว่างชิ้นส่วนต่าง ๆ กับฟังก์ชันของจิกสำหรับเจาะรูโดยใช้หลักการของ tree diagram ซึ่งเป็นเทคนิคการสร้างความสัมพันธ์ระหว่างมิติภายในชิ้นงาน และ loop diagram ที่แสดงความสัมพันธ์ระหว่างมิติของชิ้นงานประกอบ (part assembly) เพื่อระบุตำแหน่งของบุชบนแผ่นบุชสำหรับจิกประเภทต่าง ๆ ที่มีลักษณะการตั้งตำแหน่งแตกต่างกันโดยใช้คอมพิวเตอร์ ซึ่งจะส่งผลทำให้การออกแบบจิกมีความสะดวกมากยิ่งขึ้น และสามารถออกแบบได้โดยผู้ที่มีความรู้พื้นฐานเกี่ยวกับจิกทั่วไป แม้ว่าโปรแกรมที่ถูกพัฒนาขึ้นมีขอบเขตของการใช้งานจะครอบคลุมเฉพาะ plate jig และ template jig บางชนิดเท่านั้น แต่โปรแกรมก็แสดงให้เห็นว่าการประยุกต์ใช้คอมพิวเตอร์ในการกำหนดรายละเอียดของการ

ออกแบบนั้นเป็นไปได้ และตัวโปรแกรมเองก็สามารถขยายออกไปให้ครอบคลุมชนิดของจิก และชิ้นงานได้หลากหลายขึ้น

สาขาวิชา วิศวกรรมอุตสาหการ ลายมือชื่อนักศึกษา _____

ปีการศึกษา 2558 ลายมือชื่ออาจารย์ที่ปรึกษา _____

PHORNPAN THONGPANG : COMPUTER - AIDED DETERMINATION OF BUSH POSITION IN DRILL JIG. THESIS ADVISOR : ASST. PROF. YONGYOOTH SERMSUTI - ANUWAT, Ph.D., 180 PP.

TOLERANCE ANALYSIS/JIG DESIGN/DRILL BUSH POSITION

Although jigs and fixtures are an essence for the batch production of machined parts, the design and making of these work-holding devices is still a time consuming process and requires a great deal of experience of a skilled tool maker. For more than 40 years since an early attempt of computerizing the design of a clamping device, the practical success is still to be realized despite a lot of attempts have been reported. This could be because most of the computer programs developed lacked for addressing on the attachment of tolerances to dimensions of the device components designed.

This thesis describes a trial attempt to systematically computerize the design of template and plate drill jigs with an intention to determine bush position. The tolerance of hole-position on the workpiece, which is the functional requirement of the design, is the basis for developing the relationship among the functional dimensions on jig components. The loop diagram and the rooted-tree graph have been applied for deriving the equations relating tolerances and basic sizes of the functional dimensions to the design requirement.

The inputs required by the package include workpiece dimensions in the top orthogonal projection view, type of jig, and type of locator with nominal sizes of its components. These design data can be either interactively input into the program or read in from a prepared data file.

The program developed is able to determine hole-positions up to a maximum of 5 holes on the bush plate for rectangular parts and work with locators of either fixed side-stop pins or floating type locators, such as nesting pins, circular pockets, and slots.

Three sets of outputs can be obtained from the program with different mating allowances derived from 3 preferred classes of fit to locate the workpiece in the nesting pins or a circular pocket. Each set comprises the bush positions, and the design sizes and positions of the locators employed.

The user can also modify the results to suit his /her requirement or to suit the accuracy of the machine-tools available for making the device.

Although the program developed has a very limited capability, for confining to small parts with simple shapes of circular and rectangular plate forms, it has illustrated a promising approach to computerizing the work of jig and fixture design.

มหาวิทยาลัยเทคโนโลยีสุรนารี

School of Industrial Engineering Student's Signature _____

Academic Year 2015 Advisor's Signature _____

กิตติกรรมประกาศ

วิทยานิพนธ์นี้สำเร็จลุล่วงด้วยดี ผู้วิจัยขอกราบขอบพระคุณ บุคคล และกลุ่มบุคคลต่าง ๆ ที่ได้กรุณาให้คำปรึกษา แนะนำ ช่วยเหลือ อย่างดียิ่ง ทั้งในด้านวิชาการ และ ด้านการดำเนินงานวิจัย อาทิเช่น

ผู้ช่วยศาสตราจารย์ ดร.ยงยุทธ เสริมสุธีอนุวัฒน์ อาจารย์ที่ปรึกษาวิทยานิพนธ์ ผู้ที่ให้โอกาสทางการศึกษาแก่ผู้วิจัย ให้คำปรึกษาทางด้านวิชาการ แนะนำแนวทางการดำเนินงาน และแนวทางการปรับปรุงแก้ไขปัญหาต่าง ๆ รวมทั้งสละเวลาตรวจทานและแก้ไขวิทยานิพนธ์นี้จนเสร็จสมบูรณ์ และให้ความเมตตาในการอบรมสั่งสอนสิ่งที่เป็นประโยชน์ต่อการดำเนินชีวิตอีกหลายด้าน

รองศาสตราจารย์ ดร.พรศิริ จงกล รองคณบดีฝ่ายวิชาการ สำนักวิชาวิศวกรรมศาสตร์ ที่ให้การดูแลให้คำปรึกษาในทุก ๆ ด้านแก่ผู้วิจัยอย่างดียิ่ง ตลอดจนแนะนำและจัดหาแหล่งของทุนการศึกษาแก่ผู้วิจัย

หัวหน้าสาขาวิชาวิศวกรรมอุตสาหการ และคณาจารย์ในสาขาวิชาทุกท่าน ที่ให้คำปรึกษาให้การช่วยเหลือด้านต่าง ๆ และให้กำลังใจแก่ผู้วิจัยมาโดยตลอด

คณะกรรมการสอบวิทยานิพนธ์ทุกท่านที่ให้คำชี้แนะเพื่อให้วิทยานิพนธ์เล่มนี้สมบูรณ์มากยิ่งขึ้น

ท้ายนี้ ขอกราบขอบพระคุณบิดา มารดา และครอบครัว ที่ให้การเลี้ยงดู ส่งเสริมการศึกษาอย่างดีตลอดมา และให้กำลังใจแก่ผู้วิจัยเสมอมาจนสำเร็จการศึกษา

พรพรรณ ทองแพง

สารบัญ

หน้า

บทคัดย่อ (ภาษาไทย) ..... ก

บทคัดย่อ (ภาษาอังกฤษ) ..... ค

กิตติกรรมประกาศ ..... จ

สารบัญ ..... ฉ

สารบัญตาราง ..... ญ

สารบัญรูป ..... ฎ

บทที่

1 บทนำ 1

1.1 ความเป็นมาและความสำคัญของปัญหา 1

1.2 การจำแนกประเภท และการออกแบบจิกสำหรับเจาะรู (classification of drill jigs and design principles) 3

1.2.1 ประเภทของจิก 3

1.2.2 การออกแบบของจิก 5

1.3 แผนภาพวงรอบ (Loop diagram) 6

1.4 แผนภาพต้นไม้ (Rooted - tree graph) 7

1.5 ความสัมพันธ์ของแผนภาพวงรอบ และแผนภาพต้นไม้ในงานประกอบ 8

1.6 วัตถุประสงค์ของงานวิจัย 9

1.7 ขอบเขตการวิจัย 9

1.8 สรุป 10

2 ทบทวนวรรณกรรม 11

2.1 การใช้คอมพิวเตอร์ช่วยออกแบบอุปกรณ์ยึดชิ้นงาน 11

2.2 ความผิดพลาดในการกำหนดตำแหน่งชิ้นงาน 12

2.3 ความคลาดเคลื่อนของชิ้นงานประกอบ (assembly part components) 13

สารบัญ (ต่อ)

หน้า

2.4 สรุป 14
3 โปรแกรม CADBP 15
3.1 ลักษณะของชิ้นงานและตัวกำหนดตำแหน่งสำหรับ CADBP (forms of machined parts and locators) 15
3.2 โครงสร้างโปรแกรม (structure of CADBP) 17
3.3 สรุป 19
4 โครงสร้างโปรแกรม 20
4.1 ชนิด และหน้าที่ของตัวแปร 20
4.2 โมดูลป้อนข้อมูล (input module) 23
4.3 โมดูลสำหรับกำหนดค่าความคลาดเคลื่อนให้กับชิ้นส่วนประกอบของจิก (tolerance calculation and assignment module) 28
4.4 โมดูลสร้าง loop equation (loop diagram generation module) 29
4.4.1 โมดูลสำหรับสร้างเส้นทางของมิติ (path construct module) 29
4.4.2 โมดูลสำหรับค้นหาเส้นทางของมิติที่ต้องการทราบค่า 34
4.4.3 โมดูลสำหรับเทียบเส้นทางเพื่อสร้างสมการวงรอบ 36
4.5 โมดูลสำหรับคำนวณขนาดของมิติที่ต้องการทราบค่า (modules for tolerance allocation and basic size calculations) 38
4.5.1 การคำนวณขนาดมูลฐาน 40
4.5.2 การคำนวณค่าความคลาดเคลื่อน 42
4.6 สรุป 44
5 การทดสอบโปรแกรม 45
5.1 ตัวอย่างการคำนวณของชิ้นงานรูปเหลี่ยม แบบใช้รางกำหนดตำแหน่ง 45
5.1.1 ผลการคำนวณด้วยโปรแกรม 46
5.1.2 ผลการคำนวณโดยไม่ใช้โปรแกรม 48
5.2 ตัวอย่างการคำนวณของชิ้นงานรูปเหลี่ยม แบบใช้ Nesting pins 51
5.2.1 ผลการคำนวณด้วยโปรแกรม 51

สารบัญ (ต่อ)

หน้า

5.2.2 ผลการคำนวณโดยไม่ใช้โปรแกรม 55
5.3 ตัวอย่างการคำนวณของชิ้นงานรูปเหลี่ยม แบบใช้ Nesting pins
เมื่อชิ้นงานมี 2 รูเจาะ 57
5.3.1 ผลการคำนวณด้วยโปรแกรม 58
5.3.2 ผลการคำนวณโดยไม่ใช้โปรแกรม 59
5.4 ตัวอย่างการคำนวณของชิ้นงานรูปลักษณ์กลม
แบบใช้หมุดเป็นตัวกำหนดตำแหน่ง 61
5.4.1 ผลการคำนวณด้วยโปรแกรม 62
5.4.2 ผลการคำนวณโดยไม่ใช้โปรแกรม 64
5.5 ตัวอย่างการคำนวณของชิ้นงานรูปลักษณ์กลม
แบบใช้หมุดเป็นตัวกำหนดตำแหน่ง เมื่อชิ้นงานมีความต้องการรูเจาะ 2 รู 66
5.5.1 ผลการคำนวณด้วยโปรแกรม 68
5.5.2 ผลการคำนวณโดยไม่ใช้โปรแกรม 69
5.6 ตัวอย่างการคำนวณของชิ้นงานรูปลักษณ์กลม
แบบใช้รูเป็นตัวกำหนดตำแหน่ง 71
5.6.1 ผลการคำนวณด้วยโปรแกรม 72
5.6.2 ผลการคำนวณโดยไม่ใช้โปรแกรม 73
5.7 ตัวอย่างการคำนวณของชิ้นงานรูปลักษณ์กลม
แบบใช้รังเป็นตัวกำหนดตำแหน่ง 76
5.7.1 ผลการคำนวณด้วยโปรแกรม 77
5.7.2 ผลการคำนวณโดยไม่ใช้โปรแกรม 79
5.8 สรุป 81
6 สรุปผลการวิจัย 82
6.1 สรุป 82
6.2 ข้อจำกัดของโปรแกรม 82
6.3 แนวทางวิจัยในอนาคต 83

สารบัญ (ต่อ)

หน้า

รายการอ้างอิง 84
ภาคผนวก
ภาคผนวก ก, คู่มือการใช้งานโปรแกรม CADBP 86
ภาคผนวก ข, รายการโปรแกรม 106
ภาคผนวก ค, บทความวิชาการที่ได้รับการตีพิมพ์เผยแพร่ 175
ประวัติผู้เขียน 180

มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญตาราง

ตารางที่ **หน้า**

4.1 แสดงข้อมูลของตัวแปรสำคัญแบบ Public variables ที่ใช้ในโปรแกรม 21
4.2 แสดงองค์ประกอบของตัวแปร inputx( ) 22
4.3 แสดงข้อมูลของตัวแปรแบบ local variable ที่ใช้ในโมดูล
สำหรับสร้างเส้นทางของมิติ 31

มหาวิทยาลัยเทคโนโลยีสุรนารี

สารบัญรูป

รูปที่ **หน้า**

1.1 template jig แบบใช้สองด้าน (ยงยุทธ, 2556) 4
1.2 template jig แผ่นกลม (ยงยุทธ, 2556) 5
1.3 Loop diagram 6
1.4 ตัวอย่างชิ้นงานกลึง (Orawan & Yongyooth, 2009) 7
1.5 กราฟต้นไม้แสดงความสัมพันธ์ของมิติ (X1, X2,…, X8) 8
1.6 ตัวอย่างชิ้นงานประกอบ 8
1.7 แผนภาพต้นไม้ และแผนภาพต้นไม้รวม
(ก) แผนภาพต้นไม้ของชิ้นส่วน a, b และ c 9
(ข) แผนภาพต้นไม้รวมของชิ้นส่วน a, b และ c 10
2.1 องค์ประกอบจิก ฟิกซ์เจอร์ที่ก่อให้เกิดความผิดพลาด (error) 13
3.1 แผ่นชิ้นงานมีรูปลักษณ์กำหนดตำแหน่งเป็นผิวระนาบ 15
3.2 แผ่นชิ้นงานมีรูปลักษณ์กำหนดตำแหน่งเป็นรูหรือปุ่มทรงกระบอก 16
3.3 ชนิดของตัวกำหนดตำแหน่ง 16
3.4 ปุ่ม, รูและหลุมกำหนดตำแหน่ง 16
3.5 โครงสร้างของ CADBP 18
3.6 ภาพรวมผังการทำงานของ CADBP 19
4.1 ขั้นตอนการรับข้อมูล (data input method) 23
4.2 ชิ้นงานตัวอย่างสำหรับใช้กับ template jig แบบ nesting pin 25
4.3 template jig แบบ nesting pin สำหรับชิ้นงานตามแบบในรูป 4.2 25
4.4 หน้าจอ CADBP>>1 สำหรับชิ้นงานตัวอย่างที่ 4.2 26
4.5 หน้าจอ CADBP>>2 สำหรับชิ้นงานตัวอย่างที่ 4.2 27
4.6 การทำงานของฟังก์ชันในการหาค่าเบี่ยงเบน 28
4.7 ชิ้นงานกลึง 4 มิติ 30
4.8 แผนภาพต้นไม้ของชิ้นงานกลึง 4 มิติ 30

สารบัญรูป (ต่อ)

รูปที่ **หน้า**

4.9 เส้นทาง (path) ความสัมพันธ์มิติของเพลา 30
4.10 ขั้นตอนทำงานในส่วนการสร้างเส้นทางความสัมพันธ์ของมิติ 33
4.11 ขั้นตอนทำงานในโมดูลการค้นหาเส้นทางของมิติที่ต้องการทราบค่า 35
4.12 การเทียบ path 36
4.13 ขั้นตอนทำงานในส่วนการเทียบเส้นทางเพื่อสร้างสมการวงรอบ 37
4.14 การเตรียมข้อมูลการคำนวณขนาดของมิติที่ต้องการทราบค่า 39
4.15 ขั้นตอนทำงานในส่วนการคำนวณขนาดมูลฐานของมิติที่ต้องการทราบ 41
4.16 ขั้นตอนทำงานในส่วนการคำนวณค่าความคลาดเคลื่อนของมิติที่ต้องการทราบ 43
5.1 แบบงานสำหรับใช้ราง และ Nesting pin 45
5.2 แบบร่างงานบนจิกแบบรางในการระบุผิวอ้างอิง 46
5.3 แบบร่างงานบนจิกแบบ Nesting pin ในการระบุผิวอ้างอิงจากแบบงานจริง 51
5.4 แบบร่างงานบนจิกแบบ Nesting pin ในการระบุผิวอ้างอิงจากแบบงานจริง
เมื่อพิจารณา 2 รูเจาะ 57
5.5 แบบงานสำหรับใช้รูเป็นรูปลักษณ์กำหนดตำแหน่ง 61
5.6 แบบร่างงานบนจิกแบบใช้หมุดกำหนดตำแหน่ง ในการระบุผิวอ้างอิง 61
5.7 แผนการผลิตชิ้นงานกลม 2 รูเจาะ 66
5.8 แบบงานสำหรับใช้รูเป็นรูปลักษณ์กำหนดตำแหน่งเมื่อต้องการรูเจาะ 2 รู 67
5.9 แบบร่างงานบนจิกแบบใช้หมุดเป็นตัวกำหนดตำแหน่ง
สำหรับ 2 รูเจาะ ในการระบุผิวอ้างอิง 67
5.10 แผนการผลิตชิ้นงานกลมแบบมีปุ่มยื่น (spigot) 71
5.11 แบบงานสำหรับใช้ หมุด หรือ spigot เป็นรูปลักษณ์กำหนดตำแหน่ง 71
5.12 แบบร่างงานบนจิกแบบใช้รูเพื่อเป็นตัวกำหนดตำแหน่งในการระบุผิวอ้างอิง 72
5.13 แบบงานสำหรับใช้รังโดยมีผิวขอบงานเป็นรูปลักษณ์กำหนดตำแหน่ง 76
5.14 แบบร่างงานบนจิกแบบใช้รังในการกำหนดตำแหน่งในการระบุผิวอ้างอิงจาก 76

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในกระบวนการตัดโลหะจะต้องมีการใช้อุปกรณ์ยึดชิ้นงานเสมอ ซึ่งอาจเป็นอุปกรณ์มาตรฐาน เช่น หัวจับ (chucks) ปากกาจับงาน (mechanical vises) และหัวแบ่ง (dividing head) หรืออาจใช้อุปกรณ์ที่ออกแบบสำหรับใช้เฉพาะงานที่เรียกว่า จิก (jigs) และ ฟิกซ์เจอร์ (fixtures) โดยทั่วไปแล้วในการออกแบบจิก และฟิกซ์เจอร์มีความเกี่ยวโยงไปถึงการวินิจฉัยองค์ประกอบร่วมต่าง ๆ คือ ตัวยึด (clamp), ตัวกำหนดตำแหน่ง (locator), และจุดรองรับ (support points) ที่เกี่ยวข้องกับฟังก์ชันการใช้งาน (Leopold, 2008) ฟิกซ์เจอร์ที่เหมาะสมต้องสามารถควบคุมมิติและความคลาดเคลื่อนที่เกิดขึ้น และมีความมั่นคงในการจับยึดชิ้นงาน สิ่งที่ต้องคำนึงถึงเป็นหลักในการออกแบบคือการกำหนดตำแหน่งที่มีความเหมาะสมกับรูปแบบของชิ้นงาน การเลือกรูปแบบตัวกำหนดตำแหน่งที่ดี การวิเคราะห์ความคลาดเคลื่อน และสามารถใช้การได้จริง ดังนั้นในการออกแบบฟิกซ์เจอร์นอกจากจะต้องอาศัยประสบการณ์ของผู้ที่มีความชำนาญในการออกแบบ และสร้างจิก ฟิกซ์เจอร์แล้ว ยังเป็นงานที่มีหลายส่วนที่ต้องลองผิดลองถูกเพื่อให้ได้คำตอบขององค์ประกอบที่เหมาะสมของอุปกรณ์ยึด ด้วยเหตุนี้จึงมีการประยุกต์ใช้คอมพิวเตอร์เข้ามาช่วยในการออกแบบฟิกซ์เจอร์

การใช้โปรแกรมคอมพิวเตอร์ในการช่วยออกแบบอุปกรณ์ยึดชิ้นงานถือเป็นเครื่องมือที่นิยมนำมาใช้อย่างกว้างขวาง จนมีคำเฉพาะสำหรับการประยุกต์ใช้คอมพิวเตอร์ในแนวนี้หลายคำ เช่น การออกแบบฟิกซ์เจอร์อัตโนมัติ (Automated Fixture Design, AFD) และการใช้คอมพิวเตอร์ช่วยออกแบบฟิกซ์เจอร์ (Computer – Aided Fixture Design, CAFD) โปรแกรมคอมพิวเตอร์เหล่านี้จะสามารถจำแนกออกได้เป็นจำพวกใหญ่ๆคือ การออกแบบฟิกซ์เจอร์ด้วยโมเดลของแข็ง (solid model) โปรแกรมออกแบบฟิกซ์เจอร์อัตโนมัติชนิดดัดแปลงจากของเดิม (Variant fixture design) โปรแกรมออกแบบฟิกซ์เจอร์อัตโนมัติชนิดโปรแกรมสร้าง (Generative fixture design) และการวิเคราะห์ความคลาดเคลื่อนในการออกแบบฟิกซ์เจอร์ด้วยคอมพิวเตอร์ (ยงยุทธ, 2556)

แม้ว่าโปรแกรมคอมพิวเตอร์จะช่วยในการออกแบบจิก และฟิกซ์เจอร์ได้มาก แต่โปรแกรมส่วนใหญ่จะเกี่ยวข้องกับการออกแบบเบื้องต้นของฟิกซ์เจอร์ (preliminary design) เช่นงานของ A.Y.C. Nee และคณะ, 1995 : ที่พัฒนาโปรแกรมประกอบโมดุลล่าฟิกซ์เจอร์ (modular fixture)

ควบคู่ไปกับการพัฒนาโปรแกรมวางแผนกระบวนการผลิต โดย Nee ได้เสนอแนวความคิดของการออกแบบฟิกซ์เจอร์ด้วยคอมพิวเตอร์อย่างเป็นระบบ ทำให้เกิดการพัฒนาโปรแกรมในลักษณะนี้ขึ้นภายหลังอีกหลายโปรแกรม เช่น Vukelic และคณะ (2011) ที่พัฒนาโปรแกรมสร้างฟิกซ์เจอร์โดยใช้วิธีการ 2 อย่างร่วมกันคือวิธีการดัดแปลง และวิธีการสร้าง (variant and generative approaches) โดยอาศัยข้อมูลจากรายละเอียดของชิ้นงาน กระบวนการผลิต และข้อจำกัดการใช้งาน ร่วมกับระบบฐานข้อมูล ผลลัพธ์ของโปรแกรมออกมาในรูปโมเดลของแข็ง และแบบทางวิศวกรรม หรือในอีกทิศทางหนึ่ง เกี่ยวข้องกับความผันแปรของตำแหน่งชิ้นงานเนื่องจากการจับยึดชิ้นงาน เช่นงานของ Raghu และ Melkote (2004) ที่ใช้โปรแกรมคอมพิวเตอร์ช่วยในการวิเคราะห์แบบจำลองในการศึกษาอิทธิพลของลำดับในการออกแรงยึดชิ้นงานซึ่งมีผลต่อความผิดพลาดของตำแหน่งงาน (workpiece location error) นอกจากนั้นแล้วยังมีงานของ Siebenaler และ Melkote, 2006 ที่ได้ศึกษาอิทธิพลขององค์ประกอบของฟิกซ์เจอร์ และชิ้นงาน (fixture – workpiece system) ที่ส่งผลให้เกิดการเสียรูปของชิ้นงาน โดยอาศัยระเบียบวิธีไฟไนต์เอลิเมนต์ (Finite Element Analysis, FEA) ในการวิเคราะห์การเสียรูปของชิ้นงานเมื่อมีการปรับเปลี่ยนค่าของพารามิเตอร์ที่เกี่ยวข้องในการจับยึด แต่เป็นที่น่าสังเกตเห็นได้ว่าโปรแกรมที่สามารถกำหนดขนาดและความคลาดเคลื่อนของมิติมีอยู่น้อย ซึ่งในเรื่องนี้ต้องอาศัยวิธีการวิเคราะห์องค์ประกอบต่างๆที่มีอิทธิพลต่อฟังก์ชันการใช้งานของอุปกรณ์ยึดชิ้นงาน ซึ่งเป็นเรื่องของการระบุขนาด และการกระจาย หรือแบ่งความคลาดเคลื่อนให้กับมิติต่างๆ (tolerance allocation) เพื่อให้สามารถตอบสนองความต้องการใช้งาน (functional requirement) ของอุปกรณ์ที่ออกแบบ โดยเฉพาะอย่างยิ่งในการออกแบบจิกสำหรับเจาะรู ตำแหน่งของบุชบนแผ่นบุชถือเป็นองค์ประกอบที่สำคัญของการสร้างจิก เนื่องจากส่งผลโดยตรงกับฟังก์ชันการใช้งานของจิกนั่นคือตำแหน่งของรูเจาะจากตำแหน่งอ้างอิง (locating feature) ซึ่งเท่าที่ผู้วิจัยทราบการหาตำแหน่งของบุชยังคงมีการศึกษาอยู่ในขอบเขตที่จำกัด

งานวิจัยนี้จึงมีเป้าหมายหลักที่จะพัฒนาเทคนิคในการสร้างความสัมพันธ์ระหว่างชิ้นส่วนต่าง ๆ กับฟังก์ชันของจิกสำหรับเจาะรูโดยใช้หลักการของ tree diagram ซึ่งเป็นเทคนิคการสร้างความสัมพันธ์ระหว่างมิติภายในชิ้นงาน และ loop diagram ที่แสดงความสัมพันธ์ระหว่างมิติของชิ้นงานประกอบ (part assembly) ซึ่งได้มีการกล่าวถึงในหนังสือของ (Drake, 1999; Gladman, 1972; Liggett, 1993) เป็นพื้นฐานการคำนวณ เพื่อระบุตำแหน่งของบุชบนแผ่นบุชสำหรับจิกประเภทต่าง ๆ ที่มีลักษณะการตั้งตำแหน่งแตกต่างกันโดยใช้คอมพิวเตอร์ ซึ่งจะส่งผลทำให้การออกแบบจิกมีความสะดวกมากยิ่งขึ้น และสามารถออกแบบได้โดยผู้ที่มีความรู้พื้นฐานเกี่ยวกับจิก และฟิกซ์เจอร์โดยทั่วไป

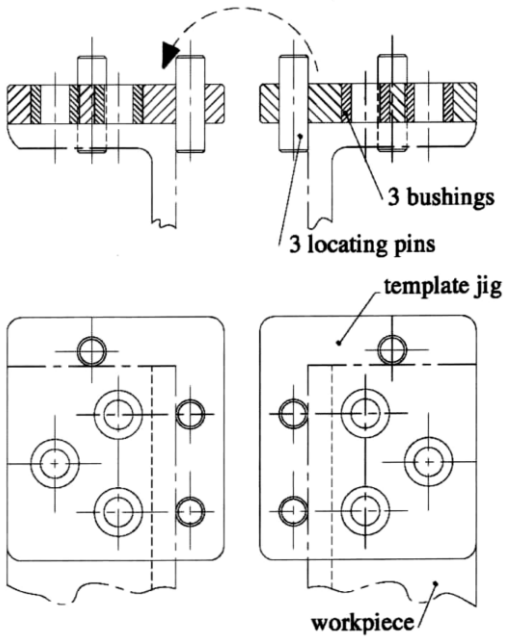
1.2 การจำแนกประเภท และการออกแบบจิกสำหรับเจาะรู (Classification of drill jigs and design principles)

1.2.1 ประเภทของจิก

จิกเป็นอุปกรณ์ยึดชิ้นงานชนิดหนึ่งที่ถูกนำมาใช้กับการผลิตชิ้นงานที่มีการผลิตเป็นรุ่น (batch production) และต้องสามารถสับเปลี่ยนกันได้ระหว่างชิ้นส่วนเดียวกัน เพราะจิกทำหน้าที่ในการจับยึดชิ้นงานให้อยู่ในตำแหน่งเดียวกันและถูกตัดในสภาวะที่ใกล้เคียงกัน นั่นทำให้สามารถควบคุมความสม่ำเสมอของงานได้มากกว่าชิ้นส่วนที่แยกกันผลิต

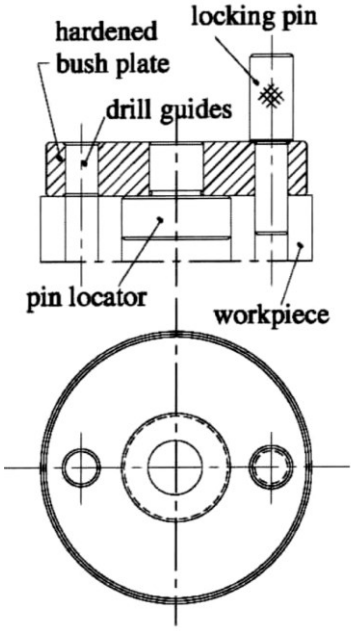
เนื่องจากการใช้งานจิกที่หลากหลายทำให้ยากที่จะจัดประเภท หรือกลุ่มของจิกอย่างชัดเจนเป็นมาตรฐาน การจำแนกประเภทของจิกจึงแบ่งแยกได้หลายประเภท อาจจำแนกโดยพิจารณาจากประเภทการใช้งาน กระบวนการ หรือผลิตภัณฑ์ที่ได้จากจิก (Tool & Engineers, 1962) ซึ่งจิกพื้นฐานสำหรับเจาะรูได้แก่ ลีฟจิก (leaf jig), ทัมเบิลจิก (tumble jig), เทมเพลตจิก (template jig), เพลตจิก (plate jig), อินเด็กซิ่งจิก (indexing jig), ยูนิเวอร์แซลจิก (universal jig), และไวส์จิก (vise jig) และอีกหลากหลายชนิด ในที่นี้จะจำแนกจิกเป็น 2 ประเภทตามลักษณะของการกำหนดตำแหน่งชิ้นงาน คือ เทมเพลตจิก และเพลตจิก ชนิดแรกกำหนดตำแหน่งโดยการลอยตัวชิ้นงานภายในตัวกำหนดตำแหน่ง (floating – type locators) และชนิดหลัง กำหนดตำแหน่งชิ้นงานกับตัวกำหนดตำแหน่ง (fixed – type locators)

เทมเพลตจิก (template jig) ส่วนใหญ่จะประกอบด้วยแผ่นบุช (bush plate) เพียงแผ่นเดียวที่มีรูนำร่องดอกสว่าน ซึ่งอาจจะมีบุช หรือไม่มีบุช โดยอาจใช้การชุบแข็งแผ่นบุชทั้งแผ่นแทน และปล่อยให้ชิ้นงานลอยตัวในตัวกำหนดตำแหน่ง (alternate locators) สำหรับการจับยึดชิ้นงานอาจมีกลไกการยึดชิ้นงานอยู่ภายในตัวจิกหรือไม่ก็ได้ หรืออาจใช้วิธียึดจากกลไกภายนอกชิ้นงานที่ใช้กับจิกชนิดนี้มักมีความคลาดเคลื่อนที่ต่ำ และมีความถูกต้องของรูปแบบของกลุ่มรู (hole pattern) ค่อนข้างสูง ซึ่งจะเห็นตัวอย่างของเทมเพลตจิกดังรูปที่ 1.1 และ 1.2



รูปที่ 1.1 template jig แบบใช้สองด้าน (ยงยุทธ, 2556)

เพลตจิก (plate jig) เป็นจิกที่มีแผ่นบุชเป็นโครงสร้างหลักโดยอาจออกแบบให้แผ่นบุชมีลักษณะเป็นราง หรือสลักที่ขอบข้างเพื่อเป็นตัวกำหนดตำแหน่ง โดยจิกชนิดนี้จะมีกลไกการยึดชิ้นงานอยู่ในตัว เพื่อให้ชิ้นงานติดกับตัวกำหนดตำแหน่ง ซึ่งโครงสร้างของจิกชนิดนี้อาจจะมีลักษณะคล้ายรูปตัวอักษร T, L หรือ Z หรืออาจเป็นแผ่น จึงมีชื่อเรียกที่แตกต่างกันออกไปตามลักษณะโครงสร้าง เช่น เทเบิลจิก (table jig) เป็นจิกที่มีขาตั้ง 4 ขา โดยใส่หรือถอดงานด้วยการหงายจิกให้ขาชี้ขึ้น และแซนด์วิชจิก (sandwich jig) เป็นเพลตจิกที่เหมาะกับการใช้กับชิ้นงานบาง ที่มีโอกาสที่จะบิดได้ในระหว่างการเจาะ ลักษณะเด่นของจิกชนิดนี้คือ มีแผ่นประกบ (back plate) สวมอยู่กับหมุดกำหนดตำแหน่งเพื่อเป็นตัวลดการบิดงอของชิ้นงาน เป็นต้น (Henriksen, 1993 และยงยุทธ, 2556)



รูปที่ 1.2 template jig แผ่นกลม (ยงยุทธ, 2556)

1.2.2 การออกแบบจิก

หลักการออกแบบอุปกรณ์จับยึดชิ้นงานมีวิวัฒนาการอย่างต่อเนื่อง ทั้งนี้การกำหนดหลักการขึ้นเป็นเพียงเพื่อให้การออกแบบจิก และฟิกซ์เจอร์มีความเป็นระบบสามารถศึกษาและปฏิบัติตามได้ ซึ่งจุดที่สำคัญในการพิจารณาเพื่อนำหลักการเหล่านั้นมาใช้มีดังนี้ (ยงยุทธ, 2556)

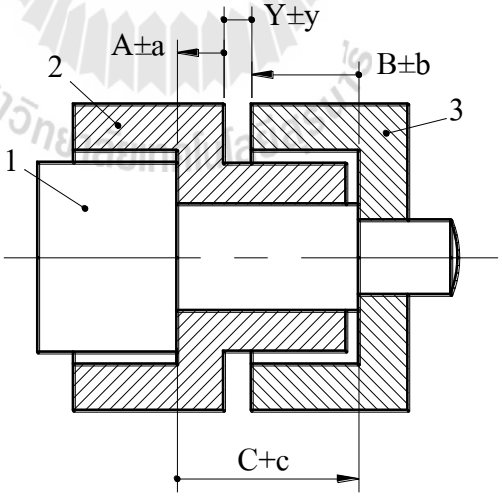
1. พิจารณาความคลาดเคลื่อน, ขนาดและความเที่ยงตรงของชิ้นงาน, น้ำหนัก, จำนวนผลิต, และจำนวนด้านที่ต้องเจาะรู ซึ่งทำให้สามารถกำหนดชนิดของจิกที่เหมาะสมได้
2. กำหนดตำแหน่งชิ้นงาน และกำหนดให้ผิวที่ต้องเจาะรูอยู่ในลักษณะที่เหมาะสมกับการทำงานของเครื่องมือกล และพยายามให้เกิดความรวดเร็วในการทำงาน
3. ออกแบบวิธีการยึดชิ้นงาน เพื่อให้ชิ้นงานไม่ขยับหรือเคลื่อนที่จากผิวกำหนดตำแหน่งในขณะถูกตัด
4. บางกรณี ต้องพิจารณาใช้ตัวรองรับเพื่อบังคับ ไม่ให้ชิ้นงานหมุน หรือบิด หรือเสียรูป เมื่อได้รับแรงที่เกิดจากการยึดหรือจากการตัดโลหะ
5. ออกแบบบุชนำดอกสว่าน (drill bush) ให้เหมาะสมกับการทำงานและจำนวนผลิต
6. ออกแบบตัวจิกเพื่อเป็นจุดรวมในการติดตั้งส่วนต่างๆ
7. ป้องกันการใส่ชิ้นงานผิดพลาด (กันโง่, Fool-proof)

8. ออกแบบให้ง่าย (simple design) ไม่ซับซ้อน และพยายามใช้ชิ้นส่วนมาตรฐานให้มากที่สุด

หลักการกว้าง ๆ เหล่านี้สามารถสรุปได้เป็นกิจกรรมของการออกแบบ 3 อย่างคือ : พิจารณาแบบของชิ้นงาน, กำหนดตำแหน่งชิ้นงาน (workpiece location) และยึดชิ้นงานให้คงที่ (clamping) ซึ่งจะต้องเป็นไปเพื่อจำกัด และกำจัดการเคลื่อนที่อิสระ 6 ชนิด (6 degree of freedom) ของชิ้นงานให้หมดไป เพื่อให้บุช (drill bush) อยู่ ณ. ตำแหน่งที่สามารถนำดอกสว่านไปจุดที่ต้องการบนชิ้นงาน

จะเห็นได้ว่า ไม่ว่าจะเป็นจิกประเภทใดก็ตาม ตำแหน่งของรูที่ถูกเจาะจะได้รับอิทธิพลมาจากขนาด และตำแหน่งของตัวกำหนดตำแหน่งทั้งสิ้น ซึ่งจากประสบการณ์ของการออกแบบ และสร้างจิกเท่านั้น จึงจะสามารถกำหนดขนาด และความคลาดเคลื่อนของชิ้นส่วนต่าง ๆ ของจิกได้ลงตัวจนบังคับรูเจาะให้อยู่ในขอบเขตของตำแหน่งที่ต้องการได้ แต่ในปัจจุบันมีเทคนิควิเคราะห์ความคลาดเคลื่อนของขนาดของชิ้นงานประกอบหลายอย่างที่เราสามารถนำมาประยุกต์ใช้สร้างความสัมพันธ์ระหว่างตำแหน่งรูเจาะที่ต้องการในแบบ กับมิติต่าง ๆ ของชิ้นส่วนของจิกได้ งานวิจัยนี้จึงเป็นความพยายามที่จะพัฒนา หรือประยุกต์เทคนิค หรือวิธีการอย่างเป็นระบบที่จะช่วยในการออกแบบโดยอาศัยการทำงานผ่านโปรแกรมคอมพิวเตอร์

1.3 แผนภาพวงรอบ (Loop diagram)



รูปที่ 1.3 Loop diagram

แผนภาพวงรอบที่เขียนอยู่บนรูป 1.3 เป็นแผนภาพที่ใช้แสดงความสัมพันธ์ของมิติบนงานประกอบ โดยแผนภาพจะแสดงให้เห็นความสัมพันธ์ระหว่างมิติต่างๆ ที่เกี่ยวข้องกับมิติความต้องการใช้งานของงานงานประกอบ ซึ่งในทางการออกแบบแล้ว แผนภาพวงรอบจะถูกเปลี่ยนเป็นความสัมพันธ์ทางคณิตศาสตร์ในรูปของอสมการ ที่เรียกว่า สมการวงรอบ (loop equation) (Drake, 1999; Gladman, 1972; Liggett, 1993)

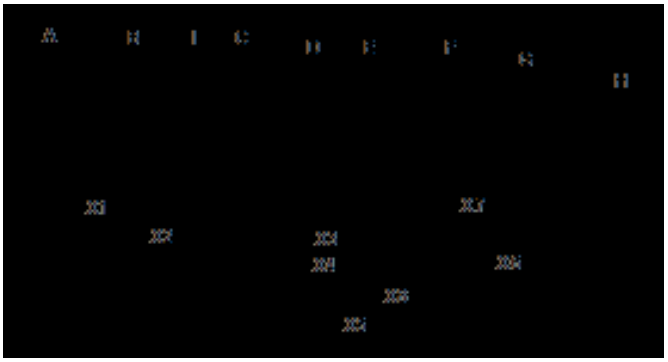
สมการวงรอบ (loop equation) เกิดจากการเปลี่ยนแผนภาพวงรอบให้เป็นสมการทางคณิตศาสตร์ 2 สมการ คือสมการสำหรับขนาดมูลฐาน (loop equation for basic sizes) และสมการสำหรับความคลาดเคลื่อน (loop equation for tolerances) โดยในแต่ละสมการจะประกอบไปด้วยตัวแปรที่แทนค่ามิติใช้งาน (functional dimension) และ ความต้องการใช้งานของงานประกอบนั้น ๆ (functional requirement) สมการทั้งสองจึงเป็นหลักของการหาขนาดมูลฐาน และค่าความคลาดเคลื่อน ของมิติใช้งานทั้งหลายที่เกี่ยวข้องกับความต้องการใช้งานที่พิจารณา

เทคนิคการใช้แผนภาพวงรอบจึงทำให้การหาขนาดของมิติบนงานประกอบทำได้โดยง่าย และมีกฎเกณฑ์ที่ชัดเจนสามารถอธิบายได้โดยใช้สมการทางคณิตศาสตร์ แต่การใช้แผนภาพวงรอบจะใช้อธิบายความสัมพันธ์ได้เฉพาะมิติระหว่างชิ้นส่วนบนงานประกอบเท่านั้น ไม่ครอบคลุมถึงการอธิบายความสัมพันธ์ของมิติหลายมิติที่อยู่บนชิ้นส่วนเดียวกัน

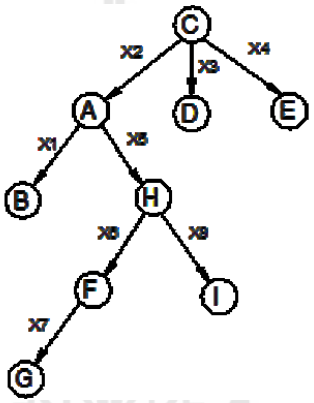
1.4 แผนภาพต้นไม้ (Rooted-tree graph)

เทคนิคแผนภาพต้นไม้ได้ถูกนำมาใช้สร้างความสัมพันธ์ของรอยตัดในการวางแผนกระบวนการผลิตด้วย Tolerance charts เป็นครั้งแรกโดย Whybrew และคณะ (1990) และต่อมามีการใช้ในการสร้างความสัมพันธ์ของมิติบนชิ้นส่วน เมื่อมีการเปลี่ยนแปลงเดตัม (Orawan & Yongyooth, 2009) ซึ่งความสัมพันธ์ที่แสดงมีความเป็นระบบและทำให้ตรวจสอบการคำนวณกลับได้โดยง่าย

แผนภาพต้นไม้สามารถใช้แสดงความสัมพันธ์ของมิติในชิ้นงานเดียวกัน ตัวอย่างเช่น ในรูป 1.4 แสดงมิติ X1, X2, …, X8 ซึ่งสามารถเขียนเป็นแผนภาพต้นไม้ได้ดังในรูปที่ 1.5

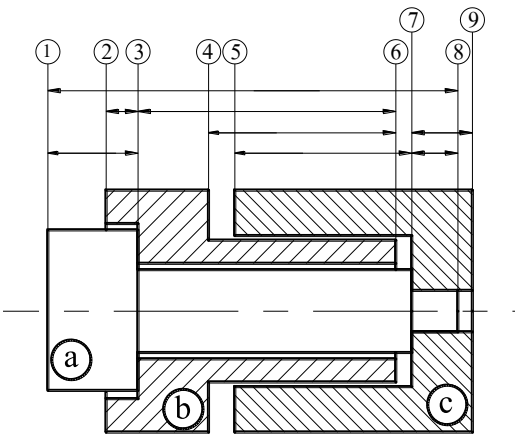


รูปที่ 1.4 ตัวอย่างชิ้นงานกลึง(Orawan & Yongyooth, 2009)



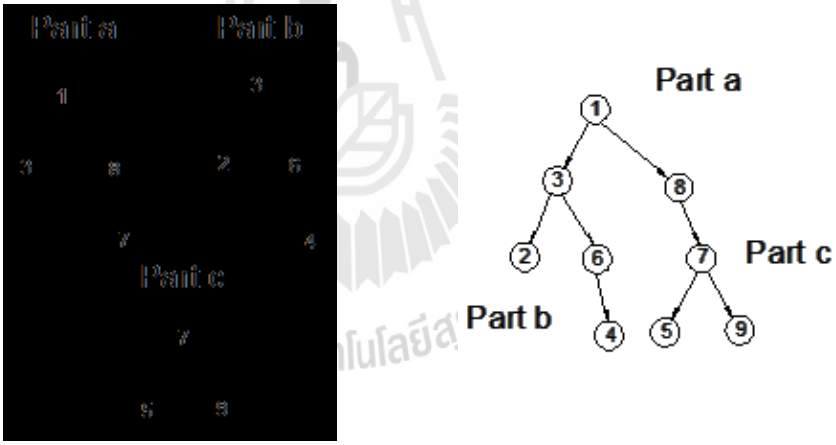
รูปที่ 1.5 กราฟต้นไม้แสดงความสัมพันธ์ของมิติ (X1, X2, …, X8)

1.5 ความสัมพันธ์ของแผนภาพวงรอบ และแผนภาพต้นไม้ในงานประกอบ



รูปที่ 1.6 ตัวอย่างชิ้นงานประกอบ

นอกจากความสัมพันธ์ระหว่างมิติบนชิ้นส่วนใดๆ ที่สามารถอธิบายได้ด้วยแผนภาพต้นไม้แล้ว มิติบนแต่ละชิ้นส่วนยังมีความเชื่อมโยงกันเนื่องจากการประกอบ ตัวอย่างเช่น ชิ้นงานประกอบรูปที่ 1.6 ประกอบด้วยชิ้นส่วน 3 ชิ้น คือ a, b และ c มีมิติที่สามารถแสดงความสัมพันธ์เป็นแผนภาพต้นไม้ได้ดังรูปที่ 1.7 (ก) ซึ่งจะเห็นว่า a และ b มีโหนดที่ใช้ร่วมกันคือ โหนด 3 เช่นเดียวกับ a และ c ที่มีโหนดร่วมกันคือ โหนด 7 หรือมองได้ว่าที่โหนด 3 และ 7 เป็นตำแหน่งที่หน้าตัดของชิ้นส่วน 2 ชิ้นสัมผัสกันจึงสามารถรวม 3 แผนภาพเข้าด้วยกันเป็นแผนภาพต้นไม้รวมดังรูปที่ 1.7 (ข) จิกเป็นงานประกอบประเภทหนึ่งซึ่งขนาดของชิ้นส่วนประกอบมีผลทำให้เกิดการเปลี่ยนแปลงของความต้องการใช้งาน (functional requirement) ในขณะประกอบ เพื่อให้ผลลัพธ์จากการออกแบบคือชิ้นงานเจาะที่มีตำแหน่งถูกต้อง มิติบนแผนภาพวงรอบจึงต้องถูกเชื่อมโยงไปยังมิติบนชิ้นส่วนอื่นๆ ที่เกี่ยวข้องกับมิติบนแผนภาพวงรอบนั้น โดยแผนภาพต้นไม้ของชิ้นส่วนจะถูกเชื่อมโยงเข้ากับแผนภาพวงรอบที่โหนดของมิติบนแผนภาพที่มีการใช้อ้างอิงบอกขนาดบนชิ้นส่วน



(ก) แผนภาพต้นไม้ของชิ้นส่วน a, b และ c

(ข) แผนภาพต้นไม้รวมของชิ้นส่วน a, b และ c

รูปที่ 1.7 แผนภาพต้นไม้ และแผนภาพต้นไม้รวม

1.6 วัตถุประสงค์ของงานวิจัย

งานวิจัยนี้มีวัตถุประสงค์เพื่อ

1. เพื่อเขียนโปรแกรมคอมพิวเตอร์ช่วยในการกำหนดตำแหน่งบุชในอุปกรณ์ยึดชิ้นงานสำหรับงานเจาะ

2. เพื่อศึกษาความสัมพันธ์ระหว่างชิ้นงานกับจิกที่ใช้ในการจับยึด
3. เพื่อทดลองศึกษาความเป็นไปได้ในการออกแบบด้วยโปรแกรมคอมพิวเตอร์โดยอัตโนมัติ

1.7 ขอบเขตการวิจัย

งานวิจัยนี้จะครอบคลุมเฉพาะ plate jig, และ template jig บางชนิดเท่านั้น

1.8 สรุป

การใช้คอมพิวเตอร์ช่วยในการออกแบบอุปกรณ์ยึดชิ้นงานจำพวกจิก และฟิกซ์เจอร์เป็นที่นิยมกันอย่างกว้างขวาง แต่โปรแกรมส่วนใหญ่มักจะใช้ในการออกแบบเบื้องต้น และเป็นที่น่าสังเกตว่าโปรแกรมที่สามารถกำหนดขนาด และความคลาดเคลื่อนของมิติมีอยู่น้อย

เทคนิคแผนภาพวงรอบ และแผนภาพต้นไม้ เป็นเทคนิคที่ใช้เพื่อแสดงความสัมพันธ์ของมิติ โดยเทคนิคแรกเป็นเทคนิคที่ใช้แสดงความสัมพันธ์ระหว่างมิติใช้งานของชิ้นส่วนในชุดงานประกอบที่ก่อให้เกิดความต้องการใช้งาน และอีกเทคนิคหนึ่งใช้แสดงความสัมพันธ์ของมิติในชิ้นงานเดียวกัน จิกเป็นงานประกอบประเภทหนึ่ง ดังนั้นงานวิจัยนี้จึงต้องการนำความสัมพันธ์ระหว่างเทคนิคทั้งสองมาใช้เป็นหลักการพื้นฐานในเขียนโปรแกรมคอมพิวเตอร์ช่วยในการกำหนดตำแหน่งบุชในอุปกรณ์ยึดชิ้นงานสำหรับงานเจาะ

มหาวิทยาลัยเทคโนโลยีสุรนารี

บทที่ 2
ทบทวนวรรณกรรม

งานออกแบบอุปกรณ์ยึดชิ้นงานมีวิวัฒนาการมานานพร้อม ๆ กับการพัฒนาเครื่องมือกล เป็นงานศิลปะที่ต้องอาศัยฝีมือ ความชำนาญเฉพาะด้าน ดังนั้นในช่วงของกลางศตวรรษที่ 20 เป็นต้นมา มีผู้พยายามวางระบบการออกแบบอุปกรณ์เช่นนี้ขึ้น ดังจะเห็นได้จากงานของ Cole (1946), Doyle (1950) และ Henriksen (1973) เป็นต้น ต่อมาเมื่อมีการประยุกต์ใช้คอมพิวเตอร์กับงานทางด้านวิศวกรรมอย่างแพร่หลาย “Programmable Conformable Clamps” (1982) จึงเกิดขึ้น หลังจากนั้นงานวิจัยทางด้านการออกแบบอุปกรณ์ยึดชิ้นงาน หรือ จิกและฟิกซ์เจอร์ก็ถูกพัฒนาอย่างต่อเนื่อง

ในบทนี้กล่าวถึงการใช้คอมพิวเตอร์ช่วยออกแบบอุปกรณ์ยึดชิ้นงานโดยสังเขป ซึ่งจะแยกกล่าวเป็น 3 ส่วนคือ (1) โปรแกรมคอมพิวเตอร์ที่เกี่ยวข้องกับการสร้างรูปแบบทั่วไปของอุปกรณ์ยึด ซึ่งรวมไปถึง variant และ generative approaches, (2) ความผิดพลาดของตำแหน่งชิ้นงาน และ (3) การวิเคราะห์ความคลาดเคลื่อนในการออกแบบอุปกรณ์ยึดชิ้นงาน

2.1 การใช้คอมพิวเตอร์ช่วยออกแบบอุปกรณ์ยึดชิ้นงาน

การออกแบบจิก และฟิกซ์เจอร์เป็นงานที่ต้องใช้ความชำนาญสูงในการออกแบบ เนื่องจากเป็นงานออกแบบที่ต้องการความเที่ยงตรง แต่ด้วยเทคโนโลยีทางคอมพิวเตอร์ที่พัฒนาขึ้น จึงมีการนำคอมพิวเตอร์ไปประยุกต์ใช้ในงานทางด้านวิศวกรรม ทั้งนี้รวมถึงงานออกแบบจิก และฟิกซ์เจอร์ด้วยเช่นกัน การประยุกต์ใช้โปรแกรมคอมพิวเตอร์ช่วยออกแบบฟิกซ์เจอร์ ซึ่งส่วนใหญ่ยังเป็นเพียงการออกแบบในเบื้องต้น เช่นการใช้คอมพิวเตอร์ช่วยสร้างโมเดลของแข็ง (solid models) ตัวอย่างเช่น โปรแกรม SolidWorks หรือ Autodesk Inventor เป็นต้น โปรแกรมออกแบบเช่นนี้ ยังมีการพัฒนาต่อเนื่องซึ่งมีงานวิจัยในด้านนี้ออกมาให้เห็นเป็นจำนวนมาก เช่น Nee และคณะ (1995) พยายามพัฒนาโปรแกรมคอมพิวเตอร์ช่วยออกแบบในแนวทางการผสมผสานของโปรแกรม 2 รูปแบบคือ ระหว่างโปรแกรมชนิดดัดแปลง และแกรมชนิดสร้าง โดยใช้ส่วนประกอบหลักคือ; (1) ฐานข้อมูลเกี่ยวกับชิ้นงาน, (2) ฐานข้อมูลเกี่ยวกับฟิกซ์เจอร์ และ (3) ส่วนการสร้างและปรับแต่งฟิกซ์เจอร์ให้เหมาะสมกับชิ้นงาน นอกจากนั้นแล้ว Vukelic (2011) ใช้เทคนิคในแนวทางเดียวกับ Nee ร่วมกับการใช้หลักปัญญาประดิษฐ์ (Artificial Intelligence, AI) โดยอาศัยข้อมูลจากรายละเอียดของชิ้นงาน กระบวนการผลิต และข้อจำกัดการใช้งาน ร่วมกับระบบฐานข้อมูล ผลลัพธ์

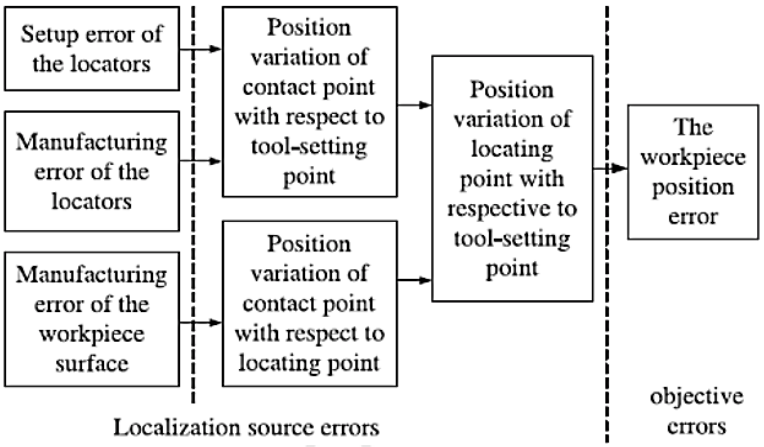
ของโปรแกรมออกมาในรูปโมเดลของแข็ง และแบบทางวิศวกรรม แต่อย่างไรก็ตามโปรแกรมส่วนใหญ่ที่มีอยู่ยังไม่สามารถกำหนดขนาด และความคลาดเคลื่อนของมิติได้ ซึ่งการพิจารณาขนาด และความคลาดเคลื่อนที่เหมาะสมให้กับฟีกซ์เจอร์ถือเป็นเรื่องสำคัญยิ่งต่อการออกแบบฟีกซ์เจอร์

2.2 ความผิดพลาดในการกำหนดตำแหน่งชิ้นงาน

การออกแบบชิ้นส่วนในงานประกอบต้องให้ความสำคัญกับการกำหนดค่าความคลาดเคลื่อนให้กับชิ้นส่วนแต่ละชิ้นในขั้นตอนออกแบบ เพราะหากให้ความคลาดเคลื่อนได้ไม่เหมาะสม จะทำให้เกิดปัญหาความคลาดเคลื่อนสะสม (tolerance stacks) และมีผลกระทบต่อความต้องการใช้งานของชุดงานประกอบ ซึ่งนำไปสู่ความเสียหายในการผลิต ดังนั้นในการออกแบบผู้ออกแบบจึงต้องทราบว่าส่วนประกอบใดของงานประกอบที่มีผล หรือไม่มีผลต่อการเปลี่ยนแปลงของความต้องการใช้งาน (functional dimensions หรือ non – functional dimensions) จิกก์เป็นงานประกอบเช่นกัน การพิจารณาส่วนนี้จึงมีความสำคัญ Qin และคณะ (2006) ได้ทำการศึกษาความสัมพันธ์ที่ก่อให้เกิดความผิดพลาดเนื่องจากการติดตั้งตัวกำหนดตำแหน่ง, ความผิดพลาดของการผลิตตัวกำหนดตำแหน่ง และความบกพร่องของผิวชิ้นงานเนื่องจากกระบวนการผลิต ได้พบว่าค่าความผิดพลาดที่เกิดขึ้นบนชิ้นงานภายหลังจากการแปรรูปโดยใช้อุปกรณ์ช่วยจับยึด มีความสัมพันธ์กับความผิดพลาดทั้ง 3 ชนิด ดังในแผนภาพของรูปที่ 2.1 คือ (1) ความผิดพลาดของการประกอบและติดตั้งตัวกำหนดตำแหน่ง (2) ความผิดพลาดของขนาดตัวกำหนดตำแหน่งเนื่องจากกระบวนการผลิต และ (3) ความผิดพลาดของขนาดและรูปทรงเรขาคณิตของชิ้นงาน ซึ่งความผิดพลาด 2 ชนิดแรก ทำให้เกิดความผันแปรของระยะระหว่างจุดสัมผัสของตัวกำหนดตำแหน่งกับดอกสว่านหรือคัตเตอร์ ส่วนความผิดพลาดในขนาด รูปร่าง และผิวชิ้นงาน ทำให้ผิวชิ้นงานแต่ละชิ้นสัมผัสกับผิวตัวกำหนดตำแหน่งแตกต่างกัน นั่นคือเกิดความผิดพลาดในตำแหน่งที่ตั้งของชิ้นงานโดยรวมในแง่ที่สัมพันธ์กับเครื่องมือตัด

โมเดลของ Qin และคณะ ไม่ได้ให้รายละเอียดเกี่ยวกับการประกอบกันของแต่ละชิ้นส่วนมากนัก แต่ก็แสดงให้เห็นว่า การประกอบกันของชิ้นส่วนต่าง ๆ มีผลต่อตำแหน่งชิ้นงาน ซึ่งมีผลต่อตำแหน่งรู หรือรูปลักษณ์ที่ถูกตัดด้วยดอกสว่าน หรือคัตเตอร์โดยตรง ดังจะเห็นจากโมเดลที่มาจากการพิจารณาเวกเตอร์ความสัมพันธ์ของพิกัดกำหนดทั่วไปที่จุดใดๆ (global coordinate system), พิกัดกำหนดของชิ้นงาน (workpiece coordinate system), และพิกัดกำหนดของตัวกำหนดตำแหน่ง (locator coordinate system) นอกจากแนวความคิดของ Qin แล้ว ยังมีการใช้ V – Block เป็นตัวกำหนดตำแหน่งชิ้นงานที่มีผิวโค้ง หรือทรงกระบอก เพื่อบังคับแนวเส้นผ่านศูนย์กลางให้อยู่กึ่งกลางระหว่างแขนของ V – Block แต่ไม่สามารถควบคุมระดับความสูงของเส้นผ่านศูนย์กลาง

หน้าตัดทุกชิ้นงานให้คงที่ได้ เพราะระดับความสูงของเส้นผ่านศูนย์กลางของหน้าตัดขึ้นกับขนาดเส้นผ่านศูนย์กลางของชิ้นงาน (Henriksen, 1973)



รูปที่ 2.1 องค์ประกอบจิก ฟิกซ์เจอร์ที่ก่อให้เกิดความผิดพลาด (error)

2.3 ความคลาดเคลื่อนของชิ้นงานประกอบ (assembly part components)

ปัญหาความคลาดเคลื่อนสะสม (tolerance stacks) เป็นปัญหาที่เกิดกับงานประกอบอยู่เสมอ ไม่เว้นแม้กระทั่งในอุปกรณ์ยึดชิ้นงานอย่างจิก และฟิกซ์เจอร์ จึงต้องอาศัยวิธีการวิเคราะห์ หรือการสังเคราะห์ความคลาดเคลื่อน (tolerance analysis or tolerance synthesis) กับงานประกอบ ซึ่งวิธีการทั้งสองเป็นหลักสำหรับกระจายความคลาดเคลื่อน (tolerance allocation) ที่มีความแตกต่างกันคือ วิธี tolerance analysis เป็นการสร้างความสัมพันธ์ในความคลาดเคลื่อนของมิติของชิ้นส่วนแต่ละชิ้นที่มีต่อหน้าที่ใช้งานของชุดงานประกอบซึ่งเรียกว่า ความต้องการใช้งาน (functional requirement) และใช้ความสัมพันธ์ดังกล่าวเป็นหลักของการกระจายค่าความคลาดเคลื่อน ให้กับแต่ละมิติที่เกี่ยวข้อง ส่วน tolerance synthesis เป็นวิธีการกำหนดความคลาดเคลื่อนให้กับมิติของชิ้นงานแต่ละชิ้นโดยไม่ได้คำนึงถึงความสัมพันธ์ในความคลาดเคลื่อนระหว่างมิติของชิ้นงาน และคำนวณค่าความคลาดเคลื่อนสะสมภายหลัง หรือใช้วิธีวัดตรวจสอบความต้องการใช้งานโดยตรง ในด้านความพยายามในการประยุกต์ใช้เทคนิคการวิเคราะห์ความคลาดเคลื่อนกับการออกแบบฟิกซ์เจอร์ที่เห็นได้ชัดเจนคือ งานของ Choudhuri และ De Meter (1999) ที่ใช้วิธี vector analysis โดยการกำหนดเวกเตอร์ของตัวกำหนดตำแหน่งที่จุดสัมผัสบนชิ้นงาน และสร้างสมการของเวกเตอร์ที่เป็นผลจากการเปลี่ยนแปลงของเวกเตอร์เมื่อตัวกำหนดตำแหน่งเองมีการเปลี่ยนแปลงไป นอกจากนั้นแล้วความเปลี่ยนแปลงทางเรขาคณิตของชิ้นงานก็มีผลต่อตัวกำหนดตำแหน่งจึงถูกนำมามีผลกับเวกเตอร์ของตัวกำหนดตำแหน่ง

2.4 สรุป

แนวทางการใช้คอมพิวเตอร์ช่วยในการออกแบบอุปกรณ์ยึดชิ้นงานในปัจจุบันมี 2 แนวทาง หลักคือ โปรแกรมชนิดดัดแปลง และชนิดสร้าง (variant and generative approaches) ซึ่งเป็นรูปแบบที่แพร่หลายในโปรแกรมออกแบบโมเดลของแข็งสำหรับจิก และฟิกซ์เจอร์ และสามารถนำมาเป็นแนวทางสำหรับประยุกต์ใช้กับโปรแกรมคอมพิวเตอร์ช่วยออกแบบได้อีกหลากหลาย แต่สำหรับการสร้างโปรแกรมออกแบบที่สามารถระบุขนาด และความคลาดเคลื่อน ต้องให้ความสำคัญกับการพิจารณาอิทธิพลของการประกอบกันของชิ้นส่วนต่าง ๆ ที่มีผลต่อตำแหน่งชิ้นงานและส่งผลโดยตรงต่อความผิดพลาดที่เกิดขึ้นกับตำแหน่งรูที่ถูกเจาะด้วยดอกสว่านร่องบิด (twist drill)

Cole, C. B. (1946). *Tool Design*: American Technical Society.

Cutkosky, M. R., Kurokawa, E., & Wright, P. K. (1982). PROGRAMMABLE CONFORMABLE CLAMPS. *Technical Paper - Society of Manufacturing Engineers. MS*, 8p.

Doyle, L. E. (1950). *Tool engineering: analysis and procedure*: Prentice-Hall.

Henriksen, E. K. (1973). *Jig and Fixture Design Manual*: Industrial Press.

Nee, A. Y. C., Whybrew, K., & kumar, A. S. (1995). *Advanced Fixture Design for FMS*: Springer-Verlag.

Qin, G. H., Zhang, W. H., Wan, M., & Qin, G. H. (2006). A mathematical approach to analysis and optimal design of a fixture locating scheme. *The International Journal of Advanced Manufacturing Technology, 29*(3-4), 349-359. doi: 10.1007/s00170-005-2509-0

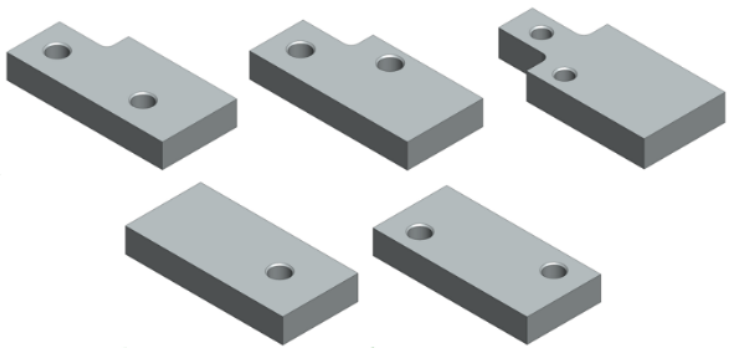
บทที่ 3
โปรแกรม CADBP

โปรแกรมคอมพิวเตอร์ช่วยกำหนดตำแหน่งบุชพัฒนาขึ้นในที่นี้ใช้ชื่อโปรแกรมว่า "Computer – Aided Determination of Bush Position" และใช้ตัวย่อว่า "CADBP" (อ่าน CAD – B – P) ซึ่งเขียนคำสั่งในการทำงานด้วยภาษาไมโครซอร์ฟวิชวลเบสิค ในบทนี้กล่าวถึงภาพรวมของโปรแกรม CADBP, ลักษณะทั่วไปของชิ้นงาน, ชนิดของตัวกำหนดตำแหน่ง, และโครงสร้างของโปรแกรม

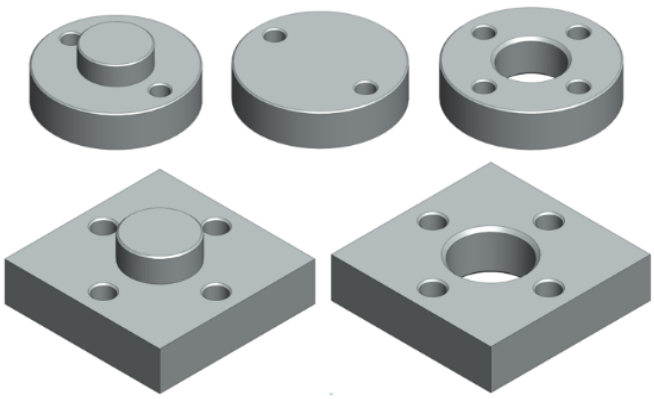
CADBP เป็นความพยายามที่จะใช้คอมพิวเตอร์ช่วยออกแบบชิ้นงานอย่างเป็นรูปธรรมในทางปฏิบัติ ซึ่งเป็นงานที่มีผู้ศึกษาในขอบเขตที่จำกัด โดยเฉพาะอย่างยิ่งการใช้เทคนิคของการวิเคราะห์ความคลาดเคลื่อนของชุดงานประกอบด้วยแล้ว มีการศึกษาที่น้อยมาก ดังนั้นอาจกล่าวได้ว่า CADBP เป็นงานทดลองช่วงแรก ๆ ของการประยุกต์ใช้เทคนิคของงานประกอบกับอุปกรณ์ยึดชิ้นงาน เช่น จิกสำหรับเจาะรู

3.1 ลักษณะของชิ้นงานและตัวกำหนดตำแหน่งสำหรับ CADBP (forms of machined parts and locators)

โปรแกรม CADBP สามารถใช้ได้กับการออกแบบ plate jig, และ template jig บางชนิด สำหรับชิ้นงานที่มีลักษณะเป็นแผ่นดังในรูป 3.1 และ 3.2

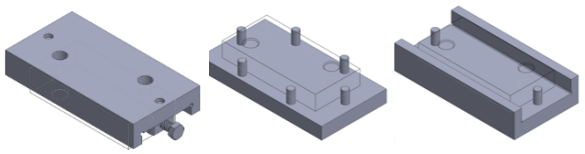


รูปที่ 3.1 แผ่นชิ้นงานมีรูปลักษณ์กำหนดตำแหน่งเป็นผิวระนาบ



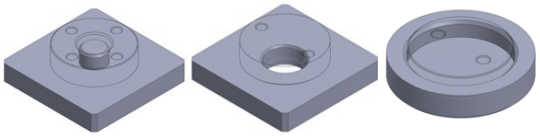
รูปที่ 3.2 แผ่นชิ้นงานมีรูปลักษณ์กำหนดตำแหน่งเป็นรูหรือปุ่มทรงกระบอก

จิกที่ได้รับการออกแบบสำหรับชิ้นงานในรูป 3.1 อาจแบบที่ใช้ตัวกำหนดตำแหน่งซึ่งสัมผัสชิ้นงานตลอดเวลา (fixed – contact type locator) (รูป 3.3 ก) หรือ ใช้ตัวกำหนดตำแหน่งแบบลอยตัวหรืออาจสัมผัสกับผิวงานเพียงบางจุด (floating-type or alternate locator) ซึ่งอาจเป็นแบบรังที่เกิดจากหมุดล้อมรอบชิ้นงาน (nesting pins) (รูป 3.3 ข) หรือแบบหมุดผสมร่องกำหนดตำแหน่ง (รูป 3.3 ค) ในกรณีของ floating – type locators เช่นนี้ ถือว่า ไม่มีแรงกดชิ้นงานให้ติดกับตัวกำหนดตำแหน่ง ชิ้นงานอาจเบี่ยงเบนไปจากตำแหน่งที่ถูกต้องบ้างภายในขอบเขตของตำแหน่งและขนาดของตัวกำหนดตำแหน่งและรูปลักษณ์กำหนดตำแหน่งของชิ้นงาน สำหรับจำนวนรูเจาะในชิ้นงานในเบื้องต้นนี้ กำหนดไว้ไม่เกิน 5 รูสำหรับแผ่นชิ้นงานมีรูปลักษณ์กำหนดตำแหน่งเป็นผิวระนาบและ 4 รูสำหรับแผ่นชิ้นงานมีรูปลักษณ์กำหนดตำแหน่งเป็นรูหรือปุ่มทรงกระบอก โดยมีรูใดรูหนึ่งอ้างอิงตำแหน่งจากผิวงาน ส่วนรูอื่นๆ อ้างอิงตำแหน่งซึ่งกันและกัน



(ก) (ข) (ค)

รูปที่ 3.3 ชนิดของตัวกำหนดตำแหน่ง



(ก) (ข) (ค)

รูปที่ 3.4 ปุ่ม, รูและหลุมกำหนดตำแหน่ง

รูปที่ 3.2 แสดงลักษณะของชิ้นงานที่เหมาะกับ template jig และสามารถหาตำแหน่งบุชได้ด้วย CADBP ชิ้นงานเหล่านี้ต้องการตัวกำหนดตำแหน่งแบบสลับผิวสัมผัสชิ้นงาน

(alternate locator) ซึ่งอาจเป็นแบบใดแบบหนึ่งในรูปที่ 3.4 ขึ้นอยู่กับรูปลักษณ์กำหนดตำแหน่งของชิ้นงาน (locating feature)

3.2 โครงสร้างโปรแกรม (structure of CADBP)

โปรแกรม CADBP ประกอบด้วยโปรแกรมย่อยที่ใช้ในการทำงานของระบบซึ่งสามารถแบ่งออกได้เป็น 7 โมดูล ดังนี้

(1) Input module: ทำหน้าที่รับข้อมูลที่ผู้ใช้ป้อนเข้าสู่โปรแกรม หรืออ่านจากไฟล์ข้อมูลและเก็บเข้าสู่ฐานข้อมูลเพื่อใช้ในการประมวลผล

(2) Tolerance calculation and assignment module: ทำหน้าที่คำนวณค่าความผันแปรของตำแหน่งงานในจิก และส่งผลลัพธ์จากการคำนวณเป็นข้อมูลสำหรับการทำงานในโมดูลถัดไป โมดูลนี้ประกอบด้วยโปรแกรมย่อยสำหรับกำหนดค่าความคลาดเคลื่อนของขนาดชิ้นงานหรือตำแหน่งของรูปลักษณ์ในชิ้นงาน เนื่องจากความคลาดเคลื่อนของกระบวนการผลิตชิ้นส่วน

(3) Loop diagram generation module: ทำหน้าที่สร้าง loop diagram โดยใช้เทคนิคการสร้างเส้นทาง (path) ความสัมพันธ์ของมิติ การทำงานของโมดูลอาศัยโปรแกรมย่อย 2 ส่วนที่สำคัญคือ (1) โปรแกรมย่อยที่ทำหน้าที่สร้าง path และ (2) โปรแกรมย่อยที่ทำหน้าที่ในการค้นหา path

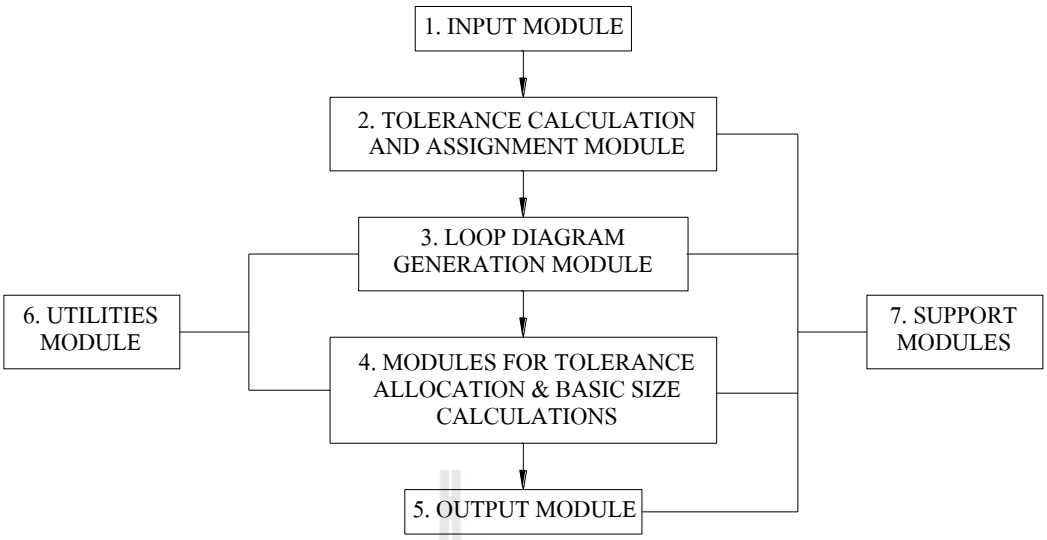
(4) Modules for tolerance allocation and basic size calculations: ทำหน้าที่หาขนาด และความคลาดเคลื่อนให้กับมิติที่ผู้ใช้ต้องการทราบค่า ประกอบด้วยโปรแกรมย่อยสำหรับคำนวณค่าความคลาดเคลื่อน และสำหรับคำนวณค่าขนาดมูลฐาน

(5) Output module: ทำหน้าที่แสดงผลลัพธ์จากการประมวลผลของโปรแกรมทั้งหมดออกทางหน้าจอแสดงผล

(6) Utilities module: โมดูลทั่วไปจะประกอบด้วยโปรแกรมย่อยที่นำไปใช้ร่วมกันหลายโมดูล เช่น โปรแกรมย่อยสำหรับค้นหาเกรดความคลาดเคลื่อนมาตรฐาน (International Tolerance grade – IT number) และค่าเบี่ยงเบนมาตรฐาน (fundamental deviation) ตามข้อกำหนดความฟิตของงานสวม (fits specification) เป็นต้น

(7) Support modules: ทำหน้าที่ควบคุมการทำงานนอกเหนือจากการคำนวณ และประมวลผล เช่น การเปิดไฟล์, บันทึกไฟล์ และการพิมพ์รายงาน เป็นต้น

รูป 3.5 แสดงให้เห็นถึงความเชื่อมโยงในการทำงานระหว่างโมดูลทั้งระบบ CADBP สามารถอ่านข้อมูลจากไฟล์ข้อมูล (data file) ซึ่งผู้ใช้จะต้องจัดเตรียมข้อมูลให้มีฟอร์แมตเดียวกับการป้อนข้อมูลโดยตรง



รูปที่ 3.5 โครงสร้างของ CADBP

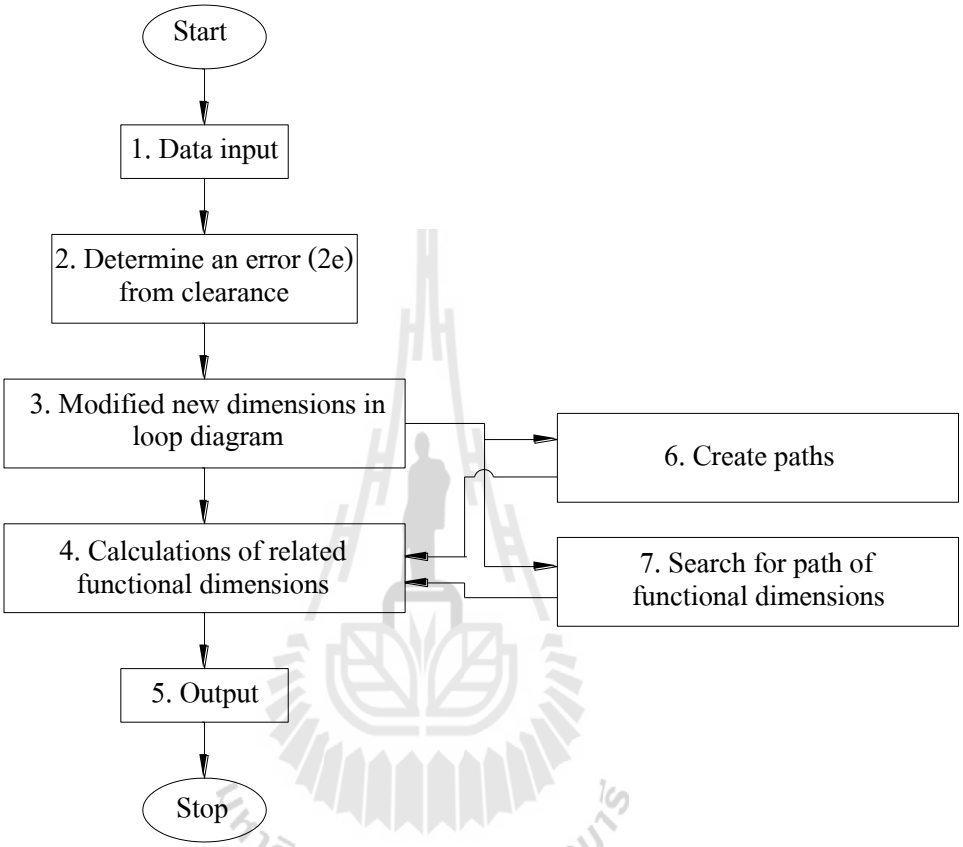
การทำงานโดยสังเขปของโปรแกรมมีขั้นตอนดังนี้: กรอบหมายเลข 1 โปรแกรมจะเริ่มต้นรับข้อมูล จากการโต้ตอบจากผู้ใช้โดยตรง หรือจากไฟล์ข้อมูล ข้อมูลจะถูกบันทึกในรูปแบบไฟล์ Text (.txt) เพื่อให้เรียกใช้ในการประมวลผลภายหลังและสามารถเปิดอ่านได้โดยไม่ต้องผ่าน CADBP

ในขั้นต่อไป ในกรอบหมายเลข 2 โปรแกรมจะเริ่มทำการประมวลผลจากการหาค่าความคลาดเคลื่อนในการประกอบของชิ้นงานและจิก ทั้งนี้ค่าความคลาดเคลื่อนดังกล่าวจะถูกเปลี่ยนแปลงเป็นค่าเบี่ยงเบนของตำแหน่งของชิ้นงานในขณะที่ถูกประกอบอยู่ในจิก ซึ่งจะถูกใช้เป็นส่วนหนึ่งของมิติในแผนภาพวงรอบ โปรแกรมจะทำการคำนวณค่ามิติที่ผู้ใช้ต้องการด้วยวิธีการสร้างสมการวงรอบ ในกรอบหมายเลข 3โดยอาศัยเทคนิคการค้นหาเส้นทาง (path) ย้อนกลับจากโหนดใด ๆ ไปสู่โหนดตั้งต้น (root node) บนแผนภาพต้นไม้ เพื่อให้ทราบมิติที่เป็นมิติใช้งานทั้งหมดที่เกี่ยวข้องกับมิติที่เราต้องการทราบค่า มิติที่เกี่ยวข้องในวงรอบจะถูกนำมาคำนวณค่าขนาดมูลฐาน และกระจายค่าค่าความคลาดเคลื่อนของมิติที่ต้องการทราบค่า (กรอบหมายเลข 4)

ผลลัพธ์จากการคำนวณจะถูกแสดง (กรอบหมายเลข 5)โดยระบุข้อมูลประเภทของจิกที่ออกแบบ ขนาดมูลฐาน และค่าความคลาดเคลื่อน ซึ่งอาจเป็นแบบเบี่ยงเบนด้านเดียว (unilateral tolerancing) สำหรับมิติของส่วนที่เป็นงานประกอบ (mating part) และเป็นแบบเบี่ยงเบนสองด้าน (bilateral tolerancing) สำหรับขนาดของมิติอื่นที่ไม่ใช่ส่วนที่เป็นงานประกอบ นอกจากนั้นยังแสดงขนาดของมิติใช้งานทั้งหมดที่เกี่ยวข้องกับมิติที่ต้องการทราบค่า และหากการคำนวณให้ผลลัพธ์การคำนวณและแสดงผลที่หน้าจอแสดงผลว่า เกิดปัญหาความคลาดเคลื่อนสะสม

เกิดขึ้น ผู้ใช้จะสามารถทราบได้ว่าจะเปลี่ยนแปลงการออกแบบได้จากมิติใดที่เกี่ยวข้องที่แสดงทางผลลัพธ์

รูปที่ 3.6 เป็น flowchart สรุปลำดับการทำงานของ CADBP ดังที่ได้อธิบายมา



รูปที่ 3.6 ภาพรวมผังการทำงานของ CADBP

3.3 สรุป

โปรแกรม CADBP ประกอบด้วยโปรแกรมย่อย 8 โมดูล คือ Input module, Loop diagram generation module, Tolerance calculation and assignment module, Tolerance allocation module, and Basic size calculation module, Output module, Utilities module, และ Support modules

การทำงานของ CADBP เริ่มจากการหาความผันแปรของตำแหน่งงานในขณะการประกอบ, สร้างแผนภาพวงรอบ, คำนวณขนาดและค่าความคลาดเคลื่อนของความต้องการใช้งาน และแสดงผล ตามลำดับ ซึ่งรายละเอียดของการทำงานในแต่ละโมดูลจะอยู่ในบทที่ 4

บทที่ 4
โครงสร้างโปรแกรม

ในบทนี้กล่าวถึงรายละเอียดของโมดูล หรือโปรแกรมย่อย (sub program) ที่มีความสำคัญในการทำงานของ CADBP และอธิบายตัวแปรต่าง ๆ ที่ใช้ในการเขียนคำสั่ง รวมไปถึงขั้นตอนวิธีการ (algorithm) และผังการทำงาน (flow chart) ของโปรแกรมย่อยทั้งหมดที่กล่าวไว้ในบทที่ 3

4.1 ชนิด และหน้าที่ของตัวแปร

ตัวแปรที่ใช้ในการเขียนโปรแกรม CADBP ประกอบด้วยตัวแปรที่ประกาศใช้เป็นแบบ Public variables และ Local variables. ตัวแปรแบบ Public ใช้กับตัวแปรหลัก ๆ ที่ทำหน้าที่เก็บค่าหรือข้อมูลที่ถูกนำไปใช้ร่วมกันในหลายโปรแกรมย่อย หรือหลายโมดูล ซึ่งรายละเอียดของตัวแปรแบบ Public แสดงในตารางที่ 4.1 ส่วนตัวแปร local variables ใช้ได้เฉพาะในโปรแกรมย่อย ซึ่งจะแสดงรายละเอียดของตัวแปรที่สำคัญสำหรับแต่ละโมดุลในภายหลัง

ในเรื่องของชนิดข้อมูลสำหรับโปรแกรมนี้ นอกจากจะใช้ string, integer, และ single แล้ว, CADBP ยังใช้ Dimension ซึ่งเป็นฐานข้อมูลเชิงโครงสร้าง (data structure) ชนิดหนึ่งที่รวมเอาข้อมูลพื้นฐานหลายชนิดมาประกอบรวมกันให้รองรับการทำงานในลักษณะของชุดข้อมูล ที่ในแต่ละชุดประกอบด้วยข้อมูลหลายประเภท หรือเรียกว่าหลายฟิลด์ (field) โดยที่แต่ละฟิลด์อาจเป็น String, Integer, หรือ Single เป็นต้น ในตารางที่ 4.2 แสดงรายการของฟิลด์ที่อยู่ในโครงสร้างข้อมูลของ inputx( ) ซึ่งเป็นตัวแปรชนิด Dimension และเป็น public variable

ตารางที่ 4.1 แสดงข้อมูลของตัวแปรสำคัญแบบ Public variables ที่ใช้ในโปรแกรม

ชื่อตัวแปร	**ชนิด**	**ขนาด**	**ชนิดของข้อมูล**	**หน้าที่ของตัวแปร**
inputx( )	Dimension	ขึ้นอยู่กับจำนวนมิติ ($\leq$ 20 elements)	หลายชนิด	เก็บข้อมูลของแต่ละมิติจากข้อมูลรับเข้า
Req( )	Array 1 มิติ	2 elements	String	เป็นตัวแปรสำหรับเก็บผิวของมิติที่ผู้ใช้ต้องการทราบค่าจากโปรแกรม
FReq( )	Array 1 มิติ	2 elements	String	เป็นตัวแปรสำหรับเก็บผิวของมิติที่เป็นความต้องการใช้จากแบบของงาน (ที่ต้องการออกแบบจิก)
NumFReq	Integer	$\leq$ 5	Integer	เป็นตัวแปรสำหรับเก็บจำนวนรูเจาะของแบบงาน
Basic_result	Single	$\leq$ 999.11111	Single (real)	เป็นตัวแปรสำหรับเก็บค่าผลลัพธ์ของขนาดมูลฐานจากการคำนวณ
Tol_result	Single	$\leq$ 999.11111	Single (real)	เป็นตัวแปรสำหรับเก็บค่าผลลัพธ์ของค่าความคลาดเคลื่อนจากการคำนวณ
Po( )	Array 1 มิติ	ขึ้นอยู่กับจำนวนมิติ ($\leq$ 20 elements)	Single	เป็นตัวแปรสำหรับเก็บตำแหน่งของโหนดซึ่งแทนผิวหรือรูปลักษณ์อ้างอิงการบอกเดตัมของมิติ

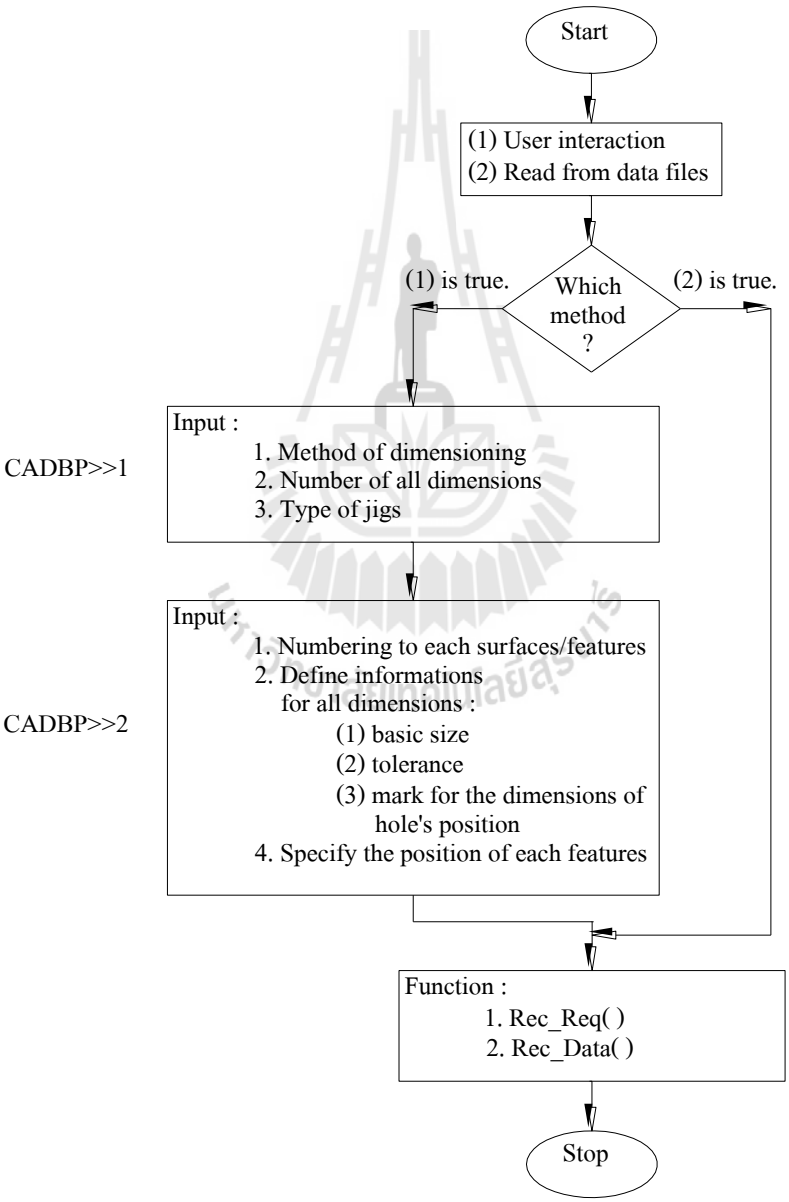
ตารางที่ 4.2 แสดงองค์ประกอบของตัวแปร inputx( )

ชื่อฟิลด์	ขนาด	ชนิดของข้อมูล	หน้าที่ของฟิลด์
F	1 ตัวอักษร	String	เป็นตัวแปรสำหรับเก็บผิวข้างหนึ่งของมิติความยาวหนึ่ง
T	1 ตัวอักษร	String	เป็นตัวแปรสำหรับเก็บผิวอีกข้างหนึ่งของมิติความยาวที่ฟิลด์ F อ้างถึง
PosF	$\leq$ 448	Integer	เป็นตัวแปรสำหรับเก็บตำแหน่งของโหนดเริ่มต้นที่เป็นเดตัมอ้างอิงในการบอกมิติ
PosT	$\leq$ 448	Integer	เป็นตัวแปรสำหรับเก็บตำแหน่งของโหนดสุดท้ายที่เป็นเดตัมอ้างอิงในการบอกมิติ
Dims	$\leq$ 999.11111	Single (real)	เป็นตัวแปรสำหรับเก็บขนาดมูลฐานของมิติรับเข้า
Dimtol	$\leq$ 999.11111	Single (real)	เป็นตัวแปรสำหรับเก็บค่าความคลาดเคลื่อนของมิติรับเข้า
Signcal	-	Integer	เป็นดัชนี (indicating variable or index) ตำแหน่งการวางเดตัมอ้างอิงการบอกมิติรับเข้า โดยมีค่าเป็น 1 เมื่อ ตำแหน่งโหนดในตัวแปร F อยู่ทางซ้ายมือของโหนดในตัวแปร T และในทางกลับกันค่าจะเป็น -1
Signtol	-	Integer	เป็นดัชนีค่าความคลาดเคลื่อนของมิติที่จะใช้จัดสรรให้กับมิติใช้งาน สำหรับมิติที่เป็นความต้องการใช้งานจะมีค่า Signtol เป็น -1

4.2 โมดูลป้อนข้อมูล (Input module)

โมดูลป้อนข้อมูลมีหน้าที่รับข้อมูลจากผู้ใช้ และเก็บเข้าสู่ฐานข้อมูลตัวแปรต่าง ๆ เพื่อใช้ประมวลผล ขั้นตอนการทำงานของโมดูลรับข้อมูลเป็นไปตามแผนภาพในรูป 4.1

การรับข้อมูลของโปรแกรมทำได้ 2 วิธี คือโดยการโต้ตอบกับผู้ใช้ (user interaction) และโดยการอ่านจากไฟล์ข้อมูลที่ผู้ใช้จัดเตรียมไว้ สำหรับวิธีแรก โปรแกรมจะรับข้อมูลผ่านหน้าจอ 2 หน้าจอ คือ CADBP>>1 และ CADBP>>2



รูปที่ 4.1 ขั้นตอนการรับข้อมูล (data input method)

หน้าจอ CADBP>>1

หน้าจอ CADBP>>1 ใช้รับข้อมูลเข้าสู่โปรแกรมเพื่อให้โปรแกรมสามารถจำแนกลักษณะของจิกที่ต้องการออกแบบตามข้อกำหนด โดยผู้ใช้ต้องทำการระบุข้อมูลดังต่อไปนี้

(1) วิธีการบอกขนาด : ผู้ใช้ต้องทำการเลือกรูปแบบในการบอกขนาดของชิ้นงาน ซึ่งมีให้เลือก 2 รูปแบบ คือ แบบ Face to Face คือการบอกขนาดชิ้นงานอ้างอิงจากผิวระนาบถึงผิวระนาบ เช่น การให้ขนาดงานสี่เหลี่ยมผืนผ้า เป็นต้น และแบบ Diameter คือการบอกขนาดของชิ้นงานเป็นเส้นผ่านศูนย์กลาง เช่น การให้ขนาดงานกลม และระหว่างผิวโค้งหนึ่งถึงอีกผิวหนึ่ง เป็นต้น

(2) จำนวนมิติ

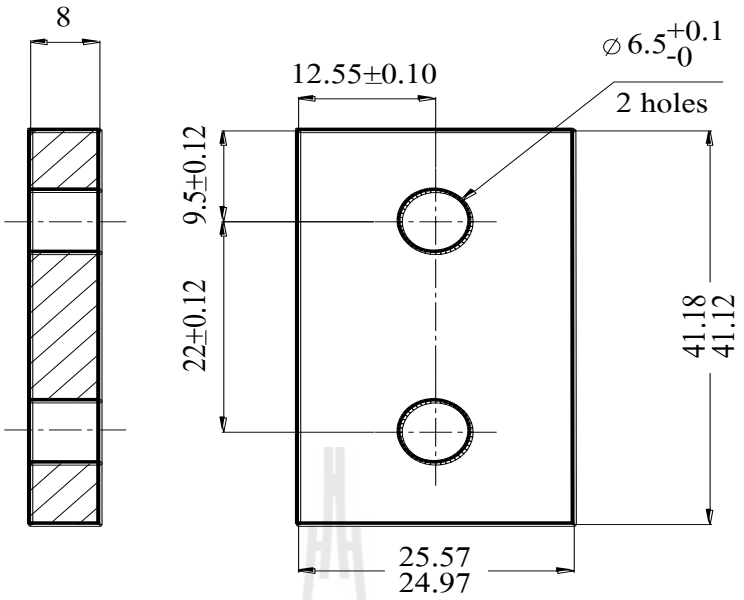
(3) ลักษณะตัวกำหนดตำแหน่ง

เมื่อการระบุวิธีการบอกขนาดในข้อ (1) เป็นแบบ Face to Face จะนำไปสู่ลักษณะของตัวกำหนดตำแหน่งสลับ(alternate locator) ที่โปรแกรมให้ทางเลือกไว้ คือ

- รัง/ราง (slot) : ใช้ผิวระนาบเป็นตัวกำหนดตำแหน่ง
- ใช้หมุดทำรัง/ราง (nesting pin) : ใช้ผิวหมุดเป็นตัวกำหนด

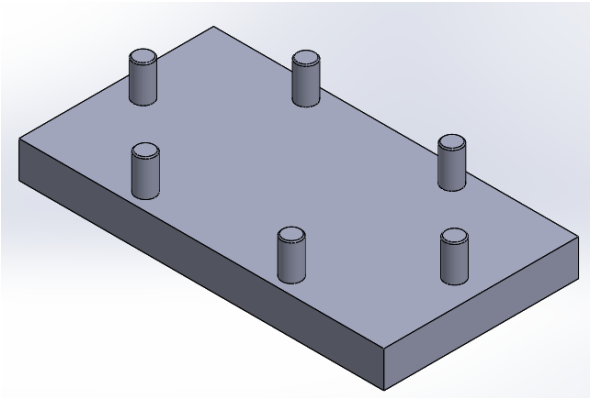
ในกรณีที่การระบุวิธีการบอกขนาด (1) เป็นแบบ Diameter จะสามารถเลือกให้ลักษณะตัวกำหนดตำแหน่งได้ดังต่อไปนี้

- หมุด (pin locator) : ใช้หมุดเป็นตัวกำหนดตำแหน่ง เมื่อชิ้นงานมีรูปลักษณ์กำหนดตำแหน่งเป็นรูเจาะ
- รู (hole locator) : ใช้รูเป็นตัวกำหนดตำแหน่ง เมื่อชิ้นงานมีรูปลักษณ์กำหนดตำแหน่งเป็นปลั๊ก/หมุด (spigot)
- รัง (nesting) : ใช้ผิวโค้งรอบนอกชิ้นงานเป็นผิวกำหนดตำแหน่ง

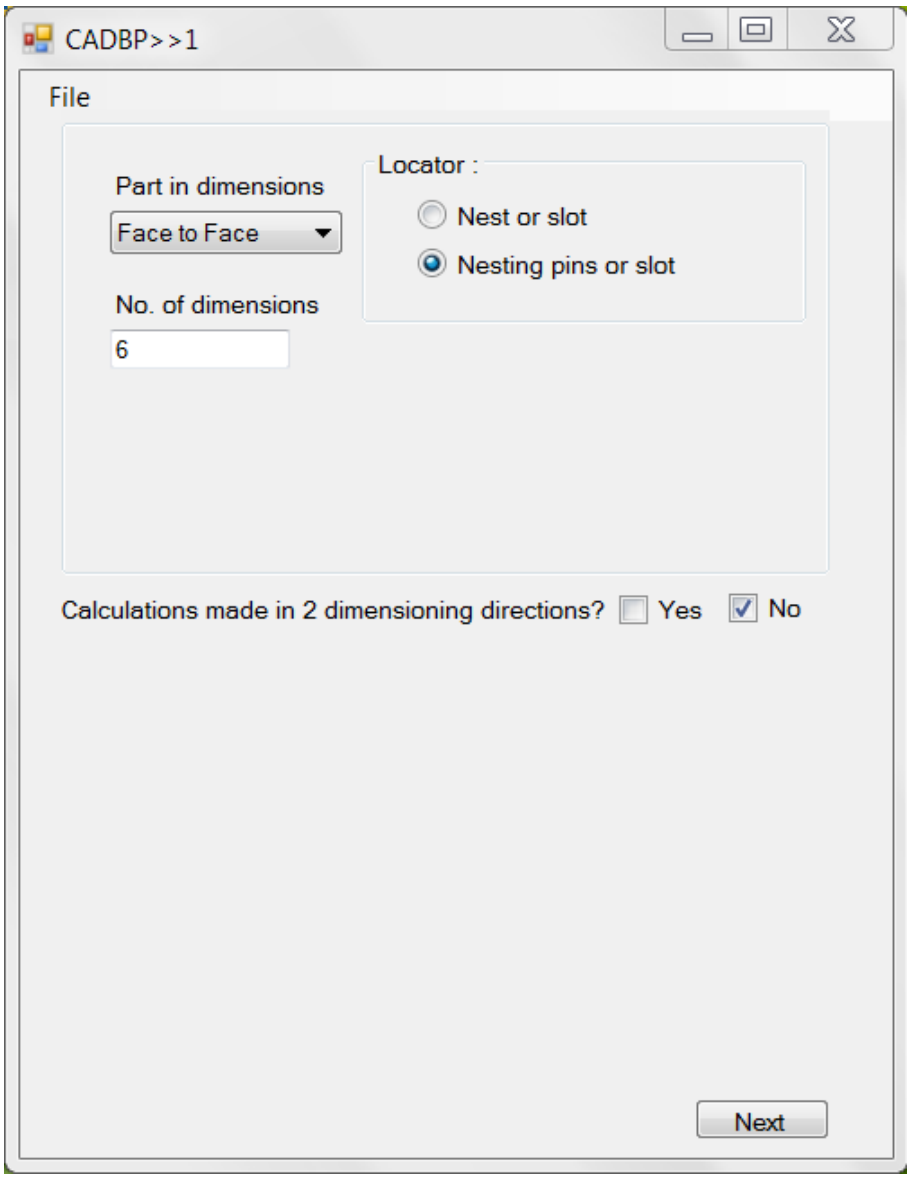


รูปที่ 4.2 ชิ้นงานตัวอย่างสำหรับใช้กับ template jig แบบ nesting pin

ตัวอย่างเช่น สมมุติว่าต้องการป้อนข้อมูลของชิ้นงานในรูป 4.2 เพื่อออกแบบ template jig แบบใช้ nesting pins (ในรูป 4.3) พิจารณาเฉพาะด้านยาวของชิ้นงานในรูป 4.2 เพื่อป้อนข้อมูลลงในหน้าจอ CADBP>>1 จากความต้องการ template jig แบบ nesting pin และด้วยลักษณะของชิ้นงาน ผู้ใช้ต้องทำการเลือกวิธีการบอกขนาดเป็นแบบ Face to Face เนื่องจากการให้ขนาดมิติของชิ้นงานมีเดตัมอ้างอิงมิติที่บอกขนาดชิ้นงานจะอยู่บนผิวระนาบ 2 ผิว และเลือกใช้ตัวกำหนดตำแหน่งแบบใช้หมุดทำรัง/ราง (nesting pin) ซึ่งจะแสดงหน้าจอให้เห็นตามรูปที่ 4.4



รูปที่ 4.3 template jig แบบ nesting pin สำหรับชิ้นงานตามแบบในรูป 4.2



รูปที่ 4.4 หน้าจอ CADBP>>1 สำหรับชิ้นงานตัวอย่างที่ 4.2

<u>หน้าจอ CADBP>>2</u>

ในหน้าจอ CADBP>>2 มีช่องสำหรับป้อนรายละเอียดของมิติ ซึ่งจำนวนชุดของช่องว่างจะขึ้นอยู่กับจำนวนมิติที่ผู้ใช้ระบุในหน้าจอ CADBP>>1 ผู้ใช้จะต้องเตรียมภาพ plan view ที่ต้องการวิเคราะห์ความคลาดเคลื่อน และมีรายละเอียดที่ผู้ใช้ต้องให้ข้อมูลสำหรับแต่ละมิติดังต่อไปนี้

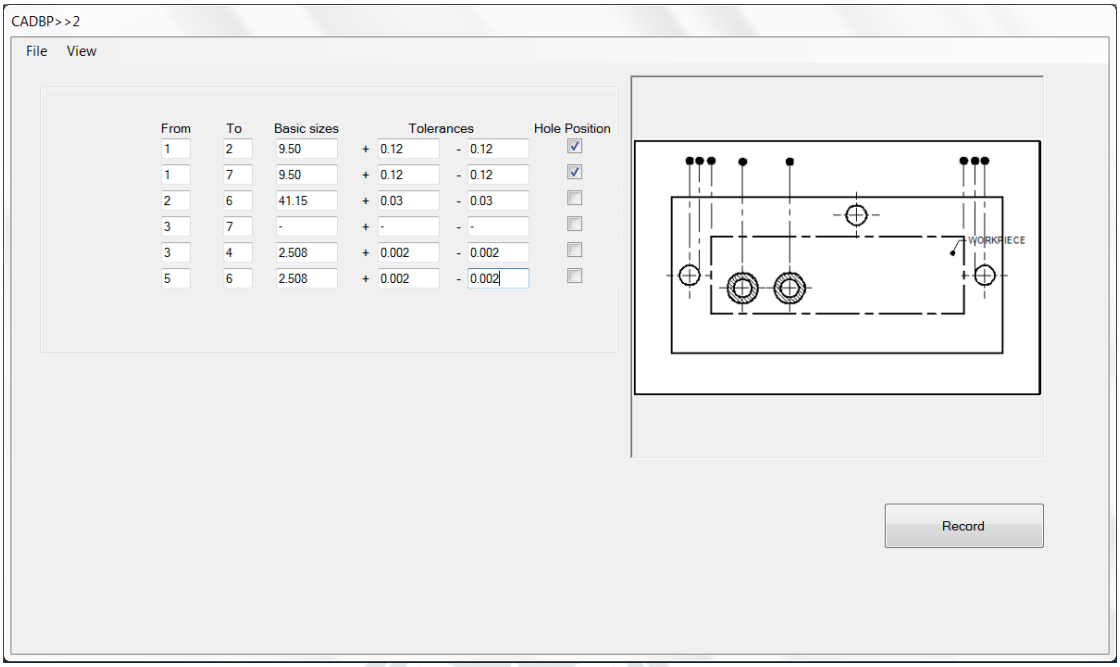
(1) หมายเลขผิวของมิติที่เป็นเดตัมในการบอกขนาด

(2) ขนาดมูลฐานของมิติ

(3) ค่าความคลาเคลื่อนของมิติ

(4) ระบุตำแหน่งด้วยเครื่องหมายถูก ใน checkbox หลังมิติ ที่แสดงตำแหน่งรูเจาะ

สำหรับมิติที่ผู้ใช้ต้องการทราบค่า (requirement) ผู้ใช้ต้องทำการให้ข้อมูลในข้อ 1 โดยในส่วนอื่น ๆ ให้ใส่ขีด, (-) หรือเว้นว่างไว้ โปรแกรมจะทราบได้ทันทีว่าเป็นมิติที่ผู้ใช้ต้องการให้คำนวณหาค่า

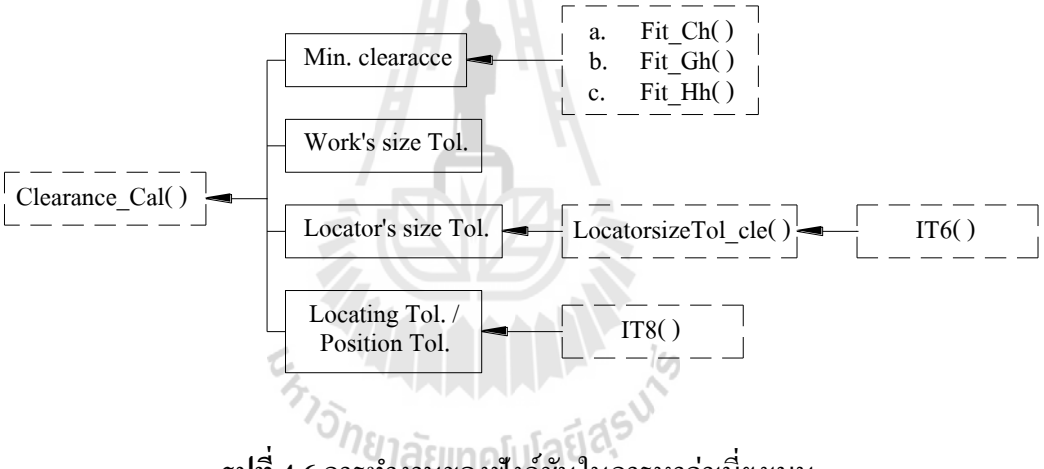


รูปที่ 4.5 หน้าจอ CADBP>>2 สำหรับชิ้นงานตัวอย่างที่ 4.2

จากรูปที่ 4.5 จะเห็นจำนวนช่องว่างสำหรับกรอกรายละเอียดของมิติปรากฏขึ้นให้ผู้ใช้กรอกข้อมูลทั้งหมด 6 ชุด ในที่นี้ตำแหน่งของรูเจาะทั้ง 2 คือมิติด้านบนที่ถูกทำเครื่องหมายในช่อง Hole Position และค่าที่ต้องการคำนวณคือมิติ 1 – 3 หรือในที่นี้กำหนดไว้ก่อนการคำนวณคือระยะจากของหมุดทางด้านซ้ายถึงตำแหน่งบุชสำหรับรูเจาะทางด้านซ้าย ซึ่งระบุเครื่องหมาย “ - ” ในช่องของขนาด และความคลาดเคลื่อน

4.3 โมดูลสำหรับกำหนดค่าความคลาดเคลื่อนให้กับชิ้นส่วนประกอบของจิก (tolerance calculation and assignment module)

โมดูลการหาค่าความคลาดเคลื่อนในการประกอบของชิ้นงานและจิกทำหน้าที่หาค่าสำหรับมิติใหม่ ที่จะนำไปเพิ่มความสัมพันธ์ระหว่างชิ้นงาน และจิก ซึ่งค่าของมิตินี้ขึ้นอยู่กับความผันแปรของขนาดงาน, ความผันแปรของขนาดชิ้นส่วนประกอบจิกอันเนื่องมาจากกระบวนการผลิต, ความผันแปรของตำแหน่งในการประกอบชิ้นส่วนประกอบจิก, ฟิตนิยมที่ผู้ออกแบบกำหนดเพื่อการสวมระหว่างตัวกำหนดตำแหน่ง และรูปลักษณ์กำหนดตำแหน่ง รวมทั้งค่าเผื่อสำหรับการสวมประกอบชิ้นงานเข้ากับจิก (minimum clearance) โปรแกรมจะทำการคำนวณค่าและแสดงผล 3 เงื่อนไขตามความคลาดเคลื่อน IT6 เนื่องจากเป็นเกรดความคลาดเคลื่อนสำหรับงานเจียรนัย และชิ้นส่วนของอุปกรณ์ยึดชิ้นงานส่วนมากจะเจียรนัย ส่วนย่อยต่างๆในการหาค่าความเบี่ยงเบนของตำแหน่งในงานประกอบจิกได้ดังนี้



รูปที่ 4.6 การทำงานของฟังก์ชันในการหาค่าเบี่ยงเบน

รูปที่ 4.6 ฟังก์ชัน Clearance( ) แสดงส่วนของการทำงานย่อย ๆ ของฟังก์ชันซึ่งทำหน้าที่ในการหาค่าช่องว่างสูงสุดในการประกอบชิ้นงานเข้ากับรัง ฟังก์ชันนี้จะส่งกลับค่าของผลรวมค่าความคลาดเคลื่อนที่สะสมจากของ ค่าช่องว่างเผื่อน้อยที่สุดจากการประกอบ (minimum clearance), ค่าความคลาดเคลื่อนของชิ้นงานตามข้อกำหนดของแบบงาน, ค่าความคลาดเคลื่อนของขนาดตัวกำหนดตำแหน่ง และค่าความคลาดเคลื่อนของตำแหน่งตัวกำหนดตำแหน่ง ฟังก์ชันจะเก็บผลลัพธ์ในตัวแปร Clearance ซึ่งเป็นตัวแปรชนิด single

ส่วนของค่าช่องว่างเผื่อน้อยที่สุดจากการประกอบ (minimum clearance) นั้น เกิดจากการค่าช่องว่างน้อยสุดเมื่อคำนวณตามฟิตนิยม ในส่วนนี้จึงมีฟังก์ชั่นเพื่อหาค่าดังกล่าว คือฟังก์ชัน Fit_Ch( ), Fit_Gh( ) และ Fit_Hh( ) ข้อกำหนดความฟิตทั้ง 3 แบบ เป็นแบบสวมคลอนจากช่องว่าง

ระหว่างงานสวมมากไปน้อย (slack fits, medium fits และ normal fits) ตามลำดับ ในกรณีที่โปรแกรมตรวจพบปัญหาความคลาดเคลื่อนสะสม ข้อกำหนดความฟิตที่มีค่าช่องว่างในการสวมน้อยลงจะถูกนำมาใช้คำนวณต่อ หรือในกรณีที่ไม่เกิดปัญหาการคำนวณลักษณะนี้ก็จะสร้างทางเลือกในการออกแบบให้ผู้ผลิตเลือกใช้ชุดผลลัพธ์การออกแบบให้อยู่ในความสามารถของกระบวนการผลิตที่มีอยู่ได้

ในฟังก์ชัน Clearance( ) ต้องใช้ค่าความคลาดเคลื่อนขนาดตัวกำหนดตำแหน่งที่ขึ้นอยู่กับความสามารถของกระบวนการที่ใช้ในการผลิตตัวกำหนดตำแหน่ง โปรแกรมนี้กำหนดเกรดความคลาดเคลื่อนมาตรฐานไว้ที่ IT6 เหตุผลดังที่กล่าวมาแล้วคือ ส่วนใหญ่ชิ้นส่วนของจิกจะเป็นงานเจียรนัย และ IT6 เป็นค่าความคลาดเคลื่อนกลาง ๆ สำหรับงานเจียรนัย

และในส่วนสุดท้ายของค่ามิติจากฟังก์ชัน Clearance( ) คือ ความคลาดเคลื่อนในตำแหน่งของตัวกำหนดตำแหน่ง ซึ่งอาศัยกระบวนการผลิตทางจักรกล เช่น การเจาะ เพื่ออัดยึดตัวกำหนดตำแหน่งกับฐานจิก ในส่วนนี้จึงใช้ฟังก์ชัน IT8( ) เข้ามาเกี่ยวข้องในการหาค่าความคลาดเคลื่อนจากกระบวนการผลิตแทนค่าความคลาดเคลื่อนตำแหน่งของตัวกำหนดตำแหน่ง

อันที่จริง IT ต่าง ๆ เป็นเกรดของความคลาดเคลื่อนของขนาด (size tolerance) ไม่ใช่เกรดความคลาดเคลื่อนของตำแหน่ง (position tolerance) ซึ่งขนาดของความคลาดเคลื่อนในชนิดหลังนี้ขึ้นอยู่กับความสามารถของเครื่องมือ คนงาน และเครื่องมือวัด และเนื่องจากข้อมูลชนิดนี้มีอยู่จำกัด ดังนั้นในที่นี้จึงใช้ IT8 โดยอนุโลม เนื่องจากงานสวม (mating parts) ใช้ IT6 ถึง IT12 ดังนั้นเกรดกลาง ๆ คือ IT8 – IT9 เพราะสามารถวัดได้ด้วยเครื่องมือวัดทั่ว ๆ ไป เช่น vernier และ micrometer

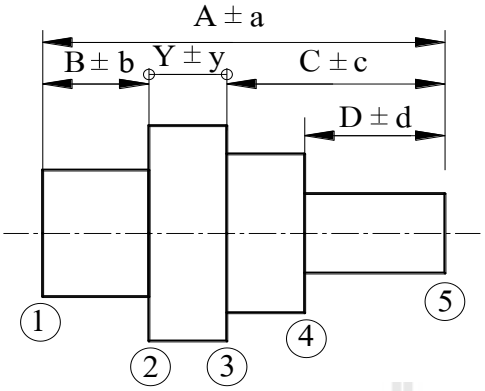
เมื่อโปรแกรมดำเนินการจนครบทุกขั้นตอน ค่าช่องว่างสะสมในตัวแปร Clearance จะถูกส่งกลับไปใช้เป็นมิติเพิ่มจากที่รับข้อมูลจากผู้ใช้ และใช้ในการดำเนินโปรแกรมในส่วนอื่นๆ

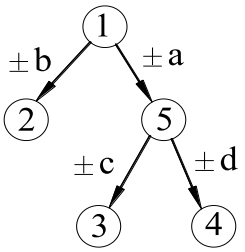
4.4 โมดูลสร้าง loop equation (loop diagram generation module)

ประกอบด้วย (1) โมดูลสำหรับสร้างเส้นทางของมิติ, (2) โมดูลสำหรับค้นหาเส้นทางของมิติที่เกี่ยวข้อง และ (3) โมดูลสำหรับเทียบเส้นทางเพื่อสร้างสมการวงรอบ

4.4.1 โมดูลสำหรับสร้างเส้นทางของมิติ (path construct module)

เส้นทางของผิวมิติ (path of dimensional faces) หมายถึง เส้นทางจากผิวใดผิวหนึ่งของมิติใดมิติหนึ่งไปยังผิวใดผิวหนึ่งที่กำหนดว่าเป็นจุดเริ่มต้นใน loop diagram, ถ้าเราทราบเส้นทางชนิดนี้จากผิวคู่ functional requirement เราสามารถจะหาเส้นทางระหว่างผิวของ functional requirement ได้ นั่นคือได้เส้นทางที่ประกอบด้วยผิวของ functional dimensions





รูปที่ 4.7 ชิ้นงานกลึง 4 มิติ **รูปที่ 4.8** แผนภาพต้นไม้ของชิ้นงานกลึง 4 มิติ

i	Path(i)
1	1
2	2 1
3	3 5 1
4	4 5 1
5	5 1

รูปที่ 4.9 เส้นทาง (path) ความสัมพันธ์มิติของเพลา

โมดูลการสร้างเส้นทาง (path) ของมิติ เป็นโปรแกรมย่อยที่ทำหน้าที่สร้างเส้นทางของผิวมิติที่ผู้ใช้ระบุ ซึ่งทุกรูปลักษณ์อ้างอิงขนาดของแต่ละมิติ จะมี path เฉพาะจากรูปลักษณ์นั้นไปหารูปลักษณ์ หรือผิวอ้างอิง และแต่ละ path จะถูกเก็บในตัวแปร Path( ) อาร์เรย์ 1 มิติชนิด String โดยมีความยาวเท่ากับจำนวนรูปลักษณ์ หรือผิวอ้างอิงของมิติ

ตารางที่ 4.3 แสดงข้อมูลของตัวแปรแบบ local variable ที่ใช้ในโมดูลสำหรับสร้างเส้นทางของมิติ

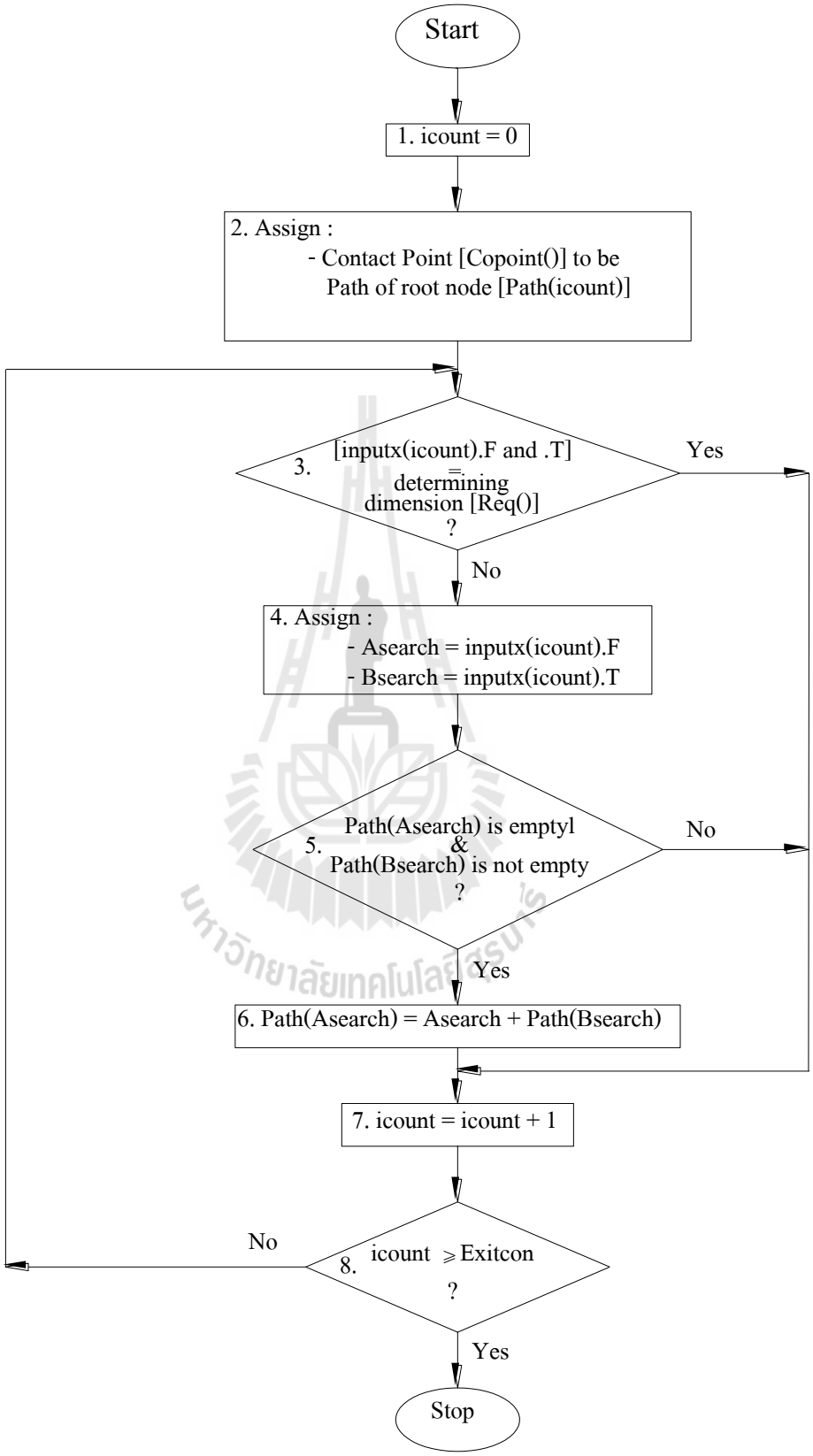
ชื่อตัวแปร	ชนิด	ขนาด	ชนิดของข้อมูล	หน้าที่ของตัวแปร
Path( )	Array 1 มิติ	ขึ้นอยู่กับจำนวนมิติ ($\leq$ 20 elements)	String	เก็บหมายเลขโหนดของแต่ละ path
Exitcon	Integer	$\leq$ 20	Integer	จำนวนโหนดหรือผิวอ้างอิงของมิติ ซึ่งมีค่าขึ้นอยู่กับประเภทของจิก เป็นตัวแปรสำหรับควบคุมรอบการวนลูปในการสร้าง path ตาม
Copoint( )	Array 1 มิติ	2 elements	Integer	เป็นตัวแปรบ่งชี้ตำแหน่งโหนดซึ่งเป็นจุดสัมผัสระหว่างงานและจิกแต่ละประเภท
Asearch	String	$\leq$ 20 ตัวอักษร	String	ตัวแปรสำหรับเก็บผิวข้างหนึ่งของมิติมิติสำหรับใช้ในการเทียบ path
Bsearch	String	$\leq$ 20 ตัวอักษร	String	ตัวแปรสำหรับเก็บผิวอีกข้างของมิติสำหรับใช้ในการเทียบ path

ในที่นี้จะใช้รูปที่ 4.7 ของชิ้นงานกลึง (มีรายละเอียดดังอธิบายในบทที่ 2) เป็นตัวอย่างเพื่อแสดงเทคนิคการสร้าง path ชิ้นงานนี้ประกอบด้วยผิวอ้างอิงของการบอกขนาด (ผิวในแนวดิ่ง) ทั้งหมด 5 ผิว และมีการระบุขนาดของมิติ 4 มิติคือ A, B, C, และ D ความสัมพันธ์ของทั้ง 4 มิติสามารถเขียนในรูปแผนภาพต้นไม้ในรูปที่ 4.8 ซึ่งมีผิวอ้างอิงของแต่ละมิติ แสดงเป็นโหนด (node) ของแผนภาพ หน้าที่ของโมดูลนี้ก็คือ สร้างเส้นทาง (path) จากแต่ละโหนดไปยังโหนดใดโหนดหนึ่ง ที่เป็นโหนดเริ่มต้นของแผนภาพ (root node) ตัวอย่างเช่น สำหรับชิ้นงานนี้ถ้าผู้ใช้โปรแกรมระบุให้โหนดหมายเลข 1 เป็น root node โมดูลนี้จะทำการค้นหา path จากโหนดอื่น ๆ ไปยัง root node นี้ จนกว่าจะได้ path ทั้งหมดครบตามตารางในรูปที่ 4.9

สำหรับการสร้าง path มีขั้นตอนการทำงานตามแผนภูมิ (flow chart) ในรูปที่ 4.10 โดยมีคำอธิบายตัวแปรที่สำคัญแสดงไว้ในตารางที่ 4.3 และมีรายละเอียดดังนี้

1) กำหนดให้ icount มีค่าเริ่มต้นเป็นศูนย์ และนับจำนวนรอบการทำงานของโปรแกรม และมีค่าสูงสุดเท่ากับ Exitcon ซึ่งก็คือจำนวนผิวทั้งหมดที่ถูกกำหนดตำแหน่งด้วยขนาดมิติ
2) กำหนด root node ในที่นี้ถือว่าความเบี่ยงเบนของดอกสว่านเนื่องจาก clearance ในรูของบุชมีค่าน้อย เพราะฉะนั้นแนวแกนของรูเจาะและแนวแกนของบุชเป็นตำแหน่งเดียวกัน ซึ่งก็คือตำแหน่งสัมผัสของจิกและชิ้นงาน ตำแหน่งนี้จะถูกเก็บในตัวแปร Copoint(0) และถูกใช้เป็น root node ในการค้นหา path
3) ตรวจสอบว่า inputx(icount).F และinputx(icount).T ของที่แต่ละ icount ไม่เท่ากับ Req( ) ; มิติที่ต้องการทราบค่าจะไม่สามารถนำมาเป็นส่วนหนึ่งของ path ได้ จึงทำขั้นตอนถัดไป แต่ถ้าเท่ากับ Req( ) จะข้ามไปทำตามขั้นตอนที่ 7
4) เก็บค่าตัวแปรสตริง inputx(icount).F และ inputx(icount).T ที่แต่ละรอบ icount ในตัวแปร Asearch และ Bsearch ตามลำดับ
5) ตรวจสอบเงื่อนไข ถ้า [Path(Asearch)] มีค่าแล้ว และ ไม่พบค่าของ [Path(Bsearch)] จะทำการข้ามขั้นตอนที่ 6 แต่ถ้าพบ path ของโหนดในตัวแปร Bsearch จะทำตามขั้นตอนถัดไป
6) นำสตริงใน Path(Bsearch) ต่อท้ายสตริง Asearch ซึ่งจะได้เป็น path ใหม่ คือ path(Asearch)
7) เพิ่มค่า icount และทำการตรวจสอบว่า icount และทำการวนลูปจนกว่าค่า icount เกิน Exitcon

จะได้ path ของโหนดที่เป็นจุดปลายของมิติ เรียงต่อกันไปยัง root node และเป็น path ที่เริ่มต้นด้วยผิวที่อยู่ในฟิลด์ F ของ inputx( ) ทั้งหมด ส่วน path ที่เริ่มต้นด้วยผิวในฟิลด์ T จะเกิดจากวิธีการ (routine) เช่นเดียวกัน แต่กำหนดให้ Asearch = inputx(icount).T และ Bsearch = inputx(icount).F ในขั้นตอนที่ 3

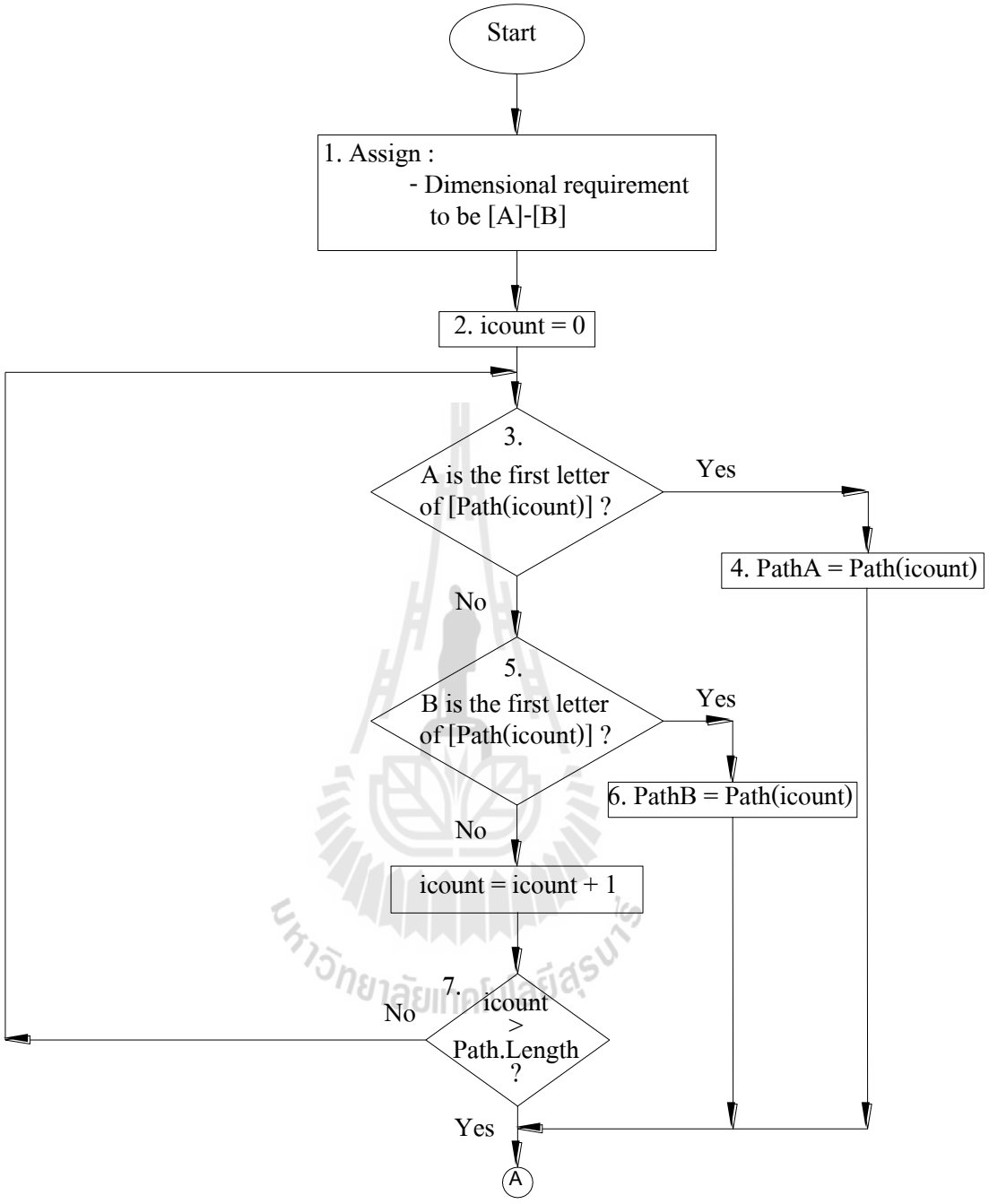


รูปที่ 4.10 ขั้นตอนทำงานในส่วนการสร้างเส้นทางความสัมพันธ์ของมิติ

4.4.2 โมดูลสำหรับค้นหาเส้นทางของมิติที่ต้องการทราบค่า

โปรแกรมส่วนนี้เป็นส่วนต่อเนื่องจากหัวข้อ 4.4.2 ทำหน้าที่ค้นหา path จากโหนดที่เป็นจุดเริ่มต้นและจุดสุดท้ายในการอ้างอิงบอกขนาดมิติที่ต้องการทราบค่าที่ผู้ใช้ทำการป้อนข้อมูลในหน้าจอรับข้อมูล โดยส่งค่าโหนดอ้างอิงทั้งสองในตัวแปร Req( ) เก็บในตัวแปรสตริง A และ B ตามลำดับ หลังจากนั้นจะนำสตริงทั้งสองค่าไปเทียบกับตำแหน่งแรกบนสายสตริงของ path ที่ได้ดำเนินการสร้าง และเก็บข้อมูลไว้ก่อนหน้านี้ ในฟังก์ชันนี้จะได้ path ของโหนดทั้งสองในตัวแปร Req( ) เก็บไว้ในตัวแปรสตริงชื่อ PathA และ PathB โดยขั้นตอนการทำงานของโปรแกรม เป็นไปตามแผนภูมิในรูปที่ 4.11 ซึ่งสามารถอธิบายได้ดังนี้

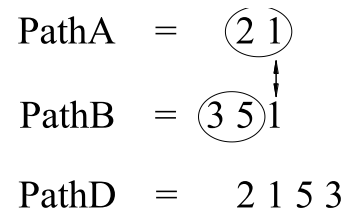
1) เรียกฟังก์ชัน Path_Search โดยส่งค่า Req(0) และ Req(1) เก็บอยู่ในตัวแปรสตริง A และ B
2) กำหนดให้ icount มีค่าเริ่มต้นเป็นศูนย์ และทำหน้าที่บ่งชี้จำนวนรอบการทำงานของโปรแกรม มีค่าสูงสุดเท่ากับจำนวนผิวที่ถูกระบุด้วยมิติ
3) ทำการตรวจสอบ, ถ้า A เทียบกับอักขระตัวแรกของสายสตริง Path(icount) แล้วเป็นค่าเดียวกัน จะทำตามขั้นตอนต่อไป แต่ถ้าไม่จะข้ามไปทำตามขั้นตอนที่ 6
4) ตัวแปร PathA มีค่าเท่ากับ Path(icount)
5) ทำการตรวจสอบ, ถ้าสตริง B เทียบกับอักขระตัวแรกของสายสตริง Path(icount) แล้วเป็นค่าเดียวกัน จะทำตามขั้นตอนต่อไป แต่ถ้าไม่จะข้ามไปทำตามขั้นตอนที่ 7
6) ตัวแปร PathB มีค่าเท่ากับ Path(icount)
7) ตรวจสอบเงื่อนไขว่า icount ยังไม่เกินค่าจำนวนความยาว string ในตัวแปร Path จึงทำการเพิ่มค่า icount และวนรอบการทำงานจนค่า icount เกินค่าจำนวนตัวแปรอาร์เรย์ของตัวแปร Path



รูปที่ 4.11 ขั้นตอนทำงานในโมดูลการค้นหาเส้นทางของมิติที่ต้องการทราบค่า

จากการทำงานของโปรแกรมในส่วนนี้จะได้ PathA และ PathB ซึ่งถูกค้นหาในลักษณะการหาเส้นทางผ่านแผนภาพต้นไม้ เมื่อได้ path ทั้งสอง ต้องนำมาเทียบกันเพื่อหา loop ของมิติทั้งหมด ซึ่งเป็นมิติใช้งานในการคำนวณ โดยสามารถหาได้จากโปรแกรมในโมดูลของการเทียบเส้นทางเพื่อสร้างสมการวงรอบ โดยจะทำงานต่อเนื่องจากในส่วนนี้

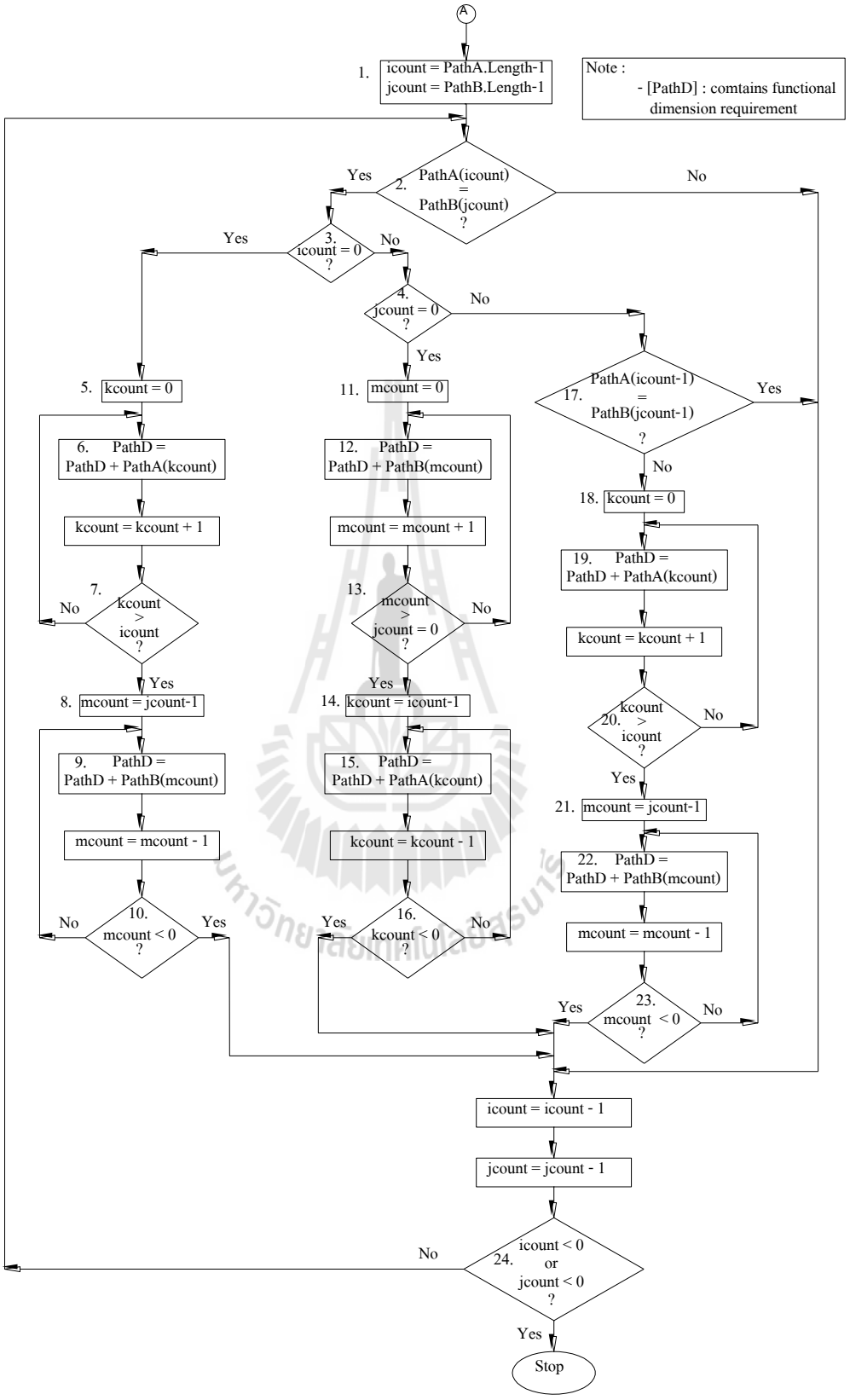
4.4.3 โมดูลสำหรับเทียบเส้นทางเพื่อสร้างสมการวงรอบ



รูปที่ 4.12 การเทียบ path

หลังจากการค้นหา path แล้ว PathA และ PathB ที่ได้จากการค้นหาจะถูกนำมาดำเนินการต่อตามขั้นตอนของโปรแกรมในส่วนการเทียบเส้นทางเพื่อสร้างสมการวงรอบ โหนดอ้างอิงของมิติที่ประกอบในสมการวงรอบจะถูกเก็บในตัวแปรสตริง PathD โดยการดำเนินการในส่วนนี้ของโปรแกรมถือเป็นส่วนสำคัญที่ทำให้ทราบมิติใช้งาน (functional dimension) มิติใช้งานทั้งหมดจะเป็นมิติที่นำมาคำนวณหาขนาดมูลฐาน (basic dimension) ค่าความคลาดเคลื่อน (tolerance) ของมิติที่ต้องการทราบค่า ขั้นตอนการทำงานของโปรแกรมมีขั้นตอนการทำงานโดยเริ่มจากการเทียบ PathA และ PathB ทีละตำแหน่งของอักขระบนสายสตริง เริ่มจากตำแหน่งสุดท้ายของสายสตริงทั้งคู่ ดังตัวอย่างดังรูปที่ 4.12 ถ้าตำแหน่งดังกล่าวมีสตริงตัวเดียวกัน และสตริงในตำแหน่งถัดไปทางซ้ายของ path ทั้งสองไม่ซ้ำกัน จะได้ PathD เท่ากับ สายสตริงตั้งแต่ตำแหน่งแรกของ PathA จนถึงตำแหน่งที่ทำการเทียบ ต่อด้วยสายสตริงของ PathB ในตำแหน่งถัดจากตำแหน่งที่ทำการเทียบไปทางซ้ายจนถึงตำแหน่งแรก โดยสามารถอธิบายตามแผนผังในรูปที่ 4.13

เส้นทางลัพธ์ (path) ที่ได้จากโปรแกรมส่วนนี้จะทำให้ทราบมิติที่เกี่ยวข้องกับมิติที่ต้องการทราบค่าทั้งหมด จากตัวอย่างชิ้นงานในรูป 4.7 มิติที่ต้องการทราบค่าคือ 2 – 3 ซึ่งเมื่อผ่านการดำเนินการของโปรแกรม ทำให้ทราบได้ว่า มิติ 2 – 1, 1 – 5 และ 5 – 3 เป็นมิติใช้งานที่มีความเกี่ยวเนื่องกันตาม loop diagram โดยสายสตริงใน PathD จะถูกนำไปเป็นข้อมูลนำเข้าในส่วนของโปรแกรมคำนวณขนาดของมิติที่ต้องการทราบค่าต่อไป



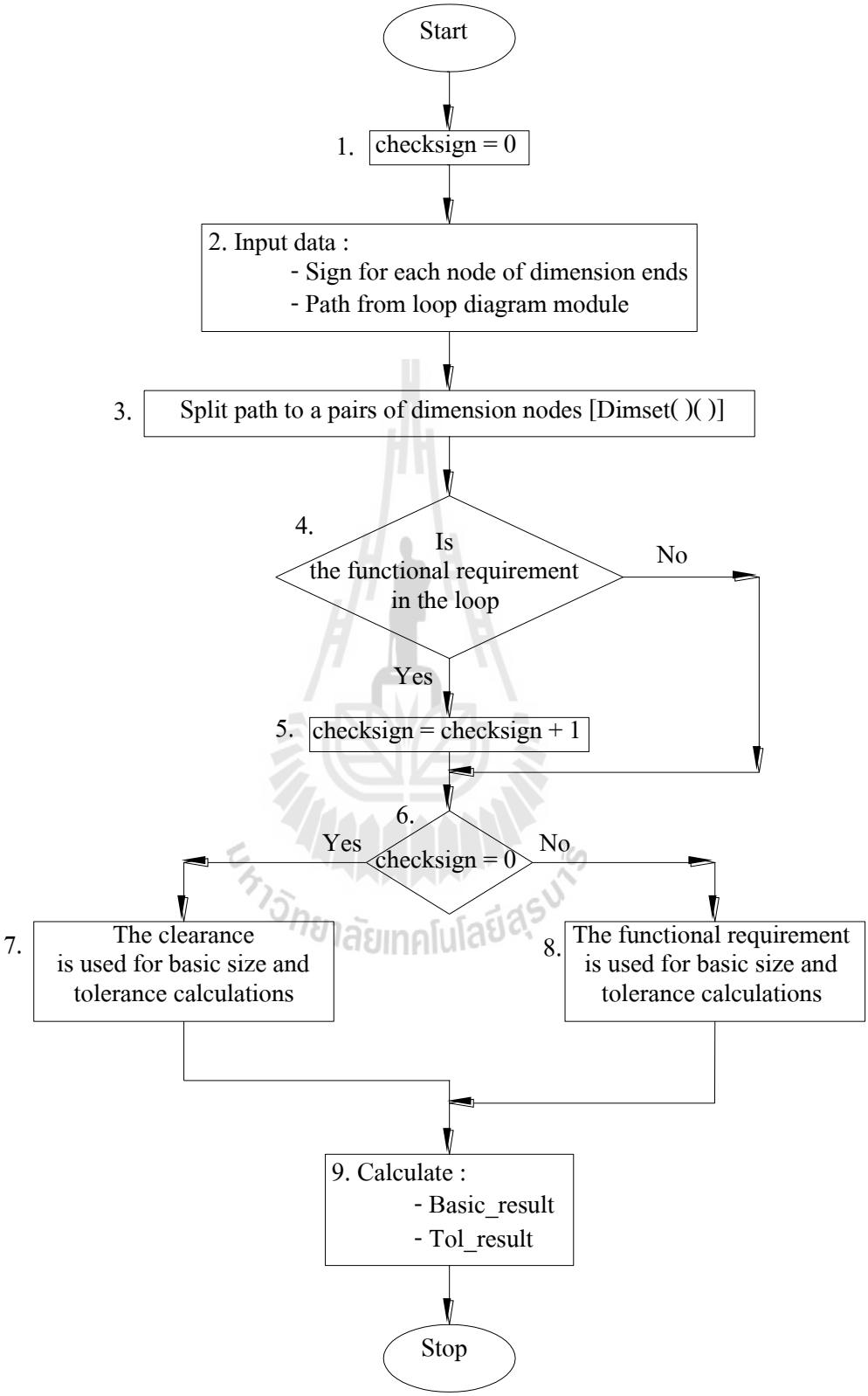
รูปที่ 4.13 ขั้นตอนทำงานในส่วนการเทียบเส้นทางเพื่อสร้างสมการวงรอบ

4.5 โมดูลสำหรับคำนวณขนาดของมิติที่ต้องการทราบค่า (modules for tolerance allocation and basic size calculations)

การคำนวณในโมดูลนี้ แบ่งออกเป็น 2 ส่วนย่อย ๆ คือ ส่วนของการคำนวณขนาดมูลฐาน และส่วนของการคำนวณค่าความคลาดเคลื่อน

ขั้นตอนการทำงานในส่วนการทำงานตามรูปที่ 4.14 จะรับค่า path ของโหนดของมิติใช้งานจากโมดูลสร้างสมการวงรอบ รวมทั้งค่าตามเครื่องหมายของมิติซึ่งขึ้นอยู่กับลักษณะการวางตำแหน่งโหนดอ้างอิงการบอกมิติของผู้ใช้ตั้งแต่ขั้นตอนการรับเข้าข้อมูล (ถ้าโหนดอ้างอิงแรกทางซ้ายมีค่าตำแหน่งการวางโหนดน้อยกว่าค่าตำแหน่งการวางโหนดอ้างอิงที่สองฝั่งขวา ค่าเครื่องหมายของมิตินั้นจะมีค่าเท่ากับ +1 แต่ถ้าโหนดอ้างอิงแรกทางซ้ายมีค่าตำแหน่งการวางโหนดมากกว่าค่าตำแหน่งการวางโหนดอ้างอิงที่สองฝั่งขวา ค่าเครื่องหมายของมิตินั้นจะมีค่าเท่ากับ -1) ซึ่งค่าเครื่องหมายของมิติจะเก็บอยู่ในตัวแปร inputx( ).Signcal ซึ่งจะถูกนำไปปรับค่าใน inputx( ).Dims ใด ๆ ให้ค่าสอดคล้องกับทิศทางการบอกขนาด

เมื่อรับค่า PathD จากโมดูลสมการวงรอบแล้ว โมดูลนี้จะนำมาแยกออกเป็นคู่โหนดในการบอกมิติ และเก็บค่าคู่โหนดในตัวแปรอาร์เรย์สตริง Dimset( )( ) ทีละคู่จนครบทั้งสาย แล้วนำไปคำนวณหาค่าขนาดมูลฐาน และความคลาดเคลื่อนต่อไป แต่ก่อนจะคำนวณค่าผลลัพธ์ โปรแกรมจะต้องทำการตรวจสอบเงื่อนไขว่า มิติที่ผู้ใช้ต้องการทราบค่านั้นมีมิติที่ตรงกับค่าความต้องการใช้งานเป็นมิติประกอบในมิติใช้งานในการคำนวณค่าหรือไม่ เพื่อให้ทราบตัวตั้งค่าในการกระจายค่าความคลาดเคลื่อน ทั้งนี้ ถ้า Checksign ไม่เท่ากับศูนย์ แสดงว่ามีมิติที่เป็นความต้องการใช้งานอยู่ใน Dimset( ) ตัวใดตัวหนึ่ง ค่าความคลาดเคลื่อนของตำแหน่งในการประกอบจะถูกนำมาเป็นตัวตั้งในการกระจายความคลาดเคลื่อนแทนค่าจากตำแหน่งรูเจาะบนชิ้นงานที่มาจากผู้ใช้ระบุ เพราะเป็นการหาองค์ขนาดองค์ประกอบอื่นที่เกี่ยวข้องกับขนาดงาน ไม่ใช่เกี่ยวข้องกับตำแหน่งรูเจาะบนชิ้นงาน



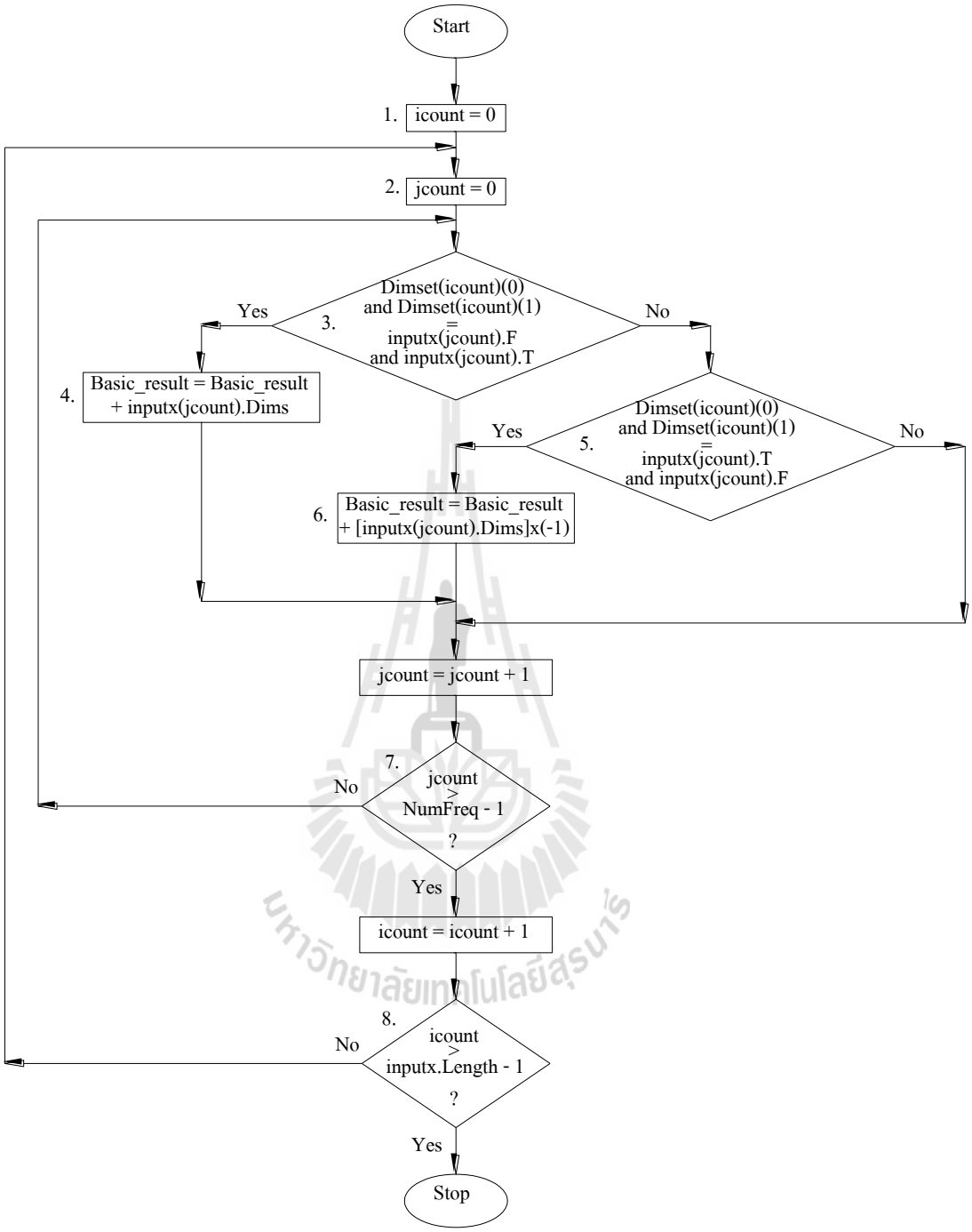
รูปที่ 4.14 การเตรียมข้อมูลการคำนวณขนาดของมิติที่ต้องการทราบค่า

องค์ประกอบที่ใช้ในการคำนวณค่าคือ ค่าขนาดมูลฐานของมิติซึ่งติดเครื่องหมายบอกทิศทางการบอกขนาดของมิติที่ใช้ในส่วนย่อยการคำนวณขนาดมูลฐาน และมิติที่เป็นตัวตั้งสำหรับกระจายค่าความคลาดเคลื่อนในส่วนย่อยของการคำนวณค่าความคลาดเคลื่อนที่ได้จากการตรวจสอบตัวแปร Checksign โดยทั้งสองค่าจะถูกคำนวณทีละส่วนคือ ขั้นตอนในการคำนวณขนาดมูลฐานซึ่งเก็บค่าผลลัพธ์ในตัวแปร Basic_result และความคลาดเคลื่อนลัพธ์จะเก็บในตัวแปร Tol_result

4.5.1 การคำนวณขนาดมูลฐาน

การคำนวณขนาดมูลฐานจะรวมค่าขนาดมูลฐานของมิติแต่ละคู่ในตัวแปร Dimset( ) ใด ๆ ตามค่าเครื่องหมาย ขั้นตอนการทำงานของโปรแกรมสามารถอธิบายตามผังการทำงานในรูปที่ 4.15 มีขั้นตอนดังนี้

1) กำหนดให้ icount มีค่าเริ่มต้นเป็นศูนย์
2) กำหนดให้ jcount มีค่าเริ่มต้นเป็นศูนย์
3) ตรวจสอบคู่โหนด Dimset(icount) , ถ้า Dimset(icount)(0) และ Dimset(icount)(1) เท่ากับ inputx(jcount).F และ inputx(jcount).T ทำตามขั้นตอนถัดไป แต่ถ้าไม่เท่ากันข้ามไปทำขั้นตอนที่ 5
4) Basic_result จะเท่ากับผลรวมของ Basic_result กับ Dimset(icount) ข้ามขั้นตอนที่ 5 และ 6
5) ตรวจสอบคู่โหนด Dimset(icount) , ถ้า Dimset(icount)(0) และ Dimset(icount)(1) เท่ากับ inputx(jcount).T และ inputx(jcount).F
6) Basic_result จะเท่ากับผลรวมของ Basic_result กับ Dimset(icount) โดย Dimset(icount) มีค่าเป็นลบ
7) เพิ่มค่า jcount ทีละ 1, ถ้า jcount มากกว่า NumFreq – 1 ทำตามขั้นตอนต่อไป แต่ถ้าไม่ และกลับไปทำตามขั้นตอนที่ 3
8) เพิ่มค่า icount ทีละ 1, ถ้า icount มากกว่า inputx.Length - 1 ทำตามขั้นตอนต่อไป แต่ถ้าไม่ และกลับไปทำตามขั้นตอนที่ 3



รูปที่ 4.15 ขั้นตอนทำงานในส่วนการคำนวณขนาดมูลฐานของมิติที่ต้องการทราบ

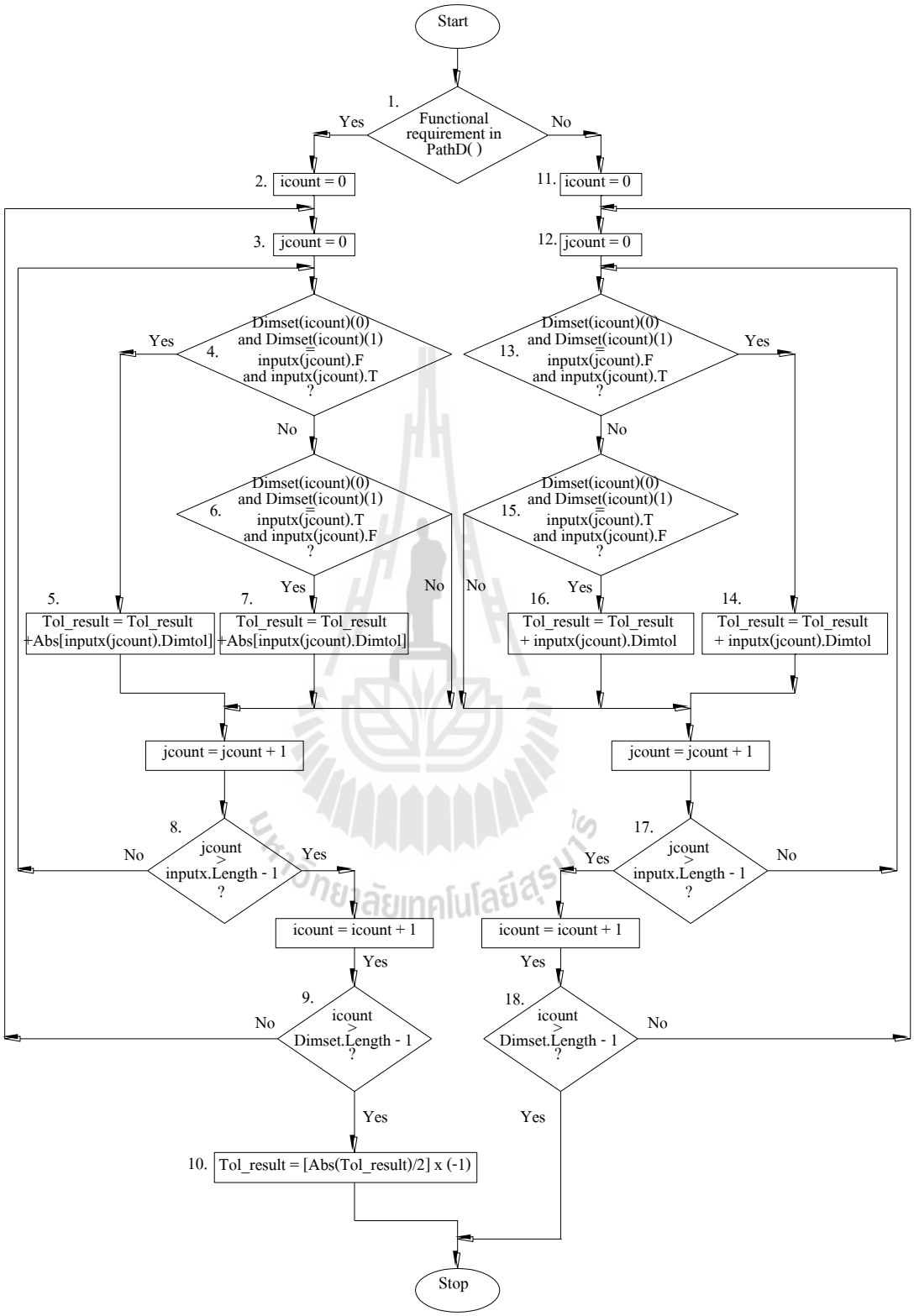
4.5.2 การคำนวณค่าความคลาดเคลื่อน

การคำนวณในส่วนนี้เป็นการกระจายค่าความคลาดเคลื่อนของมิติที่เป็นความต้องการใช้งานให้กับมิติใช้งานทั้งหมดที่อยู่ในคู่โหนดทุกคู่ใน Dimset( ) ที่ไม่ใช่คู่โหนดอ้างอิงมิติที่เป็นความต้องการใช้งาน ซึ่งรายละเอียดขั้นตอนการดำเนินการของโปรแกรมจะเป็นไปตามผังการทำงานในรูปที่ 4.14

ภาพรวมของขั้นตอนการหาค่าความคลาดเคลื่อนคือ โปรแกรมจะทำเทียบคู่โหนด Dimset( )( ) ทีละคู่เช่นเดียวกับการหาค่าขนาดมูลฐาน โดยการเทียบคู่โหนดกับสตริงของ inputx( ).F - inputx( ).T และ inputx( ).T - inputx( ).F และทำการรวมค่าความคลาดเคลื่อนสะสมแต่ละรอบที่ตัวแปร Tol_result โดยก่อนที่เราจะเทียบคู่โหนดเช่นเดียวกับการหาขนาดมูลฐาน โปรแกรมจะตรวจสอบก่อนว่า คู่โหนดทั้งหมดที่อยู่ใน Dimset( ) ไม่ใช่ตำแหน่งของรูเจาะทั้งหมด เนื่องจากว่าก่อนหน้านี้ค่าความคลาดเคลื่อนของมิติที่ไม่ใช่ตำแหน่งรูเจาะซึ่งหรือมิติที่เป็นความต้องการใช้งาน (มิติตั้งต้นสำหรับการกระจายค่าความคลาดเคลื่อน) จะมีค่าเป็นลบ และเป็นค่าบวกในกรณีที่เป็นมิติของความต้องการใช้งาน หากเป็นการรวมกันทั้งหมดของมิติบอกตำแหน่งรูเจาะต้องรวมค่าความคลาดเคลื่อนโดยใช้ค่าสัมบูรณ์ของ inputx( ).Dimtol แต่หากไม่ใช่กรณีดังกล่าว โปรแกรมจะทำการรวมค่าความคลาดเคลื่อนสะสมใน Tol_result ตามเครื่องหมายที่กำหนดไปจนครบรอบการทำงาน

สุดท้ายแล้วเมื่อสิ้นสุดการทำงานของส่วนการคำนวณค่าขนาดมิติจะได้ผลลัพธ์เป็นค่าขนาดมูลฐาน และค่าความคลาดเคลื่อนของมิติที่ผู้ใช้ต้องการทราบค่า ซึ่งเมื่อลำดับโปรแกรมทำงานต่อไป ก็จะถึงส่วนที่ทำหน้าที่แสดงผลจากการคำนวณค่าเหล่านี้ โดยค่าผลลัพธ์จะแสดงออกทั้งหมด 3 ชุด ตามค่าความฟิต 3 แบบ โดยส่วนแสดงผลยังแสดงมิติใช้งานในวงรอบการคำนวณทั้งหมดด้วย

มหาวิทยาลัยเทคโนโลยีสุรนารี



รูปที่ 4.16 ขั้นตอนทำงานในส่วนการคำนวณค่าความคลาดเคลื่อนของมิติที่ต้องการทราบ

4.6 สรุป

ในบทนี้อธิบายรายละเอียดของตัวแปรต่าง ๆ รวมถึงแผนภาพขั้นตอนการทำงานของโปรแกรมของโมดูลหลักในโปรแกรม ประกอบไปด้วย;โมดูลป้อนข้อมูล (input module), โมดูลสร้าง loop equation (loop diagram generation module), โมดูลสำหรับกำหนดค่าความคลาดเคลื่อนในการประกอบของชิ้นงานและจิก (tolerance calculation and assignment module) และ โมดูลสำหรับคำนวณขนาดของมิติที่ต้องการทราบค่า (modules for tolerance allocation and basic size calculations)

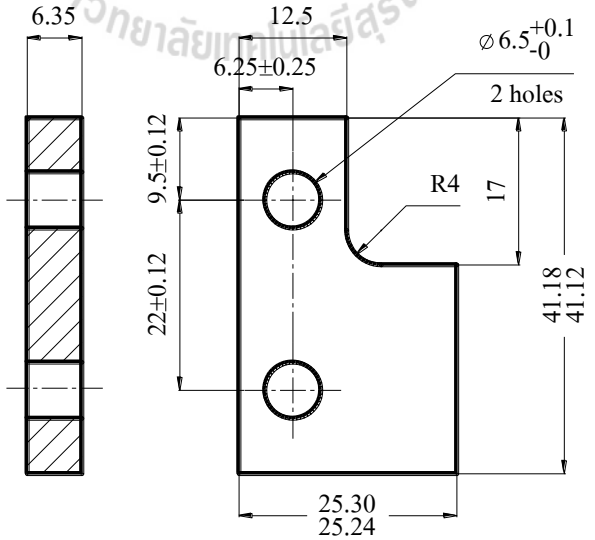
มหาวิทยาลัยเทคโนโลยีสุรนารี

บทที่ 5
การทดสอบโปรแกรม

ในบทนี้จะเป็นส่วนแสดงข้อมูลจากการทดสอบการทำงานของโปรแกรม CADBP โดยใช้ตัวอย่างชิ้นงานที่ต้องการออกแบบจิกประเภทต่าง ๆ สำหรับเจาะรู การทดสอบนี้จะนำผลการคำนวณที่ได้จากโปรแกรมมาแสดงเปรียบเทียบผลลัพธ์จากการคำนวณตามทฤษฎีด้วยวิธีการแบบเดิม ผลการทดสอบเป็นไปดังนี้

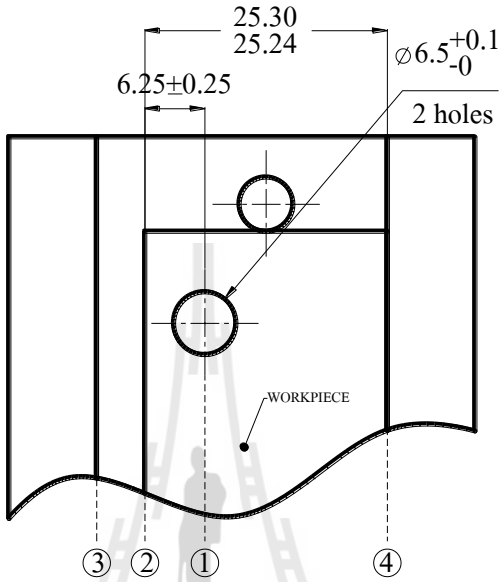
5.1 ตัวอย่างการคำนวณของชิ้นงานรูปเหลี่ยม แบบใช้รางกำหนดตำแหน่ง

ตัวอย่างชิ้นงานดังรูปที่ 5.1 นำมาดัดแปลงจากตัวอย่างในหนังสือของยงยุทธ (2556) เพื่อใช้แสดงวิธีการหาตำแหน่งบุชด้วยโปรแกรม CADBP จากรูปชิ้นงานทำด้วย spacer bar ถูกตัดจากแท่งเหล็ก และได้รับการเจียรนัยขอบทั้ง 4 ด้าน เป็นสี่เหลี่ยมผืนผ้า กว้าง $\frac{25.30}{25.24}$ mm., ยาว $\frac{41.18}{41.12}$ mm. ต้องการเจาะรูขนาด $\varnothing$ 6.5 mm. จำนวน 2 รู จากลักษณะชิ้นงานจะเลือกใช้ขอบชิ้นงานเป็นรูปลักษณ์กำหนดตำแหน่ง ใช้รางเป็นตัวกำหนดตำแหน่งตามด้านกว้าง และใช้หมุดทำรังเพื่อกำหนดตำแหน่งตามด้านยาว



รูปที่ 5.1 แบบงานสำหรับใช้ราง และ Nesting pin

ในการทดสอบโปรแกรมในส่วนนี้จะทำการทดสอบหาตำแหน่งบุชสำหรับเจาะรูด้านบนของชิ้นงานเพียงรูเดียว โดยควบคุมตำแหน่งรูเจาะตามแนวระดับด้วยการใช้ผิวด้านข้างของงานเป็นรู้ลักษณ์กำหนดตำแหน่งด้วยราง และแบบร่างสำหรับคำนวณหาตำแหน่งบุชแสดงดังรูปที่ 5.2



รูปที่ 5.2 แบบร่างงานบนจิกแบบรางในการระบุผิวอ้างอิง

5.1.1 ผลการคำนวณด้วยโปรแกรม

ผลลัพธ์ต่อไปนี้แสดงค่าเป็นข้อมูลจากหน้าจอแสดงผลลัพธ์ของโปรแกรม โดยแสดงรูปแบบของจิก ขนาด และความคลาดเคลื่อนของตำแหน่งบุช (1 – 3) และทำการเพิ่มเติมมิติ คือ ขนาดราง (3 - 4) พร้อมทั้งมิติที่เกี่ยวข้องในสมการวงรอบ ดังนี้

```
>>>>>>>> HORIZONTAL AXIS <<<<<<<<
งานเหลี่ยมโดยใช้ราง


Preferred Fit  C8_h8
Bush position along with axis (1-3) is
          6.3965    + .2135 / - .2135


From     To        Basic sizes         Tolerances
```

1	2	6.25	0.25
2	3	0.1465	0.0365

Preferred Fit G8_h8

Bush position along with axis (1-3) is

6.2935 + .2135 / - .2135

From	To	Basic sizes	Tolerances
1	2	6.25	0.25
2	3	0.0435	0.0365

Preferred Fit H8_h8

Bush position along with axis (1-3) is

6.2865 + .2135 / - .2135

From	To	Basic sizes	Tolerances
1	2	6.25	0.25
2	3	0.0365	0.0365

>>>>>>>> HORIZONTAL AXIS <<<<<<<<

งานเหลี่ยมโดยใช้ราง

Preferred Fit C8_h8

Locator size/position along with axis (3-4) is

25.4100 + .0130 / - 0

From	To	Basic sizes	Tolerances
3	2	0.1465	0.0365
2	4	25.27	0.03

มหาวิทยาลัยเทคโนโลยีสุรนารี

Preferred Fit G8_h8

Locator size/position along with axis (3-4) is

25.3070 + .0130 / - 0

From	To	Basic sizes	Tolerances
3	2	0.0435	0.0365
2	4	25.27	0.03

Preferred Fit H8_h8

Locator size/position along with axis (3-4) is

25.3000 + .0130 / - 0

From	To	Basic sizes	Tolerances
3	2	0.0365	0.0365
2	4	25.27	0.03

5.1.2 ผลการคำนวณโดยไม่ใช้โปรแกรม

เมื่อทำการคำนวณด้วยหลักการเดียวกับที่ใช้ในโปรแกรมทีละขั้นตอนจะได้คำตอบเช่นเดียวกับการคำนวณด้วยโปรแกรม CADBP ดังต่อไปนี้

Max. Clearance (2e) = Min. Clearance + Size Tol. for work + Size Tol. for locator

<u>ฟิตนิยม Ch</u>

2e = Ch (25.75) + (2 x 0.03) + IT6 (25.27)

= 0.11 + 0.06 + 0.013

= 0.183

N = (Max. Clearance + Min. Clearance) x $\frac{1}{2}$

= (0.183+ 0.11) x $\frac{1}{2}$ = 0.1465

± n = (Max. Clearance - Min. Clearance) x $\frac{1}{2}$

= (0.183- 0.11) x $\frac{1}{2}$ = 0.0365

ค่าจากการคำนวณที่มิติ (1 – 3) มาจาก มิติ (1 – 2) และ (2 – 3)

ดังนั้น ค่าจากการคำนวณที่มิติ (1 – 3) = (6.25 + 0.1465) ± (0.25 - 0.0365)

= 6.3965 ± 0.2135

ค่าจากการคำนวณที่มิติ (3 – 4) มาจาก มิติ (4 – 2) และ (2 – 3)

ดังนั้น ค่าจากการคำนวณที่มิติ (3 – 4) = (0.1465 – 25.27) ± (0.0365 – 0.030)

= 25.4165 ± 0.0065

หรือ = $25.41^{+0.013}_{-0}$

<u>ฟิตนิยม Gh</u>

2e = Gh (25.75) + (2 x 0.03) + IT6 (25.27)

= 0.007 + 0.06 + 0.013

= 0.08

N = (Max. Clearance + Min. Clearance) x $\frac{1}{2}$

= (0.08 + 0.007) x $\frac{1}{2}$ = 0.0435

± n = (Max. Clearance - Min. Clearance) x $\frac{1}{2}$

= (0.08 + 0.007) x $\frac{1}{2}$ = 0.0365

ค่าจากการคำนวณที่มิติ (1 – 3) มาจาก มิติ (1 – 2) และ (2 – 3)

ดังนั้น ค่าจากการคำนวณที่มิติ (1 – 3) = (6.25 + 0.0435) ± (0.25 - 0.0365)

= 6.2935 ± 0.2135

ค่าจากการคำนวณที่มิติ (3 – 4) มาจาก มิติ (4 – 2) และ (2 – 3)

ดังนั้น ค่าจากการคำนวณที่มิติ (3 – 4) = (0.0435 – 25.27) ± (0.0365 – 0.030)

= 25.3135 ± 0.0065

หรือ = $25.307^{+0.013}_{-0}$

<u>ฟิตนิยม Hh</u>

2e = Hh (25.75) + (2 x 0.03) + IT6 (25.27)

= 0 + 0.06 + 0.013

= .073

N = (Max. Clearance + Min. Clearance) x $\frac{1}{2}$

= (0.073 + 0) x $\frac{1}{2}$ = 0.0365

$\pm n$ = (Max. Clearance - Min. Clearance) x $\frac{1}{2}$

= (0.073 - 0) x $\frac{1}{2}$ = 0.0365

ค่าจากการคำนวณที่มิติ (1 – 3) มาจาก มิติ (1 – 2) และ (2 – 3)

ดังนั้น ค่าจากการคำนวณที่มิติ (1 – 3) = (6.25 + 0.0365) ± (0.25 - 0.0365)

= 6.2865 ± 0.2135

ค่าจากการคำนวณที่มิติ (3 – 4) มาจาก มิติ (4 – 2) และ (2 – 3)

ดังนั้น ค่าจากการคำนวณที่มิติ (3 – 4) = (0.0365 – 25.27) ± (0.0365 – 0.030)

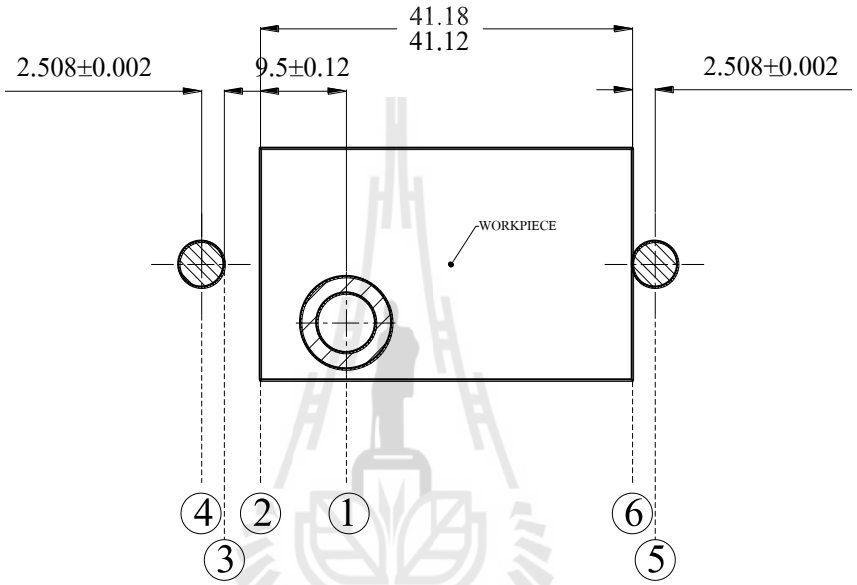
= 25.3065 ± 0.0065

หรือ = $25.30^{+0.013}_{-0}$

มหาวิทยาลัยเทคโนโลยีสุรนารี

5.2 ตัวอย่างการคำนวณของชิ้นงานรูปเหลี่ยม แบบใช้ Nesting pins

ตัวอย่างนี้เป็นตัวอย่างชิ้นงานเดียวกับในหัวข้อ 5.1 จะทำการทดสอบหาตำแหน่งบุชสำหรับเจาะรูด้านบนของชิ้นงานโดยควบคุมระยะในแนวตั้งฉากกับแนวระดับตามรูป 5.1 ด้วยการใช้ผิวด้านข้างของงานเป็นรูปลักษณ์กำหนดตำแหน่งด้วยหมุดกำหนดตำแหน่ง 2 ตัว ซึ่งจะได้แบบร่างการสำหรับคำนวณหาตำแหน่งบุชแสดงดังรูป 5.3



รูปที่ 5.3 แบบร่างงานบนจิกแบบ Nesting pin ในการระบุผิวอ้างอิงจากแบบงานจริง

5.2.1 ผลการคำนวณด้วยโปรแกรม

ผลลัพธ์ต่อไปนี้แสดงค่าเป็นข้อมูลจากหน้าจอแสดงผลลัพธ์ของโปรแกรม โดยแสดงรูปแบบของจิก ขนาด และความคลาดเคลื่อนของตำแหน่งบุชโดยอ้างอิงผิวหมุด (1 – 3) และทำการเพิ่มเติมมิติคือ ตำแหน่งบุชที่อ้างอิงตำแหน่งกึ่งกลางหมุด (1 – 4) และขนาดรัง (4 – 5) พร้อมทั้งมิติที่เกี่ยวข้องในสมการวงรอบ ดังนี้

```
>>>>>>>> HORIZONTAL AXIS <<<<<<<<
งานเหลี่ยมโดยใช้หมุดเป็นรัง


Preferred Fit  C8_h8
Bush position along with axis (1-3) is
```

9.6815 + .0685 - .0685

From	To	Basic sizes	Tolerances
1	2	9.5	0.12
2	3	0.1815	0.0515

Preferred Fit G8_h8

Bush position along with axis (1-3) is

9.5605 + .0685 - .0685

From	To	Basic sizes	Tolerances
1	2	9.5	0.12
2	3	0.0605	0.0515

Preferred Fit H8_h8

Bush position along with axis (1-3) is

9.5515 + .0685 - .0685

From	To	Basic sizes	Tolerances
1	2	9.5	0.12
2	3	0.0515	0.0515

>>>>>>>> HORIZONTAL AXIS <<<<<<<<

งานเหลี่ยมโดยใช้หมุดเป็นรัง

Preferred Fit C8_h8

Bush position along with axis (1-4) is

12.1895 + .0665 - .0665

From	To	Basic sizes	Tolerances
1	2	9.5	0.12
2	3	0.1815	0.0515
3	4	2.508	0.002

Preferred Fit G8_h8

Bush position along with axis (1-4) is

12.0685 + .0665 - .0665

From	To	Basic sizes	Tolerances
1	2	9.5	0.12
2	3	0.0605	0.0515
3	4	2.508	0.002

Preferred Fit H8_h8

Bush position along with axis (1-4) is

12.0595 + .0665 - .0665

From	To	Basic sizes	Tolerances
1	2	9.5	0.12
2	3	0.0515	0.0515
3	4	2.508	0.002

>>>>>>>> HORIZONTAL AXIS <<<<<<<<

งานเหลี่ยมโดยใช้หมุดเป็นรัง

Preferred Fit C8_h8

Locator size/position along with axis (4-5) is

46.3475 + .0175 - .0175

มหาวิทยาลัยเทคโนโลยีสุรนารี

From	To	Basic sizes	Tolerances
4	3	2.508	0.002
3	2	0.1815	0.0515
2	6	41.15	0.03
6	5	2.508	0.002

Preferred Fit G8_h8

Locator size/position along with axis (4-5) is

46.2265 + .0175 - .0175

From	To	Basic sizes	Tolerances
4	3	2.508	0.002
3	2	0.0605	0.0515
2	6	41.15	0.03
6	5	2.508	0.002

Preferred Fit H8_h8

Locator size/position along with axis (4-5) is

46.2175 + .0175 - .0175

From	To	Basic sizes	Tolerances
4	3	2.508	0.002
3	2	0.0515	0.0515
2	6	41.15	0.03
6	5	2.508	0.002

มหาวิทยาลัยเทคโนโลยีสุรนารี

5.2.2 ผลการคำนวณโดยไม่ใช้โปรแกรม

เมื่อทำการคำนวณด้วยหลักการเดียวกับที่ใช้ในโปรแกรมทีละขั้นตอนจะได้คำตอบเช่นเดียวกับการคำนวณด้วยโปรแกรม CADBP ดังต่อไปนี้

Max. Clearance (2e) = Min. Clearance + Size Tol. for work
+ Size Tol. for locator + Pin Pos. Tol.

<u>ฟิตนิยม Ch</u>

2e = Ch (41.15) + (2 x 0.03) + (2 x 0.002) + IT8 (2.508 + 42.15 + 2.508)
= 0.13 + 0.06 + 0.004 + 0.039
= 0.233

N = (Max. Clearance + Min. Clearance) x $\frac{1}{2}$
= (0.233 + 0.13) x $\frac{1}{2}$ = 0.1815

± n = (Max. Clearance - Min. Clearance) x $\frac{1}{2}$
= (0.233 - 0.13) x $\frac{1}{2}$ = 0.0515

ค่าจากการคำนวณที่มิติ (1 – 3) มาจาก มิติ (1 – 2) และ (2 – 3)
ดังนั้น ค่าจากการคำนวณที่มิติ (1 – 3) = (9.5 + 0.1815) ± (0.12 - 0.0515)
= 9.6815 ± 0.0685

ค่าจากการคำนวณที่มิติ (1 – 4) มาจาก มิติ (1 – 2), (2 – 3) และ (3 – 4)
ดังนั้น ค่าจากการคำนวณที่มิติ (31 – 4) = (9.5 + 0.1815 + 2.508)
± (0.12 – 0.0515 – 0.002)
= 12.1895 ± 0.0665

ค่าจากการคำนวณที่มิติ (4 – 5) มาจาก มิติ (4 – 3), (3 – 2), (2 – 6) และ (6 – 5)
ดังนั้น ค่าจากการคำนวณที่มิติ (4 – 5) = (2.508 + 0.1815 + 41.15 + 2.508)
± (0.0515 - 0.002 – 0.03 - 0.002)
= 46.3475 ± 0.0175

<u>ฟิตนิยม Gh</u>

2e = Ch (41.15) + (2 x 0.03) + (2 x 0.002) + IT8 (2.508 + 42.15 + 2.508)
= 0.009 + 0.06 + 0.004 + 0.039
= 0.112

N = (Max. Clearance + Min. Clearance) x $\frac{1}{2}$

= (0.112 + 0.009) x $\frac{1}{2}$ = 0.0605

± n = (Max. Clearance - Min. Clearance) x $\frac{1}{2}$

= (0.112 - 0.009) x $\frac{1}{2}$ = 0.0515

ค่าจากการคำนวณที่มิติ (1 – 3) มาจาก มิติ (1 – 2) และ (2 – 3)

ดังนั้น ค่าจากการคำนวณที่มิติ (1 – 3) = (9.5 + 0.0605) ± (0.12 - 0.0515)

= 9.5605 ± 0.0685

ค่าจากการคำนวณที่มิติ (1 – 4) มาจาก มิติ (1 – 2), (2 – 3) และ (3 – 4)

ดังนั้น ค่าจากการคำนวณที่มิติ (1 – 4) = (9.5 + 0.0605 + 2.508)

± (0.12 – 0.0515 – 0.002)

= 12.0685 ± 0.0665

ค่าจากการคำนวณที่มิติ (4 – 5) มาจาก มิติ (4 – 3), (3 – 2), (2 – 6) และ (6 – 5)

ดังนั้น ค่าจากการคำนวณที่มิติ (4 – 5) = (2.508 + 0.0605 + 41.15 + 2.508)

± (0.0515 - 0.002 – 0.03 - 0.002)

= 46.2265 ± 0.0175

<u>ฟิตนิยม Hh</u>

2e = Hh(41.15) + (2 x 0.03) + (2 x 0.002) + IT8 (2.508 + 42.15 + 2.508)

= 0 + 0.06 + 0.004 + 0.039

= 0.103

N = (Max. Clearance + Min. Clearance) x $\frac{1}{2}$

= (0.103 + 0) x $\frac{1}{2}$ = 0.0515

± n = (Max. Clearance - Min. Clearance) x $\frac{1}{2}$

= (0.103 - 0) x $\frac{1}{2}$ = 0.0515

ค่าจากการคำนวณที่มิติ (1 – 3) มาจาก มิติ (1 – 2) และ (2 – 3)

ดังนั้น ค่าจากการคำนวณที่มิติ (1 – 3) = (9.5 + 0.0515) ± (0.12 - 0.0515)

= 9.5515 ± 0.0685

ค่าจากการคำนวณที่มิติ (1 – 4) มาจาก มิติ (1 – 2), (2 – 3) และ (3 – 4)

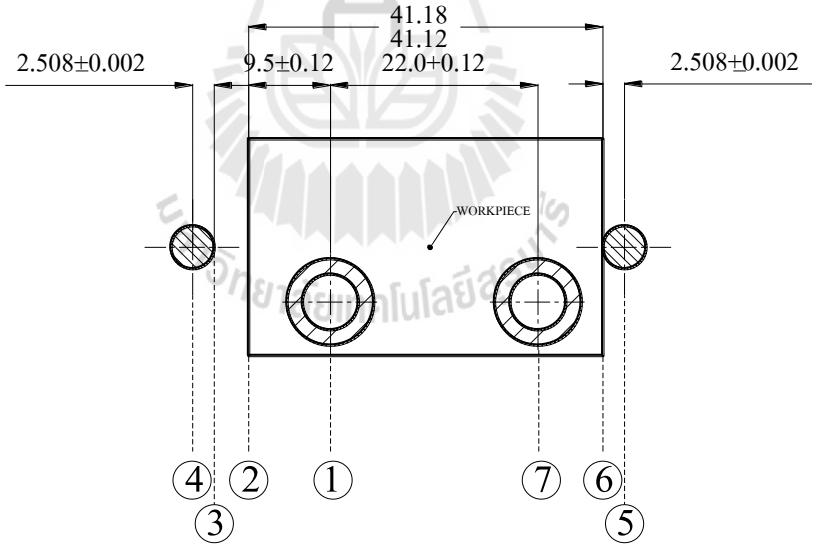
ดังนั้น ค่าจากการคำนวณที่มิติ (3 – 4) = (9.5 + 0.0515 + 2.508)
± (0.12 – 0.0515 – 0.002)
= 12.0595 ± 0.0665

ค่าจากการคำนวณที่มิติ (4 – 5) มาจาก มิติ (4 – 3), (3 – 2), (2 – 6) และ (6 – 5)

ดังนั้น ค่าจากการคำนวณที่มิติ (4 – 5) = (2.508 + 0.0515 + 41.15 + 2.508)
± (0.0515 - 0.002 - 0.03 - 0.002)
= 46.2175 ± 0.0175

5.3 ตัวอย่างการคำนวณของชิ้นงานรูปเหลี่ยม แบบใช้ Nesting pin เมื่อชิ้นงานมี 2 รูเจาะ

ตัวอย่างนี้เป็นตัวอย่างชิ้นงานเดียวกับในรูป 5.1 ต้องการหาตำแหน่งบุชสำหรับเจาะรูด้านล่างของชิ้นงาน โดยแบบของงานระบุตำแหน่งรูเจาะด้านล่าง อ้างอิงจากตำแหน่งรูเจาะด้านบน ดังแสดงใน แบบร่างจิกในรูป 5.4 จะใช้ผิวด้านข้างของงานเป็นรูปลักษณ์กำหนดตำแหน่งด้วยหมุดกำหนดตำแหน่ง 2 ตัว



รูปที่ 5.4 แบบร่างงานบนจิกแบบ Nesting pin ในการระบุผิวอ้างอิงจากแบบงานจริง เมื่อพิจารณา 2 รูเจาะ

5.3.1 ผลการคำนวณด้วยโปรแกรม

ผลลัพธ์ต่อไปนี้แสดงค่าเป็นข้อมูลจากหน้าจอแสดงผลลัพธ์ของโปรแกรม โดยแสดงรูปแบบของจิก ขนาด และความคลาดเคลื่อนของตำแหน่งบุชโดยอ้างอิงผิวหมุด (3 – 7) พร้อมทั้งมิติที่เกี่ยวข้องในสมการวงรอบ ดังนี้

>>>>>>>> HORIZONTAL AXIS <<<<<<<<

งานเหลี่ยมโดยใช้หมุดเป็นรัง

Preferred Fit C8_h8

Bush position along with axis (3-7) is

19.1815 + .1885 - .1885

From	To	Basic sizes	Tolerances
3	2	0.1815	0.0515
2	1	9.5	0.12
1	7	9.5	0.12

Preferred Fit G8_h8

Bush position along with axis (3-7) is

19.0605 + .1885 - .1885

From	To	Basic sizes	Tolerances
3	2	0.0605	0.0515
2	1	9.5	0.12
1	7	9.5	0.12

Preferred Fit H8_h8

Bush position along with axis (3-7) is

19.0515 + .1885 - .1885

From	To	Basic sizes	Tolerances
3	2	0.0515	0.0515

มหาวิทยาลัยเทคโนโลยีสุรนารี

2	1	9.5	0.12
1	7	9.5	0.12

5.3.2 ผลการคำนวณโดยไม่ใช้โปรแกรม

เมื่อทำการคำนวณด้วยหลักการเดียวกับที่ใช้ในโปรแกรมทีละขั้นตอนจะได้คำตอบเช่นเดียวกับการคำนวณด้วยโปรแกรม CADBP ดังต่อไปนี้

Max. Clearance (2e) = Min. Clearance + Size Tol. for work
+ Size Tol. for locator + Pin Pos. Tol.

<u>ฟิตนิยม Ch</u>

2e = Ch (41.15) + (2 x 0.03) + (2 x 0.002) + IT8 (2.508 + 41.15 + 2.508)
= 0.13 + 0.06 + 0.004 + 0.039
= 0.233

N = (Max. Clearance + Min. Clearance) x $\frac{1}{2}$
= (0.233 + 0.13) x $\frac{1}{2}$ = 0.1815

± n = (Max. Clearance - Min. Clearance) x $\frac{1}{2}$
= (0.233 - 0.13) x $\frac{1}{2}$ = 0.0515

ค่าจากการคำนวณที่มิติ (3 – 7) มาจาก มิติ (3 – 2), (2 – 1) และ (1 – 7)

ดังนั้น ค่าจากการคำนวณที่มิติ (3 – 7) = (0.1815 + 9.5 + 9.5)
± (0.12 + 0.12 – 0.0515)
= 19.1815 ± 0.1885

<u>ฟิตนิยม Gh</u>

2e = Ch (41.15) + (2 x 0.03) + (2 x 0.002) + IT8 (2.508 + 41.15 + 2.508)
= 0.009 + 0.06 + 0.004 + 0.039
= 0.112

N = (Max. Clearance + Min. Clearance) x $\frac{1}{2}$
= (0.112 + 0.009) x $\frac{1}{2}$ = 0.0605

$\pm n$ = (Max. Clearance - Min. Clearance) x $\frac{1}{2}$

= (0.112 - 0.009) x $\frac{1}{2}$ = 0.0515

ค่าจากการคำนวณที่มิติ (3 – 7) มาจาก มิติ (3 – 2), (2 – 1) และ (1 – 7)

ดังนั้น ค่าจากการคำนวณที่มิติ (3 – 7) = (0.0605 + 9.5 + 9.5)

$\pm$ (0.12 + 0.12 - 0.0515)

= 19.0605 $\pm$ 0.1885

<u>ฟิตนิยม Hh</u>

2e = Hh(41.15) + (2 x 0.03) + (2 x 0.002) + IT8 (2.508 + 41.15 + 2.508)

= 0 + 0.06 + 0.004 + 0.039

= 0.103

N = (Max. Clearance + Min. Clearance) x $\frac{1}{2}$

= (0.103 + 0) x $\frac{1}{2}$ = 0.0515

$\pm n$ = (Max. Clearance - Min. Clearance) x $\frac{1}{2}$

= (0.103 - 0) x $\frac{1}{2}$ = 0.0515

ค่าจากการคำนวณที่มิติ (3 – 7) มาจาก มิติ (3 – 2), (2 – 1) และ (1 – 7)

ดังนั้น ค่าจากการคำนวณที่มิติ (3 – 7) = (0.0515 + 9.5 + 9.5)

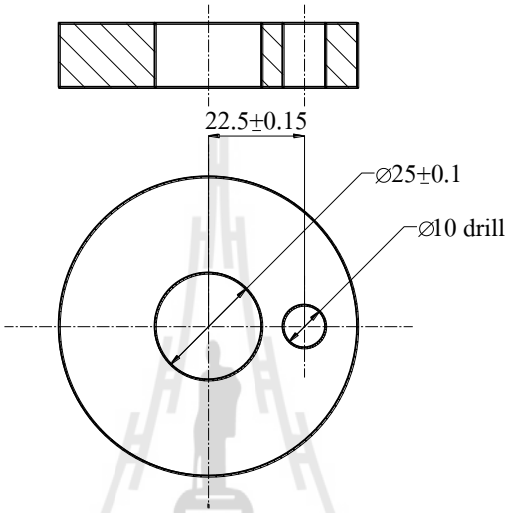
$\pm$ (0.12 + 0.12 - 0.0515)

= 19.0515 $\pm$ 0.1885

มหาวิทยาลัยเทคโนโลยีสุรนารี

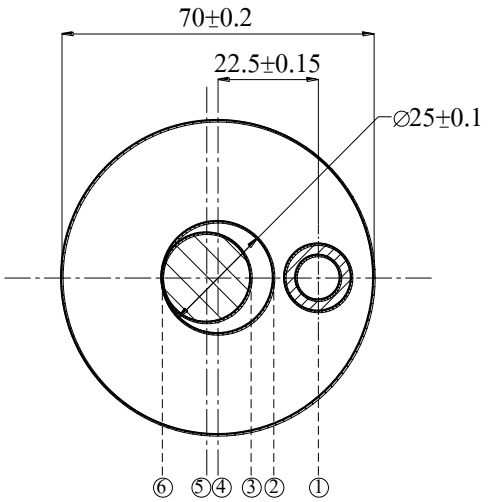
5.4 ตัวอย่างการคำนวณของชิ้นงานรูปลักษณ์กลม แบบใช้หมุดเป็นตัวกำหนดตำแหน่ง

รูปที่ 5.5 ชิ้นงานเหล็กกล้าได้รับการตัดขึ้นรูป (machining) และเจาะรู (drill) Ø 25±0.1 mm ก่อนนำมาทำขั้นตอนสุดท้ายคือเจาะรู Ø10 mm โดยอยู่ในตำแหน่งที่ห่างจากศูนย์กลางรู Ø 25 mm เป็นระยะ 22.5 ± 0.15 mm



รูปที่ 5.5 แบบงานสำหรับใช้รูเป็นรูปลักษณ์กำหนดตำแหน่ง

จากแบบงานตำแหน่งรู Ø 10 mm อ้างอิงตำแหน่งจากรูเจาะตรงกลาง จึงเลือกใช้หมุดเป็นตัวกำหนดตำแหน่ง โดยสามารถร่างแบบของจิก และกำหนดผิวอ้างอิงได้ตามรูป 5.6



รูปที่ 5.6 แบบร่างงานบนจิกแบบใช้หมุดกำหนดตำแหน่ง ในการระบุผิวอ้างอิง

5.4.1 ผลการคำนวณด้วยโปรแกรม

ผลลัพธ์ต่อไปนี้แสดงค่าเป็นข้อมูลจากหน้าจอแสดงผลลัพธ์ของโปรแกรม โดยแสดงรูปแบบของจิก ขนาด และความคลาดเคลื่อนของตำแหน่งบุชโดยอ้างอิงจากศูนย์กลางของหมุดกำหนดตำแหน่ง (1 – 5) และทำการเพิ่มเติมมิติคือ ขนาดเส้นผ่านศูนย์กลางของตัวกำหนดตำแหน่ง (3 – 6) พร้อมทั้งมิติที่เกี่ยวข้องในสมการวงรอบ ดังนี้

>>>>>>>> HORIZONTAL AXIS <<<<<<<<

งานกลมโดยใช้หมุดเป็นตัวกำหนดตำแหน่ง

Preferred Fit C8_h8

Bush position along with axis (1-5) is

22.5000 + .0435 / - .0435

From	To	Basic sizes	Tolerances
1	4	22.5	0.15
4	5	0	0.1065

Preferred Fit G8_h8

Bush position along with axis (1-5) is

22.5000 + .0435 / - .0435

From	To	Basic sizes	Tolerances
1	4	22.5	0.15
4	5	0	0.1065

Preferred Fit H8_h8

Bush position along with axis (1-5) is

22.5000 + .0435 / - .0435

From	To	Basic sizes	Tolerances
1	4	22.5	0.15

มหาวิทยาลัยเทคโนโลยีสุรนารี

4	5	0	0.1065

>>>>>>>> HORIZONTAL AXIS <<<<<<<<

งานกลมโดยใช้หมุดเป็นตัวกำหนดตำแหน่ง

Preferred Fit C8_h8

Locator size/position along with axis (3-6) is

24.7900 + 0 / - .0130

From	To	Basic sizes	Tolerances
6	2	25	0.1
2	3	0.2165	0.1065

Preferred Fit G8_h8

Locator size/position along with axis (3-6) is

24.8930 + 0 / - .0130

From	To	Basic sizes	Tolerances
6	2	25	0.1
2	3	0.1135	0.1065

Preferred Fit H8_h8

Locator size/position along with axis (3-6) is

24.9000 + 0 / - .0130

From	To	Basic sizes	Tolerances
6	2	25	0.1
2	3	0.1065	0.1065

มหาวิทยาลัยเทคโนโลยีสุรนารี

5.4.2 ผลการคำนวณโดยไม่ใช้โปรแกรม

เมื่อทำการคำนวณด้วยหลักการเดียวกับที่ใช้ในโปรแกรมทีละขั้นตอนจะได้คำตอบเช่นเดียวกับการคำนวณด้วยโปรแกรม CADBP ดังต่อไปนี้

Max. Clearance (2e) = Min. Clearance + Size Tol. for work + Size Tol. for locator

<u>ฟิตนิยม Ch</u>

2e = Ch (25) + (2 x 0.1) + IT6 (25)

= 0.11 + 0.2 + 0.013

= 0.323

N = (Max. Clearance + Min. Clearance) x $\frac{1}{2}$

= (0.323 + 0.11) x $\frac{1}{2}$ = 0.2165

± n = (Max. Clearance - Min. Clearance) x $\frac{1}{2}$

= (0.323 - 0.11) x $\frac{1}{2}$ = 0.1065

± m = (Max. Clearance - Min. Clearance) x $\frac{1}{2}$

= (0.323 - 0.11) x $\frac{1}{2}$ = 0.1065

ค่าจากการคำนวณที่มิติ (1 – 5) มาจาก มิติ (1 – 4) และ (4 – 5)

ดังนั้น ค่าจากการคำนวณที่มิติ (1 – 5) = (22.5 + 0) ± (0.15 - 0.1065)

= 22.5 ± 0.0435

ค่าจากการคำนวณที่มิติ (3 – 6) มาจาก มิติ (3 – 2) และ (2 – 6)

ดังนั้น ค่าจากการคำนวณที่มิติ (3 – 6) = (25 – 0.2165) ± (0.1065 – 0.1)

= 24.7835 ± 0.0065

หรือ = $24.79^{+0}_{-0.013}$

<u>ฟิตนิยม Gh</u>

2e = Gh (25) + (2 x 0.1) + IT6 (25)

= 0.007 + 0.2 + 0.013

= 0.220

N = (Max. Clearance + Min. Clearance) x $\frac{1}{2}$

= (0.220 + 0.007) x $\frac{1}{2}$ = 0.1135

± n = (Max. Clearance - Min. Clearance) x $\frac{1}{2}$

= (0.220 - 0.007) x $\frac{1}{2}$ = 0.1065

± m = (Max. Clearance - Min. Clearance) x $\frac{1}{2}$

= (0.220 - 0.007) x $\frac{1}{2}$ = 0.1065

ค่าจากการคำนวณที่มิติ (1 – 5) มาจาก มิติ (1 – 4) และ (4 – 5)

ดังนั้น ค่าจากการคำนวณที่มิติ (1 – 5) = (22.5 + 0) ± (0.15 - 0.1065)

= 22.5 ± 0.0435

ค่าจากการคำนวณที่มิติ (3 – 6) มาจาก มิติ (3 – 2) และ (2 – 6)

ดังนั้น ค่าจากการคำนวณที่มิติ (3 – 6) = (25 – 0.1135) ± (0 .1065 – 0.1)

= 24.8865 ± 0.0065

หรือ = $24.893^{+0}_{-0.013}$

<u>ฟิตนิยม Hh</u>

2e = Hh (25) + (2 x 0.1) + IT6 (25)

= 0 + 0.2 + 0.013

= 0.213

N = (Max. Clearance + Min. Clearance) x $\frac{1}{2}$

= (0.213 + 0) x $\frac{1}{2}$ = 0.1065

± n = (Max. Clearance - Min. Clearance) x $\frac{1}{2}$

= (0.213 - 0) x $\frac{1}{2}$ = 0.1065

± m = (Max. Clearance - Min. Clearance) x $\frac{1}{2}$

= (0.213 - 0) x $\frac{1}{2}$ = 0.1065

ค่าจากการคำนวณที่มิติ (1 – 5) มาจาก มิติ (1 – 4) และ (4 – 5)

มหาวิทยาลัยเทคโนโลยีสุรนารี

ดังนั้น ค่าจากการคำนวณที่มิติ (1 – 5) = (22.5 + 0) ± (0.15 - 0.1065)

= 22.5 ± 0.0435

ค่าจากการคำนวณที่มิติ (3 – 6) มาจาก มิติ (3 – 2) และ (2 – 6)

ดังนั้น ค่าจากการคำนวณที่มิติ (3 – 6) = (25 – 0.1065) ± (0 .1065 – 0.1)

= 24.8935 ± 0.0065

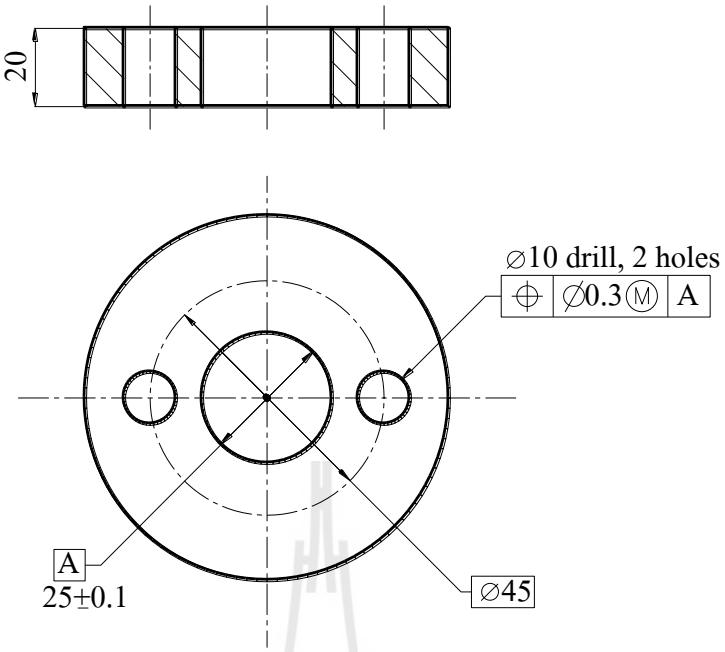
หรือ = $24.9^{+0}_{-0.013}$

5.5 ตัวอย่างการคำนวณของชิ้นงานรูปลักษณ์กลม แบบใช้หมุดเป็นตัวกำหนดตำแหน่ง เมื่อชิ้นงานมีความต้องการรูเจาะ 2 รู

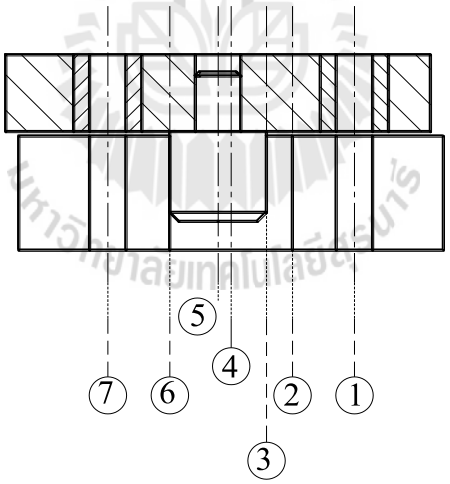
แผนการผลิต (production plan)	
10	ตัดจาก stock Ø 70 mm ยาว 38 mm
20	กลึงปาดหน้าทั้งสองด้าน
30	คว้านรู Ø 25 x 15 mm ตลอดชิ้นงาน
40	ตรวจสอบความเรียบร้อย

รูปที่ 5.7 แผนการผลิตชิ้นงานกลม 2 รูเจาะ

จากรูปที่ 5.8 เป็นแบบของชิ้นงานเหล็กกล้าคาร์บอนต่ำ ผ่านการตัด (machining) มาทุกส่วนแล้วก่อนที่จะนำมาทำการเจาะรูขนาด Ø10 mm จำนวน 2 รู โดยแผนการผลิตชิ้นงานก่อนนำมาทำการเจาะเป็นดังรูปที่ 5.7 และแบบร่างการสำหรับคำนวณหาตำแหน่งบุชแสดงดังรูปที่ 5.9



รูปที่ 5.8 แบบงานสำหรับใช้รูเป็นรูปลักษณ์กำหนดตำแหน่งเมื่อต้องการรูเจาะ 2 รู



รูปที่ 5.9 แบบร่างงานบนจิกแบบใช้หมุดเป็นตัวกำหนดตำแหน่ง
สำหรับ 2 รูเจาะ ในการระบุผิวอ้างอิง

5.5.1 ผลการคำนวณด้วยโปรแกรม

ผลลัพธ์ต่อไปนี้แสดงค่าเป็นข้อมูลจากหน้าจอแสดงผลลัพธ์ของโปรแกรม โดยแสดงรูปแบบของจิก ขนาด และความคลาดเคลื่อนของตำแหน่งบุชทีละตำแหน่งโดยในที่นี้เลือกหาตำแหน่งโหนด 7 โดยตำแหน่งรูเจาะจะอ้างอิงศูนย์กลางของรูกำหนดตำแหน่ง มิติที่ต้องการหาค่าคือ (5 – 7) นอกจากนั้นลัพธ์จากโปรแกรมยังแสดงมิติที่เกี่ยวข้องในสมการวงรอบ ดังนี้

>>>>>>>> HORIZONTAL AXIS <<<<<<<<

งานกลมโดยใช้หมุดเป็นตัวกำหนดตำแหน่ง

Preferred Fit C8_h8

Bush position along with axis (5-7) is

22.5000 + .0435 / - .0435

From	To	Basic sizes	Tolerances
7	4	22.5	0.15
4	5	0	0.1065

Preferred Fit G8_h8

Bush position along with axis (5-7) is

22.5000 + .0435 / - .0435

From	To	Basic sizes	Tolerances
7	4	22.5	0.15
4	5	0	0.1065

Preferred Fit H8_h8

Bush position along with axis (5-7) is

22.5000 + .0435 / - .0435

มหาวิทยาลัยเทคโนโลยีสุรนารี

From	To	Basic sizes	Tolerances
7	4	22.5	0.15
4	5	0	0.1065

5.5.2 ผลการคำนวณโดยไม่ใช้โปรแกรม

เมื่อทำการคำนวณด้วยหลักการเดียวกับที่ใช้ในโปรแกรมทีละขั้นตอนจะได้คำตอบเช่นเดียวกับการคำนวณด้วยโปรแกรม CADBP ดังต่อไปนี้

Max. Clearance (2e) = Min. Clearance + Size Tol. for work + Size Tol. for locator

<u>ฟิตนิยม Ch</u>

2e = Ch (25) + (2 x 0.1) + IT6 (25)

= 0.11 + 0.2 + 0.013

= 0.323

N = (Max. Clearance + Min. Clearance) x $\frac{1}{2}$

= (0.323 + 0.11) x $\frac{1}{2}$ = 0.2165

± n = (Max. Clearance - Min. Clearance) x $\frac{1}{2}$

= (0.323 - 0.11) x $\frac{1}{2}$ = 0.1065

± m = (Max. Clearance - Min. Clearance) x $\frac{1}{2}$

= (0.323 - 0.11) x $\frac{1}{2}$ = 0.1065

ค่าจากการคำนวณที่มิติ (5 – 7) มาจาก มิติ (5 – 4) และ (4 – 7)

ดังนั้น ค่าจากการคำนวณที่มิติ (5 – 7) = (22.5 + 0) ± (0.15 - 0.1065)

= 22.5 ± 0.0435

<u>ฟิตนิยม Gh</u>

2e = Gh (25) + (2 x 0.1) + IT6 (25)

= 0.007 + 0.2 + 0.013

= 0.220

N = (Max. Clearance + Min. Clearance) x $\frac{1}{2}$

= (0.220 + 0.007) x $\frac{1}{2}$ = 0.1135

± n = (Max. Clearance - Min. Clearance) x $\frac{1}{2}$

= (0.220 - 0.007) x $\frac{1}{2}$ = 0.1065

± m = (Max. Clearance - Min. Clearance) x $\frac{1}{2}$

= (0.220 - 0.007) x $\frac{1}{2}$ = 0.1065

ค่าจากการคำนวณที่มิติ (5 – 7) มาจาก มิติ (5 – 4) และ (4 – 7)

ดังนั้น ค่าจากการคำนวณที่มิติ (5 – 7) = (22.5 + 0) ± (0.15 - 0.1065)

= 22.5 ± 0.0435

<u>ฟิตนิยม Hh</u>

2e = Hh (25) + (2 x 0.1) + IT6 (25)

= 0 + 0.2 + 0.033

= 0.213

N = (Max. Clearance + Min. Clearance) x $\frac{1}{2}$

= (0.213 + 0) x $\frac{1}{2}$ = 0.1065

± n = (Max. Clearance - Min. Clearance) x $\frac{1}{2}$

= (0.213 - 0) x $\frac{1}{2}$ = 0.1065

± m = (Max. Clearance - Min. Clearance) x $\frac{1}{2}$

= (0.213 - 0) x $\frac{1}{2}$ = 0.1065

ค่าจากการคำนวณที่มิติ (5 – 7) มาจาก มิติ (5 – 4) และ (4 – 7)

ดังนั้น ค่าจากการคำนวณที่มิติ (5 – 7) = (22.5 + 0) ± (0.15 - 0.1065)

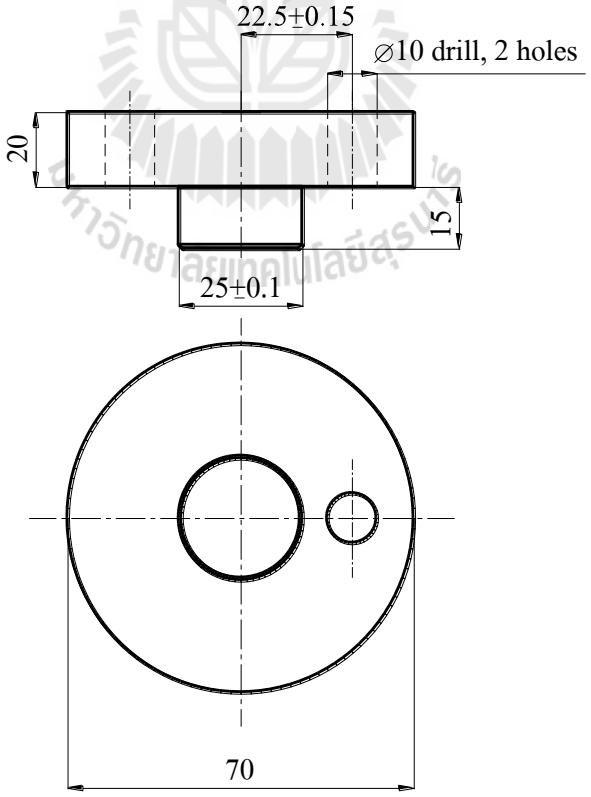
= 22.5 ± 0.0435

5.6 ตัวอย่างการคำนวณของชิ้นงานรูปลักษณ์กลม แบบใช้รูเป็นตัวกำหนดตำแหน่ง

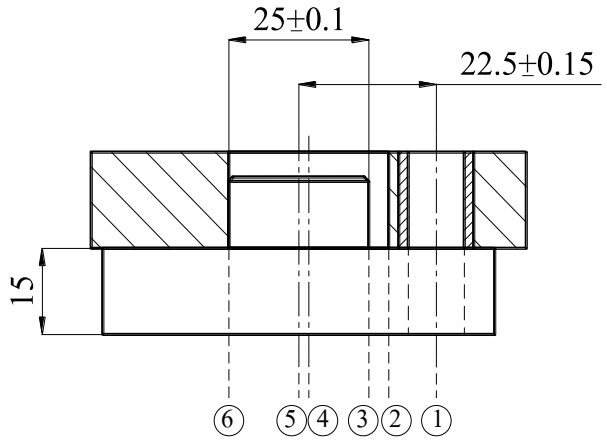
แผนการผลิต (production plan)	
10	ตัดจาก stock Ø 70 mm ยาว 38 mm
20	กลึงปาดหน้า และปอก Ø 25 x 15 mm
30	กลึงปาดหน้าให้ได้ความหนา 20 mm
40	ลบขอบคม (deburr)
50	ตรวจสอบความเรียบร้อย

รูปที่ 5.10 แผนการผลิตชิ้นงานกลม แบบมีปุ่มยื่น (spigot)

จากรูปที่ 5.11 เป็นแบบของชิ้นงานเหล็กกล้าคาร์บอนต่ำ ผ่านการตัด (machining) มาทุกส่วนแล้วก่อนที่จะนำมาทำการเจาะรูขนาด Ø10 mm เท่านั้น โดยแผนการผลิตชิ้นงานก่อนนำมาทำการเจาะเป็นดังรูปที่ 5.10 และแบบร่างการสำหรับคำนวณหาตำแหน่งบุชแสดงดังรูปที่ 5.12



รูปที่ 5.11 แบบงานสำหรับใช้ หมุด หรือ spigot เป็นรูปลักษณ์กำหนดตำแหน่ง



รูปที่ 5.12 แบบร่างงานบนจิกแบบใช้รูเพื่อเป็นตัวกำหนดตำแหน่งในการระบุผิวอ้างอิง

5.6.1 ผลการคำนวณด้วยโปรแกรม

ผลลัพธ์ต่อไปนี้แสดงค่าเป็นข้อมูลจากหน้าจอแสดงผลลัพธ์ของโปรแกรม โดยแสดงรูปแบบของจิก ขนาด และความคลาดเคลื่อนของตำแหน่งบุชโดยอ้างอิงจากศูนย์กลางของรูกำหนดตำแหน่ง (1 – 4) และทำการเพิ่มเติมมิติคือ ขนาดเส้นผ่านศูนย์กลางของรูกำหนดตำแหน่ง (2 – 6) พร้อมทั้งมิติที่เกี่ยวข้องในสมการวงรอบ ดังนี้

```
>>>>>>>> HORIZONTAL AXIS <<<<<<<<
งานกลมโดยใช้รูเป็นตัวกำหนดตำแหน่ง


Preferred Fit  C8_h8
Bush position along with axis (1-4) is
        22.5000  + .0435 / - .0435


From    To      Basic sizes     Tolerances
1       5       22.5            0.15
5       4       0               0.1065


Preferred Fit  G8_h8
Bush position along with axis (1-4) is
        22.5000  + .0435 / - .0435
```

From	To	Basic sizes	Tolerances
1	5	22.5	0.15
5	4	0	0.1065

Preferred Fit H8_h8

Bush position along with axis (1-4) is

22.5000 + .0435 / - .0435

From	To	Basic sizes	Tolerances
1	5	22.5	0.15
5	4	0	0.1065

5.6.2 ผลการคำนวณโดยไม่ใช้โปรแกรม

เมื่อทำการคำนวณด้วยหลักการเดียวกับที่ใช้ในโปรแกรมทีละขั้นตอนจะได้คำตอบเช่นเดียวกับการคำนวณด้วยโปรแกรม CADBP ดังต่อไปนี้

Max. Clearance (2e) = Min. Clearance + Size Tol. for work + Size Tol. for locator

<u>ฟิตนิยม Ch</u>

$2e = Ch(25) + (2 \times 0.1) + IT6(25)$

$= 0.11 + 0.2 + 0.013$

$= 0.323$

$N = (\text{Max. Clearance} + \text{Min. Clearance}) \times \frac{1}{2}$

$= (0.323 + 0.11) \times \frac{1}{2} = 0.2165$

$\pm n = (\text{Max. Clearance} - \text{Min. Clearance}) \times \frac{1}{2}$

$= (0.323 - 0.11) \times \frac{1}{2} = 0.1065$

$\pm m = (\text{Max. Clearance} - \text{Min. Clearance}) \times \frac{1}{2}$

$= (0.323 - 0.11) \times \frac{1}{2} = 0.1065$

ค่าจากการคำนวณที่มิติ (1 – 4) มาจาก มิติ (1 – 5) และ (5 – 4)

ดังนั้น ค่าจากการคำนวณที่มิติ (1 – 4) = (22.5 - 0) ± (0.15 - 0.1065)

= 22.5 ± 0.0435

ค่าจากการคำนวณที่มิติ (2 – 6) มาจาก มิติ (2 – 3) และ (3 – 6)

ดังนั้น ค่าจากการคำนวณที่มิติ (2 – 6) = (25 + 0.2165) ± (0.1065 – 0.1)

= 25.2165 ± 0.0065

ฟิตนิยม Gh

2e = Gh (25) + (2 x 0.1) + IT6 (25)

= 0.007 + 0.2 + 0.013

= 0.220

N = (Max. Clearance + Min. Clearance) x $\frac{1}{2}$

= (0.220 + 0.007) x $\frac{1}{2}$ = 0.1135

± n = (Max. Clearance - Min. Clearance) x $\frac{1}{2}$

= (0.220 - 0.007) x $\frac{1}{2}$ = 0.1065

± m = (Max. Clearance - Min. Clearance) x $\frac{1}{2}$

= (0.24 - 0.007) x $\frac{1}{2}$ = 0.1065

ค่าจากการคำนวณที่มิติ (1 – 4) มาจาก มิติ (1 – 5) และ (5 – 4)

ดังนั้น ค่าจากการคำนวณที่มิติ (1 – 4) = (22.5 - 0) ± (0.15 - 0.1065)

= 22.5 ± 0.0435

ค่าจากการคำนวณที่มิติ (2 – 6) มาจาก มิติ (2 – 3) และ (3 – 6)

ดังนั้น ค่าจากการคำนวณที่มิติ (2 – 6) = (25 + 0.1135) ± (0.1065 – 0.1)

= 25.1135 ± 0.0065

ฟิตนิยม Hh

2e = Hh (25) + (2 x 0.1) + IT6 (25)

= 0 + 0.2 + 0.013

= 0.213

N = (Max. Clearance + Min. Clearance) x $\frac{1}{2}$

= (0.213 + 0) x $\frac{1}{2}$ = 0.1065

± n = (Max. Clearance - Min. Clearance) x $\frac{1}{2}$

= (0.213 - 0) x $\frac{1}{2}$ = 0.1065

± m = (Max. Clearance - Min. Clearance) x $\frac{1}{2}$

= (0.213 - 0) x $\frac{1}{2}$ = 0.1065

ค่าจากการคำนวณที่มิติ (1 – 4) มาจาก มิติ (1 – 5) และ (5 – 4)

ดังนั้น ค่าจากการคำนวณที่มิติ (1 – 4) = (22.5 - 0) ± (0.15 - 0.1065)

= 22.5 ± 0.0435

ค่าจากการคำนวณที่มิติ (2 – 6) มาจาก มิติ (2 – 3) และ (3 – 6)

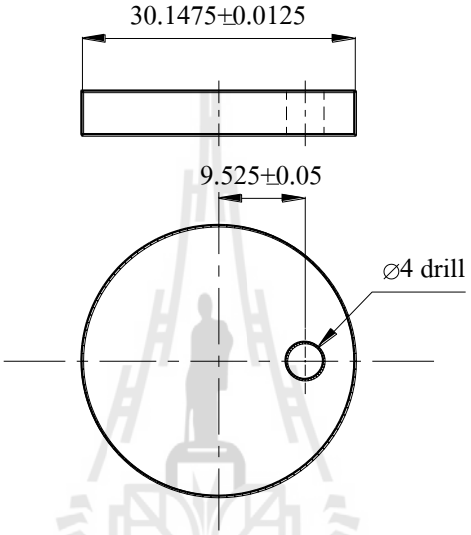
ดังนั้น ค่าจากการคำนวณที่มิติ (2 – 6) = (25 + 0.1065) ± (0.1065 – 0.1)

= 25.1065 ± 0.0065

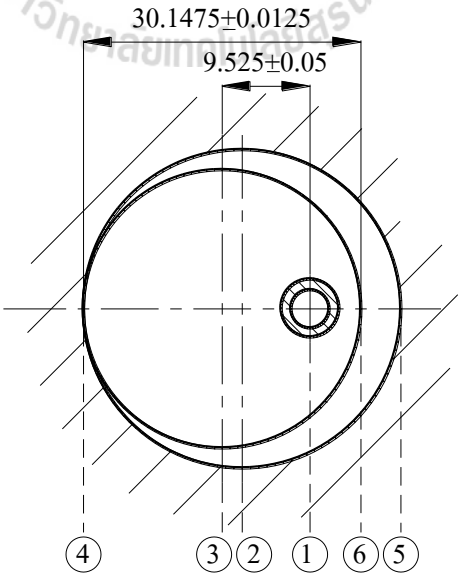
มหาวิทยาลัยเทคโนโลยีสุรนารี

5.7 ตัวอย่างการคำนวณของชิ้นงานรูปลักษณ์กลม แบบใช้รังเป็นตัวกำหนดตำแหน่ง

จากรูป 5.13 เป็นแบบของชิ้นงานเหล็กกล้า AISI 1020 เลื่อยตัดจากแท่งเหล็กกล้า (bar stock) ขนาด $\varnothing \frac{30.160}{30.135}$ mm จากนั้นชิ้นงานจะถูกกลึงปาดหน้า และลบคมขอบงาน โดยขั้นตอนสุดท้ายชิ้นงานจะถูกนำมาเจาะรู Ø 4 mm โดยแบบร่างการสำหรับคำนวณหาตำแหน่งบุชแสดงดังรูปที่ 5.14



รูปที่ 5.13 แบบงานสำหรับใช้รังโดยมีผิวขอบงานเป็นรูปลักษณ์กำหนดตำแหน่ง



รูปที่ 5.14 แบบร่างงานบนจิกแบบใช้รังในการกำหนดตำแหน่งในการระบุผิวอ้างอิงจาก

5.7.1 ผลการคำนวณด้วยโปรแกรม

ผลลัพธ์ต่อไปนี้แสดงค่าเป็นข้อมูลจากหน้าจอแสดงผลลัพธ์ของโปรแกรม โดยแสดงรูปแบบของจิก ขนาด และความคลาดเคลื่อนของตำแหน่งบุชโดยอ้างอิงจากศูนย์กลางรังกำหนดตำแหน่ง (1 – 2) และทำการเพิ่มมิติคือ ขนาดเส้นผ่านศูนย์กลางของรังกำหนดตำแหน่ง (4 – 5) พร้อมทั้งมิติที่เกี่ยวข้องในสมการวงรอบ ดังนี้

>>>>>>>> HORIZONTAL AXIS <<<<<<<<

งานกลมโดยใช้รัง

Preferred Fit C8_h8

Bush position along with axis (1-2) is

9.5250 + .0295 / - .0295

From	To	Basic sizes	Tolerances
1	3	9.525	0.05
3	2	0	0.0205

Preferred Fit G8_h8

Bush position along with axis (1-2) is

9.5250 + .0295 / - .0295

From	To	Basic sizes	Tolerances
1	3	9.525	0.05
3	2	0	0.0205

Preferred Fit H8_h8

Bush position along with axis (1-2) is

9.5250 + .0295 / - .0295

มหาวิทยาลัยเทคโนโลยีสุรนารี

From	To	Basic sizes	Tolerances
1	3	9.525	0.05
3	2	0	0.0205

>>>>>>>> HORIZONTAL AXIS <<<<<<<<

งานกลมโดยใช้รัง

Preferred Fit C8_h8

Locator size/position along with axis (4-5) is

30.2800 + .0160 / - 0

From	To	Basic sizes	Tolerances
4	6	30.1475	0.0125
6	5	0.1405	0.0205

Preferred Fit G8_h8

Locator size/position along with axis (4-5) is

30.1690 + .0160 / - 0

From	To	Basic sizes	Tolerances
4	6	30.1475	0.0125
6	5	0.0295	0.0205

Preferred Fit H8_h8

Locator size/position along with axis (4-5) is

30.1600 + .0160 / - 0

From	To	Basic sizes	Tolerances
4	6	30.1475	0.0125
6	5	0.0205	0.0205

มหาวิทยาลัยเทคโนโลยีสุรนารี

5.7.2 ผลการคำนวณโดยไม่ใช้โปรแกรม

เมื่อทำการคำนวณด้วยหลักการเดียวกับที่ใช้ในโปรแกรมทีละขั้นตอนจะได้คำตอบเช่นเดียวกับการคำนวณด้วยโปรแกรม CADBP ดังต่อไปนี้

Max. Clearance (2e) = Min. Clearance + Size Tol. for work + Size Tol. for locator

ฟิตนิยม Ch

2e = Ch (30.1475) + (2 x 0.0125) + IT6 (30.1475)

= 0.12 + 0.025 + 0.016

= 0.161

N = (Max. Clearance + Min. Clearance) x $\frac{1}{2}$

= (0.161 + 0.12) x $\frac{1}{2}$ = 0.1405

± n = (Max. Clearance - Min. Clearance) x $\frac{1}{2}$

= (0.161 - 0.12) x $\frac{1}{2}$ = 0.0205

± m = (Max. Clearance - Min. Clearance) x $\frac{1}{2}$

= (0.161 - 0.12) x $\frac{1}{2}$ = 0.0205

ค่าจากการคำนวณที่มิติ (1 – 2) มาจาก มิติ (1 – 3) และ (3 – 2)

ดังนั้น ค่าจากการคำนวณที่มิติ (1 – 2) = (9.525 - 0) ± (0.05 - 0.0205)

= 9.525 ± 0.0295

ค่าจากการคำนวณที่มิติ (4 – 5) มาจาก มิติ (4 – 6) และ (6 – 5)

ดังนั้น ค่าจากการคำนวณที่มิติ (4 – 5) = (30.1475 + 0.1405)

± (0.0205 – 0.0125)

= 30.288 ± 0.008

= 30.28^{+0016}_{-0}

ฟิตนิยม Gh

2e = Gh (30.1475) + (2 x 0.0125) + IT6 (30.1475)

= 0.009 + 0.025 + 0.016

= 0.050

N = (Max. Clearance + Min. Clearance) x $\frac{1}{2}$

= (0.050 + 0.009) x $\frac{1}{2}$ = 0.095

± n = (Max. Clearance - Min. Clearance) x $\frac{1}{2}$

= (0.050 - 0.009) x $\frac{1}{2}$ = 0.0205

± m = (Max. Clearance - Min. Clearance) x $\frac{1}{2}$

= (0.073 - 0.009) x $\frac{1}{2}$ = 0.0205

ค่าจากการคำนวณที่มิติ (1 – 2) มาจาก มิติ (1 – 3) และ (3 – 2)

ดังนั้น ค่าจากการคำนวณที่มิติ (1 – 2) = (9.525 - 0) ± (0.05 - 0.0205)

= 9.525 ± 0.0295

ค่าจากการคำนวณที่มิติ (4 – 5) มาจาก มิติ (4 – 6) และ (6 – 5)

ดังนั้น ค่าจากการคำนวณที่มิติ (4 – 5) = (30.1475 + 0.0295)

± (0.0205 – 0.0125)

= 30.177 ± 0.008

= 30.169^{+0016}_{-0}

<u>ฟิตนิยม Hh</u>

2e = Hh (30.1475) + (2 x 0.0125) + IT6 (30.1475)

= 0 + 0.025 + 0.016

= 0.041

N = (Max. Clearance + Min. Clearance) x $\frac{1}{2}$

= (0.041 + 0) x $\frac{1}{2}$ = 0.0205

± n = (Max. Clearance - Min. Clearance) x $\frac{1}{2}$

= (0.041 - 0) x $\frac{1}{2}$ = 0.0205

± m = (Max. Clearance - Min. Clearance) x $\frac{1}{2}$

= (0.041 - 0) x $\frac{1}{2}$ = 0.0205

ค่าจากการคำนวณที่มิติ (1 – 2) มาจาก มิติ (1 – 3) และ (3 – 2)

ดังนั้น ค่าจากการคำนวณที่มิติ (1 – 2) = (9.525 - 0) ± (0.05 - 0.0205)

= 9.525 ± 0.0295

ค่าจากการคำนวณที่มิติ (4 – 5) มาจาก มิติ (4 – 6) และ (6 – 5)

ดังนั้น ค่าจากการคำนวณที่มิติ (4 – 5) = (30.1475 + 0.0205)

± (0.0205 – 0.0125)

= 30.168 ± 0.008

= 30.160^{+0016}_{-0}

5.8 สรุป

จากการทดสอบโปรแกรมด้วยการเปรียบเทียบผลลัพธ์ที่ได้จากโปรแกรมในหน้าจอแสดงผล กับผลลัพธ์ที่เกิดจากการคำนวณเป็นขั้นตอนโดยไม่ใช้โปรแกรมพบว่า จากตัวอย่างการคำนวณจิกที่มีรูปแบบแตกต่างกัน ค่าที่ได้จากการคำนวณของโปรแกรมทั้งหมดให้ผลลัพธ์ตรงกับการคำนวณโดยที่ไม่ใช้โปรแกรม

มหาวิทยาลัยเทคโนโลยีสุรนารี

บทที่ 6
สรุปผลการวิจัย

6.1 สรุป

การใช้คอมพิวเตอร์ช่วยในการกำหนดตำแหน่งบุชในการออกแบบจิกสำหรับเจาะรูอาศัยข้อสรุปของหลักการโครงข่ายแผนภาพต้นไม้ (Skeleton of rooted tree graph) ในการสร้างความสัมพันธ์ระหว่างความต้องการของชิ้นงานในแบบกับตัวกำหนดตำแหน่ง และตำแหน่งบุชในจิกที่ใช้ในการจับยึด ด้วยความสัมพันธ์เช่นนี้ (CADBP, Computer – Aided Determination of Bush Position) ที่ถูกพัฒนาขึ้นไม่เพียงแต่จะแสดงค่าผลลัพธ์ของขนาดมิติที่ผู้ใช้โปรแกรมต้องการครั้งละ 3 ชุดทางเลือก (เรียงลำดับตามฟิตนิยมของงานสวม 3 ระดับตั้งแต่สวมคลอนมากถึงสวมคลอนน้อยที่สุด) ยังแสดงมิติใช้งานในวงรอบที่ส่งผลกับความต้องการใช้งานเมื่อมีการเปลี่ยนแปลงมิตินั้นด้วย ทำให้ผู้ใช้งานสามารถทำการเปลี่ยนแปลง หรือปรับปรุงแบบงานให้เหมาะสมยิ่งขึ้นหากผู้ออกแบบมีกระบวนการผลิตที่มีความสามารถสูงกว่าเกรดความคลาดเคลื่อนมาตรฐาน IT6 ที่กำหนดใช้กับโปรแกรมนี้

กล่าวได้ว่า CADBP เป็นผลของงานวิจัยในขั้นทดลองเพื่อหาวิธีการใหม่ ๆ สำหรับออกแบบจิกเจาะรู ซึ่งบรรลุวัตถุประสงค์ของการวิจัยเป็นส่วนใหญ่ แม้ว่าจะไม่สามารถครอบคลุมปัญหาการออกแบบจิกทุกประเภท หรือจำกัดเฉพาะรูปทรงชิ้นงานง่าย ๆ ก็ตาม แต่โปรแกรมนี้ก็สามารถแสดงให้เห็นว่า เทคนิคของ tolerance analysis สามารถนำมาใช้เป็นพื้นฐานเชื่อมโยงกันของมิติในชิ้นส่วนต่าง ๆ ของจิก และนำไปสู่ตำแหน่งของบุชที่ต้องการได้

6.2 ข้อจำกัดของโปรแกรม

เนื่องจากโปรแกรมที่พัฒนาขึ้นนี้ใช้หลักการพิจารณาความสัมพันธ์ในความคลาดเคลื่อนแบบสี่เหลี่ยม (rectangular tolerance) ของมิติเท่านั้น ดังนั้นความคลาดเคลื่อนของมิติในเชิงเรขาคณิตที่จะใช้ป้อนเข้าสู่โปรแกรมจึงต้องถูกแปลงให้เป็นความคลาดเคลื่อนแบบสี่เหลี่ยมก่อน นอกจากนั้นแล้วประเภทของจิกในโปรแกรมยังครอบคลุมเฉพาะ plate jig, และ template jig บางชนิดเท่านั้น

6.3 แนวทางวิจัยในอนาคต

ในปัจจุบันการระบุความคลาดเคลื่อนเชิงเรขาคณิตลงในแบบงานได้รับความนิยมเป็นอย่างมาก ด้วยวัตถุประสงค์ที่สามารถแสดงการใช้งานรูปลักษณ์ได้ชัดเจน และยังช่วยให้การตรวจรับชิ้นงานตรงตามวัตถุประสงค์การใช้งานด้วย โปรแกรมช่วยออกแบบจิกจึงควรรองรับการออกแบบที่ชิ้นงานมีการให้ความคลาดเคลื่อนเชิงเรขาคณิตด้วย นอกจากนั้นแล้วด้วยเทคโนโลยีที่ทันสมัยการออกแบบชิ้นส่วนย่อมมีความซับซ้อนขึ้น โปรแกรมช่วยกำหนดตำแหน่งสำหรับการออกแบบจิกต้องสามารถออกแบบได้โดยมีรูปแบบของจิกที่หลากหลาย และครอบคลุมลักษณะชิ้นงานต่างๆ มากขึ้นด้วย

มหาวิทยาลัยเทคโนโลยีสุรนารี

รายการอ้างอิง

ยงยุทธ เสริมสุธีอนุวัฒน์ (2556). **หลักวิศวกรรมเครื่องมือเบื้องต้น: การออกแบบจิก, ฟิกซ์เจอร์และเกจตรวจสอบ**. โรงพิมพ์ 21 เซ็นจูรี่.

Cole, C. B. (1946). **Tool Design**. American Technical Society

Cutkosky, M. R., Kurokawa, E., & Wright, P. K. (1982). **Programmable Conformable Clamps**. Technical Paper - Society of Manufacturing Engineers. MS: 8p.

Doyle, L. E. (1950). **Tool engineering: analysis and procedure**. Prentice-Hall.

Drake, P. (1999). **Dimensioning and Tolerancing Handbook**. McGraw-Hill Education.

Gladman, C. A. (1972). **Manual for geometric analysis of engineering designs**. Sydney: Australian, trade publications.

Henriksen, E. K. (1973). **Jig and Fixture Design Manual**. Industrial Press.

Leopold, J. (2008). **Clamping Modeling - State-of-the-Art and Future Trends**. In C. Xiong, H. Liu, Y.

Liggett, J. V. (1993). **Dimensional variation management handbook: a guide for quality, design, and manufacturing engineers**. Prentice Hall.

Nee, A. Y. C., Whybrew, K., & kumar, A. S. (1995). **Advanced Fixture Design for FMS**. Springer - Verlag.

Orawan, N., & Yongyooth, S. (2009). **An optimization technique for tolerance allocation when datum for dimensioning changes**. Paper presented at the IE Network Conference 2009.

Qin, G. H., Zhang, W. H., Wan, M., & Qin, G. H. (2006). **A mathematical approach to analysis and optimal design of a fixture locating scheme**. The International Journal of Advanced Manufacturing Technology. 29(3-4): 349-359.

Raghu, A., & Melkote, S. N. (2004). **Analysis of the effects of fixture clamping sequence on part location errors**. International Journal of Machine Tools and Manufacture. 44(4): 373-382.

Sermsuti-Anuwat, Y. (1992). **Computer-aided Process Planning and Fixture Design (CAPPFD)**. A Thesis Submitted for the Degree of Doctor of Philosophy in Mechanical Engineering at the University of Canterbury, Christchurch, New Zealand: University of Canterbury.

Sermsuti-anuwat, Y. (2009). **Milling fixture design: a tolerance analysis approach**. International Journal of Mechanical Engineering Education. 37(2): 111-117.

Sermsuti-anuwat, Y. (2014). Circulation Notes, **Concepts of rooted tree graph for establishing dimensional relationship between part components and assembly function and their application in jig design**. Dated 08/09/2014.

Siebenaler, S. P., & Melkote, S. N. (2006). **Prediction of workpiece deformation in a fixture system using the finite element method**. International Journal of Machine Tools and Manufacture. 46(1): 51-58.

Tool, A. S. o., & Engineers, M. (1962). **Handbook of fixture design: a practical reference book of workholding principles and designs for all classes of machining, assembly, and inspection**. McGraw-Hill.

Whybrew, K., Britton, G. A., Robinson, D. F., & Sermsuti-anuwat, Y. (1990). **A graph-theoretic approach to tolerance charting**. The International Journal of Advanced Manufacturing Technology. 5(2): 175-183.

มหาวิทยาลัยเทคโนโลยีสุรนารี

ภาคผนวก ก

คู่มือการใช้งานโปรแกรม CADBP

คู่มือการใช้งานโปรแกรม CADBP

โปรแกรม “ CADBP ” (Computer – aided determination of bush position) สามารถนำมาใช้ในการหาตำแหน่งบุช หรือขนาดของมิติใช้งานที่เกี่ยวข้องกับตำแหน่งบุช ซึ่งผู้ใช้โปรแกรมต้องมีพื้นฐานในการเลือกใช้ประเภทของจิกกับลักษณะงานที่ต้องการนำมาใช้กับโปรแกรม โดยมีข้อมูลนำเข้าที่ผู้ใช้ต้องเตรียมก่อนการใช้โปรแกรมดังนี้

(1) แบบของชิ้นงานที่ระบุขนาด และตำแหน่งของรูเจาะบนชิ้นงาน

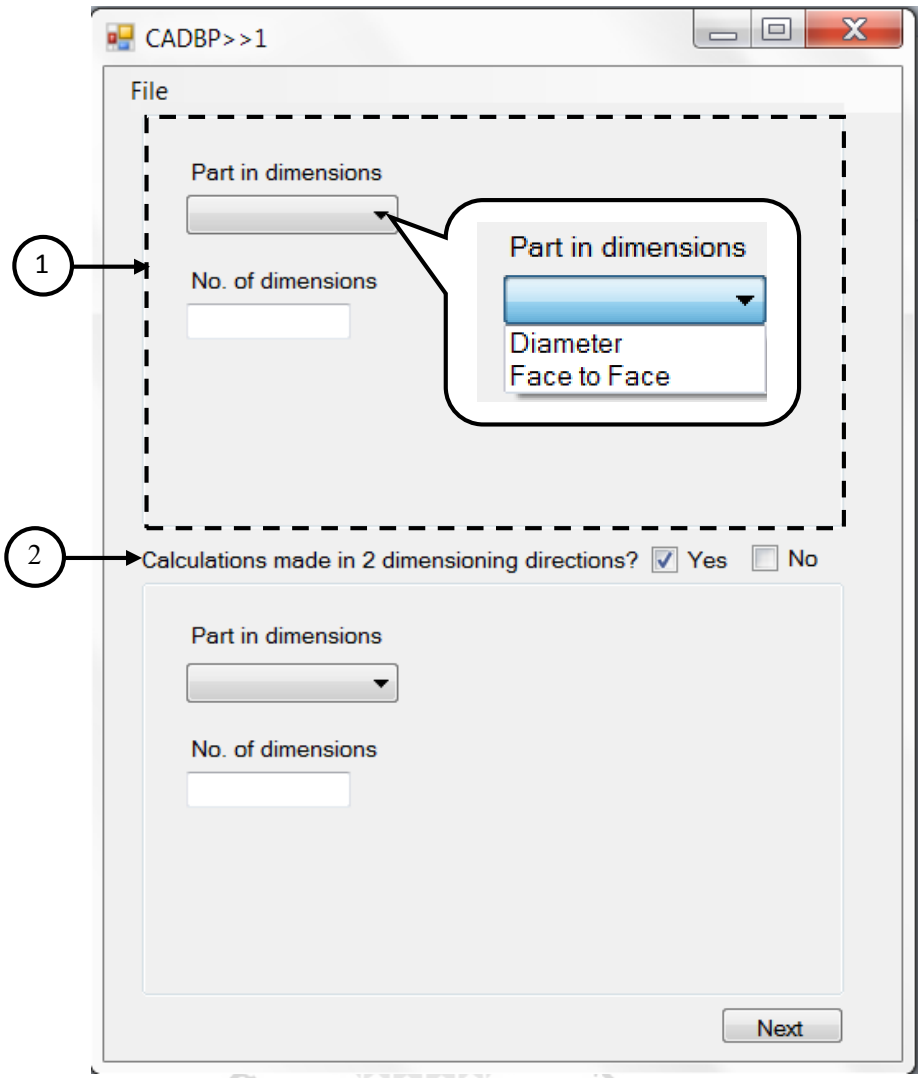
(2) ประเภทของจิก (ลักษณะชิ้นงาน, รูปลักษณ์กำหนดตำแหน่งของชิ้นงาน (locating feature))

โปรแกรมนี้ จะประกอบด้วยฟอร์ม (window form) สำหรับรับข้อมูล 2 ฟอร์ม และฟอร์มสำหรับแสดงผล โดยแต่ละฟอร์มจะมีแถบเมนูสำหรับสั่งงาน ลักษณะกล่องรับข้อมูล และแสดงผลต่างกันไป การใช้งานฟอร์มจะเริ่มจากการระบุข้อมูลที่ฟอร์ม “ CADBP>>1 ”, “ CADBP>>2 ” และเมื่อประมวลผลเรียบร้อยโปรแกรมจะแสดงผลลัพธ์ที่ “ CADBP>>OUTPUT ” ตามลำดับ

1 การป้อนข้อมูล (Input data)

การป้อนข้อมูลจะอยู่ในส่วนของหน้าต่าง “ CADBP>>1 ” และ “ CADBP>>2 ” เมื่อผู้ใช้เข้าสู่โปรแกรมจะปรากฏหน้าจอรับข้อมูลแรกชื่อว่า “ CADBP>>1 ”ในลักษณะดังรูปที่ M1 หน้าจอนี้จะประกอบด้วยแถบเมนู ส่วนรับข้อมูลเข้า และปุ่มดำเนินการ

ในหน้าจอรับข้อมูลแรกดังรูปที่ M1, (CADBP>>1) จะทำหน้าที่ในการรับข้อมูลเบื้องต้นประกอบไปด้วยวิธีการบอกขนาดบอกขนาด และจำนวนมิติที่ผู้ใช้ต้องการระบุ



มหาวิทยาลัยเทคโนโลยีสุรนารี

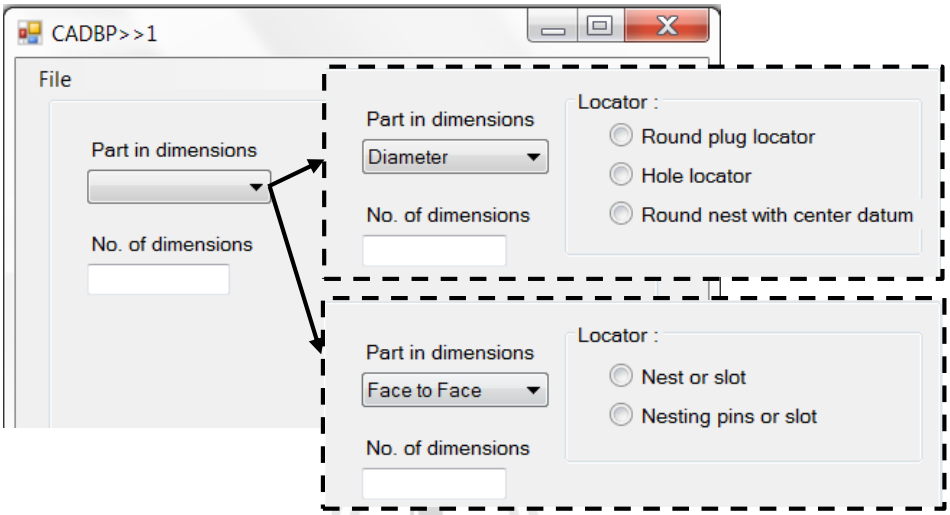
รูปที่ M1 ส่วนรับข้อมูลหน้าต่าง CADBP>>1

ในการเลือกวิธีการบอกขนาดชิ้นงานสามารถเลือกใช้ได้ตามลักษณะงาน และการอ้างอิงรูปลักษณ์ในการให้ขนาดบนมิติที่บอกขนาดชิ้นงานที่ผู้ใช้ต้องการออกแบบจิกดังรูปที่ M2 คือ

Diameter ใช้ในกรณีที่ชิ้นงานมีลักษณะกลม
หรือมีผิวโค้งเป็นรูปลักษณ์อ้างอิงมิติ

Face to Face ใช้สำหรับกรณีงานเป็นรูปหลายเหลี่ยม
หรือมีผิวระนาบเป็นรูปลักษณ์อ้างอิงมิติ

วิธีการบอกขนาดของมิติที่ถูกเลือกจะบ่งบอกถึงรูปลักษณ์ของชิ้นงาน ทำให้จำแนกลักษณะของจิกที่สามารถใช้งานกับรูปลักษณ์ของงานแต่ละแบบได้แตกต่างกัน เมื่อเลือกรูปแบบวิธีการบอกขนาดจะปรากฏลักษณะการกำหนดตำแหน่งที่เป็นไปได้ตามลักษณะงานให้เลือกทางด้านขวามือ ดังรูปที่ M2 เพื่อให้ผู้ใช้เลือกโดยการทำเครื่องหมายด้านหน้าตัวเลือกนั้นๆ



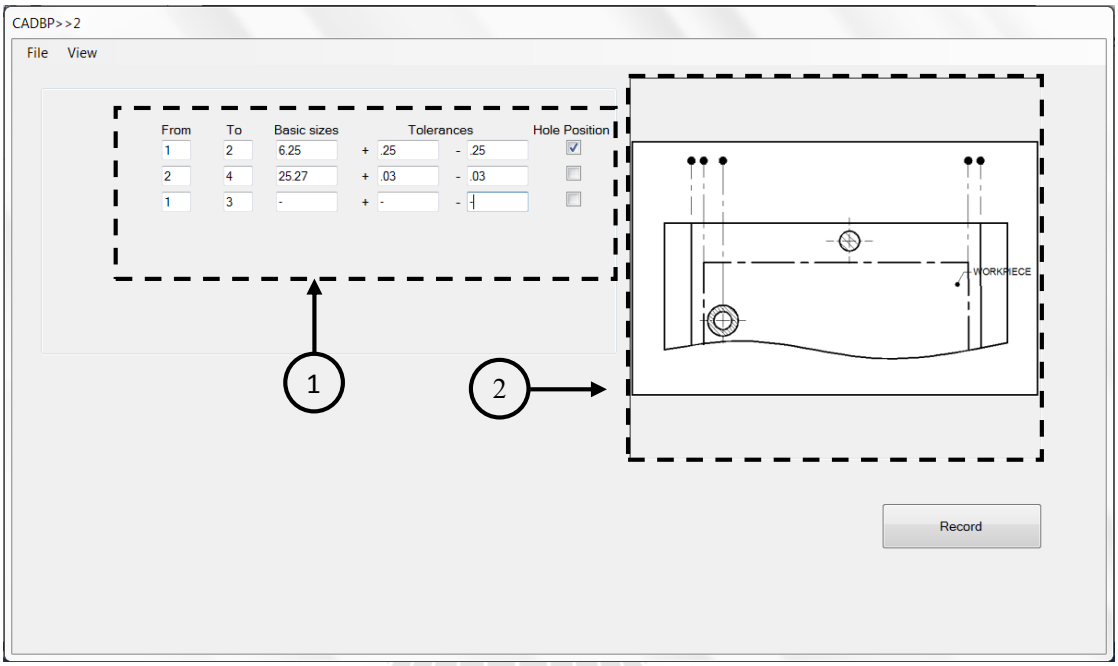
รูปที่ M2 ตัวเลือกการบอกขนาดมิติ

จากตารางในรูปที่ M3 แสดงลักษณะการกำหนดตำแหน่งของจิกที่สามารถใช้งานได้กับรูปลักษณ์งานตามแต่ที่ผู้ใช้ระบุจากวิธีการบอกการบอกขนาดชิ้นงาน คือการบอกลักษณะเป็นเส้นผ่านศูนย์กลาง (Diameter) หรือ จากผิวระนาบหนึ่งไปสู่ผิวระนาบหนึ่ง (Face to Face)

วิธีการบอกขนาดชิ้นงาน	**การกำหนดตำแหน่ง (locating)**
Diameter	หมุดเป็นตัวกำหนดตำแหน่ง, (Round plug locator)
	รูเป็นตัวกำหนดตำแหน่ง, (Hole locator)
	รัง (Round nest with center datum)
Plane - Plane	รัง/ราง, (Nest or slot)
	ใช้หมุดทำรัง/ราง, (Nesting pins or slot)

รูปที่ M3 ตัวเลือกการกำหนดตำแหน่ง

เมื่อระบุข้อมูลในส่วนรับข้อมูลทั้งหมดแล้ว หากผู้ใช้ต้องการเพิ่มแนวแกนของการคำนวณอีก 1 มิติ ให้ทำเครื่องหมายที่หมายเลข 2 ในรูปที่ M1 ส่วนรับข้อมูลจะปรากฏขึ้นอีก 1 ชุด สำหรับรับข้อมูลในกรณีที่ต้องการคำนวณใน 2 มิติพร้อมกัน ข้อมูลเบื้องต้นที่ระบุในหน้าต่างนี้จะนำเข้าสู่หน้าต่างถัดไป โดยปุ่ม “Next” ที่ด้านล่างของหน้าต่าง เมื่อกดปุ่มแล้วจะปรากฏหน้าจอถัดไปซึ่งใช้รับรายละเอียดของข้อมูลป้อนเข้าสู่โปรแกรมดังรูปที่ M4



รูปที่ M4 ส่วนรับข้อมูลหน้าต่าง CADBP>>2

หน้าจอ CADBP>>2 ทำหน้าที่รับรายละเอียดข้อมูลคือ โหนดอ้างอิงมิติ ขนาดของมิติ ความคลาดเคลื่อนของมิติ และตำแหน่งของรูเจาะจากการทำเครื่องหมายช่องท้ายสุดของมิตินั้น และมิติที่ผู้ใช้ต้องการทราบขนาดจะเว้นช่องว่างหรือขีด (-) ที่ Basic Sizes และ Tolerances โดยจำนวนแถวที่ปรากฏขึ้นเพื่อใช้ป้อนมิติลงไปขึ้นอยู่กับจำนวนมิติทั้งหมดที่ผู้ใช้ป้อนข้อมูลในหน้าแรก เมื่อป้อนข้อมูลในส่วนที่ 1 เรียบร้อยแล้วให้กดปุ่ม “Record” เพื่อเก็บข้อมูลรอการประมวลผล และโปรแกรมจะถามหาค่าตำแหน่งโหนดที่ผู้ใช้อ้างถึงในการให้ค่าของมิติ ผู้ใช้จะต้องระบุตำแหน่งแต่ละโหนดเรียงจากโหนดที่ 1 ตามลำดับไปเรื่อยๆจนครบจำนวนโหนดสูงสุดที่ผู้ใช้อ้างถึง เป็นการสิ้นสุดการป้อนข้อมูล

2 การบันทึกข้อมูล

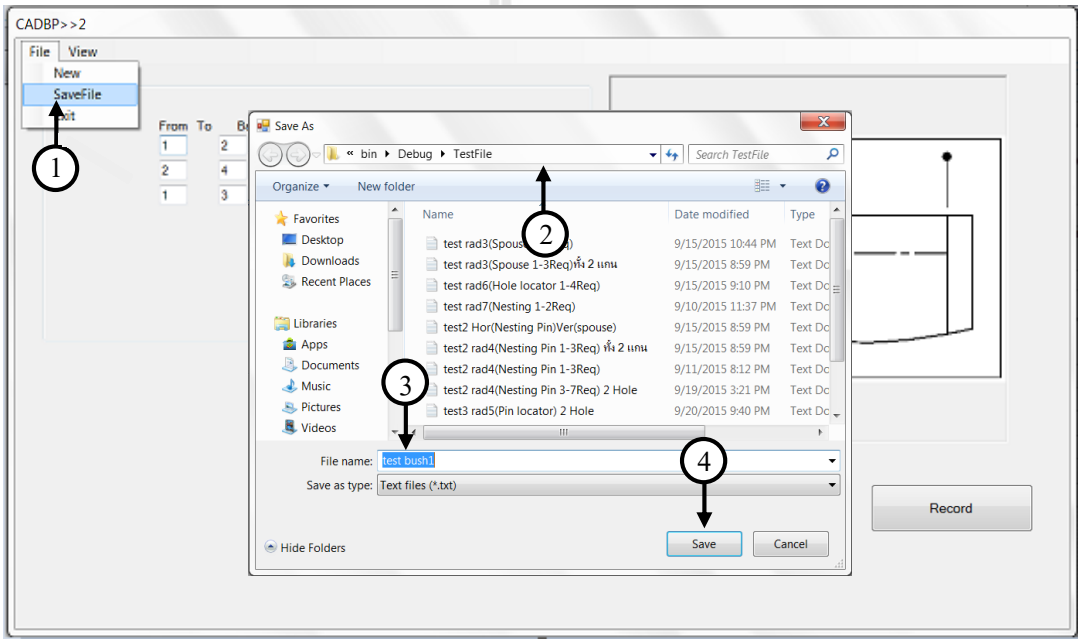
หลังจากที่ผู้ใช้ป้อนข้อมูลเสร็จเรียบร้อยแล้ว ผู้ใช้สามารถทำการบันทึกข้อมูลไว้ในหน่วยความจำได้ที่คำสั่งบนแถบเมนูในหน้าจอ CADBP>>2โดยมีขั้นตอนการบันทึกดังนี้

1) คลิกที่คำสั่ง File > SaveFile บนแถบเมนู จะมีหน้าต่าง Save As แสดงขึ้นมาดังรูปที่ 5

2) เลือกโฟลเดอร์ที่ต้องการบันทึกข้อมูล

3) ป้อนชื่อไฟล์ที่ต้องการบันทึกข้อมูลลงในช่อง File name

4) คลิกที่ปุ่ม Save ข้อมูลจะถูกบันทึกเป็นไฟล์นามสกุล .txt



รูปที่ M5 หน้าต่างคำสั่งบันทึกข้อมูล

3 การเปิดไฟล์ข้อมูล

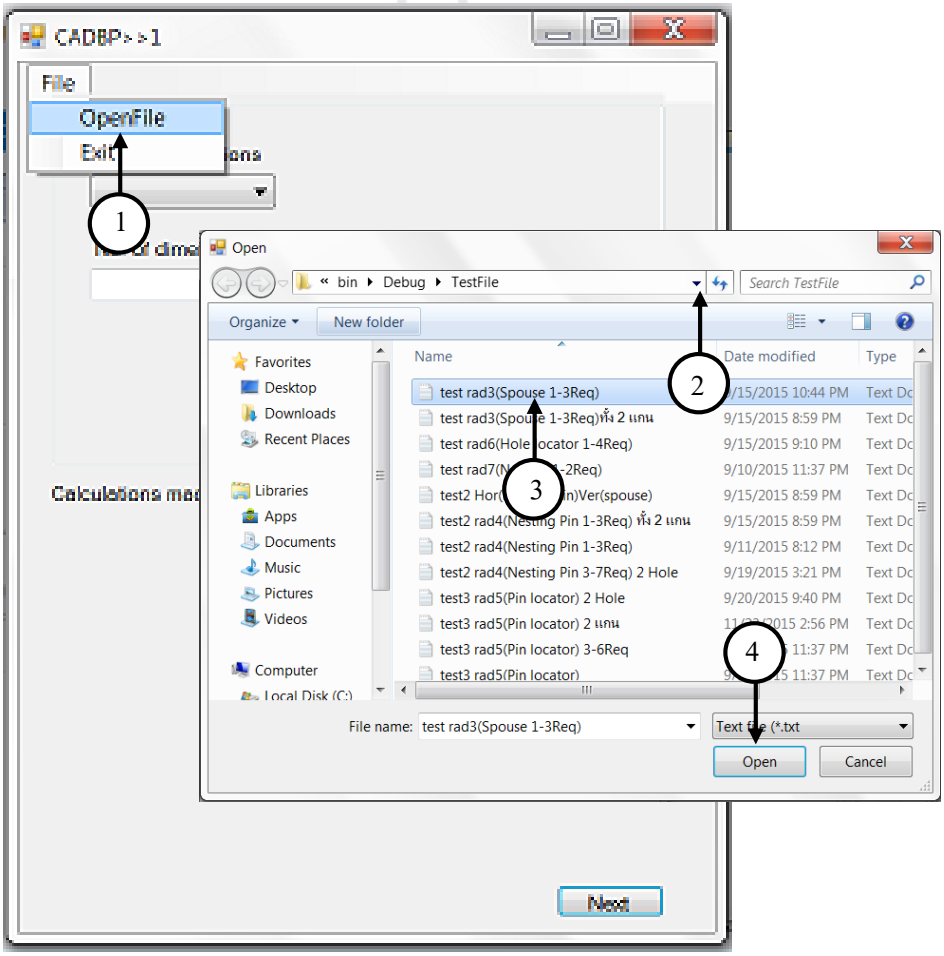
เมื่อผู้ใช้ต้องการเปิดไฟล์ข้อมูลที่มีการบันทึกไว้แล้วในหน่วยความจำขึ้นมาใช้งาน ข้อมูลต่าง ๆ จากหน่วยความจำจะถูกอ่านและใส่ลงในส่วนรับเข้าข้อมูลบนหน้าจอทั้งหมด คำสั่งเปิดไฟล์จะอยู่บนแถบเมนูในหน้าจอ CADBP>>1 โดยมีขั้นตอนการเปิดไฟล์ดังนี้

1) คลิกที่คำสั่ง File > OpenFile บนแถบเมนู จะมีหน้าต่าง Open แสดงขึ้นมาดังรูปที่ M6

2) เลือกโฟลเดอร์ที่เก็บข้อมูลไว้ข้อมูล

3) เลือกไฟล์ที่ต้องการเปิด

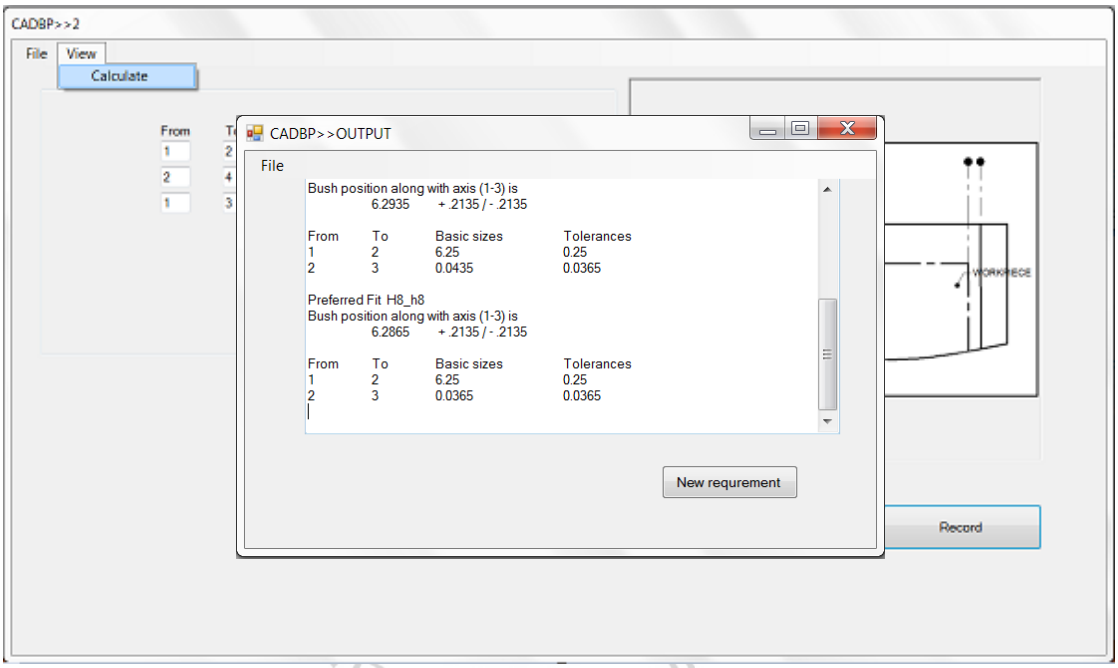
4) คลิกที่ปุ่ม Open ข้อมูลจะถูกป้อนให้กับส่วนรับข้อมูลที่หน้าต่างรับข้อมูลทั้งหมด



รูปที่ M6 หน้าต่างคำสั่งเปิดไฟล์ข้อมูล

4 การหาผลลัพธ์ และการแสดงผล

เมื่อต้องการคำนวณหาผลลัพธ์หลังจากการเปิดไฟล์ข้อมูล หรือป้อนข้อมูลรับเข้าจนครบ ผู้ใช้จะใช้คำสั่งที่แถบเมนู View > Calculate บนหน้าต่าง CADBP>>2 ผลลัพธ์จะปรากฏบนกล่องข้อความในหน้าจอ CADBP>>OUTPUT ดังรูปที่ M7 โดยจะแสดงคำตอบเป็นชุด ตามความฟิตที่กำหนดคือ Ch, Gh และ Hh ซึ่งในกรณีเกิดปัญหาความคลาดเคลื่อนสะสม (tolerance stacks) โปรแกรมจะแจ้งเตือนปัญหา และแสดงผลลัพธ์เฉพาะความฟิตถัดไปที่ไม่เกิดปัญหาเท่านั้น



มหาวิทยาลัยเทคโนโลยีสุร

รูปที่ M7 หน้าต่างแสดงผลลัพธ์, CADBP>>OUTPUT

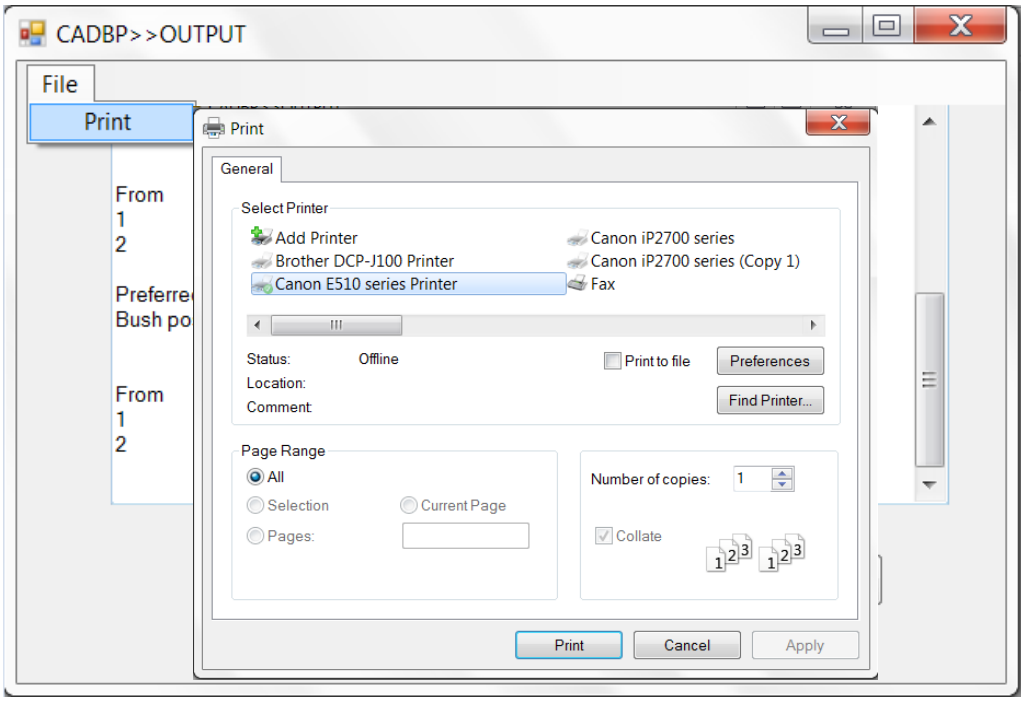
5 การสั่งพิมพ์

การสั่งพิมพ์ข้อมูลสามารถทำได้ที่หน้าจอแสดงผล CADBP>>OUTPUT โดยมีขั้นตอนการสั่งพิมพ์ข้อมูลดังนี้

1) คลิกที่คำสั่ง File > Print บนแถบเมนู จะมีหน้าต่าง Print แสดงขึ้นมาดังรูปที่ M8

2) เลือกเครื่องพิมพ์ จำนวนในการพิมพ์ หรือปรับแต่งคุณสมบัติในการพิมพ์

3) คลิกที่ปุ่ม Print ผลลัพธ์ที่แสดงที่หน้าจอ OUTPUT จะพิมพ์ออกทางเครื่องพิมพ์



รูปที่ M8 หน้าต่างคำสั่งพิมพ์

6 การเริ่มงานใหม่

ในขณะการทำงานหากผู้ใช้ต้องการเริ่มงานใหม่ให้ผู้ใช้กลับเข้าสู่หน้าต่าง CADBP>>2 และเลือกคำสั่ง File > New ข้อมูลทั้งหมดที่อยู่ในส่วนรับข้อมูลจะถูกล้าง และโปรแกรมจะเริ่มต้นใหม่โดยปรากฏหน้าต่าง CADBP>>1 ผู้ใช้จะสามารถใช้งานใหม่ได้จากหน้าต่างดังกล่าว

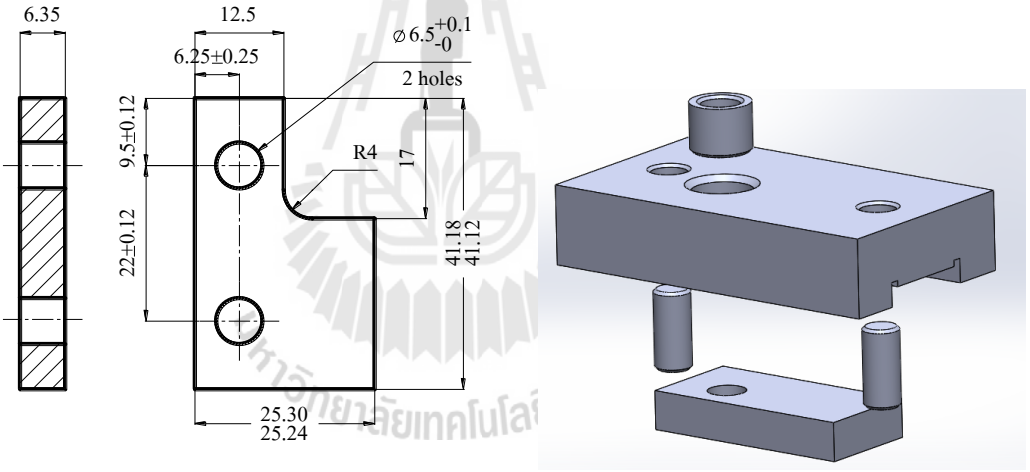
7 การออกจากโปรแกรม

การออกจากโปรแกรมสามารถทำได้โดยการเลือกใช้คำสั่งที่แถบเมนู File > Exit หรือเลือกเครื่องหมายกากบาท (X) ที่มุมบนด้านขวาของหน้าต่างการทำงาน

8 ขั้นตอนการใช้งานโปรแกรม

ขั้นตอนการใช้งานโปรแกรมจะแสดงให้เห็นจากตัวอย่างของชิ้นงาน โดยชิ้นงานที่เลือกใช้เป็นลักษณะงานสี่เหลี่ยมผืนผ้า ในที่นี้จะแสดงตัวอย่างการใช้งานโดยใช้ลักษณะตัวตั้งตำแหน่งที่ต่างกันระหว่างแกนนอน กับแกนตั้ง ซึ่งจะแสดงให้เห็นทั้งวิธีโต้ตอบกับโปรแกรมโดยตรง และใช้ไฟล์ข้อมูลในการส่งข้อมูล

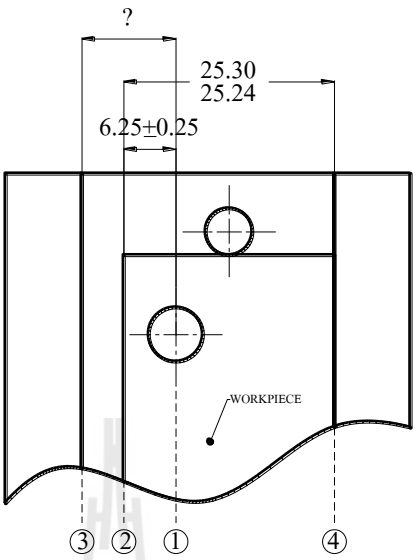
8.1 ตัวอย่าง



รูปที่ M9 ชิ้นงานตัวอย่างควบคุมตำแหน่ง 2 มิติ **รูปที่ M10** แบบของจิกควบคุมทั้ง 2 มิติ

ตัวอย่างชิ้นงานดังรูปที่ M9 เป็นตัวอย่างชิ้นงานเดียวกับในหัวข้อ 6.1 เพื่อใช้แสดงวิธีการใช้โปรแกรม CADBP หาค่าตำแหน่งบุช จากรูปชิ้นงานทำด้วย spacer bar ถูกตัดจากแท่งเหล็ก และได้รับการเจียรนัยขอบทั้ง 4 ด้าน เป็นสี่เหลี่ยมผืนผ้า กว้าง $\frac{25.30}{25.24}$ mm., ยาว $\frac{41.18}{41.12}$ mm. ต้องการเจาะรูขนาด $\varnothing$ 6.5 mm. 2 รู จากลักษณะชิ้นงานจะเลือกใช้ขอบของชิ้นงานเป็นรูปลักษณ์กำหนดตำแหน่ง ใช้รางเป็นตัวกำหนดตำแหน่งตามด้านกว้าง และหมุดทำรังตามด้านยาว ดังลักษณะรูปที่ M10

8.1.1 ใช้รางกำหนดตำแหน่งตามระยะความกว้างชิ้นงาน

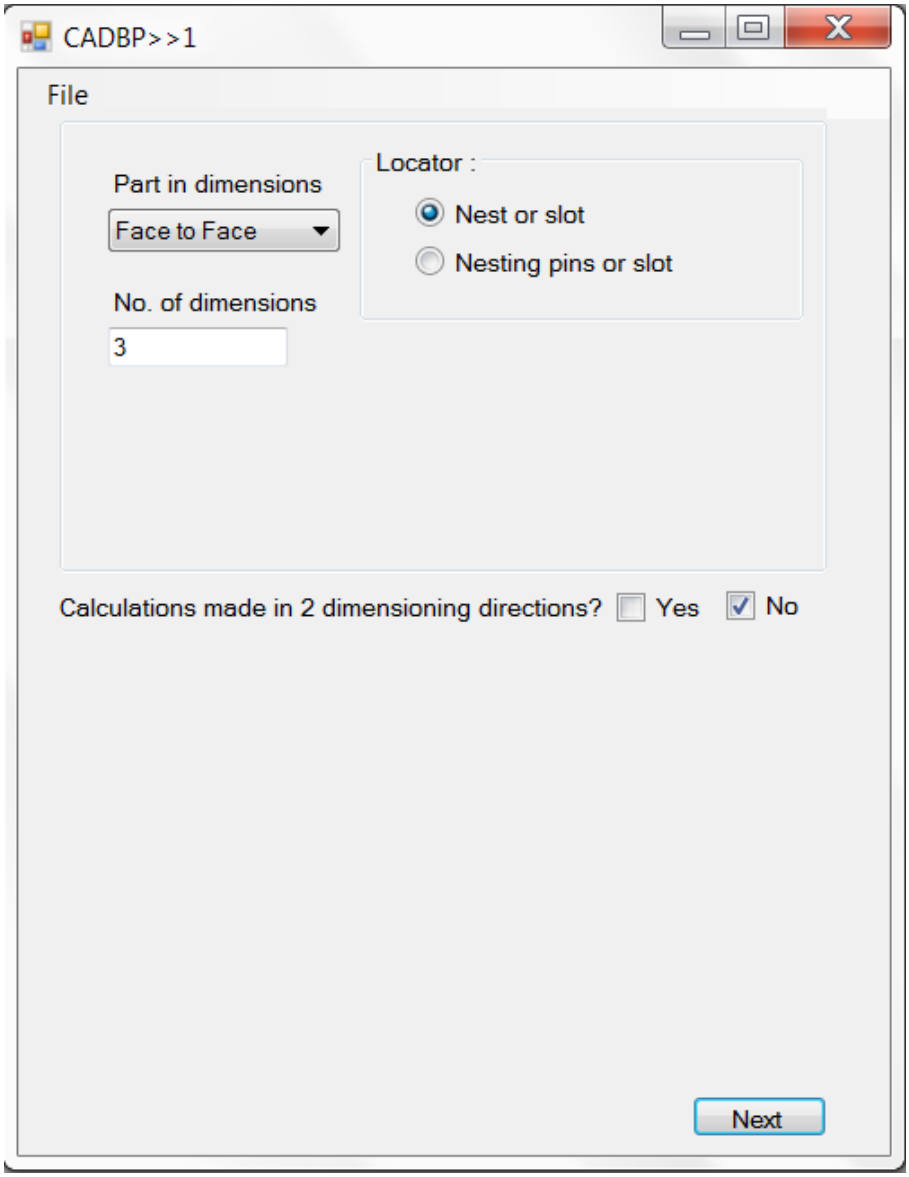


รูปที่ M11 แบบร่างจิกแบบใช้ราง/รัง

เริ่มจากผู้ใช้ป้อนข้อมูลวิธีการบอกขนาดของมิติ การกำหนดตำแหน่ง จำนวนมิติ ขนาดของมิติ และตำแหน่งในการกำหนดผิวอ้างอิงจากแบบงานในรูป 11 ลงบนหน้าจอรับข้อมูล CADBP>>1 และ CADBP>>2 หลังจากนั้นเมื่อสั่งให้โปรแกรมคำนวณ จะได้ผลลัพธ์จากการคำนวณแสดงในหน้าจอ CADBP>>OUTPUT รายละเอียดการใช้งานกับตัวอย่างเป็นดังนี้

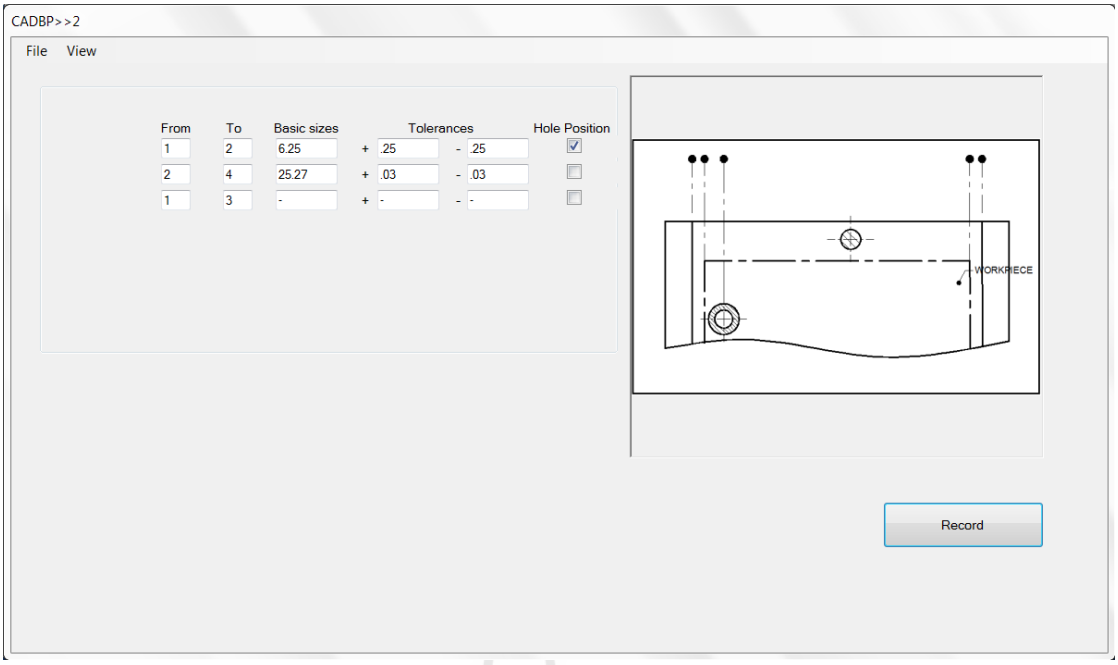
1) ระบุข้อมูลในโปรแกรม (สำหรับกรณีไม่มีไฟล์ข้อมูล)

ในหน้าจอ CADBP>>1 ทำการเลือกวิธีการบอกขนาดของมิติระหว่างผิวระนาบด้วยการเลือก ComboBox เป็น “Face to Face”, จำนวนมิติที่จะทำการระบุทั้งหมด และรูปลักษณ์กำหนดตำแหน่ง ดังรูปที่ 12 แล้วกดปุ่ม “Next”



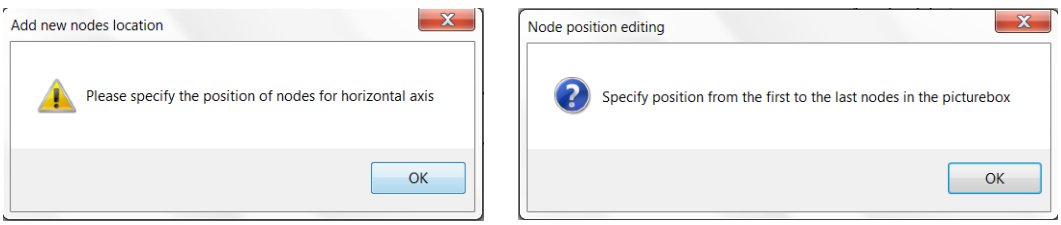
รูปที่ M12 ส่วนรับข้อมูลหน้าต่าง CADBP>>1 สำหรับตัวอย่างรูปที่ 11

2) กรอกรายละเอียดของแต่ละมิติ รวมทั้งมิติที่ต้องการทราบค่า แต่ให้ใช้เครื่องหมาย “ - ” ในช่องขนาด ดังรูปที่ M13 แล้วกดปุ่ม “Record” เพื่อทำการเก็บชื่อเดตัมอ้างอิง มิติแต่ละชุด และทำเครื่องหมายในช่องหลังมิติเป็นตำแหน่งรูเจาะ



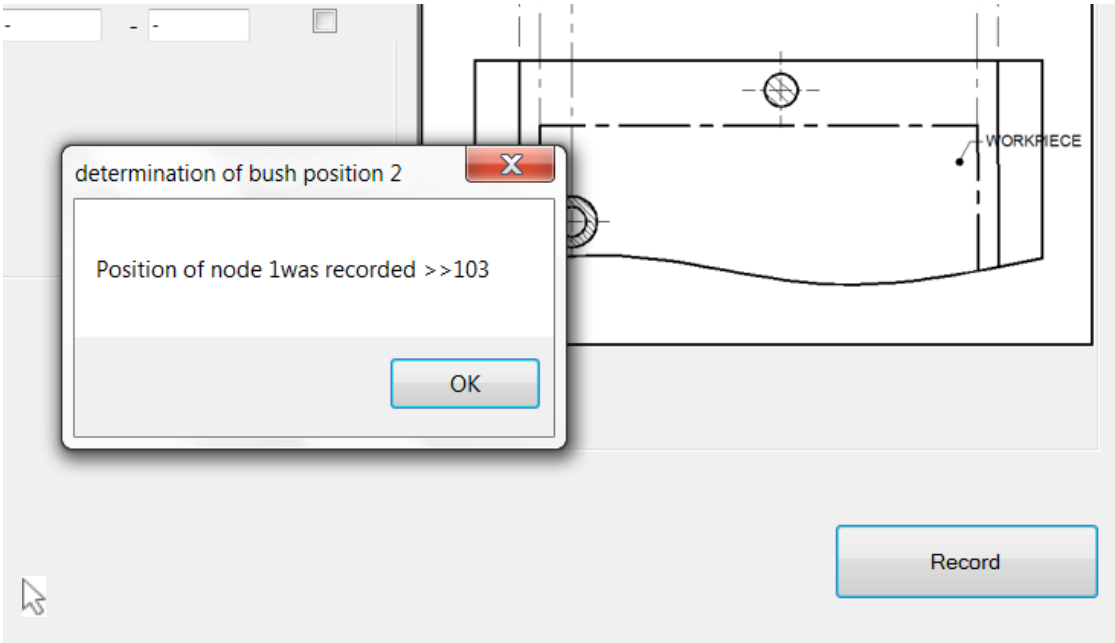
รูปที่ M13 ส่วนรับข้อมูลหน้าต่าง CADBP>>2 สำหรับตัวอย่างรูปที่ M11

3) กดปุ่ม “Record” ข้อมูลรับเข้าในส่วนของโหนดอ้างอิง และค่าขนาดมูลฐานและความคลาดเคลื่อนของมิติจะถูกเก็บเข้าโปรแกรม โดยเมื่อกดปุ่มข้อความแจ้งเตือนการระบุโหนดจะแสดงดังรูปที่ M14 หลังจากยอมรับข้อความแจ้งเตือนทั้งหมด ให้ทำการระบุตำแหน่งโหนดทีละโหนดและกดยอมรับข้อความตามรูป M14 จนครบจำนวนโหนดที่ต้องการตามแต่รูปแบบจิกจะมีปุ่มกดบันทึกตำแหน่งแสดงขึ้น กดปุ่ม “Record new positions” เป็นการเสร็จกระบวนการนำเข้าข้อมูลสู่โปรแกรมแบบโต้ตอบกับโปรแกรมโดยตรง



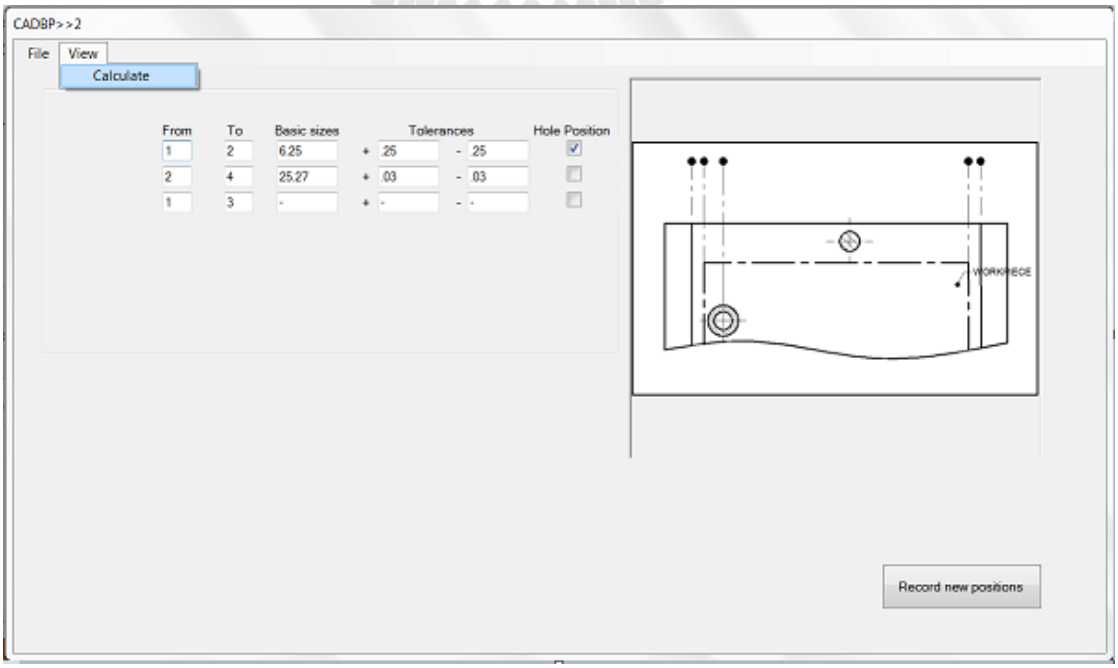
(ก) **(ข)**

รูปที่ M14 กล่องข้อความแจ้งเตือน สำหรับตัวอย่างรูปที่ 11

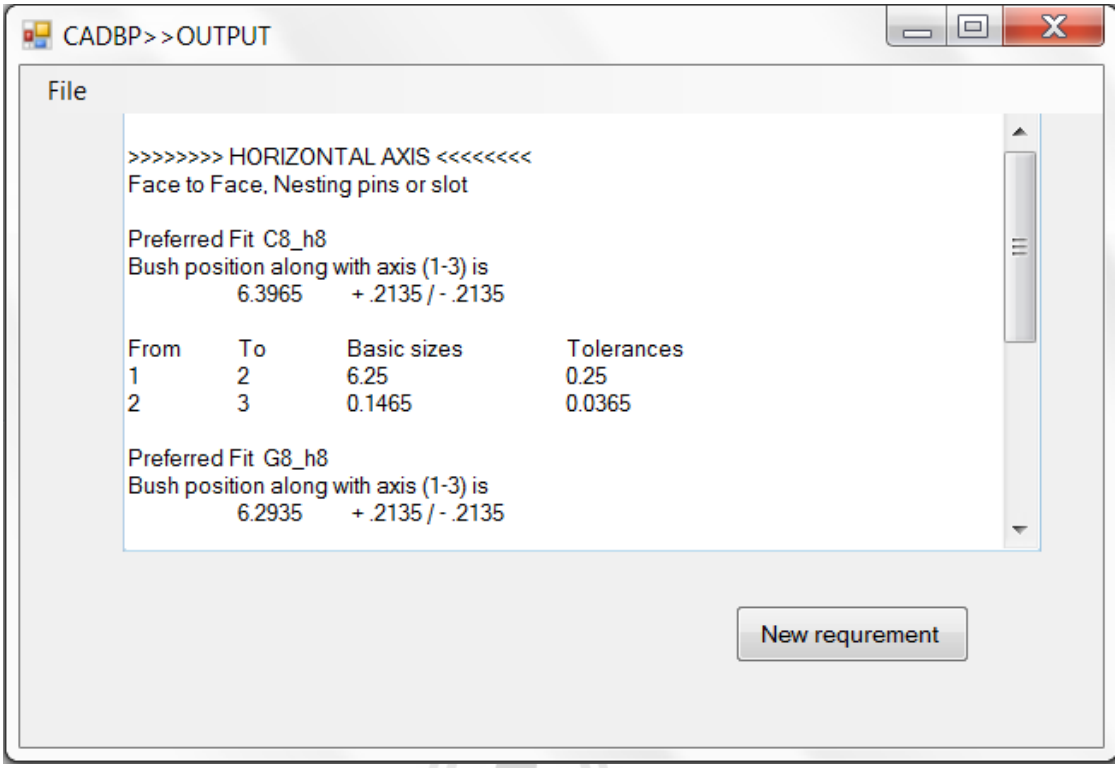


รูปที่ M15 กล่องข้อความแสดงการระบุตำแหน่งโหนด สำหรับตัวอย่างรูปที่ 11

4) เลือกคำสั่งคำนวณค่าจาก View > Calculate ดังรูปที่ M16 ค่าจากการคำนวณจะถูกแสดงในหน้าจอ CADBP>>OUTPUT ดังรูปที่ M17



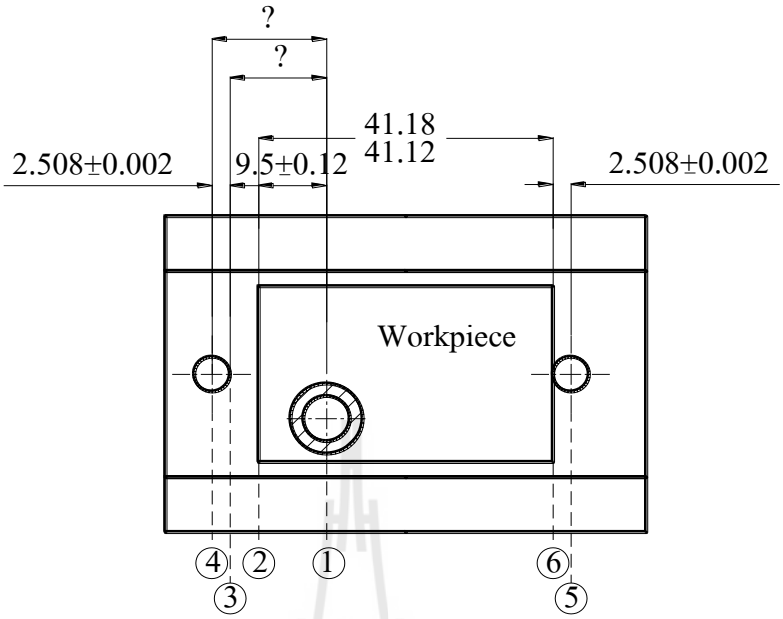
รูปที่ M16 คำสั่ง “Calculate” ในหน้าจอ CADBP>>2 สำหรับตัวอย่างรูปที่ M11



รูป M17 หน้าต่างแสดงผลลัพธ์, CADBP>>OUTPUT สำหรับตัวอย่างรูปที่ M11

การแสดงผลลัพธ์จะแสดงออกมาเป็น 3 ทางเลือกของการออกแบบ คือ ที่การเลือกค่าช่องว่างการประกอบ (minimum clearance) จากฟิตนิยม จากช่องว่างมากสุดไปน้อยสุดคือ Ch, Gh และ Hh ตามลำดับ เช่น สำหรับค่าของมิติ 1-3 ที่ฟิตนิยม Ch จะให้ค่าขนาดของมิติคือ 6.3965 $\pm$ 0.2135 พร้อมแสดงมิติที่เกี่ยวข้องในวงรอบการคำนวณสำหรับนำไปปรับปรุงค่าข้อมูลรับเข้ากรณีเกิดปัญหาความคลาดเคลื่อนสะสม

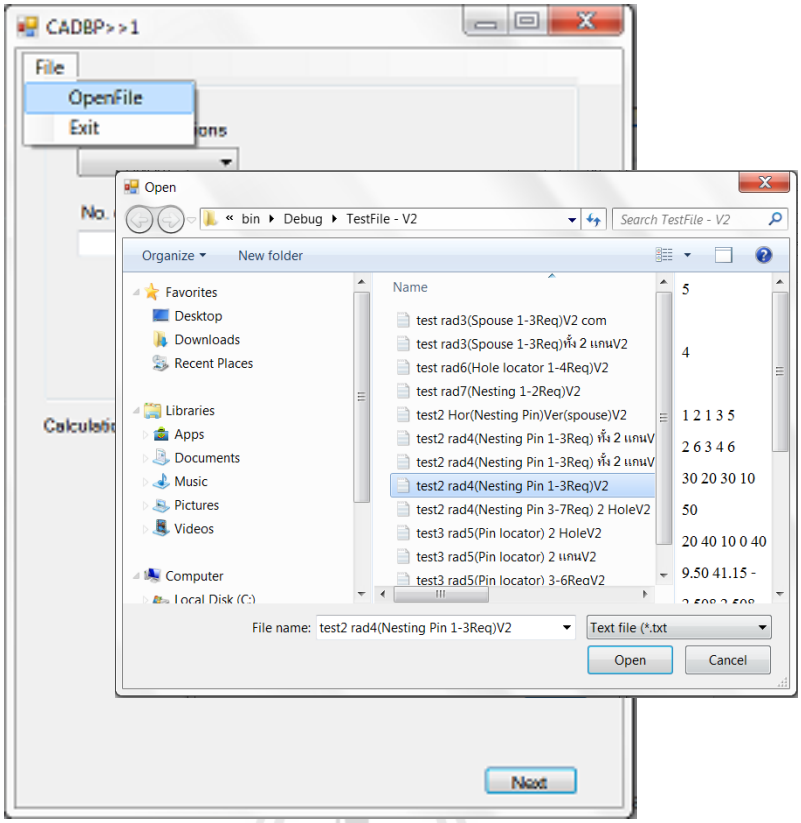
8.1.2 ใช้หมุดเป็นรังกำหนดตำแหน่งตามระยะความยาวชิ้นงาน



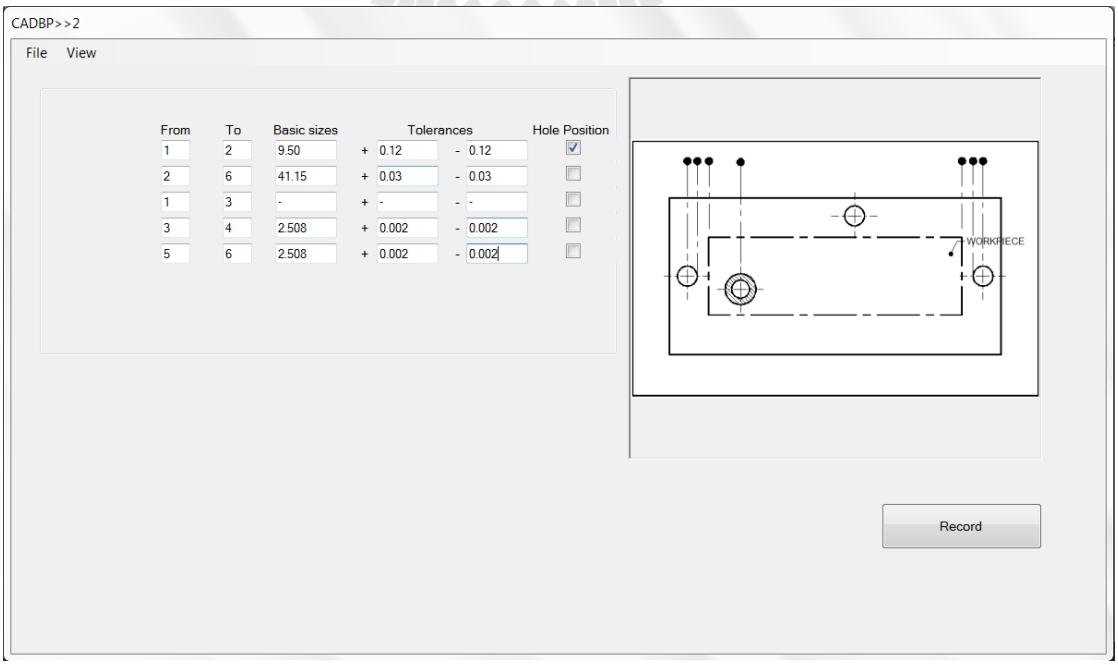
รูปที่ M18 แบบร่างจิกแบบใช้หมุดทำรัง

สำหรับตัวอย่างนี้จะแสดงข้อแตกต่างในกรณีข้อมูลรับเข้าเป็นไฟล์ข้อมูล และเพิ่มเติมในส่วนของการคำนวณมิติอื่นในจิกรูปลักษณ์เดิม ซึ่งจะได้แสดงผ่านตัวอย่างตามแบบงานในรูปที่ M18 ขั้นตอนการใช้งานเป็นดังนี้

1) ค้นหาไฟล์โดยเลือก File > OpenFile ที่หน้า CADBP>>1 จะมีหน้า Open แสดงขึ้นดังรูป M19 ให้ทำการเลือกไฟล์ที่ต้องการ แล้วกด “Open” หน้าจอ CADBP>>2 จะถูกแสดงตามรูปที่ 20

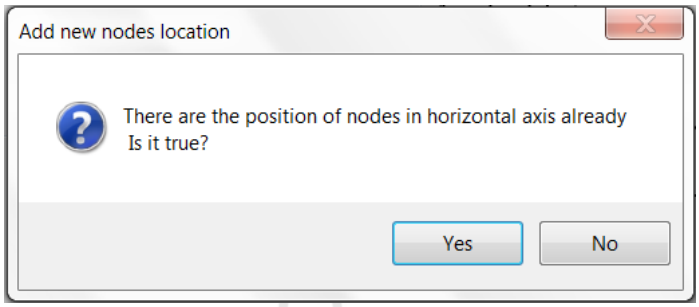


รูปที่ M19 หน้าต่างคำสั่งเปิดไฟล์ข้อมูล สำหรับตัวอย่างรูปที่ 18

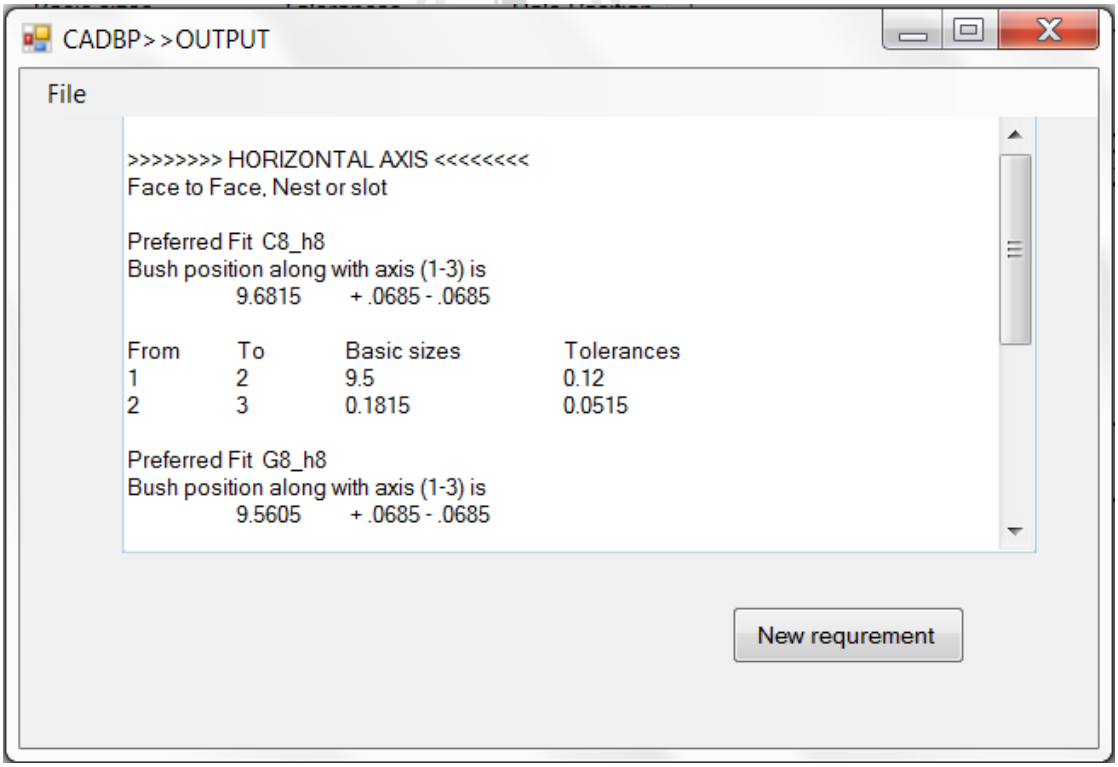


รูปที่ M20 ส่วนรับข้อมูลหน้าต่าง CADBP>>2 สำหรับตัวอย่างรูปที่ 18

2) ทำการบันทึกข้อมูลโดยปุ่ม “Record” แต่ในที่นี้จะมีกล่องข้อความแจ้งเตือน “มีการระบุตำแหน่งของโหนดอยู่แล้ว คุณมั่นใจว่าจะไม่มีการแก้ไขข้อมูล” ตามที่แสดงในรูป M21 หากไม่มีการเปลี่ยนแปลงเดตัมบอกขนาดให้กด “Yes” แต่หากต้องการเปลี่ยนแปลงเดตัมการบอกขนาด หรือเปลี่ยนแปลงมิติที่ต้องการทราบค่าให้กด “No” แล้วทำการบันทึกตำแหน่ง และกดคำนวณค่าเช่นเดียวกับในตัวอย่างแรก จะได้ผลลัพธ์ดังแสดงในรูป M22

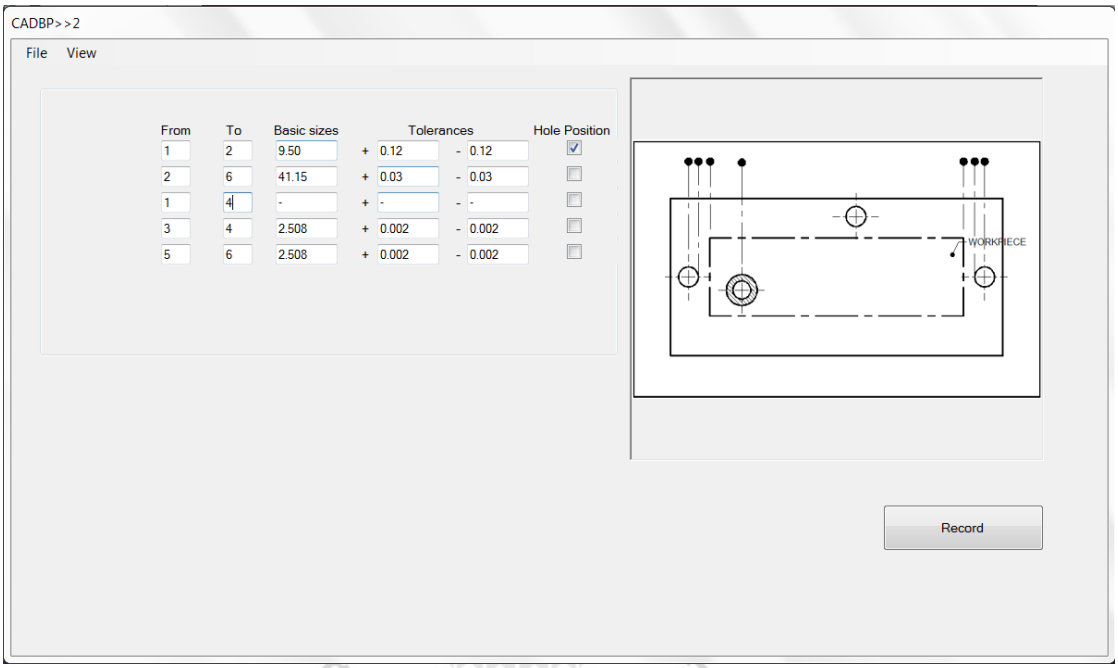


รูปที่ M21 กล่องข้อความแจ้งเตือน สำหรับตัวอย่างรูปที่ M18

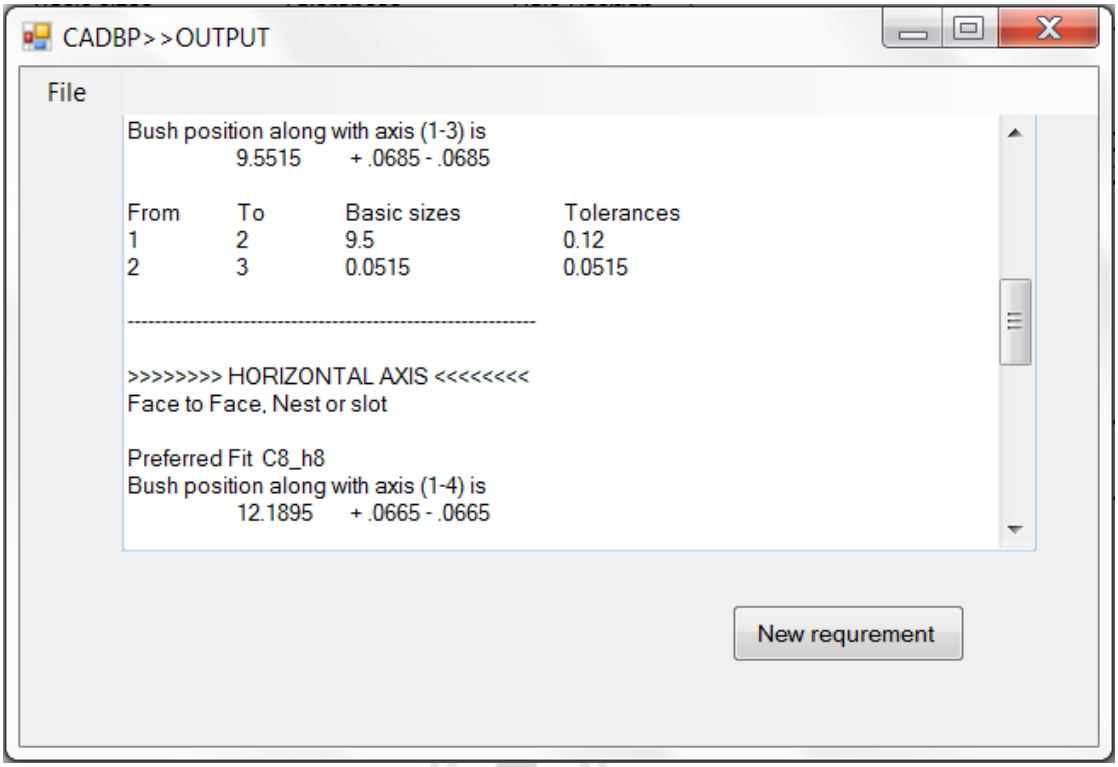


รูปที่ M22 หน้าต่างแสดงผลลัพธ์, CADBP>>OUTPUT สำหรับตัวอย่างรูปที่ M18

สำหรับการเปลี่ยนแปลง หรือสลับโหนดอ้างอิง ผู้ใช้สามารถเลือกปุ่ม “New requirement” ในหน้า CADBP>>OUTPUT แล้วทำการเปลี่ยนแปลงค่าในหน้าจอ CADBP>>2 แล้วดำเนินการเช่นเดียวกับการหาค่ามิติในครั้งแรก ผลลัพธ์จากการคำนวณของมิติใหม่จะถูกแสดงต่อจากมิติเก่า ในลักษณะเดียวกัน แสดงต่อท้ายผลลัพธ์เช่น เพิ่มจากคำนวณมิติ 1 – 4 จากครั้งแรกที่คำนวณมิติ 1 – 3 มิติที่ต้องการคำนวณจะถูกเปลี่ยนจาก 1 – 3 เป็น 1 – 4 ดังรูปที่ M23 และแสดงผลลัพธ์ต่อท้ายของเดิมดังรูปที่ M24



รูปที่ M23 ส่วนรับข้อมูลหน้าต่าง CADBP>>2 ของการคำนวณมิติเพิ่มเติม สำหรับตัวอย่างรูปที่ M18



รูปที่ M24 หน้าต่างแสดงผลลัพธ์, CADBP>>OUTPUT ของการคำนวณมิติเพิ่มเติม สำหรับตัวอย่างรูปที่ M18

มหาวิทยาลัยเทคโนโลยีสุรนารี

ภาคผนวก ข

รายการโปรแกรม

Variable

```
Public inputx() As Dimension

Structure Dimension
    Public F As String
    Public Fadd As Boolean
    Public T As String
    Public Tadd As Boolean
    Public Dims As Single
    Public Dimtol As Single
    Public Signcal As Integer
    Public PosF As Integer
    Public PosT As Integer
    Public SignTol As Integer
End Structure

Public Req() As String
Public FReq() As String
Public Po() As Single
Public NumFReq As Integer
Public Jigtype As Integer
Public Basic_result As Single
Public Tol_result As Single

Public txtF() As TextBox
Public txtT() As TextBox
Public CheckHF() As CheckBox
Public SignP() As Label
Public SignM() As Label
Public txtdimX() As TextBox
Public txtTolXUp() As TextBox
Public txtTolXLo() As TextBox
Public txtFV() As TextBox
Public txtTV() As TextBox
Public CheckVF() As CheckBox
Public FReqStr() As String
Public FReqStrV() As String
Public SignPV() As Label
Public SignMV() As Label
Public txtdimXV() As TextBox
Public txtTolXUpV() As TextBox
Public txtTolXLoV() As TextBox

Public Changeaxis As Integer
Public PicFileName As String
Public ArrPos() As String
Public VarArrpos() As Integer
Public Clearance As Single = 0
Public Path() As String
Public IndexTab As Boolean
Public Max As Integer
Public Path_Result As String
Public Side As String = ""
Public Shift_to As String = ""

Public pic_name As String
Public pic_count As Integer = 0
Public Click_pic As Integer = 0
Public Click_picV As Integer = 0
Public getpo() As Single
Public getpoV() As Single
```

Note : CADBP>>1 is “Form1”
CABBP>>2 is “Form3”
CADBP>>OUTPUT is “Form4”

Input Module

```
Public NodePo() As String
Public NodePoV() As String
Public PosFrom() As String
Public PosFromV() As String
Public PosTo() As String
Public PosToV() As String
Public fromfile As Boolean = False
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
Private Sub ComboBox1_SelectedIndexChanged(sender As Object, e As EventArgs) Handles
ComboBox1.SelectedIndexChanged
    If ComboBox1.SelectedIndex = 0 Then
        GroupBox3.Visible = False
        GroupBox2.Top = 21
        GroupBox2.Left = 167
        GroupBox2.Visible = True

    ElseIf ComboBox1.SelectedIndex = 1 Then
        GroupBox3.Visible = True
        GroupBox2.Top = 21
        GroupBox2.Left = 167
        GroupBox2.Visible = False
    End If
End Sub

Private Sub ComboBox2_SelectedIndexChanged(sender As Object, e As EventArgs) Handles
ComboBox2.SelectedIndexChanged
    If ComboBox2.SelectedIndex = 0 Then
        GroupBox6.Visible = False
        GroupBox5.Top = 21
        GroupBox5.Left = 167
        GroupBox5.Visible = True

    ElseIf ComboBox2.SelectedIndex = 1 Then
        GroupBox6.Visible = True
        GroupBox5.Top = 21
        GroupBox5.Left = 167
        GroupBox5.Visible = False
    End If
End Sub

Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Me.Hide()
    Add_textbox()
    If CheckBox1.Checked = True Then
        Add_textboxV()
    End If
    Form3.Show()
Public NodePo() As String
Public NodePoV() As String
Public PosFrom() As String
Public PosFromV() As String
Public PosTo() As String
Public PosToV() As String
Public fromfile As Boolean = False

Private Sub ComboBox1_SelectedIndexChanged(sender As Object, e As EventArgs) Handles
ComboBox1.SelectedIndexChanged
    If ComboBox1.SelectedIndex = 0 Then
        GroupBox3.Visible = False
        GroupBox2.Top = 21
        GroupBox2.Left = 167
        GroupBox2.Visible = True

     ElseIf ComboBox1.SelectedIndex = 1 Then
        GroupBox3.Visible = True
        GroupBox2.Top = 21
        GroupBox2.Left = 167
        GroupBox2.Visible = False
    End If
End Sub

Private Sub ComboBox2_SelectedIndexChanged(sender As Object, e As EventArgs) Handles
ComboBox2.SelectedIndexChanged
    If ComboBox2.SelectedIndex = 0 Then
        GroupBox6.Visible = False
        GroupBox5.Top = 21
        GroupBox5.Left = 167
        GroupBox5.Visible = True
    ElseIf ComboBox2.SelectedIndex = 1 Then
        GroupBox6.Visible = True
        GroupBox5.Top = 21
        GroupBox5.Left = 167
```

```
        GroupBox5.Visible = False
    End If
End Sub

Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Me.Hide()
    Add_textbox()
    If CheckBox1.Checked = True Then
        Add_textboxV()
    End If
    Form3.Show()
End Sub

Public B As Integer = 0
Public Sub changeB()
    B = 1
End Sub

Private Sub det_newdim_Click(sender As Object, e As EventArgs) Handles det_newdim.Click
    Outbox.AppendText(vbNewLine & "----------" & vbNewLine)
    changeB()
    Changeaxis = 0
    Form1.fromfile = False
    Form3.Show()
    Me.Hide()

End Sub

Public Sub Add_textbox()
    Dim Numtxt As Integer = CInt(txtDim.Text)
    ReDim txtF(Numtxt - 1)
    ReDim txtT(Numtxt - 1)
    ReDim txtdimX(Numtxt - 1)
    ReDim txtTolXUp(Numtxt - 1)
    ReDim txtTolXLo(Numtxt - 1)
    ReDim CheckHF(Numtxt - 1)
    ReDim SignP(Numtxt - 1)
    ReDim SignM(Numtxt - 1)
    ReDim inputx(Numtxt - 1)
    For i As Integer = 0 To Numtxt - 1
        txtF(i) = New TextBox()
        txtF(i).Height = 22
        txtF(i).Width = 32
        txtF(i).Top = 63 + i * 28
        txtF(i).Left = 131

        txtT(i) = New TextBox()
        txtT(i).Height = 22
        txtT(i).Width = 32
        txtT(i).Top = 63 + i * 28
        txtT(i).Left = 198

        txtdimX(i) = New TextBox()
        txtdimX(i).Height = 22
        txtdimX(i).Width = 67
        txtdimX(i).Top = 63 + i * 28
        txtdimX(i).Left = 255

        txtTolXUp(i) = New TextBox()
        txtTolXUp(i).Height = 22
        txtTolXUp(i).Width = 67
        txtTolXUp(i).Top = 63 + i * 28
        txtTolXUp(i).Left = 365

        txtTolXLo(i) = New TextBox()
        txtTolXLo(i).Height = 22
        txtTolXLo(i).Width = 67
        txtTolXLo(i).Top = 63 + i * 28
        txtTolXLo(i).Left = 462

        SignP(i) = New Label()
        SignP(i).Height = 22
        SignP(i).Width = 25
        SignP(i).Top = 66 + i * 28
        SignP(i).Left = 345
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        SignP(i).Text = "+"

        SignM(i) = New Label()
        SignM(i).Height = 22
        SignM(i).Width = 25
        SignM(i).Top = 66 + i * 28
        SignM(i).Left = 447
        SignM(i).Text = "-"

        CheckHF(i) = New CheckBox()
        CheckHF(i).Top = 60 + i * 28
        CheckHF(i).Left = 570

        Form3.GroupBox1.Controls.Add(txtF(i))
        Form3.GroupBox1.Controls.Add(txtT(i))
        Form3.GroupBox1.Controls.Add(txtdimX(i))
        Form3.GroupBox1.Controls.Add(txtTolXUp(i))
        Form3.GroupBox1.Controls.Add(txtTolXLo(i))
        Form3.GroupBox1.Controls.Add(SignP(i))
        Form3.GroupBox1.Controls.Add(SignM(i))
        Form3.GroupBox1.Controls.Add(CheckHF(i))
    Next
End Sub

Public Sub Add_textboxV()
    Dim NumtxtV As Integer = CInt(txtDimV.Text)

    ReDim txtFV(NumtxtV - 1)
    ReDim txtTV(NumtxtV - 1)
    ReDim txtdimXV(NumtxtV - 1)
    ReDim txtTolXUpV(NumtxtV - 1)
    ReDim txtTolXLoV(NumtxtV - 1)
    ReDim SignPV(NumtxtV - 1)
    ReDim SignMV(NumtxtV - 1)
    ReDim CheckVF(NumtxtV - 1)

    For i = 0 To NumtxtV - 1
        txtFV(i) = New TextBox()
        txtFV(i).Height = 22
        txtFV(i).Width = 32
        txtFV(i).Top = 63 + i * 28
        txtFV(i).Left = 131

        txtTV(i) = New TextBox()
        txtTV(i).Height = 22
        txtTV(i).Width = 32
        txtTV(i).Top = 63 + i * 28
        txtTV(i).Left = 198

        txtdimXV(i) = New TextBox()
        txtdimXV(i).Height = 22
        txtdimXV(i).Width = 67
        txtdimXV(i).Top = 63 + i * 28
        txtdimXV(i).Left = 255

        txtTolXUpV(i) = New TextBox()
        txtTolXUpV(i).Height = 22
        txtTolXUpV(i).Width = 67
        txtTolXUpV(i).Top = 63 + i * 28
        txtTolXUpV(i).Left = 365

        txtTolXLoV(i) = New TextBox()
        txtTolXLoV(i).Height = 22
        txtTolXLoV(i).Width = 67
        txtTolXLoV(i).Top = 63 + i * 28
        txtTolXLoV(i).Left = 462

        SignPV(i) = New Label()
        SignPV(i).Height = 22
        SignPV(i).Width = 25
        SignPV(i).Top = 66 + i * 28
        SignPV(i).Left = 345
        SignPV(i).Text = "+"

        SignMV(i) = New Label()
        SignMV(i).Height = 22
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        SignMV(i).Width = 25
        SignMV(i).Top = 66 + i * 28

        SignMV(i).Left = 447
        SignMV(i).Text = "-"

        CheckVF(i) = New CheckBox()
        CheckVF(i).Top = 60 + i * 28
        CheckVF(i).Left = 570

        Form3.GroupBox2.Controls.Add(txtFV(i))
        Form3.GroupBox2.Controls.Add(txtTV(i))
        Form3.GroupBox2.Controls.Add(txtdimXV(i))
        Form3.GroupBox2.Controls.Add(txtTolXUpV(i))
        Form3.GroupBox2.Controls.Add(txtTolXLoV(i))
        Form3.GroupBox2.Controls.Add(SignPV(i))
        Form3.GroupBox2.Controls.Add(SignMV(i))
        Form3.GroupBox2.Controls.Add(CheckVF(i))
    Next
End Sub

Private Sub CheckBox1_CheckedChanged(sender As Object, e As EventArgs) Handles CheckBox1.CheckedChanged
    GroupY.Visible = True
    Form3.GroupBox2.Visible = True
    If ComboBox1.SelectedIndex = 0 Then
        ComboBox2.SelectedIndex = 0
        txtDimV.Text = txtDim.Text
        txtDimV.ReadOnly = True
        txtWV.Text = txtW.Text
        txtWV.ReadOnly = True
    End If
End Sub

Private Sub Form3_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    Changeaxis = 0 'รอบแรกโหลดฟอร์มขึ้นใหม่ เริ่มแกนนอนเสมอ
    SelectForm()
End Sub

Public Sub SelectForm()
    Dim work As String = ""

    If Form1.ComboBox1.SelectedItem = "Length" Then
        work = "ชิ้นงานเหลี่ยม"
    ElseIf Form1.ComboBox1.SelectedItem = "Diameter" Then
        work = "ชิ้นงานกลม"
    End If
    PicConPos()
    End If
    PictureBox1.ImageLocation = PicFileName
End Sub

Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
Nextread:
    Rec_Req()
End Sub

Private Sub PictureBox1_Click(sender As Object, e As System.Windows.Forms.MouseEventArgs) Handles
 PictureBox1.Click
    Dim W As Single

    If pic_count = 0 Then
        ReDim getpo(Max - 1)
    End If
    Form1.fromfile = False
    Click_pic = Click_pic + 1
    If Changeaxis <> 1 Then
        W = e.X
    ElseIf Changeaxis = 1 Then
        W = e.Y
    End If
    MsgBox("ตำแหน่ง node " & Click_pic & "ถูกเก็บค่าแล้ว >>" & W)
    getpo(pic_count) = W
    pic_count = pic_count + 1
    If Click_pic = Max And Changeaxis <> 1 Then
```

```
        Button1.Visible = False
        Node_spec.Visible = True
    ElseIf Click_pic = Max And Changeaxis = 1 Then
        Getnewchangepo()
    End If
End Sub

Private Sub Node_spec_Click(sender As Object, e As EventArgs) Handles Node_spec.Click
NextNewPo:
    Form4.B = 0
    Getnewchangepo()
End Sub

Public Sub Getnewchangepo()
    Dim upperbound As Integer
    Dim icount, jcount As Integer
    Dim icountdim As Integer

    If Changeaxis <> 1 Then
        upperbound = CInt(Form1.txtDim.Text)
    ElseIf Changeaxis = 1 Then
        upperbound = CInt(Form1.txtDimV.Text)
    End If
    For j As Integer = 0 To upperbound - 1
        For i As Integer = 0 To Max - 1
            If CInt(inputx(j).F) = i + 1 Then
                inputx(j).PosF = getpo(i)
            End If
        Next
        For i = 0 To Max - 1
            If CInt(inputx(j).T) = i + 1 Then
                inputx(j).PosT = getpo(i)
            End If
        Next
    Next
    Max = FindMaxFea()
    For icountdim = 0 To inputx.Length - 1
        inputx(icountdim).Fadd = False
        inputx(icountdim).Tadd = False
    Next

    ReDim Po(Max)
    For icount = 1 To Max
        For jcount = 0 To inputx.Length - 1
            If icount = inputx(jcount).F Then
                If Po(icount) = 0 Then
                    Po(icount) = inputx(jcount).PosF
                End If
            End If
            If icount = inputx(jcount).T Then
                If Po(icount) = 0 Then
                    Po(icount) = inputx(jcount).PosT
                End If
            End If
        Next
    Next
    ReDim ArrPos(Max - 1)
    ArrPos = sortstr()
    Set_Po()
    If Changeaxis <> 1 Then
        Node_spec.Visible = False
        Button1.Visible = True
    End If
    Click_pic = 0
    pic_count = 0
    If Changeaxis = 1 Then
        Calculate()
    End If
End Sub

Public Sub NumHoleCheck()
    If Changeaxis <> 1 Then
        NumFReq = 0
```

```
        For icount As Integer = 0 To CInt(Form1.txtDim.Text) - 1
            If CheckHF(icount).Checked = True Then
                NumFReq = NumFReq + 1
            End If
        Next
    ElseIf Changeaxis = 1 Then
        NumFReq = 0
        For icount As Integer = 0 To CInt(Form1.txtDimV.Text) - 1
            If CheckVF(icount).Checked = True Then
                NumFReq = NumFReq + 1
            End If
        Next
    End If
End Sub

Public Sub Rec_Data()
    Dim A, B As Single
    If Changeaxis = 0 Then
        For i As Integer = 0 To CInt(Form1.txtDim.Text) - 1
            inputx(i).F = txtF(i).Text
            inputx(i).T = txtT(i).Text
            If txtdimX(i).Text = " " Or txtdimX(i).Text = "-" Then
                inputx(i).Dims = 0
            Else
                inputx(i).Dims = CSng(txtdimX(i).Text)
            End If
            If txtTolXUp(i).Text = " " Or txtTolXUp(i).Text = "-" Then
                A = 0
            ElseIf txtTolXUp(i).Text <> " " Or txtTolXUp(i).Text <> "-" Then
                A = CSng(txtTolXUp(i).Text)
            End If
            If txtTolXLo(i).Text = " " Or txtTolXLo(i).Text = "-" Then
                B = 0
            ElseIf txtTolXLo(i).Text <> " " Or txtTolXLo(i).Text <> "-" Then
                B = CSng(txtTolXLo(i).Text)
            End If
            inputx(i).Dimtol = (A + B) / 2
        Next
    ElseIf Changeaxis = 1 Then
        For i As Integer = 0 To CInt(Form1.txtDimV.Text) - 1
            inputx(i).F = txtFV(i).Text
            inputx(i).T = txtTV(i).Text
            If txtdimXV(i).Text = " " Or txtdimXV(i).Text = "-" Then
                inputx(i).Dims = 0
            Else
                inputx(i).Dims = CSng(txtdimXV(i).Text)
            End If
            If txtTolXUpV(i).Text = " " Or txtTolXUpV(i).Text = "-" Then
                A = 0
            ElseIf txtTolXUpV(i).Text <> " " Or txtTolXUpV(i).Text <> "-" Then
                A = CSng(txtTolXUpV(i).Text)
            End If
            If txtTolXLoV(i).Text = " " Or txtTolXLoV(i).Text = "-" Then
                B = 0
            ElseIf txtTolXLoV(i).Text <> " " Or txtTolXLoV(i).Text <> "-" Then
                B = CSng(txtTolXLoV(i).Text)
            End If
            inputx(i).Dimtol = (A + B) / 2
        Next
    End If

    If Changeaxis <> 1 Then
        For j As Integer = 0 To CInt(Form1.txtDim.Text) - 1
            inputx(j).F = inputx(j).F.Trim
            inputx(j).T = inputx(j).T.Trim
        Next
    ElseIf Changeaxis = 1 Then
        For j As Integer = 0 To CInt(Form1.txtDimV.Text) - 1
            inputx(j).F = inputx(j).F.Trim
            inputx(j).T = inputx(j).T.Trim
        Next
    End If
```

```
    If Form1.fromfile = True Then
      If Changeaxis <> 1 Then
        For i As Integer = 0 To CInt(Form1.txtDim.Text - 1)
          inputx(i).PosF = Form1.PosFrom(i)
          inputx(i).PosT = Form1.PosTo(i)
        Next
      ElseIf Changeaxis = 1 Then
        For i As Integer = 0 To CInt(Form1.txtDimV.Text - 1)
          inputx(i).PosF = Form1.PosFromV(i)
          inputx(i).PosT = Form1.PosToV(i)
        Next
      End If
    End If

  End Sub

  Public Sub Rec_Req()
    Dim i As Integer
    Dim j As Integer
    Dim ResultMsg As DialogResult

    If Form1.fromfile = False And Form4.B = 0 Then
      ReDim inputx(CInt(Form1.txtDim.Text) - 1)
    End If
    If Changeaxis = 0 And Form4.B = 1 Then
      ReDim inputx(CInt(Form1.txtDim.Text) - 1)
    End If
    If Changeaxis = 0 And Form4.B = 0 Then
      ReDim inputx(CInt(Form1.txtDim.Text) - 1)
    ElseIf Changeaxis = 1 And Form4.B = 0 Then
      ReDim inputx(CInt(Form1.txtDimV.Text) - 1)
    End If
    Rec_Data()
    Max = FindMaxFea()
    For i = 0 To inputx.Length - 1
      inputx(i).Fadd = False
      inputx(i).Tadd = False
    Next
    j = 0
    'ระบุ requirement ที่เราต้องการ
    If Changeaxis <> 1 Then
      For i = 0 To txtF.Length - 1
        If txtdimX(i).Text = "-" Then
          Req(0) = txtF(i).Text
          Req(1) = txtT(i).Text
        End If
      Next
    ElseIf Changeaxis = 1 Then
      For i = 0 To txtFV.Length - 1
        If txtdimXV(i).Text = "-" Then
          Req(0) = txtFV(i).Text
          Req(1) = txtTV(i).Text
        End If
      Next

    End If
    'ระบุ requirement ของงาน
    NumFReq = 0
    If Changeaxis <> 1 Then
      For i = 0 To txtdimX.Length - 1
        If CheckHF(i).Checked = True Then
          NumFReq = NumFReq + 1
        End If
      Next
      ReDim FReq(NumFReq - 1)
      j = 0
      For i = 0 To txtF.Length - 1
        If CheckHF(i).Checked = True Then
          FReq(j) = inputx(i).F & inputx(i).T
          j = j + 1
        End If
      Next
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        ElseIf Changeaxis = 1 Then
            For i = 0 To txtdimXV.Length – 1
                If CheckVF(i).Checked = True Then
                    NumFReq = NumFReq + 1
                End If
            Next
            ReDim FReq(NumFReq - 1)
            j = 0
            For i = 0 To txtFV.Length - 1
                If CheckVF(i).Checked = True Then
                    FReq(j) = inputx(i).F & inputx(i).T
                    j = j + 1
                End If
            Next
        End If

If Changeaxis <> 1 Then
  If inputx(0).PosF <> 0 Then

            ResultMsg = MessageBox.Show("There are the position of nodes in horizontal axis already" & vbNewLine & " Is
    it true?", "Add new nodes location", _
                                        MessageBoxButtons.YesNo, _
                                        MessageBoxIcon.Question, _
                                        MessageBoxDefaultButton.Button1)
            Select Case ResultMsg
                Case DialogResult.Yes


                Case DialogResult.No
                    MessageBox.Show("Specify position from the first to the last nodes in the picturebox", "Node position
    editing", _
                                        MessageBoxButtons.OK, _
                                        MessageBoxIcon.Question)

            End Select
        Else
            MessageBox.Show("Please specify the position of nodes for horizontal axis", "Add new nodes location", _
                                        MessageBoxButtons.OK, _
                                        MessageBoxIcon.Exclamation)
            MessageBox.Show("Specify position from the first to the last nodes in the picturebox", "Node position editing", _
                                        MessageBoxButtons.OK, _
                                        MessageBoxIcon.Question)         End If
ElseIf Changeaxis = 1 Then
  If inputx(0).PosF <> 0 Then
            ResultMsg = MessageBox.Show("There are the position of nodes in vertical axis already" & vbNewLine & " Is it
    true?", "Add new nodes location", _
                                        MessageBoxButtons.YesNo, _
                                        MessageBoxIcon.Question, _
                                        MessageBoxDefaultButton.Button1)
            Select Case ResultMsg
                Case DialogResult.Yes
                    Calculate()

                Case DialogResult.No
                    MessageBox.Show("Specify position from the first to the last nodes in the picturebox", "Node position
    editing", _
                                        MessageBoxButtons.OK, _
                                        MessageBoxIcon.Question)

            End Select
        Else
            MessageBox.Show("Please specify the position of nodes for vertical axis", "Add new nodes location", _
                                        MessageBoxButtons.OK, _
                                        MessageBoxIcon.Exclamation)
            MessageBox.Show("Specify position from the first to the last nodes in the picturebox", "Node position editing", _
                                        MessageBoxButtons.OK, _
                                        MessageBoxIcon.Question)
        End If
End If
      Click_pic = 0

    End Sub
```

```
Public Function FindMaxFea() As Integer
    Dim MaxF, MaxT, Max As Integer

    If Changeaxis <> 1 Then
        MaxF = inputx(0).F
        For icount = 0 To CInt(Form1.txtDim.Text) - 2
            If inputx(icount + 1).F > MaxF Then
                MaxF = inputx(icount + 1).F
            End If
        Next
        MaxT = inputx(0).T
        For icount = 0 To CInt(Form1.txtDim.Text) - 2
            If inputx(icount + 1).T > MaxT Then
                MaxT = inputx(icount + 1).T
            End If
        Next
        If MaxF > MaxT Then
            Max = MaxF
        Else
            Max = MaxT
        End If
    ElseIf Changeaxis = 1 Then
        MaxF = inputx(0).F
        For icount = 0 To CInt(Form1.txtDimV.Text) - 2
            If inputx(icount + 1).F > MaxF Then
                MaxF = inputx(icount + 1).F
            End If
        Next
        MaxT = inputx(0).T
        For icount = 0 To CInt(Form1.txtDimV.Text) - 2
            If inputx(icount + 1).T > MaxT Then
                MaxT = inputx(icount + 1).T
            End If
        Next
        If MaxF > MaxT Then
            Max = MaxF
        Else
            Max = MaxT
        End If
    End If

    Return Max
End Function

Public Sub Set_Po()
    ReDim Po(Max)
    Dim icount As Integer
    If Changeaxis <> 1 Then
        For icount = 1 To Max
            For jcount = 0 To CInt(Form1.txtDim.Text - 1)
                If icount = inputx(jcount).F Then
                    If Po(icount) = 0 Then
                        Po(icount) = inputx(jcount).PosF
                    End If
                End If
                If icount = inputx(jcount).T Then
                    If Po(icount) = 0 Then
                        Po(icount) = inputx(jcount).PosT
                    End If
                End If
            Next
        Next
    ElseIf Changeaxis = 1 Then
        For icount = 1 To Max
            For jcount = 0 To CInt(Form1.txtDimV.Text - 1)
                If icount = inputx(jcount).F Then
                    If Po(icount) = 0 Then
                        Po(icount) = inputx(jcount).PosF
                    End If
                End If
                If icount = inputx(jcount).T Then
                    If Po(icount) = 0 Then
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
                    Po(icount) = inputx(jcount).PosT
                End If
            End If
        Next
    Next
End If

If Form1.fromfile = True And Form4.B = 0 Then
    If Changeaxis <> 1 Then
        For icount = 1 To Max
            If Po(icount) = 0 Then
                Po(icount) = CSng(Form1.NodePo(icount - 1))
            End If
        Next
    ElseIf Changeaxis = 1 Then
        For icount = 1 To Max
            If Po(icount) = 0 Then
                Po(icount) = CSng(Form1.NodePoV(icount - 1))
            End If
        Next
    End If
Else
    Po(0) = 0.000000001
    For icount = 1 To Max
        If Po(icount) = 0 Then
            Po(icount) = getpo(icount - 1)
        End If
    Next
    If Form1.fromfile = False Then
        If Changeaxis <> 1 Then
            ReDim Form1.NodePo(FindMaxFea() - 1)
            For icount = 0 To Form1.NodePo.Length - 1
                Form1.NodePo(icount) = Po(icount + 1)
            Next
        ElseIf Changeaxis = 1 Then
            ReDim Form1.NodePoV(FindMaxFea() - 1)
            For icount = 0 To Form1.NodePoV.Length - 1
                Form1.NodePoV(icount) = Po(icount + 1)
            Next

        End If
    End If
End If
End Sub

'เรียงลำดับ feature ที่ใช้บอกขนาด
Public Function sortstr() As String()
    Dim Added As Integer = 1
    Dim j As Integer
    Dim A As Integer = 1
    Dim StrA As String = " "
    Dim ArrA() As String
    Dim k As Integer = 0

    ReDim ArrA(Max - 1)
    Array.Sort(Po)
    For i As Integer = 1 To Max
        ArrA(k) = CStr(Po(i))
        k = k + 1
    Next
    Return ArrA
End Function
End Module
```

Tolerance Calculation and Assignment Module

```
Public ComPos As String
Public Sub Contact_selection()
    Dim icount As Integer
    Dim jcount As Integer
    Dim kcount As Integer
    Dim Repeat As Integer = 0
```

```
Dim ReqType As String = ""
Dim CPos As Integer

Set_Po()
ArrPos = sortstr()

    If Changeaxis <> 1 Then
      If Form1.RadioButton3.Checked = True Then
         For icount = 0 To CInt(Form1.txtDim.Text) - 1
            If inputx(icount).Dims <> 0 And inputx(icount).Dimtol <> 0 Then
               If CheckHF(icount).Checked = False Then
                  If ArrPos(0) = inputx(icount).PosF Then
                     Shift_to = "Left"
                     Exit For
                  ElseIf ArrPos(0) = inputx(icount).PosT Then
                     Shift_to = "Left"
                     Exit For
                  ElseIf ArrPos(ArrPos.Length - 1) = inputx(icount).PosF Then
                     Shift_to = "Right"
                     Exit For
                  ElseIf ArrPos(ArrPos.Length - 1) = inputx(icount).PosT Then
                     Shift_to = "Right"
                     Exit For
                  End If
               End If
            End If
         Next

      ElseIf Form1.RadioButton4.Checked = True Then
         For icount = 0 To CInt(Form1.txtDim.Text) - 1
            If inputx(icount).Dims <> 0 And inputx(icount).Dimtol <> 0 Then
               If ArrPos(1) = inputx(icount).PosF Then
                  Repeat = Repeat + 1
               ElseIf ArrPos(1) = inputx(icount).PosT Then
                  Repeat = Repeat + 1
               End If
            End If
         Next
         If Repeat > 1 Then
            Shift_to = "Left"
         ElseIf Repeat = 1 Then
            Shift_to = "Right"
         End If

      ElseIf Form1.RadioButton5.Checked = True Then
      Repeat = 0
      ReqType = ""
         If CInt(Form1.txtDim.Text) <= 3 Then
            For icount = 0 To CInt(Form1.txtDim.Text) - 1
               If CheckHF(icount).Checked = True Then
                  If inputx(icount).PosF < inputx(icount).PosT Then
                     If inputx(icount).PosF = ArrPos(0) Then
                        Side = "รูเจาะอยู่ซ้าย"
                     ElseIf inputx(icount).PosT = ArrPos(ArrPos.Length - 1) Then
                        Side = "รูเจาะอยู่ขวา"
                     End If
                  ElseIf inputx(icount).PosT < inputx(icount).PosF Then

                     If inputx(icount).PosT = ArrPos(0) Then
                        Side = "รูเจาะอยู่ซ้าย"
                     ElseIf inputx(icount).PosF = ArrPos(ArrPos.Length - 1) Then
                        Side = "รูเจาะอยู่ขวา"
                     End If
                  End If
               End If
            Next

            Select Case Side
               Case "รูเจาะอยู่ซ้าย"
                  For icount = 0 To CInt(Form1.txtDim.Text) - 1
                     If inputx(icount).Dims = 0 And inputx(icount).Dimtol = 0 Then
                        If inputx(icount).PosF < inputx(icount).PosT And inputx(icount).PosF = ArrPos(0) Then
```

```
                ReqType = "Case1"
            ElseIf inputx(icount).PosF < inputx(icount).PosT And inputx(icount).PosT <> ArrPos(0) Then
                ReqType = "Case2"
            ElseIf inputx(icount).PosF > inputx(icount).PosT And inputx(icount).PosT = ArrPos(0) Then
                ReqType = "Case1"
            ElseIf inputx(icount).PosF > inputx(icount).PosT And inputx(icount).PosF <> ArrPos(0) Then
                ReqType = "Case2"
            End If
        End If
    Next

    Select Case ReqType
        Case "Case1"
            For icount = 0 To CInt(Form1.txtDim.Text) - 1
                If CheckHF(icount).Checked = True Then
                    If inputx(icount).PosF < inputx(icount).PosT Then
                        CPos = inputx(icount).PosT
                    ElseIf inputx(icount).PosF > inputx(icount).PosT Then
                        CPos = inputx(icount).PosF
                    End If
                End If
            Next
            For icount = 0 To CInt(Form1.txtDim.Text) - 1
                If inputx(icount).Dims = 0 And inputx(icount).Dimtol = 0 Then
                    If inputx(icount).PosF < inputx(icount).PosT Then
                        If inputx(icount).PosT > CPos Then
                            Shift_to = "Left"
                        ElseIf inputx(icount).PosT < CPos Then
                            Shift_to = "Right"
                        End If
                    ElseIf inputx(icount).PosF > inputx(icount).PosT Then
                        If inputx(icount).PosF > CPos Then
                            Shift_to = "Left"
                        ElseIf inputx(icount).PosF < CPos Then
                            Shift_to = "Right"
                        End If
                    End If
                End If
            Next
        Case "Case2"
            Repeat = 0
            For icount = 0 To CInt(Form1.txtDim.Text) - 1
                If inputx(icount).Dims = 0 And inputx(icount).Dimtol = 0 Then
                    If inputx(icount).PosF < inputx(icount).PosT Then
                        CPos = inputx(icount).PosT
                    ElseIf inputx(icount).PosT < inputx(icount).PosF Then
                        CPos = inputx(icount).PosF
                    End If
                End If
            Next
            For icount = 0 To CInt(Form1.txtDim.Text) - 1
                If CheckHF(icount).Checked = False Then
                    If inputx(icount).PosF < inputx(icount).PosT Then
                        If inputx(icount).PosT = CPos Then
                            Shift_to = "Left"
                            Repeat = 1
                        End If
                    ElseIf inputx(icount).PosT < inputx(icount).PosF Then
                        If inputx(icount).PosT = CPos Then
                            Shift_to = "Left"
                            Repeat = 1
                        End If
                    End If
                End If
                If Repeat = 0 Then
                    Shift_to = "Right"
                End If
            Next
    End Select
Case "รูเจาะอยู่ขวา"
    For icount = 0 To CInt(Form1.txtDim.Text) - 1
        If inputx(icount).Dims = 0 And inputx(icount).Dimtol = 0 Then
```

```
                    If inputx(icount).PosF < inputx(icount).PosT And inputx(icount).PosT = ArrPos(ArrPos.Length - 1)
Then
                        ReqType = "Case1"
                    ElseIf inputx(icount).PosF < inputx(icount).PosT And inputx(icount).PosT <>
ArrPos(ArrPos.Length - 1) Then
                        ReqType = "Case2"
                    ElseIf inputx(icount).PosF > inputx(icount).PosT And inputx(icount).PosF = ArrPos(ArrPos.Length
- 1) Then
                        ReqType = "Case1"
                    ElseIf inputx(icount).PosF > inputx(icount).PosT And inputx(icount).PosF <>
ArrPos(ArrPos.Length - 1) Then
                        ReqType = "Case2"
                    End If
                End If
            Next

            Select Case ReqType
                Case "Case1"
                    For icount = 0 To CInt(Form1.txtDim.Text) - 1
                        If CheckHF(icount).Checked = True Then
                            If inputx(icount).PosF < inputx(icount).PosT Then
                                CPos = inputx(icount).PosF
                            ElseIf inputx(icount).PosF > inputx(icount).PosT Then
                                CPos = inputx(icount).PosT
                            End If
                        End If
                    Next
                    For icount = 0 To CInt(Form1.txtDim.Text) - 1
                        If inputx(icount).Dims = 0 And inputx(icount).Dimtol = 0 Then
                            If inputx(icount).PosF < inputx(icount).PosT Then
                                If inputx(icount).PosF > CPos Then
                                    Shift_to = "Left"
                                    Exit For
                                ElseIf inputx(icount).PosF < CPos Then
                                    Shift_to = "Right"
                                    Exit For
                                End If
                            ElseIf inputx(icount).PosF > inputx(icount).PosT Then
                                If inputx(icount).PosT > CPos Then
                                    Shift_to = "Left"
                                    Exit For
                                ElseIf inputx(icount).PosT < CPos Then
                                    Shift_to = "Right"
                                    Exit For
                                End If
                            End If
                        End If
                    Next
                Case "Case2"
                    For icount = 0 To CInt(Form1.txtDim.Text) - 1
                        If inputx(icount).Dims = 0 And inputx(icount).Dimtol = 0 Then
                            If inputx(icount).PosF = ArrPos(0) Then
                                Shift_to = "Right"
                            ElseIf inputx(icount).PosT = ArrPos(0) Then
                                Shift_to = "Right"
                            End If
                        End If
                    Next
                    If ReqType = "" Then
                        Shift_to = "Left"
                    End If
            End Select
        End Select

        Else
        'กรณีที่จำนวนรูเจาะมี 2 รู
        Side = "2รูเจาะ"
            ReqType = ""
        kcount = 0

        Do While kcount <= 1
            For jcount = 0 To CInt(Form1.txtDim.Text) - 1
```

```
                If CheckHF(jcount).Checked = True Then
                    If kcount = 0 Then
                        CPos = inputx(jcount).PosF
                    Else
                        CPos = inputx(jcount).PosT
                    End If
                End If
                For icount = 0 To CInt(Form1.txtDim.Text) - 1
                    If inputx(icount).Dims = 0 And inputx(icount).Dimtol = 0 Then
                        If inputx(icount).PosF = CPos Then
                            ReqType = "Case1"
                        ElseIf inputx(icount).PosT = CPos Then
                            ReqType = "Case1"
                        End If
                    End If
                    kcount = kcount + 1
                Next
            Next
        Loop

        If ReqType = "" Then
            ReqType = "Case2"
        End If
    End If

    Select Case ReqType
        Case "Case1"
            For icount = 0 To CInt(Form1.txtDim.Text) - 1
                If inputx(icount).Dims = 0 And inputx(icount).Dimtol = 0 Then
                    If ArrPos(4) = inputx(icount).PosF Then
                        Shift_to = "Left"
                    ElseIf ArrPos(2) = inputx(icount).PosF Then
                        Shift_to = "Right"
                    End If
                End If
            Next
        Case "Case2"
            For icount = 0 To CInt(Form1.txtDim.Text) - 1
                If CheckHF(icount).Checked = False Then
                    If inputx(icount).Dims <> 0 And inputx(icount).Dimtol <> 0 Then
                        If inputx(icount).PosF < inputx(icount).PosT Then
                            CPos = inputx(icount).PosF
                            For jcount = 0 To CInt(Form1.txtDim.Text) - 1
                                If inputx(jcount).Dims = 0 And inputx(jcount).Dimtol = 0 Then
                                    If inputx(jcount).PosF = CPos Then
                                        Shift_to = "Left"
                                    ElseIf inputx(jcount).PosT = CPos Then
                                        Shift_to = "Right"
                                    End If
                                End If
                            Next
                            CPos = inputx(icount).PosT
                            For jcount = 0 To CInt(Form1.txtDim.Text) - 1
                                If inputx(jcount).Dims = 0 And inputx(jcount).Dimtol = 0 Then
                                    If inputx(jcount).PosF = CPos Then
                                        Shift_to = "Left"
                                    ElseIf inputx(jcount).PosT = CPos Then
                                        Shift_to = "Right"
                                    End If
                                  End If
                                  Next
                    End If
                    End If
                    End If
                    Next
    End Select

ElseIf Form1.RadioButton6.Checked = True Then
    Repeat = 0
    If CInt(Form1.txtDim.Text) <= 3 Then
        For icount = 0 To CInt(Form1.txtDim.Text) - 1
            If CheckHF(icount).Checked = True Then
```

```
                If inputx(icount).PosF < inputx(icount).PosT Then
                    If inputx(icount).PosF = ArrPos(0) Then
                        Side = "รูเจาะอยู่ซ้าย"
                    ElseIf inputx(icount).PosT = ArrPos(ArrPos.Length - 1) Then
                        Side = "รูเจาะอยู่ขวา"
                    End If
                ElseIf inputx(icount).PosT < inputx(icount).PosF Then
                    If inputx(icount).PosT = ArrPos(0) Then
                        Side = "รูเจาะอยู่ซ้าย"
                    ElseIf inputx(icount).PosF = ArrPos(ArrPos.Length - 1) Then
                        Side = "รูเจาะอยู่ขวา"
                    End If
                End If
            End If
        Next

        Select Case Side
            Case "รูเจาะอยู่ซ้าย"
                For icount = 0 To CInt(Form1.txtDim.Text) - 1
                    If inputx(icount).Dims = 0 And inputx(icount).Dimtol = 0 Then
                        If inputx(icount).PosF < inputx(icount).PosT And inputx(icount).PosF = ArrPos(0) Then
                            ReqType = "Case1"
                        ElseIf inputx(icount).PosF < inputx(icount).PosT And inputx(icount).PosT <> ArrPos(0) Then
                            ReqType = "Case2"
                        ElseIf inputx(icount).PosF > inputx(icount).PosT And inputx(icount).PosT = ArrPos(0) Then
                            ReqType = "Case1"
                        ElseIf inputx(icount).PosF > inputx(icount).PosT And inputx(icount).PosF <> ArrPos(0) Then
                            ReqType = "Case2"
                        End If
                    End If
                Next
                Select Case ReqType
                    Case "Case1"
                        For icount = 0 To CInt(Form1.txtDim.Text) - 1
                            If CheckHF(icount).Checked = True Then
                                If inputx(icount).PosF < inputx(icount).PosT Then
                                    CPos = inputx(icount).PosT
                                ElseIf inputx(icount).PosF > inputx(icount).PosT Then
                                    CPos = inputx(icount).PosF
                                End If
                            End If
                        Next
                        For icount = 0 To CInt(Form1.txtDim.Text) - 1
                            If inputx(icount).Dims = 0 And inputx(icount).Dimtol = 0 Then
                                If inputx(icount).PosF < inputx(icount).PosT Then
                                    If inputx(icount).PosT > CPos Then
                                        Shift_to = "Left"
                                    ElseIf inputx(icount).PosT < CPos Then
                                        Shift_to = "Right"
                                    End If
                                ElseIf inputx(icount).PosF > inputx(icount).PosT Then
                                    If inputx(icount).PosF > CPos Then
                                            Shift_to = "Left"
                                    ElseIf inputx(icount).PosF < CPos Then
                                        Shift_to = "Right"
                                    End If
                                End If
                            End If
                        Next
                    Case "Case2"
                        Repeat = 0
                        For icount = 0 To CInt(Form1.txtDim.Text) - 1
                            If inputx(icount).Dims = 0 And inputx(icount).Dimtol = 0 Then
                                If inputx(icount).PosF < inputx(icount).PosT Then
                                    CPos = inputx(icount).PosF
                                ElseIf inputx(icount).PosT < inputx(icount).PosF Then
                                    CPos = inputx(icount).PosT
                                End If
                            End If
                        Next
                        For icount = 0 To CInt(Form1.txtDim.Text) - 1
```

```
                        If CheckHF(icount).Checked = False Then
                            If inputx(icount).PosF < inputx(icount).PosT Then
                                If inputx(icount).PosF = CPos Then
                                    Shift_to = "Left"
                                    Repeat = 1
                                End If
                            ElseIf inputx(icount).PosT < inputx(icount).PosF Then
                                If inputx(icount).PosT = CPos Then
                                    Shift_to = "Left"
                                    Repeat = 1
                                End If
                            End If
                        End If
                        If Repeat = 0 Then
                            Shift_to = "Right"
                        End If
                    Next
                End Select
            Case "รูเจาะอยู่ขวา"
                For icount = 0 To CInt(Form1.txtDim.Text) - 1
                    If inputx(icount).Dims = 0 And inputx(icount).Dimtol = 0 Then
                        If inputx(icount).PosF < inputx(icount).PosT And inputx(icount).PosT = ArrPos(ArrPos.Length - 1)
Then
                            ReqType = "Case1"
                        ElseIf inputx(icount).PosF < inputx(icount).PosT And inputx(icount).PosT <>
ArrPos(ArrPos.Length - 1) Then
                            ReqType = "Case2"
                        ElseIf inputx(icount).PosF > inputx(icount).PosT And inputx(icount).PosF = ArrPos(ArrPos.Length
- 1) Then
                            ReqType = "Case1"
                        ElseIf inputx(icount).PosF > inputx(icount).PosT And inputx(icount).PosF <>
ArrPos(ArrPos.Length - 1) Then
                            ReqType = "Case2"
                        End If
                    End If
                Next

                Select Case ReqType
                    Case "Case1"
                        For icount = 0 To CInt(Form1.txtDim.Text) - 1
                            If CheckHF(icount).Checked = True Then
                                If inputx(icount).PosF < inputx(icount).PosT Then
                                    CPos = inputx(icount).PosF
                                ElseIf inputx(icount).PosF > inputx(icount).PosT Then
                                    CPos = inputx(icount).PosT
                                End If
                            End If
                        Next
                        For icount = 0 To CInt(Form1.txtDim.Text) - 1
                            If inputx(icount).Dims = 0 And inputx(icount).Dimtol = 0 Then
                                If inputx(icount).PosF < inputx(icount).PosT Then
                                    If inputx(icount).PosF > CPos Then
                                        Shift_to = "Left"
                                    ElseIf inputx(icount).PosF < CPos Then
                                        Shift_to = "Right"
                                    End If
                                ElseIf inputx(icount).PosF > inputx(icount).PosT Then
                                    If inputx(icount).PosT > CPos Then
                                        Shift_to = "Left"
                                    ElseIf inputx(icount).PosT < CPos Then
                                        Shift_to = "Right"
                                    End If
                                End If
                            End If
                        Next
                    Case "Case2"
                        Repeat = 0
                        For icount = 0 To CInt(Form1.txtDim.Text) - 1
                            If inputx(icount).Dims = 0 And inputx(icount).Dimtol = 0 Then
                                If inputx(icount).PosF < inputx(icount).PosT Then
                                    CPos = inputx(icount).PosF
                                ElseIf inputx(icount).PosT < inputx(icount).PosF Then
```

```
                        CPos = inputx(icount).PosT
                    End If
                End If
            Next
            For icount = 0 To CInt(Form1.txtDim.Text) - 1
                If CheckHF(icount).Checked = False Then
                    If inputx(icount).PosF < inputx(icount).PosT Then
                        If inputx(icount).PosF = CPos Then
                            Shift_to = "Left"
                            Repeat = 1
                        End If
                    ElseIf inputx(icount).PosT < inputx(icount).PosF Then
                        If inputx(icount).PosT = CPos Then
                            Shift_to = "Left"
                            Repeat = 1
                        End If
                    End If
                End If
                If Repeat = 0 Then
                    Shift_to = "Right"
                End If
            Next
        End Select
    End Select

   Else
    'กรณีที่จำนวนรูเจาะมี 2 รู
    Side = "2รูเจาะ"
    For icount = 0 To CInt(Form1.txtDim.Text) - 1
        If CheckHF(icount).Checked = False Then
            If inputx(icount).Dims <> 0 And inputx(icount).Dimtol <> 0 Then
                If inputx(icount).PosF < inputx(icount).PosT And inputx(icount).PosF = ArrPos(1) Then
                    Shift_to = "Left"
                ElseIf inputx(icount).PosF > inputx(icount).PosT And inputx(icount).PosT = ArrPos(1) Then
                    Shift_to = "Left"
                Else
                    Shift_to = "Right"
                End If
            End If
        End If
    Next
  End If
ElseIf Form1.RadioButton7.Checked = True Then
  For icount = 0 To CInt(Form1.txtDim.Text) - 1
    If CheckHF(icount).Checked = False Then
      If inputx(icount).Dims <> 0 And inputx(icount).Dimtol <> 0 Then
        If inputx(icount).PosF = ArrPos(0) Then
          Shift_to = "Left"
        ElseIf inputx(icount).PosT = ArrPos(0) Then
          Shift_to = "Left"
        ElseIf inputx(icount).PosF = ArrPos(ArrPos.Length - 1) Then
          Shift_to = "Right"
        ElseIf inputx(icount).PosT = ArrPos(ArrPos.Length - 1) Then
          Shift_to = "Right"
        End If
      End If
    End If
  Next

  If CInt(Form1.txtDim.Text) <= 3 Then
    Select Case Shift_to
      Case "Left"
        For icount = 0 To CInt(Form1.txtDim.Text) - 1
          If CheckHF(icount).Checked = True Then
            If inputx(icount).PosF < inputx(icount).PosT Then
              If inputx(icount).PosF = ArrPos(1) And inputx(icount).PosT = ArrPos(2) Then
                Side = "รูเจาะอยู่ซ้าย"
              Else
                Side = "รูเจาะอยู่ขวา"
              End If
            ElseIf inputx(icount).PosF > inputx(icount).PosT Then
```

```
                        If inputx(icount).PosT = ArrPos(1) And inputx(icount).PosF = ArrPos(2) Then
                            Side = "รูเจาะอยู่ซ้าย"
                        Else
                            Side = "รูเจาะอยู่ขวา"
                        End If
                    End If
                End If
            Next
        Case "Right"
            For icount = 0 To CInt(Form1.txtDim.Text) - 1
                If CheckHF(icount).Checked = True Then
                    If inputx(icount).PosF < inputx(icount).PosT Then
                        If inputx(icount).PosF = ArrPos(2) And inputx(icount).PosT = ArrPos(3) Then
                            Side = "รูเจาะอยู่ซ้าย"
                        Else
                            Side = "รูเจาะอยู่ขวา"
                        End If
                    ElseIf inputx(icount).PosF > inputx(icount).PosT Then
                        If inputx(icount).PosT = ArrPos(2) And inputx(icount).PosF = ArrPos(3) Then
                            Side = "รูเจาะอยู่ซ้าย"
                        Else
                            Side = "รูเจาะอยู่ขวา"
                        End If
                    End If
                End If
            Next
    End Select
  Else
    Side = "2รูเจาะ"
  End If
End If

ElseIf Changeaxis = 1 Then
 'หาตำแหน่งรูเจาะ(ซ้าย/ขวา)
 If Form1.RadioButton23.Checked = True Then
    For icount = 0 To CInt(Form1.txtDimV.Text) - 1
        If inputx(icount).Dims <> 0 And inputx(icount).Dimtol <> 0 Then
            If CheckVF(icount).Checked = False Then
                If ArrPos(0) = inputx(icount).PosF Then
                    Shift_to = "Left"
                    Exit For
                ElseIf ArrPos(0) = inputx(icount).PosT Then
                    Shift_to = "Left"
                    Exit For
                ElseIf ArrPos(ArrPos.Length - 1) = inputx(icount).PosF Then
                    Shift_to = "Right"
                    Exit For
                ElseIf ArrPos(ArrPos.Length - 1) = inputx(icount).PosT Then
                    Shift_to = "Right"
                    Exit For
                End If
            End If
        End If
    Next

 ElseIf Form1.RadioButton24.Checked = True Then
    For icount = 0 To CInt(Form1.txtDimV.Text) - 1
        If inputx(icount).Dims <> 0 And inputx(icount).Dimtol <> 0 Then
            If ArrPos(1) = inputx(icount).PosF Then
                Repeat = Repeat + 1
            ElseIf ArrPos(1) = inputx(icount).PosT Then
                Repeat = Repeat + 1
            End If
        End If
    Next
    If Repeat > 1 Then
        Shift_to = "Left"
    ElseIf Repeat = 1 Then
        Shift_to = "Right"
    End If
```

```
ElseIf Form1.RadioButton25.Checked = True Then
    ReqType = ""
    kcount = 0
    Do While kcount <= 1
        For jcount = 0 To CInt(Form1.txtDim.Text) - 1
            If CheckHF(jcount).Checked = True Then
                If kcount = 0 Then
                    CPos = inputx(jcount).PosF
                Else
                    CPos = inputx(jcount).PosT
                End If
            End If
            For icount = 0 To CInt(Form1.txtDim.Text) - 1
                If inputx(icount).Dims = 0 And inputx(icount).Dimtol = 0 Then
                    If inputx(icount).PosF = CPos Then
                        ReqType = "Case1"
                    ElseIf inputx(icount).PosT = CPos Then
                        ReqType = "Case1"
                    End If
                End If
                kcount = kcount + 1
            Next
        Next
    Loop
    If ReqType = "" Then
        ReqType = "Case2"
    End If

    'check ทิศทางการผลัก
    Select Case ReqType
        Case "Case1"
            Repeat = 0
            For icount = 0 To CInt(Form1.txtDim.Text) - 1
                If CheckHF(icount).Checked = True Then
                    If inputx(icount).PosF < inputx(icount).PosT Then
                        CPos = inputx(icount).PosT
                    ElseIf inputx(icount).PosF > inputx(icount).PosT Then
                        CPos = inputx(icount).PosF
                    End If
                End If
            Next
            For icount = 0 To CInt(Form1.txtDim.Text) - 1
                If inputx(icount).Dims = 0 And inputx(icount).Dimtol = 0 Then
                    If inputx(icount).PosT = CPos Then
                        Shift_to = "Left"
                    ElseIf inputx(icount).PosF = CPos Then
                        Shift_to = "Left"
                    Else
                        Shift_to = "Right"
                    End If
                End If
            Next
        Case "Case2"
            For icount = 0 To CInt(Form1.txtDim.Text) - 1
                If inputx(icount).Dims = 0 And inputx(icount).Dimtol = 0 Then
                    If inputx(icount).PosF = ArrPos(ArrPos.Length - 1) Then
                        Shift_to = "Left"
                    ElseIf inputx(icount).PosT = ArrPos(ArrPos.Length - 1) Then
                        Shift_to = "Left"
                    ElseIf inputx(icount).PosF = ArrPos(0) Then
                        Shift_to = "Right"
                    ElseIf inputx(icount).PosT = ArrPos(0) Then
                        Shift_to = "Right"
                    End If
                End If
            Next
    End Select

ElseIf Form1.RadioButton26.Checked = True Then
    ReqType = ""
    kcount = 0
    Do While kcount <= 1
```

```
        For jcount = 0 To CInt(Form1.txtDim.Text) - 1
            If CheckHF(jcount).Checked = True Then
                If kcount = 0 Then
                    CPos = inputx(jcount).PosF
                Else
                    CPos = inputx(jcount).PosT
                End If
            End If
            For icount = 0 To CInt(Form1.txtDim.Text) - 1
                If inputx(icount).Dims = 0 And inputx(icount).Dimtol = 0 Then
                    If inputx(icount).PosF = CPos Then
                        ReqType = "Case1"
                    ElseIf inputx(icount).PosT = CPos Then
                        ReqType = "Case1"
                    End If
                End If
                kcount = kcount + 1
            Next
        Next
    Loop
    If ReqType = "" Then
        ReqType = "Case2"
    End If

     'check ทิศทางการผลัก
    Select Case ReqType
        Case "Case1"
            Repeat = 0
            For icount = 0 To CInt(Form1.txtDim.Text) - 1
                If inputx(icount).Dims = 0 And inputx(icount).Dimtol = 0 Then
                    If inputx(icount).PosF < inputx(icount).PosT Then
                        CPos = inputx(icount).PosF
                    ElseIf inputx(icount).PosT < inputx(icount).PosF Then
                        CPos = inputx(icount).PosT
                    End If
                End If
            Next
            For icount = 0 To CInt(Form1.txtDim.Text) - 1
                If CheckHF(icount).Checked = False Then
                    If inputx(icount).PosF < inputx(icount).PosT Then
                        If inputx(icount).PosF = CPos Then
                            Shift_to = "Left"
                            Repeat = 1
                        End If
                    ElseIf inputx(icount).PosT < inputx(icount).PosF Then
                        If inputx(icount).PosT = CPos Then
                            Shift_to = "Left"
                            Repeat = 1
                        End If
                    End If
                End If
                If Repeat = 0 Then
                    Shift_to = "Right"
                End If
            Next
        Case "Case2"
            For icount = 0 To CInt(Form1.txtDim.Text) - 1
                If inputx(icount).Dims <> 0 And inputx(icount).Dimtol <> 0 Then
                    If CheckVF(icount).Checked = False Then
                        If inputx(icount).PosF = ArrPos(0) Then
                            Shift_to = "Left"
                        ElseIf inputx(icount).PosT = ArrPos(0) Then
                            Shift_to = "Left"
                        ElseIf inputx(icount).PosF = ArrPos(ArrPos.Length - 1) Then
                            Shift_to = "Right"
                        ElseIf inputx(icount).PosT = ArrPos(ArrPos.Length - 1) Then
                            Shift_to = "Right"
                        End If
                    End If
                End If
            Next
    End Select
```

```
ElseIf Form1.RadioButton27.Checked = True Then
   ReqType = ""
   kcount = 0
   Do While kcount <= 1
      For jcount = 0 To CInt(Form1.txtDim.Text) - 1
         If CheckHF(jcount).Checked = True Then
            If kcount = 0 Then
               CPos = inputx(jcount).PosF
            Else
               CPos = inputx(jcount).PosT
            End If
         End If
         For icount = 0 To CInt(Form1.txtDim.Text) - 1
            If inputx(icount).Dims = 0 And inputx(icount).Dimtol = 0 Then
               If inputx(icount).PosF = CPos Then
                  ReqType = "Case1"
               ElseIf inputx(icount).PosT = CPos Then
                  ReqType = "Case1"
               End If
            End If
            kcount = kcount + 1
         Next
      Next
   Loop
   If ReqType = "" Then
      ReqType = "Case2"
   End If

   'check ทิศทางการผลัก
   Select Case ReqType
      Case "Case1"
         Repeat = 0
         For icount = 0 To CInt(Form1.txtDim.Text) - 1
            If inputx(icount).Dims = 0 And inputx(icount).Dimtol = 0 Then
               If inputx(icount).PosF < inputx(icount).PosT Then
                  CPos = inputx(icount).PosF
               ElseIf inputx(icount).PosT < inputx(icount).PosF Then
                  CPos = inputx(icount).PosT
               End If
            End If
         Next
         For icount = 0 To CInt(Form1.txtDim.Text) - 1
            If CheckHF(icount).Checked = False Then
               If inputx(icount).PosF < inputx(icount).PosT Then
                  If inputx(icount).PosF = CPos Then
                     Shift_to = "Left"
                     Repeat = 1
                  End If
               ElseIf inputx(icount).PosT < inputx(icount).PosF Then
                  If inputx(icount).PosT = CPos Then
                     Shift_to = "Left"
                     Repeat = 1
                  End If
               End If
            End If
            If Repeat = 0 Then
               Shift_to = "Right"
            End If
         Next
      Case "Case2"
         For icount = 0 To CInt(Form1.txtDim.Text) - 1
            If inputx(icount).Dims <> 0 And inputx(icount).Dimtol <> 0 Then
                  If CheckVF(icount).Checked = False Then
                  If inputx(icount).PosF = ArrPos(0) Then
                     Shift_to = "Left"
                  ElseIf inputx(icount).PosT = ArrPos(0) Then
                     Shift_to = "Left"
                  ElseIf inputx(icount).PosF = ArrPos(ArrPos.Length - 1) Then
                     Shift_to = "Right"
                  ElseIf inputx(icount).PosT = ArrPos(ArrPos.Length - 1) Then
                     Shift_to = "Right"
```

```
                    End If
                End If
            End If
        Next
    End Select
  End If
 End If
End Sub

Public Function LocatorsizeTol_cle(ByVal A As String, B As String) As Single
  Dim locasize As Single

  If Changeaxis <> 1 Then
    If Form1.RadioButton3.Checked Then        ' add clearance จาก Tol. pin ที่ใช้ทำรัง งานเหลี่ยม
      IndexTab = False
      For icount = 0 To inputx.Length - 1 'Pin size.Tol
        If inputx(icount).PosF = A And inputx(icount).PosT = B Then
          locasize = IT6(inputx(icount).Dims)
          IndexTab = True
        ElseIf inputx(icount).PosT = A And inputx(icount).PosF = B Then
          locasize = IT6(inputx(icount).Dims)
          IndexTab = True
        End If
      Next
    End If

    If Form1.RadioButton4.Checked Then
      IndexTab = False
      For icount = 0 To inputx.Length - 1 'Pin size.Tol
        If inputx(icount).PosF = A And inputx(icount).PosT = B Then
          locasize = (inputx(icount).Dimtol) * 2
          IndexTab = True
        ElseIf inputx(icount).PosT = A And inputx(icount).PosF = B Then
          locasize = (inputx(icount).Dimtol) * 2
          IndexTab = True
        End If
      Next
    End If

    If Form1.RadioButton5.Checked Then 'add clearance จาก Tol. locator ของงานกลม ใช้ Pin locator
      IndexTab = False
      For icount = 0 To inputx.Length - 1 'Pin size.Tol
        If inputx(icount).PosF = A And inputx(icount).PosT = B Then
          locasize = IT6(inputx(icount).Dims)
          IndexTab = True
        ElseIf inputx(icount).PosT = A And inputx(icount).PosF = B Then
          locasize = IT6(inputx(icount).Dims)
          IndexTab = True
        End If
      Next
    End If

     If Form1.RadioButton6.Checked Then 'add clearance จาก Tol. locator ของงานกลม ใช้ Hole locator
      IndexTab = False
      For icount = 0 To inputx.Length - 1 'Pin size.Tol
        If inputx(icount).PosF = A And inputx(icount).PosT = B Then
          locasize = IT6(inputx(icount).Dims)
          IndexTab = True
        ElseIf inputx(icount).PosT = A And inputx(icount).PosF = B Then
          locasize = IT6(inputx(icount).Dims)
          IndexTab = True
        End If
      Next
    End If

    If Form1.RadioButton7.Checked Then 'add clearance จาก Tol. locator ของงานกลม ใช้ Hole locator
      IndexTab = False
      For icount = 0 To inputx.Length - 1 'Pin size.Tol
        If inputx(icount).PosF = A And inputx(icount).PosT = B Then
          locasize = IT6(inputx(icount).Dims)
          IndexTab = True
        ElseIf inputx(icount).PosT = A And inputx(icount).PosF = B Then
          locasize = IT6(inputx(icount).Dims)
```

```
            IndexTab = True
          End If
        Next
      End If

    ElseIf Changeaxis = 1 Then
      If Form1.RadioButton23.Checked Then
        IndexTab = False
        For icount = 0 To inputx.Length - 1 'Pin size.Tol
          If inputx(icount).PosF = A And inputx(icount).PosT = B Then
            locasize = IT6(inputx(icount).Dims)
            IndexTab = True
          ElseIf inputx(icount).PosT = A And inputx(icount).PosF = B Then
            locasize = IT6(inputx(icount).Dims)
            IndexTab = True
          End If
        Next
      End If

      If Form1.RadioButton24.Checked Then
        IndexTab = False
        For icount = 0 To inputx.Length - 1 'Pin size.Tol
          If inputx(icount).PosF = A And inputx(icount).PosT = B Then
            locasize = (inputx(icount).Dimtol) * 2
            IndexTab = True
          ElseIf inputx(icount).PosT = A And inputx(icount).PosF = B Then
            locasize = (inputx(icount).Dimtol) * 2
            IndexTab = True
          End If
        Next
      End If

      If Form1.RadioButton25.Checked Then 'add clearance จาก Tol. locator ของงานกลม ใช้ Pin locator
        IndexTab = False
        For icount = 0 To inputx.Length - 1 'Pin size.Tol
          If inputx(icount).PosF = A And inputx(icount).PosT = B Then
            locasize = IT6(inputx(icount).Dims)
            IndexTab = True
          ElseIf inputx(icount).PosT = A And inputx(icount).PosF = B Then
            locasize = IT6(inputx(icount).Dims)
            IndexTab = True
          End If
        Next
      End If

      If Form1.RadioButton26.Checked Then 'add clearance จาก Tol. locator ของงานกลม ใช้ Hole locator
        IndexTab = False
        For icount = 0 To inputx.Length - 1 'Pin size.Tol
          If inputx(icount).PosF = A And inputx(icount).PosT = B Then
            locasize = IT6(inputx(icount).Dims)
            IndexTab = True
          ElseIf inputx(icount).PosT = A And inputx(icount).PosF = B Then
            locasize = IT6(inputx(icount).Dims)
            IndexTab = True
          End If
        Next
      End If

      If Form1.RadioButton27.Checked Then 'add clearance จาก Tol. locator ของงานกลม ใช้ Hole locator
        IndexTab = False
        For icount = 0 To inputx.Length - 1 'Pin size.Tol
          If inputx(icount).PosF = A And inputx(icount).PosT = B Then
            locasize = IT6(inputx(icount).Dims)
            IndexTab = True
          ElseIf inputx(icount).PosT = A And inputx(icount).PosF = B Then
            locasize = IT6(inputx(icount).Dims)
            IndexTab = True
          End If
        Next
      End If
    End If

    Return (locasize)
```

```
End Function

Public Function Clearance_cal(ByVal Type As Integer, ByVal A As String, ByVal B As String, ByVal C As String, ByVal
D As String) As Single()

    Dim Clearance_minclear(1) As Single

    If Changeaxis <> 1 Then
        If Type = 1 Then
            IndexTab = False
            For icount = 0 To inputx.Length - 1
                If Form1.RadioButton5.Checked Then
                    Clearance = Fit_Ch(inputx(icount).Dims)
                    IndexTab = True
                ElseIf Form1.RadioButton6.Checked Then
                    Clearance = Fit_Ch(inputx(icount).Dims)
                    IndexTab = True
                ElseIf Form1.RadioButton7.Checked Then
                    Clearance = Fit_Ch(inputx(icount).Dims)
                    IndexTab = True
                Else
                    Clearance = Fit_Ch(inputx(icount).Dims)
                    IndexTab = True
                End If

            ElseIf inputx(icount).PosT = A And inputx(icount).PosF = B Then
                If Form1.RadioButton5.Checked Then
                    Clearance = Fit_Ch(inputx(icount).Dims)
                    IndexTab = True
                ElseIf Form1.RadioButton6.Checked Then
                    Clearance = Fit_Ch(inputx(icount).Dims)
                    IndexTab = True
                ElseIf Form1.RadioButton7.Checked Then
                    Clearance = Fit_Ch(inputx(icount).Dims)
                    IndexTab = True
                Else
                    Clearance = Fit_Ch(inputx(icount).Dims)
                    IndexTab = True
                End If
            End If
        Next

        ElseIf Type = 2 Then
            IndexTab = False
            For icount = 0 To inputx.Length - 1
                If inputx(icount).PosF = A And inputx(icount).PosT = B Then
                    If Form1.RadioButton5.Checked Then
                        Clearance = Fit_Gh(inputx(icount).Dims)
                        IndexTab = True
                    ElseIf Form1.RadioButton6.Checked Then
                        Clearance = Fit_Gh(inputx(icount).Dims)
                        IndexTab = True
                    ElseIf Form1.RadioButton7.Checked Then
                        Clearance = Fit_Gh(inputx(icount).Dims)
                    Else
                        Clearance = Fit_Gh(inputx(icount).Dims)
                        IndexTab = True
                    End If

                ElseIf inputx(icount).PosT = A And inputx(icount).PosF = B Then
                    If Form1.RadioButton5.Checked Then
                        Clearance = Fit_Gh(inputx(icount).Dims)
                        IndexTab = True
                    ElseIf Form1.RadioButton6.Checked Then
                        Clearance = Fit_Gh(inputx(icount).Dims)
                        IndexTab = True
                    ElseIf Form1.RadioButton7.Checked Then
                        Clearance = Fit_Gh(inputx(icount).Dims)
                        IndexTab = True
                    Else
                        Clearance = Fit_Gh(inputx(icount).Dims)
                        IndexTab = True
                    End If
```

```
            End If
        Next

    ElseIf Type = 3 Then
        IndexTab = False
        For icount = 0 To inputx.Length - 1
            If inputx(icount).PosF = A And inputx(icount).PosT = B Then
                If Form1.RadioButton5.Checked Then
                    Clearance = Fit_Hh(inputx(icount).Dims)
                    IndexTab = True
                ElseIf Form1.RadioButton6.Checked Then
                    Clearance = Fit_Hh(inputx(icount).Dims)
                    IndexTab = True
                ElseIf Form1.RadioButton7.Checked Then
                    Clearance = Fit_Hh(inputx(icount).Dims)
                    IndexTab = True
                Else
                    Clearance = Fit_Hh(inputx(icount).Dims)
                    IndexTab = True
                End If

            ElseIf inputx(icount).PosT = A And inputx(icount).PosF = B Then
                If Form1.RadioButton5.Checked Then
                    Clearance = Fit_Hh(inputx(icount).Dims)
                    IndexTab = True
                ElseIf Form1.RadioButton6.Checked Then
                    Clearance = Fit_Hh(inputx(icount).Dims)
                    IndexTab = True
                ElseIf Form1.RadioButton7.Checked Then
                    Clearance = Fit_Hh(inputx(icount).Dims)
                    IndexTab = True
                Else
                    Clearance = Fit_Hh(inputx(icount).Dims)
                    IndexTab = True
                End If
            End If
        Next
    End If
    Clearance_minclear(1) = Clearance

    ' add clearance จาก Pin Pos. Tol.(เฉพาะ nesting pin)
    If Form1.RadioButton4.Checked Then
        Dim BasePinPos As Single = 0
        For icount = 0 To inputx.Length - 1 'Pin pos.Tol
            If inputx(icount).PosF = A And inputx(icount).PosT = B Then
                BasePinPos = BasePinPos + inputx(icount).Dims
            End If
            If inputx(icount).PosF = C And inputx(icount).PosT = D Then
                BasePinPos = BasePinPos + inputx(icount).Dims
            End If
        Next

        IndexTab = False
        Clearance = Clearance + IT8(BasePinPos)
        IndexTab = True
    End If

    ' add clearance จาก Work size Tol.
    If Form1.RadioButton3.Checked Then
        For icount = 0 To inputx.Length - 1
            If inputx(icount).PosF = A And inputx(icount).PosT = B Then
                Clearance = Clearance + inputx(icount).Dimtol * 2
            ElseIf inputx(icount).PosT = A And inputx(icount).PosF = B Then
                Clearance = Clearance + inputx(icount).Dimtol * 2
            End If
        Next
    End If

    If Form1.RadioButton4.Checked Then
        For icount = 0 To inputx.Length - 1
            If inputx(icount).PosF = A And inputx(icount).PosT = B Then
                Clearance = Clearance + inputx(icount).Dimtol * 2
            ElseIf inputx(icount).PosT = A And inputx(icount).PosF = B Then
```

```
                Clearance = Clearance + inputx(icount).Dimtol * 2
            End If
        Next
    End If

    If Form1.RadioButton5.Checked Then
        For icount = 0 To inputx.Length - 1
            If inputx(icount).PosF = A And inputx(icount).PosT = B Then
                Clearance = Clearance + inputx(icount).Dimtol * 2
            ElseIf inputx(icount).PosT = A And inputx(icount).PosF = B Then
                Clearance = Clearance + inputx(icount).Dimtol * 2
            End If
        Next
    End If

    If Form1.RadioButton6.Checked Then
        For icount = 0 To inputx.Length - 1
            If inputx(icount).PosF = A And inputx(icount).PosT = B Then
                Clearance = Clearance + inputx(icount).Dimtol * 2
            ElseIf inputx(icount).PosT = A And inputx(icount).PosF = B Then
                Clearance = Clearance + inputx(icount).Dimtol * 2
            End If
        Next
    End If

    If Form1.RadioButton7.Checked Then
        For icount = 0 To inputx.Length - 1
            If inputx(icount).PosF = A And inputx(icount).PosT = B Then
                Clearance = Clearance + inputx(icount).Dimtol * 2
            ElseIf inputx(icount).PosT = A And inputx(icount).PosF = B Then
                Clearance = Clearance + inputx(icount).Dimtol * 2
            End If
        Next

    End If

    ' add clearance จาก Pin Size Tol./Locator size Tol.
    If Form1.RadioButton3.Checked Then
        Clearance = Clearance + LocatorsizeTol_cle(A, B)
    End If
    If Form1.RadioButton4.Checked Then
        Clearance = Clearance + LocatorsizeTol_cle(C, D)
    End If
    If Form1.RadioButton5.Checked Then
        Clearance = Clearance + LocatorsizeTol_cle(C, D)
    End If
    If Form1.RadioButton6.Checked Then
        Clearance = Clearance + LocatorsizeTol_cle(C, D)
    End If
    If Form1.RadioButton7.Checked Then
        Clearance = Clearance + LocatorsizeTol_cle(C, D)
    End If
ElseIf Changeaxis = 1 Then
    If Type = 1 Then
        IndexTab = False
        For icount = 0 To inputx.Length - 1
            If inputx(icount).PosF = A And inputx(icount).PosT = B Then
                If Form1.RadioButton25.Checked Then
                    Clearance = Fit_Ch(inputx(icount).Dims)
                    IndexTab = True
                ElseIf Form1.RadioButton26.Checked Then
                    Clearance = Fit_Ch(inputx(icount).Dims)
                    IndexTab = True
                ElseIf Form1.RadioButton27.Checked Then
                    Clearance = Fit_Ch(inputx(icount).Dims)
                    IndexTab = True
                Else
                    Clearance = Fit_Ch(inputx(icount).Dims)
                    IndexTab = True
                End If
            ElseIf inputx(icount).PosT = A And inputx(icount).PosF = B Then
                If Form1.RadioButton25.Checked Then
                    Clearance = Fit_Ch(inputx(icount).Dims)
```

```
                IndexTab = True
            ElseIf Form1.RadioButton26.Checked Then
                Clearance = Fit_Ch(inputx(icount).Dims)
                IndexTab = True
            ElseIf Form1.RadioButton27.Checked Then
                Clearance = Fit_Ch(inputx(icount).Dims)
                IndexTab = True
            Else
                Clearance = Fit_Ch(inputx(icount).Dims)
                IndexTab = True
            End If
        End If
    Next

ElseIf Type = 2 Then
    IndexTab = False
    For icount = 0 To inputx.Length - 1
        If inputx(icount).PosF = A And inputx(icount).PosT = B Then
            If Form1.RadioButton25.Checked Then
                Clearance = Fit_Gh(inputx(icount).Dims)
                IndexTab = True
            ElseIf Form1.RadioButton26.Checked Then
                Clearance = Fit_Gh(inputx(icount).Dims)
                IndexTab = True
            ElseIf Form1.RadioButton27.Checked Then
                Clearance = Fit_Gh(inputx(icount).Dims)
            Else
                Clearance = Fit_Gh(inputx(icount).Dims)
                IndexTab = True
            End If
        ElseIf inputx(icount).PosT = A And inputx(icount).PosF = B Then
            If Form1.RadioButton25.Checked Then
                Clearance = Fit_Gh(inputx(icount).Dims)
                IndexTab = True
            ElseIf Form1.RadioButton26.Checked Then
                Clearance = Fit_Gh(inputx(icount).Dims)
                IndexTab = True
            ElseIf Form1.RadioButton27.Checked Then
                Clearance = Fit_Gh(inputx(icount).Dims)
                IndexTab = True
            Else
                Clearance = Fit_Gh(inputx(icount).Dims)
                IndexTab = True
            End If
        End If
    Next

ElseIf Type = 3 Then
    IndexTab = False
    For icount = 0 To inputx.Length - 1
        If inputx(icount).PosF = A And inputx(icount).PosT = B Then
            If Form1.RadioButton25.Checked Then
                Clearance = Fit_Hh(inputx(icount).Dims)
                IndexTab = True
            ElseIf Form1.RadioButton26.Checked Then
                Clearance = Fit_Hh(inputx(icount).Dims)
                IndexTab = True
            ElseIf Form1.RadioButton27.Checked Then
                Clearance = Fit_Hh(inputx(icount).Dims)
                IndexTab = True
            Else
                Clearance = Fit_Hh(inputx(icount).Dims)
                IndexTab = True
            End If
        ElseIf inputx(icount).PosT = A And inputx(icount).PosF = B Then
            If Form1.RadioButton25.Checked Then
                Clearance = Fit_Hh(inputx(icount).Dims)
                IndexTab = True
            ElseIf Form1.RadioButton26.Checked Then
                Clearance = Fit_Hh(inputx(icount).Dims)
                IndexTab = True
            ElseIf Form1.RadioButton27.Checked Then
```

```
                Clearance = Fit_Hh(inputx(icount).Dims)
                IndexTab = True
            Else
                Clearance = Fit_Hh(inputx(icount).Dims)
                IndexTab = True
            End If
        End If
    Next
End If

Clearance_minclear(1) = Clearance

' add clearance จาก Pin Pos. Tol.(เฉพาะ nesting pin)
If Form1.RadioButton24.Checked Then
    Dim BasePinPos As Single = 0
    For icount = 0 To inputx.Length - 1 'Pin pos.Tol
        If inputx(icount).PosF = A And inputx(icount).PosT = B Then
            BasePinPos = BasePinPos + inputx(icount).Dims
        End If
        If inputx(icount).PosF = C And inputx(icount).PosT = D Then
            BasePinPos = BasePinPos + inputx(icount).Dims
        End If
    Next
    IndexTab = False
    Clearance = Clearance + IT8(BasePinPos)
    IndexTab = True
End If

' add clearance จาก Work size Tol.
If Form1.RadioButton23.Checked Then
    For icount = 0 To inputx.Length - 1
        If inputx(icount).PosF = A And inputx(icount).PosT = B Then
            Clearance = Clearance + inputx(icount).Dimtol * 2
        ElseIf inputx(icount).PosT = A And inputx(icount).PosF = B Then
            Clearance = Clearance + inputx(icount).Dimtol * 2
        End If
    Next
End If

If Form1.RadioButton24.Checked Then
    For icount = 0 To inputx.Length - 1
        If inputx(icount).PosF = A And inputx(icount).PosT = B Then
            Clearance = Clearance + inputx(icount).Dimtol * 2
        ElseIf inputx(icount).PosT = A And inputx(icount).PosF = B Then
            Clearance = Clearance + inputx(icount).Dimtol * 2
        End If
    Next
End If

If Form1.RadioButton25.Checked Then
    For icount = 0 To inputx.Length - 1
        If inputx(icount).PosF = A And inputx(icount).PosT = B Then
            Clearance = Clearance + inputx(icount).Dimtol * 2
        ElseIf inputx(icount).PosT = A And inputx(icount).PosF = B Then
            Clearance = Clearance + inputx(icount).Dimtol * 2
        End If
    Next
End If

If Form1.RadioButton26.Checked Then
    For icount = 0 To inputx.Length - 1
        If inputx(icount).PosF = A And inputx(icount).PosT = B Then
            Clearance = Clearance + inputx(icount).Dimtol * 2
        ElseIf inputx(icount).PosT = A And inputx(icount).PosF = B Then
            Clearance = Clearance + inputx(icount).Dimtol * 2
        End If
    Next
End If

If Form1.RadioButton27.Checked Then
    For icount = 0 To inputx.Length - 1
        If inputx(icount).PosF = A And inputx(icount).PosT = B Then
            Clearance = Clearance + inputx(icount).Dimtol * 2
```

```
            ElseIf inputx(icount).PosT = A And inputx(icount).PosF = B Then
               Clearance = Clearance + inputx(icount).Dimtol * 2
            End If
         Next
      End If

      ' add clearance จาก Pin Size Tol./Locator size Tol.
      If Form1.RadioButton23.Checked Then
         Clearance = Clearance + LocatorsizeTol_cle(A, B)
      End If
      If Form1.RadioButton24.Checked Then
         Clearance = Clearance + LocatorsizeTol_cle(C, D)
      End If
      If Form1.RadioButton25.Checked Then
         Clearance = Clearance + LocatorsizeTol_cle(C, D)
      End If
      If Form1.RadioButton26.Checked Then
         Clearance = Clearance + LocatorsizeTol_cle(C, D)
      End If
      If Form1.RadioButton27.Checked Then
         Clearance = Clearance + LocatorsizeTol_cle(C, D)
      End If
      Clearance_minclear(0) = Clearance
   End If

   Return Clearance_minclear
End Function

Public Sub Add_clearance(ByVal A As Integer, ByVal cle() As Single)
   Dim icount, jcount As Integer
   Dim Max As Integer
   Dim Org() As Integer

   Max = FindMaxFea()

   If Changeaxis <> 1 Then
      If Form1.RadioButton3.Checked = True Then
         ReDim Org(0)
      ElseIf Form1.RadioButton4.Checked = True Then
         ReDim Org(0)
      Else
         ReDim Org(1)
      End If
   ElseIf Changeaxis = 1 Then
      If Form1.RadioButton23.Checked = True Then
         ReDim Org(0)
      ElseIf Form1.RadioButton24.Checked = True Then
         ReDim Org(0)
      Else
         ReDim Org(1)
      End If
   End If

   Select Case Shift_to
      Case "Left"
         If Changeaxis <> 1 Then
            If Form1.RadioButton3.Checked = True Then
               Org(0) = (ArrPos.Length - 1)
            ElseIf Form1.RadioButton4.Checked = True Then
               Org(0) = (ArrPos.Length - 2)
            ElseIf Form1.RadioButton5.Checked = True Then
               Org(0) = 1
               Org(1) = 3
            ElseIf Form1.RadioButton6.Checked = True Then
               Org(0) = 4
               Org(1) = 2
            ElseIf Form1.RadioButton7.Checked = True Then
               Org(0) = ArrPos.Length - 1
               Org(1) = 2
            End If
         ElseIf Changeaxis = 1 Then
```

```
            If Form1.RadioButton23.Checked = True Then
               Org(0) = (ArrPos.Length - 1)
            ElseIf Form1.RadioButton24.Checked = True Then
               Org(0) = (ArrPos.Length - 2)
            ElseIf Form1.RadioButton25.Checked = True Then
               Org(0) = 1
               Org(1) = 4
            ElseIf Form1.RadioButton26.Checked = True Then
               Org(0) = 5
               Org(1) = 3
            ElseIf Form1.RadioButton27.Checked = True Then
               Org(0) = 5
               Org(1) = 3
            End If
         End If

      Case "Right"
         If Changeaxis <> 1 Then
            If Form1.RadioButton3.Checked = True Then
               Org(0) = 1
            ElseIf Form1.RadioButton4.Checked = True Then
               Org(0) = 2
            ElseIf Form1.RadioButton5.Checked = True Then
               Org(0) = 4
               Org(1) = 2
            ElseIf Form1.RadioButton6.Checked = True Then
               Org(0) = 1
               Org(1) = 3
            ElseIf Form1.RadioButton7.Checked = True Then
               Org(0) = 1
               Org(1) = 3
            End If
         ElseIf Changeaxis = 1 Then
            If Form1.RadioButton23.Checked = True Then
               Org(0) = 1
            ElseIf Form1.RadioButton24.Checked = True Then
               Org(0) = 2
            ElseIf Form1.RadioButton25.Checked = True Then
               Org(0) = 5
               Org(1) = 3
            ElseIf Form1.RadioButton26.Checked = True Then
               Org(0) = 1
               Org(1) = 4
            ElseIf Form1.RadioButton27.Checked = True Then
               Org(0) = 1
               Org(1) = 4
            End If
         End If
   End Select

   Dim NewFea() As String
   If Changeaxis <> 1 Then
      If Form1.RadioButton3.Checked = True Then
         ReDim NewFea(1)
      ElseIf Form1.RadioButton4.Checked = True Then
         ReDim NewFea(1)
      Else
         ReDim NewFea(3)
      End If
   ElseIf Changeaxis = 1 Then
      If Form1.RadioButton23.Checked = True Then
         ReDim NewFea(1)
      ElseIf Form1.RadioButton24.Checked = True Then
         ReDim NewFea(1)
      Else
         ReDim NewFea(3)
      End If
   End If

   If Changeaxis <> 1 Then
      If Form1.RadioButton3.Checked = True Then
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        For icount = 0 To Max
            If ArrPos((Org(0) - 1) + VarArrpos(0)) = Po(icount) Then
                NewFea(0) = CStr(icount)
                Exit For
            End If
        Next
        For icount = 0 To Max
            If ArrPos(Org(0) + VarArrpos(0)) = Po(icount) Then
                NewFea(1) = CStr(icount)
                Exit For
            End If
        Next

        'Ch-----------------------------------------------
        If A = 1 And Form4.B = 0 Then
            ReDim Preserve inputx(CInt(inputx.Length))
        End If
        inputx(inputx.Length - 1).F = NewFea(0)
        inputx(inputx.Length - 1).T = NewFea(1)
        inputx(inputx.Length - 1).PosF = ArrPos((Org(0) - 1) + VarArrpos(0))
        inputx(inputx.Length - 1).PosT = ArrPos(Org(0) + VarArrpos(0))
        inputx(CInt(inputx.Length - 1)).Dims = Abs((cle(0) + cle(1)) / 2)
        inputx(CInt(inputx.Length - 1)).Dimtol = Abs((cle(0) - cle(1)) / 2)
        Path_Build()
        Path_Result = Path_Search(CInt(Req(0)), CInt(Req(1)))

    End If

    'งานเหลี่ยม ใช้ pin ทำรัง
    If Form1.RadioButton4.Checked = True Then
                For icount = 0 To Max
            If ArrPos((Org(0) - 1) + VarArrpos(0)) = Po(icount) Then
                NewFea(0) = CStr(icount)
                Exit For
            End If
        Next
          For icount = 0 To Max
            If ArrPos(Org(0) + VarArrpos(0)) = Po(icount) Then
                NewFea(1) = CStr(icount)
                Exit For
            End If
        Next

    'Ch-----------------------------------------------
        If A = 1 And Form4.B = 0 Then
            ReDim Preserve inputx(CInt(inputx.Length))
        End If
        inputx(inputx.Length - 1).F = NewFea(0)
        inputx(inputx.Length - 1).T = NewFea(1)
        inputx(inputx.Length - 1).PosF = ArrPos((Org(0) - 1) + VarArrpos(0))
        inputx(inputx.Length - 1).PosT = ArrPos(Org(0) + VarArrpos(0))
        inputx(CInt(inputx.Length - 1)).Dims = Abs((cle(0) + cle(1)) / 2)
        inputx(CInt(inputx.Length - 1)).Dimtol = Abs((cle(0) - cle(1)) / 2)
        Path_Build()
        Path_Result = Path_Search(CInt(Req(0)), CInt(Req(1)))
End If

    'กลม pin locator
    If Form1.RadioButton5.Checked = True Then
        For icount = 0 To Max
            If ArrPos((Org(0) - 1) + VarArrpos(0)) = Po(icount) Then
                NewFea(0) = CStr(icount)
                Exit For
            End If
        Next
        For icount = 0 To Max
            If ArrPos(Org(0) + VarArrpos(0)) = Po(icount) Then
                NewFea(1) = CStr(icount)
                Exit For
            End If
        Next
```

```
        For icount = 0 To Max
            If ArrPos((Org(1) - 1) + VarArrpos(0)) = Po(icount) Then
                NewFea(2) = CStr(icount)
                Exit For
            End If
        Next
        For icount = 0 To Max
            If ArrPos(Org(1) + VarArrpos(0)) = Po(icount) Then
                NewFea(3) = CStr(icount)
                Exit For
            End If
        Next

        'Ch-----------------------------------------------
        If A = 1 And Form4.B = 0 Then
            ReDim Preserve inputx(CInt(inputx.Length + 1))
        End If
        inputx(inputx.Length - 2).F = NewFea(2)
        inputx(inputx.Length - 2).T = NewFea(3)
        inputx(inputx.Length - 2).PosF = ArrPos((Org(1) - 1) + VarArrpos(0))
        inputx(inputx.Length - 2).PosT = ArrPos(Org(1) + VarArrpos(0))
        inputx(CInt(inputx.Length - 2)).Dims = 0
        inputx(CInt(inputx.Length - 2)).Dimtol = Abs((cle(0) - cle(1)) / 2)
        inputx(inputx.Length - 1).F = NewFea(0)
        inputx(inputx.Length - 1).T = NewFea(1)
        inputx(inputx.Length - 1).PosF = ArrPos((Org(0) - 1) + VarArrpos(0))
        inputx(inputx.Length - 1).PosT = ArrPos(Org(0) + VarArrpos(0))
        inputx(CInt(inputx.Length - 1)).Dims = Abs((cle(0) + cle(1)) / 2)
        inputx(CInt(inputx.Length - 1)).Dimtol = Abs((cle(0) - cle(1)) / 2)
        Path_Build()
        Path_Result = Path_Search(CInt(Req(0)), CInt(Req(1)))
    End If

    'กลม 3 locator
    If Form1.RadioButton6.Checked = True Then
        For icount = 0 To Max
            If ArrPos((Org(0) - 1) + VarArrpos(0)) = Po(icount) Then
                NewFea(0) = CStr(icount)
                Exit For
            End If
        Next
        For icount = 0 To Max
            If ArrPos(Org(0) + VarArrpos(0)) = Po(icount) Then
                NewFea(1) = CStr(icount)
                Exit For
            End If
        Next
        For icount = 0 To Max
            If ArrPos((Org(1) - 1) + VarArrpos(0)) = Po(icount) Then
                NewFea(2) = CStr(icount)
                Exit For
            End If
        Next
        For icount = 0 To Max
            If ArrPos(Org(1) + VarArrpos(0)) = Po(icount) Then
                NewFea(3) = CStr(icount)
                Exit For
            End If
        Next

        'Ch-----------------------------------------------
        If A = 1 And Form4.B = 0 Then
            ReDim Preserve inputx(CInt(inputx.Length + 1))
        End If
        inputx(inputx.Length - 2).F = NewFea(2)
        inputx(inputx.Length - 2).T = NewFea(3)
        inputx(inputx.Length - 2).PosF = ArrPos((Org(1) - 1) + VarArrpos(0))
        inputx(inputx.Length - 2).PosT = ArrPos(Org(1) + VarArrpos(0))
        inputx(CInt(inputx.Length - 2)).Dims = 0
        inputx(CInt(inputx.Length - 2)).Dimtol = Abs((cle(0) - cle(1)) / 2)
        inputx(inputx.Length - 1).F = NewFea(0)
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
            inputx(inputx.Length - 1).T = NewFea(1)
            inputx(inputx.Length - 1).PosF = ArrPos((Org(0) - 1) + VarArrpos(0))
            inputx(inputx.Length - 1).PosT = ArrPos(Org(0) + VarArrpos(0))
            inputx(CInt(inputx.Length - 1)).Dims = Abs((cle(0) + cle(1)) / 2)
            inputx(CInt(inputx.Length - 1)).Dimtol = Abs((cle(0) - cle(1)) / 2)
            Path_Build()
            Path_Result = Path_Search(CInt(Req(0)), CInt(Req(1)))
        End If

        If Form1.RadioButton7.Checked = True Then
            For icount = 0 To Max
                If ArrPos((Org(0) - 1)) = Po(icount) Then
                    NewFea(0) = CStr(icount)
                    Exit For
                End If
            Next
            For icount = 0 To Max
                If ArrPos(Org(0)) = Po(icount) Then
                    NewFea(1) = CStr(icount)
                    Exit For
                End If
            Next
            For icount = 0 To Max
                If ArrPos((Org(1) - 1) + VarArrpos(1)) = Po(icount) Then
                    NewFea(2) = CStr(icount)
                    Exit For
                End If
            Next
            For icount = 0 To Max
                If ArrPos(Org(1) + VarArrpos(1)) = Po(icount) Then
                    NewFea(3) = CStr(icount)
                    Exit For
                End If
            Next

            'Ch-----------------------------------------------
            If A = 1 And Form4.B = 0 Then
                ReDim Preserve inputx(CInt(inputx.Length + 1))
            End If
            inputx(inputx.Length - 2).F = NewFea(2)
            inputx(inputx.Length - 2).T = NewFea(3)
            inputx(inputx.Length - 2).PosF = ArrPos((Org(1) - 1) + VarArrpos(1))
            inputx(inputx.Length - 2).PosT = ArrPos(Org(1) + VarArrpos(1))
            inputx(CInt(inputx.Length - 2)).Dims = 0
            inputx(CInt(inputx.Length - 2)).Dimtol = Abs((cle(0) - cle(1)) / 2)
            inputx(inputx.Length - 1).F = NewFea(0)
            inputx(inputx.Length - 1).T = NewFea(1)
            inputx(inputx.Length - 1).PosF = ArrPos((Org(0) - 1))
            inputx(inputx.Length - 1).PosT = ArrPos(Org(0))
            inputx(CInt(inputx.Length - 1)).Dims = Abs((cle(0) + cle(1)) / 2)
            inputx(CInt(inputx.Length - 1)).Dimtol = Abs((cle(0) - cle(1)) / 2)
            Path_Build()
            Path_Result = Path_Search(CInt(Req(0)), CInt(Req(1)))
        End If

    ElseIf Changeaxis = 1 Then
        'งานเหลี่ยม แบบรัง
        If Form1.RadioButton23.Checked = True Then
            For icount = 0 To Max
                If ArrPos((Org(0) - 1) + VarArrpos(0)) = Po(icount) Then
                    NewFea(0) = CStr(icount)
                    Exit For
                End If
            Next
            For icount = 0 To Max
                If ArrPos(Org(0) + VarArrpos(0)) = Po(icount) Then
                    NewFea(1) = CStr(icount)
                    Exit For
                End If
            Next

            'Ch-----------------------------------------------
```

```
        If A = 1 And Form4.B = 0 Then
            ReDim Preserve inputx(CInt(inputx.Length))
        End If
        inputx(inputx.Length - 1).F = NewFea(0)
        inputx(inputx.Length - 1).T = NewFea(1)
        inputx(inputx.Length - 1).PosF = ArrPos((Org(0) - 1) + VarArrpos(0))
        inputx(inputx.Length - 1).PosT = ArrPos(Org(0) + VarArrpos(0))
        inputx(CInt(inputx.Length - 1)).Dims = Abs((cle(0) + cle(1)) / 2)
        inputx(CInt(inputx.Length - 1)).Dimtol = Abs((cle(0) - cle(1)) / 2)
        Path_Build()
        Path_Result = Path_Search(CInt(Req(0)), CInt(Req(1)))
    End If

    'งานเหลี่ยม ใช้ pin ทำรัง
    If Form1.RadioButton24.Checked = True Then
        For icount = 0 To Max
            If ArrPos((Org(0) - 1) + VarArrpos(0)) = Po(icount) Then
                NewFea(0) = CStr(icount)
                Exit For
            End If
        Next
        For icount = 0 To Max
            If ArrPos(Org(0) + VarArrpos(0)) = Po(icount) Then
                NewFea(1) = CStr(icount)
                Exit For
            End If
        Next

        'Ch-----------------------------------------------
        If A = 1 And Form4.B = 0 Then
            ReDim Preserve inputx(CInt(inputx.Length))
        End If
        inputx(inputx.Length - 1).F = NewFea(0)
        inputx(inputx.Length - 1).T = NewFea(1)
        inputx(inputx.Length - 1).PosF = ArrPos((Org(0) - 1) + VarArrpos(0))
        inputx(inputx.Length - 1).PosT = ArrPos(Org(0) + VarArrpos(0))
        inputx(CInt(inputx.Length - 1)).Dims = Abs((cle(0) + cle(1)) / 2)
        inputx(CInt(inputx.Length - 1)).Dimtol = Abs((cle(0) - cle(1)) / 2)
        Path_Build()
        Path_Result = Path_Search(CInt(Req(0)), CInt(Req(1)))
    End If

    'กลม pin locator
    If Form1.RadioButton25.Checked = True Then
                For icount = 0 To Max
            If ArrPos(Org(0) - 1) = Po(icount) Then
                NewFea(0) = CStr(icount)
                Exit For
            End If
        Next
        For icount = 0 To Max
            If ArrPos(Org(0)) = Po(icount) Then
                NewFea(1) = CStr(icount)
                Exit For
            End If
        Next
        For icount = 0 To Max
            If ArrPos(Org(1) - 2) = Po(icount) Then
                NewFea(2) = CStr(icount)
                Exit For
            End If
        Next
        For icount = 0 To Max
            If ArrPos(Org(1)) = Po(icount) Then
                NewFea(3) = CStr(icount)
                Exit For
              End If
        Next

        'Ch-----------------------------------------------
        If A = 1 And Form4.B = 0 Then
            ReDim Preserve inputx(CInt(inputx.Length + 1))
```

```
        End If

        inputx(inputx.Length - 2).F = NewFea(2)
        inputx(inputx.Length - 2).T = NewFea(3)
        inputx(inputx.Length - 2).PosF = ArrPos(Org(1) - 2)
        inputx(inputx.Length - 2).PosT = ArrPos(Org(1))
        inputx(CInt(inputx.Length - 2)).Dims = 0
        inputx(CInt(inputx.Length - 2)).Dimtol = Abs((cle(0) - cle(1)) / 2)
        inputx(inputx.Length - 1).F = NewFea(0)
        inputx(inputx.Length - 1).T = NewFea(1)
        inputx(inputx.Length - 1).PosF = ArrPos(Org(0) - 1)
        inputx(inputx.Length - 1).PosT = ArrPos(Org(0))
        inputx(CInt(inputx.Length - 1)).Dims = Abs((cle(0) + cle(1)) / 2)
        inputx(CInt(inputx.Length - 1)).Dimtol = Abs((cle(0) - cle(1)) / 2)
        Path_Build()
        Path_Result = Path_Search(CInt(Req(0)), CInt(Req(1)))
    End If

    If Form1.RadioButton26.Checked = True Then
        For icount = 0 To Max
            If ArrPos(Org(0) - 1) = Po(icount) Then
                NewFea(0) = CStr(icount)
                Exit For
            End If
        Next
        For icount = 0 To Max
            If ArrPos(Org(0)) = Po(icount) Then
                NewFea(1) = CStr(icount)
                Exit For
            End If
        Next
        For icount = 0 To Max
            If ArrPos(Org(1) - 2) = Po(icount) Then
                NewFea(2) = CStr(icount)
                Exit For
            End If
        Next
        For icount = 0 To Max
            If ArrPos(Org(1)) = Po(icount) Then
                NewFea(3) = CStr(icount)
                Exit For
            End If
        Next

        'Ch----------------------------------------------
        If A = 1 And Form4.B = 0 Then
            ReDim Preserve inputx(CInt(inputx.Length + 1))
        End If
        inputx(inputx.Length - 2).F = NewFea(2)
        inputx(inputx.Length - 2).T = NewFea(3)
        inputx(inputx.Length - 2).PosF = ArrPos(Org(1) - 2)
        inputx(inputx.Length - 2).PosT = ArrPos(Org(1))
        inputx(CInt(inputx.Length - 2)).Dims = 0
        inputx(CInt(inputx.Length - 2)).Dimtol = Abs((cle(0) - cle(1)) / 2)
        inputx(inputx.Length - 1).F = NewFea(0)
        inputx(inputx.Length - 1).T = NewFea(1)
        inputx(inputx.Length - 1).PosF = ArrPos(Org(0) - 1)
        inputx(inputx.Length - 1).PosT = ArrPos(Org(0))
        inputx(CInt(inputx.Length - 1)).Dims = Abs((cle(0) + cle(1)) / 2)
        inputx(CInt(inputx.Length - 1)).Dimtol = Abs((cle(0) - cle(1)) / 2)
        Path_Build()
        Path_Result = Path_Search(CInt(Req(0)), CInt(Req(1)))
    End If

     If Form1.RadioButton27.Checked = True Then
        For icount = 0 To Max
            If ArrPos(Org(0) - 1) = Po(icount) Then
                NewFea(0) = CStr(icount)
                Exit For
            End If
        Next
```

```
            For icount = 0 To Max
               If ArrPos(Org(0)) = Po(icount) Then
                  NewFea(1) = CStr(icount)
                  Exit For
               End If
            Next
            For icount = 0 To Max
               If ArrPos(Org(1) - 2) = Po(icount) Then
                  NewFea(2) = CStr(icount)
                  Exit For
               End If
            Next
            For icount = 0 To Max
               If ArrPos(Org(1)) = Po(icount) Then
                  NewFea(3) = CStr(icount)
                  Exit For
               End If
            Next

            'Ch-----------------------------------------------
            If A = 1 And Form4.B = 0 Then
               ReDim Preserve inputx(CInt(inputx.Length + 1))
            End If
            inputx(inputx.Length - 2).F = NewFea(2)
            inputx(inputx.Length - 2).T = NewFea(3)
            inputx(inputx.Length - 2).PosF = ArrPos(Org(1) - 2)
            inputx(inputx.Length - 2).PosT = ArrPos(Org(1))
            inputx(CInt(inputx.Length - 2)).Dims = 0
            inputx(CInt(inputx.Length - 2)).Dimtol = Abs((cle(0) - cle(1)) / 2)
            inputx(inputx.Length - 1).F = NewFea(0)
            inputx(inputx.Length - 1).T = NewFea(1)
            inputx(inputx.Length - 1).PosF = ArrPos(Org(0) - 1)
            inputx(inputx.Length - 1).PosT = ArrPos(Org(0))
            inputx(CInt(inputx.Length - 1)).Dims = Abs((cle(0) + cle(1)) / 2)
            inputx(CInt(inputx.Length - 1)).Dimtol = Abs((cle(0) - cle(1)) / 2)
            Path_Build()
            Path_Result = Path_Search(CInt(Req(0)), CInt(Req(1)))
         End If
      End If

   End Sub
```

Loop Diagram Generation Module

```
   Public Sub Path_Build()
      Dim icount As Integer
      Dim jcount As Integer
      Dim Base() As Integer
      Dim Asearch As String
      Dim Bsearch As String
      Dim Added As Integer = 0
      ReDim Path(Max - 1)
      ReDim Base(Max - 1)
       Dim Copoint() As Integer

      For icount = 0 To Path.Length - 1
         Path(icount) = ""
      Next
      If Changeaxis <> 1 Then
         If Form1.RadioButton3.Checked = True Then
            ReDim Copoint(0)
         ElseIf Form1.RadioButton4.Checked = True Then
            ReDim Copoint(0)
         Else
            ReDim Copoint(1)
         End If
      ElseIf Changeaxis = 1 Then
         If Form1.RadioButton23.Checked = True Then
            ReDim Copoint(0)
         ElseIf Form1.RadioButton24.Checked = True Then
            ReDim Copoint(0)
```

```
    Else
      ReDim Copoint(1)
    End If
  End If

  Select Case Shift_to
    Case "Left"
      If Changeaxis <> 1 Then
        If Form1.RadioButton3.Checked = True Then
          Copoint(0) = 1
        ElseIf Form1.RadioButton4.Checked = True Then
          Copoint(0) = 2
        ElseIf Form1.RadioButton5.Checked = True Then
          Select Case Side
            Case "รูเจาะอยู่ซ้าย"
              Copoint(0) = 0
              Copoint(1) = 5
            Case "รูเจาะอยู่ขวา"
              Copoint(0) = 4
              Copoint(1) = 5
            Case "2 รูเจาะ"
              Copoint(0) = 0
              Copoint(1) = 5
          End Select
        ElseIf Form1.RadioButton6.Checked = True Then
          Select Case Side
            Case "รูเจาะอยู่ซ้าย"
              Copoint(0) = 0
              Copoint(1) = 1
            Case "รูเจาะอยู่ขวา"
              Copoint(0) = 0
              Copoint(1) = 5
            Case "2 รูเจาะ"
              Copoint(0) = 0
              Copoint(1) = 1
          End Select

        ElseIf Form1.RadioButton7.Checked = True Then
          Select Case Side
            Case "รูเจาะอยู่ซ้าย"
              Copoint(0) = 0
              Copoint(1) = 1
            Case "รูเจาะอยู่ขวา"
              Copoint(0) = 0
              Copoint(1) = 3
            Case "2 รูเจาะ"
              Copoint(0) = 0
              Copoint(1) = 1
          End Select
        End If

      ElseIf Changeaxis = 1 Then
        If Form1.RadioButton23.Checked = True Then
          Copoint(0) = (1)
        ElseIf Form1.RadioButton24.Checked = True Then
          Copoint(0) = (2)
        ElseIf Form1.RadioButton25.Checked = True Then
          Copoint(0) = 3
          Copoint(1) = 5
        ElseIf Form1.RadioButton26.Checked = True Then
          Copoint(0) = 2
          Copoint(1) = 5
        ElseIf Form1.RadioButton27.Checked = True Then
          Copoint(0) = 0
          Copoint(1) = 2
        End If
      End If

    Case "Right"
      If Changeaxis <> 1 Then
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
            If Form1.RadioButton3.Checked = True Then
               Copoint(0) = (2)
            ElseIf Form1.RadioButton4.Checked = True Then
               Copoint(0) = (3)
            ElseIf Form1.RadioButton5.Checked = True Then
               Select Case Side
                  Case "รูเจาะอยู่ซ้าย"
                     Copoint(0) = 0
                     Copoint(1) = 1
                  Case "รูเจาะอยู่ขวา"
                     Copoint(0) = 0
                     Copoint(1) = 5
                  Case "2รูเจาะ"
                     Copoint(0) = 0
                     Copoint(1) = 1
               End Select
            ElseIf Form1.RadioButton6.Checked = True Then
               Select Case Side
                  Case "รูเจาะอยู่ซ้าย"
                     Copoint(0) = 0
                     Copoint(1) = 5
                  Case "รูเจาะอยู่ขวา"
                     Copoint(0) = 4
                     Copoint(1) = 5
                  Case "2รูเจาะ"
                     Copoint(0) = 0
                     Copoint(1) = 5
               End Select
            ElseIf Form1.RadioButton7.Checked = True Then
               Select Case Side
                  Case "รูเจาะอยู่ซ้าย"
                     Copoint(0) = 2
                     Copoint(1) = 5
                  Case "รูเจาะอยู่ขวา"
                     Copoint(0) = 4
                     Copoint(1) = 5
                  Case "2รูเจาะ"
                     Copoint(0) = 2
                     Copoint(1) = 6
               End Select
            End If

         ElseIf Changeaxis = 1 Then
            If Form1.RadioButton23.Checked = True Then
               Copoint(0) = 2
            ElseIf Form1.RadioButton24.Checked = True Then
               Copoint(0) = 3
            ElseIf Form1.RadioButton25.Checked = True Then
               Copoint(0) = 0
               Copoint(1) = 5
            ElseIf Form1.RadioButton26.Checked = True Then
               Copoint(0) = 3
               Copoint(1) = 5
            ElseIf Form1.RadioButton27.Checked = True Then
               Copoint(0) = 3
               Copoint(1) = 5
            End If
         End If
      End Select

      If Changeaxis <> 1 Then
         If Form1.RadioButton3.Checked = True Then
            For icount = 0 To inputx.Length - 1
               If inputx(icount).PosF = ArrPos(Copoint(0)) Then
                  Path(CInt(inputx(icount).F) - 1) = inputx(icount).F
                  Exit For
               ElseIf inputx(icount).PosT = ArrPos(Copoint(0)) Then
                  Path(CInt(inputx(icount).T) - 1) = inputx(icount).T
                  Exit For
               End If
```

```
    Next

  ElseIf Form1.RadioButton4.Checked = True Then
    For icount = 0 To inputx.Length - 1
      If inputx(icount).PosF = ArrPos(Copoint(0)) Then
        Path(CInt(inputx(icount).F) - 1) = inputx(icount).F
        Exit For
      ElseIf inputx(icount).PosT = ArrPos(Copoint(0)) Then
        Path(CInt(inputx(icount).T) - 1) = inputx(icount).T
        Exit For
      End If
    Next
  End If

  If Form1.RadioButton5.Checked = True Then
    For icount = 0 To inputx.Length - 1
      If inputx(icount).PosF = ArrPos(Copoint(0)) Then
        Path(CInt(inputx(icount).F) - 1) = inputx(icount).F
        Exit For
      ElseIf inputx(icount).PosT = ArrPos(Copoint(0)) Then
        Path(CInt(inputx(icount).T) - 1) = inputx(icount).T
        Exit For
      End If
    Next
    For icount = 0 To inputx.Length - 1
      If inputx(icount).PosF = ArrPos(Copoint(1)) Then
        Path(CInt(inputx(icount).F) - 1) = inputx(icount).F
        Exit For
      ElseIf inputx(icount).PosT = ArrPos(Copoint(1)) Then
        Path(CInt(inputx(icount).T) - 1) = inputx(icount).T
        Exit For
      End If
    Next
  End If

  If Form1.RadioButton6.Checked = True Then
    For icount = 0 To inputx.Length - 1
      If inputx(icount).PosF = ArrPos(Copoint(0)) Then
        Path(CInt(inputx(icount).F) - 1) = inputx(icount).F
        Exit For
      ElseIf inputx(icount).PosT = ArrPos(Copoint(0)) Then
        Path(CInt(inputx(icount).T) - 1) = inputx(icount).T
        Exit For
      End If
    Next
    For icount = 0 To inputx.Length - 1
      If inputx(icount).PosF = ArrPos(Copoint(1)) Then
        Path(CInt(inputx(icount).F) - 1) = inputx(icount).F
        Exit For
      ElseIf inputx(icount).PosT = ArrPos(Copoint(1)) Then
        Path(CInt(inputx(icount).T) - 1) = inputx(icount).T
        Exit For
      End If
    Next
  End If

  If Form1.RadioButton7.Checked = True Then
    For icount = 0 To inputx.Length - 1
      If inputx(icount).PosF = ArrPos(Copoint(0)) Then
        Path(CInt(inputx(icount).F) - 1) = inputx(icount).F
        Exit For
      ElseIf inputx(icount).PosT = ArrPos(Copoint(0)) Then
        Path(CInt(inputx(icount).T) - 1) = inputx(icount).T
        Exit For
      End If
    Next
    For icount = 0 To inputx.Length - 1
      If inputx(icount).PosF = ArrPos(Copoint(1)) Then
        Path(CInt(inputx(icount).F) - 1) = inputx(icount).F
        Exit For
      ElseIf inputx(icount).PosT = ArrPos(Copoint(1)) Then
```

```
                Path(CInt(inputx(icount).T) - 1) = inputx(icount).T
                Exit For
            End If
        Next
    End If

ElseIf Changeaxis = 1 Then
    If Form1.RadioButton23.Checked = True Then
        For icount = 0 To inputx.Length - 1
            If inputx(icount).PosF = ArrPos(Copoint(0)) Then
                Path(CInt(inputx(icount).F) - 1) = inputx(icount).F
                Exit For
            ElseIf inputx(icount).PosT = ArrPos(Copoint(0)) Then
                Path(CInt(inputx(icount).T) - 1) = inputx(icount).T
                Exit For
            End If
         Next

    ElseIf Form1.RadioButton24.Checked = True Then
        For icount = 0 To inputx.Length - 1
            If inputx(icount).PosF = ArrPos(Copoint(0)) Then
                Path(CInt(inputx(icount).F) - 1) = inputx(icount).F
                Exit For
            ElseIf inputx(icount).PosT = ArrPos(Copoint(0)) Then
                Path(CInt(inputx(icount).T) - 1) = inputx(icount).T
                Exit For
            End If
        Next
    End If

     If Form1.RadioButton25.Checked = True Then
        For icount = 0 To inputx.Length - 1
            If inputx(icount).PosF = ArrPos(Copoint(0)) Then
                Path(CInt(inputx(icount).F) - 1) = inputx(icount).F
                Exit For
            ElseIf inputx(icount).PosT = ArrPos(Copoint(0)) Then
                Path(CInt(inputx(icount).T) - 1) = inputx(icount).T
                Exit For
            End If
        Next
        For icount = 0 To inputx.Length - 1
            If inputx(icount).PosF = ArrPos(Copoint(1)) Then
                Path(CInt(inputx(icount).F) - 1) = inputx(icount).F
                Exit For
            ElseIf inputx(icount).PosT = ArrPos(Copoint(1)) Then
                Path(CInt(inputx(icount).T) - 1) = inputx(icount).T
                Exit For
            End If
        Next
    End If

    If Form1.RadioButton26.Checked = True Then
        For icount = 0 To inputx.Length - 1
            If inputx(icount).PosF = ArrPos(Copoint(0)) Then
                Path(CInt(inputx(icount).F) - 1) = inputx(icount).F
                Exit For
            ElseIf inputx(icount).PosT = ArrPos(Copoint(0)) Then
                Path(CInt(inputx(icount).T) - 1) = inputx(icount).T
                Exit For
            End If
        Next
        For icount = 0 To inputx.Length - 1
            If inputx(icount).PosF = ArrPos(Copoint(1)) Then
                Path(CInt(inputx(icount).F) - 1) = inputx(icount).F
                Exit For
            ElseIf inputx(icount).PosT = ArrPos(Copoint(1)) Then
                Path(CInt(inputx(icount).T) - 1) = inputx(icount).T
                Exit For
            End If
        Next
    End If
```

```
    If Form1.RadioButton27.Checked = True Then
      For icount = 0 To inputx.Length - 1
        If inputx(icount).PosF = ArrPos(Copoint(0)) Then
          Path(CInt(inputx(icount).F) - 1) = inputx(icount).F
          Exit For
        ElseIf inputx(icount).PosT = ArrPos(Copoint(0)) Then
          Path(CInt(inputx(icount).T) - 1) = inputx(icount).T
          Exit For
        End If
      Next
      For icount = 0 To inputx.Length - 1
        If inputx(icount).PosF = ArrPos(Copoint(1)) Then
          Path(CInt(inputx(icount).F) - 1) = inputx(icount).F
          Exit For
        ElseIf inputx(icount).PosT = ArrPos(Copoint(1)) Then
          Path(CInt(inputx(icount).T) - 1) = inputx(icount).T
          Exit For
        End If
      Next
    End If
  End If

  Dim Exitcon As Integer

  If Changeaxis <> 1 Then
    If Form1.RadioButton3.Checked = True Then
      Exitcon = Max - 2
    ElseIf Form1.RadioButton4.Checked = True Then
      Exitcon = Max - 2
    Else
      Exitcon = Max - 3
    End If

  ElseIf Changeaxis = 1 Then
    If Form1.RadioButton23.Checked = True Then
      Exitcon = Max - 2
    ElseIf Form1.RadioButton24.Checked = True Then
      Exitcon = Max - 2
    Else
      Exitcon = Max - 3
    End If
  End If

  Do While Added <= Exitcon
    For icount = 0 To inputx.Length - 1
      If inputx(icount).F = Req(0) And inputx(icount).T = Req(1) Then
      ElseIf inputx(icount).T = Req(0) And inputx(icount).F = Req(1) Then
      Else
        Asearch = inputx(icount).F
        Bsearch = inputx(icount).T
        For jcount = 0 To Max - 1
          If Path(CInt(Asearch) - 1) = "" Then
            If Path(CInt(Bsearch) - 1) <> "" Then
              Path(CInt(Asearch) - 1) = Asearch & "" & Path(CInt(Bsearch) - 1)
              Added = Added + 1
              If Changeaxis = 1 Then
              End If
            End If
          End If
        Next
      End If
    Next
    For icount = 0 To inputx.Length - 1
      If inputx(icount).F = Req(0) And inputx(icount).T = Req(1) Then
      ElseIf inputx(icount).T = Req(0) And inputx(icount).F = Req(1) Then
      Else
        Bsearch = inputx(icount).F
        Asearch = inputx(icount).T
        For jcount = 0 To Max - 1
          If Path(CInt(Asearch) - 1) = "" Then
            If Path(CInt(Bsearch) - 1) <> "" Then
              Path(CInt(Asearch) - 1) = Asearch & "" & Path(CInt(Bsearch) - 1)
```

```
                    Added = Added + 1
                End If
            End If
        Next
    End If
  Next
Loop

 Added = 0
For icount = 0 To Path.Length - 1
  Path(icount) = Path(icount).Trim
Next
Form4.Show()
End Sub

Public Function Path_Search(ByVal A As Integer, ByVal B As Integer) As String
  Dim PathA As String = " "
  Dim PathB As String = " "
  Dim PathD As String = " "
  Dim icount, jcount, kcount, mcount As Integer

  For icount = 0 To Path.Length - 1
    If A = CStr(Path(icount)(0)) Then
      PathA = Path(icount)
    End If
    If B = CStr(Path(icount)(0)) Then
      PathB = Path(icount)
    End If
  Next
  PathA = PathA.Trim
  PathB = PathB.Trim
  icount = PathA.Length - 1
  jcount = PathB.Length - 1

  Do Until icount < 0 Or jcount < 0
    If PathA(icount) = PathB(jcount) Then
      If icount = 0 Or jcount = 0 Then
        If icount = 0 Then
          For kcount = 0 To icount
            PathD = PathD & PathA(kcount)
          Next
          For mcount = jcount - 1 To 0 Step -1
            PathD = PathD & PathB(mcount)
          Next
        End If
        If jcount = 0 Then
          For mcount = 0 To jcount
            PathD = PathD & PathB(mcount)
          Next
          For kcount = icount - 1 To 0 Step -1
            PathD = PathD & PathA(kcount)
          Next
        End If
      Else
        If PathA(icount - 1) = PathB(jcount - 1) Then
        Else
          For kcount = 0 To icount
            PathD = PathD & "" & PathA(kcount)
          Next
          For mcount = jcount - 1 To 0 Step -1
            PathD = PathD & "" & PathB(mcount)
          Next
        End If
      End If
    End If
    icount = icount - 1
    jcount = jcount - 1
  Loop

  PathD = PathD.Trim
  Return PathD
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
    End Function
```

Module for Tolerance Allocation and Basic Size Calculations

```
    Private Sub CalculateToolStripMenuItem_Click(sender As Object, e As EventArgs) Handles
    CalculateToolStripMenuItem.Click
NextCal:
        Calculate()
    End Sub

Public Sub Calsize_path(ByVal Path As String)
        Dim icount, jcount, kcount As Integer
        Dim Max As Integer
        Dim Checksign As Integer = 0
        Dim SignTol As Integer = -1
        Dim Dimset() As String
        Dim NumFreqInLoop As Integer = 0

        Basic_result = 0
        Tol_result = 0
        Max = FindMaxFea()

        'กำหนดเครื่องหมายของ basic size
        For icount = 0 To inputx.Length - 1
            If inputx(icount).PosF < inputx(icount).PosT Then
                inputx(icount).Signcal = 1
            ElseIf inputx(icount).PosT < inputx(icount).PosF Then
                inputx(icount).Signcal = -1
            Else
            End If
        Next
        For icount = 0 To inputx.Length - 1
            inputx(icount).Dims = inputx(icount).Dims * inputx(icount).Signcal
        Next

        'check กรณี Req เป็นตำแหน่งของ Freq
        For icount = 0 To Path.Length - 2
            For jcount = 0 To FReq.Length - 1
                If Path(icount) = FReq(jcount)(0) And Path(icount + 1) = FReq(jcount)(1) Then
                    NumFreqInLoop = NumFreqInLoop + 1
                ElseIf Path(icount) = FReq(jcount)(1) And Path(icount + 1) = FReq(jcount)(0) Then
                    NumFreqInLoop = NumFreqInLoop + 1
                End If
            Next
        Next
        jcount = 0
        For icount = 0 To Path.Length - 2
            ReDim Preserve Dimset(jcount)
            Dimset(jcount) = Path(icount) & Path(icount + 1)
            jcount = jcount + 1
        Next

        'check เครื่องหมายของ Tol
        kcount = 0
        ReDim Preserve Dimset(jcount - 1)
        Do While kcount < Dimset.Length
            For icount = 0 To inputx.Length - 1
                For jcount = 0 To NumFReq - 1
                    If Dimset(kcount)(0) = FReq(jcount)(0) And Dimset(kcount)(1) = FReq(jcount)(1) Then
                        Checksign = Checksign + 1
                    ElseIf Dimset(kcount)(0) = FReq(jcount)(1) And Dimset(kcount)(1) = FReq(jcount)(0) Then
                        Checksign = Checksign + 1
                    End If
                Next
            Next
            kcount = kcount + 1
        Loop

        'กำหนดเครื่องหมายของ Tol
        If Checksign = 0 Then
            inputx(inputx.Length - 1).Dimtol = (-1) * inputx(inputx.Length - 1).Dimtol
        ElseIf Checksign <> 0 Then
```

```
        For icount = 0 To inputx.Length - 1
            For jcount = 0 To NumFReq - 1
                If inputx(icount).F = FReq(jcount)(0) And inputx(icount).T = FReq(jcount)(1) Then
                    inputx(icount).Dimtol = (-1) * inputx(icount).Dimtol
                ElseIf inputx(icount).F = FReq(jcount)(1) And inputx(icount).T = FReq(jcount)(0) Then
                    inputx(icount).Dimtol = (-1) * inputx(icount).Dimtol
                End If
            Next
        Next
    End If
    For icount = 0 To Dimset.Length - 1
        For jcount = 0 To inputx.Length - 1
            If Dimset(icount)(0) = inputx(jcount).F And Dimset(icount)(1) = inputx(jcount).T Then
                Basic_result = Basic_result + inputx(jcount).Dims
                Exit For
            ElseIf Dimset(icount)(0) = inputx(jcount).T And Dimset(icount)(1) = inputx(jcount).F Then
                Basic_result = Basic_result + (inputx(jcount).Dims * (-1))
                Exit For
            End If
        Next
    Next
    If NumFreqInLoop = Dimset.Length Then
        For icount = 0 To Dimset.Length - 1
            For jcount = 0 To inputx.Length - 1
                If Dimset(icount)(0) = inputx(jcount).F And Dimset(icount)(1) = inputx(jcount).T Then
                    Tol_result = Tol_result + Abs(inputx(jcount).Dimtol)
                    Exit For
                ElseIf Dimset(icount)(0) = inputx(jcount).T And Dimset(icount)(1) = inputx(jcount).F Then
                    Tol_result = Tol_result + Abs(inputx(jcount).Dimtol)
                    Exit For
                End If
            Next
        Next
        Tol_result = (Abs(Tol_result) / 2) * (-1)
    ElseIf NumFreqInLoop <> Dimset.Length Then
        For icount = 0 To Dimset.Length - 1
            For jcount = 0 To inputx.Length - 1
                If Dimset(icount)(0) = inputx(jcount).F And Dimset(icount)(1) = inputx(jcount).T Then
                    Tol_result = Tol_result + inputx(jcount).Dimtol
                    Exit For
                ElseIf Dimset(icount)(0) = inputx(jcount).T And Dimset(icount)(1) = inputx(jcount).F Then
                    Tol_result = Tol_result + inputx(jcount).Dimtol
                    Exit For
                End If
            Next
        Next
    End If

End Sub

Public Sub Calculate()
    Dim icount As Integer
    Dim A As Integer = 0

    Contact_selection()
    'กำหนด B(ตำแหน่งเบี่ยงเบนเมื่อจำนวนรูเจาะเพิ่มขึ้น)
    NumHoleCheck()
    If NumFReq Mod 2 = 0 Then
        If Changeaxis <> 1 Then
            Select Shift_to
                Case "Left"
                    If Form1.RadioButton3.Checked = True Then
                        ReDim VarArrpos(1)
                        VarArrpos(0) = 0
                        VarArrpos(1) = NumFReq
                    ElseIf Form1.RadioButton4.Checked = True Then
                        ReDim VarArrpos(1)
                        VarArrpos(0) = 0
                        VarArrpos(1) = NumFReq
                    ElseIf Form1.RadioButton5.Checked = True Then
                        ReDim VarArrpos(0)
                        VarArrpos(0) = NumFReq - 1
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
            ElseIf Form1.RadioButton6.Checked = True Then
               ReDim VarArrpos(0)
               VarArrpos(0) = NumFReq - 1
            ElseIf Form1.RadioButton7.Checked = True Then
               ReDim VarArrpos(1)
               VarArrpos(0) = NumFReq
               VarArrpos(1) = NumFReq - 1
            End If

         Case "Right"
            If Form1.RadioButton3.Checked = True Then
               ReDim VarArrpos(1)
               VarArrpos(0) = 0
               VarArrpos(1) = NumFReq
            ElseIf Form1.RadioButton4.Checked = True Then
               ReDim VarArrpos(1)
               VarArrpos(0) = 0
               VarArrpos(1) = NumFReq
            ElseIf Form1.RadioButton5.Checked = True Then
               ReDim VarArrpos(0)
               VarArrpos(0) = NumFReq - 1
            ElseIf Form1.RadioButton6.Checked = True Then
               ReDim VarArrpos(0)
               VarArrpos(0) = NumFReq - 1
            ElseIf Form1.RadioButton7.Checked = True Then
               ReDim VarArrpos(1)
               VarArrpos(0) = NumFReq
               VarArrpos(1) = NumFReq - 1
              End If
      End Select
   End If

   If Changeaxis = 1 Then
      Select Case Shift_to
         Case "Left"
            If Form1.RadioButton23.Checked = True Then
               ReDim VarArrpos(1)
               VarArrpos(0) = 0
               VarArrpos(1) = NumFReq
            ElseIf Form1.RadioButton24.Checked = True Then
               ReDim VarArrpos(1)
               VarArrpos(0) = 0
               VarArrpos(1) = NumFReq
            ElseIf Form1.RadioButton25.Checked = True Then
               ReDim VarArrpos(0)
               VarArrpos(0) = 0
            ElseIf Form1.RadioButton26.Checked = True Then
               ReDim VarArrpos(0)
               VarArrpos(0) = 0
            ElseIf Form1.RadioButton27.Checked = True Then
               ReDim VarArrpos(0)
               VarArrpos(0) = 0
            End If
         Case "Right"
            If Form1.RadioButton23.Checked = True Then
               ReDim VarArrpos(1)
               VarArrpos(0) = 1
               VarArrpos(1) = NumFReq
            ElseIf Form1.RadioButton24.Checked = True Then
               ReDim VarArrpos(1)
               VarArrpos(0) = 1
               VarArrpos(1) = NumFReq
            ElseIf Form1.RadioButton25.Checked = True Then
               ReDim VarArrpos(0)
               VarArrpos(0) = 0
            ElseIf Form1.RadioButton26.Checked = True Then
               ReDim VarArrpos(0)
               VarArrpos(0) = 0
            ElseIf Form1.RadioButton27.Checked = True Then
               ReDim VarArrpos(0)
               VarArrpos(0) = 0
            End If
```

```
        End Select
    End If

    'จำนวนรูเจาะเป็นคี่
ElseIf NumFReq Mod 2 <> 0 Then
    If Changeaxis <> 1 Then
        Select Case Shift_to
            Case "Left"
                If Form1.RadioButton3.Checked = True Then
                    ReDim VarArrpos(1)
                    VarArrpos(0) = 0
                        VarArrpos(1) = NumFReq
                ElseIf Form1.RadioButton4.Checked = True Then
                    ReDim VarArrpos(1)
                    VarArrpos(0) = 0
                    VarArrpos(1) = NumFReq
                End If

                Select Case Side
                Case "รูเจาะอยู่ซ้าย"
                    If Form1.RadioButton5.Checked = True Then
                        ReDim VarArrpos(0)
                        VarArrpos(0) = NumFReq
                    ElseIf Form1.RadioButton6.Checked = True Then
                        ReDim VarArrpos(0)
                        VarArrpos(0) = NumFReq
                    ElseIf Form1.RadioButton7.Checked = True Then
                        ReDim VarArrpos(1)
                        VarArrpos(0) = NumFReq
                        VarArrpos(1) = NumFReq
                    End If
                Case "รูเจาะอยู่ขวา"
                    If Form1.RadioButton5.Checked = True Then
                        ReDim VarArrpos(0)
                        VarArrpos(0) = 0
                    ElseIf Form1.RadioButton6.Checked = True Then
                        ReDim VarArrpos(0)
                        VarArrpos(0) = 0
                    ElseIf Form1.RadioButton7.Checked = True Then
                        ReDim VarArrpos(1)
                        VarArrpos(0) = NumFReq
                        VarArrpos(1) = 0
                    End If
                End Select

            Case "Right"
                If Form1.RadioButton3.Checked = True Then
                    ReDim VarArrpos(1)
                    VarArrpos(0) = 0
                    VarArrpos(1) = NumFReq
                ElseIf Form1.RadioButton4.Checked = True Then
                    ReDim VarArrpos(1)
                    VarArrpos(0) = 0
                    VarArrpos(1) = NumFReq
                End If
                Select Case Side
                    Case "รูเจาะอยู่ซ้าย"
                        If Form1.RadioButton5.Checked = True Then
                            ReDim VarArrpos(0)
                            VarArrpos(0) = NumFReq
                        ElseIf Form1.RadioButton6.Checked = True Then
                            ReDim VarArrpos(0)
                            VarArrpos(0) = NumFReq
                        ElseIf Form1.RadioButton7.Checked = True Then
                            ReDim VarArrpos(1)
                            VarArrpos(0) = NumFReq
                            VarArrpos(1) = NumFReq
                        End If
                    Case "รูเจาะอยู่ขวา"
                        If Form1.RadioButton5.Checked = True Then
                            ReDim VarArrpos(0)
                            VarArrpos(0) = 0
```

```
                    ElseIf Form1.RadioButton6.Checked = True Then
                        ReDim VarArrpos(0)
                        VarArrpos(0) = 0
                    ElseIf Form1.RadioButton7.Checked = True Then
                        ReDim VarArrpos(1)
                        VarArrpos(0) = NumFReq
                        VarArrpos(1) = 0
                    End If
                End Select
            End Select

        ElseIf Changeaxis = 1 Then
            Select Case Shift_to
                Case "Left"
                    If Form1.RadioButton23.Checked = True Then
                        ReDim VarArrpos(1)
                        VarArrpos(0) = 0
                        VarArrpos(1) = NumFReq
                    ElseIf Form1.RadioButton24.Checked = True Then
                        ReDim VarArrpos(1)
                        VarArrpos(0) = 0
                        VarArrpos(1) = NumFReq
                    ElseIf Form1.RadioButton25.Checked = True Then
                        ReDim VarArrpos(0)
                        VarArrpos(0) = 0
                    ElseIf Form1.RadioButton26.Checked = True Then
                        ReDim VarArrpos(0)
                        VarArrpos(0) = 0
                    ElseIf Form1.RadioButton27.Checked = True Then
                        ReDim VarArrpos(0)
                        VarArrpos(0) = 0
                    End If
                Case "Right"
                    If Form1.RadioButton23.Checked = True Then
                        ReDim VarArrpos(1)
                        VarArrpos(0) = 0
                        VarArrpos(1) = NumFReq
                    ElseIf Form1.RadioButton24.Checked = True Then
                        ReDim VarArrpos(1)
                        VarArrpos(0) = 0
                        VarArrpos(1) = NumFReq
                    ElseIf Form1.RadioButton25.Checked = True Then
                        ReDim VarArrpos(0)
                        VarArrpos(0) = 0
                    ElseIf Form1.RadioButton26.Checked = True Then
                        ReDim VarArrpos(0)
                        VarArrpos(0) = 0
                    ElseIf Form1.RadioButton27.Checked = True Then
                        ReDim VarArrpos(0)
                        VarArrpos(0) = 0
                    End If
            End Select
        End If
    End If

    If Changeaxis <> 1 Then
        Select Case Shift_to
            Case "Left"
                If Form1.RadioButton3.Checked = True Then
                    Set_Po()
                    ReDim ArrPos(Max - 1)
                    ArrPos = sortstr()
                    Set_Po()
                    Add_clearance(1, Clearance_cal(1, ArrPos(0), ArrPos(1 + VarArrpos(1)), ArrPos(0), ArrPos(1 +
VarArrpos(1))))
                    A = A + 1
                    Calsize_path(Path_Result)
                    PrintOutput_path(A)
                    Print_Path()
                    Rec_Data()
                    Add_clearance(2, Clearance_cal(2, ArrPos(0), ArrPos(1 + VarArrpos(1)), ArrPos(0), ArrPos(1 +
VarArrpos(1))))
```

```
            A = A + 1
            Calsize_path(Path_Result)
            PrintOutput_path(A)
            Print_Path()
            Rec_Data()
            Add_clearance(3, Clearance_cal(3, ArrPos(0), ArrPos(1 + VarArrpos(1)), ArrPos(0), ArrPos(1 +
VarArrpos(1))))
            A = A + 1
            Calsize_path(Path_Result)
            PrintOutput_path(A)
            Print_Path()
        End If

        If Form1.RadioButton4.Checked = True Then
            Set_Po()
            ReDim ArrPos(Max - 1)
            ArrPos = sortstr()
            Set_Po()
            Add_clearance(1, Clearance_cal(1, ArrPos(1), ArrPos(2 + VarArrpos(1)), ArrPos(0), ArrPos(1)))
            A = A + 1
            Calsize_path(Path_Result)
            PrintOutput_path(A)
            Print_Path()
            Rec_Data()
            Add_clearance(2, Clearance_cal(2, ArrPos(1), ArrPos(2 + VarArrpos(1)), ArrPos(0), ArrPos(1)))
            A = A + 1
            Calsize_path(Path_Result)
            PrintOutput_path(A)
            Print_Path()
            Rec_Data()
            Add_clearance(3, Clearance_cal(3, ArrPos(1), ArrPos(2 + VarArrpos(1)), ArrPos(0), ArrPos(1)))
            A = A + 1
            Calsize_path(Path_Result)
            PrintOutput_path(A)
            Print_Path()
        End If

        If Form1.RadioButton5.Checked = True Then
            Set_Po()
            ReDim ArrPos(Max - 1)
            ArrPos = sortstr()
            Set_Po()
            Add_clearance(1, Clearance_cal(1, ArrPos(0 + VarArrpos(0)), ArrPos(4 + VarArrpos(0)), ArrPos(0 +
VarArrpos(0)), ArrPos(4 + VarArrpos(0))))
            A = A + 1
            Calsize_path(Path_Result)
            PrintOutput_path(A)
            Print_Path()
            Rec_Data()
            Add_clearance(2, Clearance_cal(2, ArrPos(0 + VarArrpos(0)), ArrPos(4 + VarArrpos(0)), ArrPos(0 +
VarArrpos(0)), ArrPos(4 + VarArrpos(0))))
            A = A + 1
            Calsize_path(Path_Result)
            PrintOutput_path(A)
            Print_Path()
            Rec_Data()
            Add_clearance(3, Clearance_cal(3, ArrPos(0 + VarArrpos(0)), ArrPos(4 + VarArrpos(0)), ArrPos(0 +
VarArrpos(0)), ArrPos(4 + VarArrpos(0))))
            A = A + 1
            Calsize_path(Path_Result)
            PrintOutput_path(A)
            Print_Path()
        End If

        If Form1.RadioButton6.Checked = True Then
            Set_Po()
            ReDim ArrPos(Max - 1)
            ArrPos = sortstr()
            Set_Po()
            Add_clearance(1, Clearance_cal(1, ArrPos(0 + VarArrpos(0)), ArrPos(3 + VarArrpos(0)), ArrPos(0 +
VarArrpos(0)), ArrPos(3 + VarArrpos(0))))
            A = A + 1
            Calsize_path(Path_Result)
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
                PrintOutput_path(A)
                Print_Path()
                Rec_Data()
                Add_clearance(2, Clearance_cal(2, ArrPos(0 + VarArrpos(0)), ArrPos(3 + VarArrpos(0)), ArrPos(0 +
VarArrpos(0)), ArrPos(3 + VarArrpos(0))))
                A = A + 1
                Calsize_path(Path_Result)
                PrintOutput_path(A)
                Print_Path()
                Rec_Data()
                Add_clearance(3, Clearance_cal(3, ArrPos(0 + VarArrpos(0)), ArrPos(3 + VarArrpos(0)), ArrPos(0 +
VarArrpos(0)), ArrPos(3 + VarArrpos(0))))
                A = A + 1
                Calsize_path(Path_Result)
                PrintOutput_path(A)
                Print_Path()
            End If

            If Form1.RadioButton7.Checked = True Then
                Set_Po()
                ReDim ArrPos(Max - 1)
                ArrPos = sortstr()
                Set_Po()
                Add_clearance(1, Clearance_cal(1, ArrPos(0), ArrPos(3 + VarArrpos(0)), ArrPos(0), ArrPos(3 +
VarArrpos(0))))
                A = A + 1
                Calsize_path(Path_Result)
                PrintOutput_path(A)
                Print_Path()
                Rec_Data()
                Add_clearance(2, Clearance_cal(2, ArrPos(0), ArrPos(3 + VarArrpos(0)), ArrPos(0), ArrPos(3 +
VarArrpos(0))))
                A = A + 1
                Calsize_path(Path_Result)
                PrintOutput_path(A)
                Print_Path()
                Rec_Data()
                Add_clearance(3, Clearance_cal(3, ArrPos(0), ArrPos(3 + VarArrpos(0)), ArrPos(0), ArrPos(3 +
VarArrpos(0))))
                A = A + 1
                Calsize_path(Path_Result)
                PrintOutput_path(A)
                Print_Path()
            End If

        Case "Right"
            If Form1.RadioButton3.Checked = True Then
                Set_Po()
                ReDim ArrPos(Max - 1)
                ArrPos = sortstr()
                Set_Po()
                Add_clearance(1, Clearance_cal(1, ArrPos(1), ArrPos(2 + VarArrpos(1)), ArrPos(1), ArrPos(2 +
VarArrpos(1))))
                A = A + 1
                Calsize_path(Path_Result)
                PrintOutput_path(A)
                  Print_Path()
                Rec_Data()
                Add_clearance(2, Clearance_cal(2, ArrPos(1), ArrPos(2 + VarArrpos(1)), ArrPos(1), ArrPos(2 +
VarArrpos(1))))
                A = A + 1
                Calsize_path(Path_Result)
                PrintOutput_path(A)
                Print_Path()
                Rec_Data()
                Add_clearance(3, Clearance_cal(3, ArrPos(1), ArrPos(2 + VarArrpos(1)), ArrPos(1), ArrPos(2 +
VarArrpos(1))))
                A = A + 1
                Calsize_path(Path_Result)
                PrintOutput_path(A)
                Print_Path()
            End If
```

```
        If Form1.RadioButton4.Checked = True Then
            Set_Po()
            ReDim ArrPos(Max - 1)
            ArrPos = sortstr()
            Set_Po()
            Add_clearance(1, Clearance_cal(1, ArrPos(2), ArrPos(3 + VarArrpos(1)), ArrPos(0), ArrPos(1)))
            A = A + 1
            Calsize_path(Path_Result)
            PrintOutput_path(A)
            Print_Path()
            Rec_Data()
            Add_clearance(2, Clearance_cal(2, ArrPos(2), ArrPos(3 + VarArrpos(1)), ArrPos(0), ArrPos(1)))
            A = A + 1
            Calsize_path(Path_Result)
            PrintOutput_path(A)
            Print_Path()
            Rec_Data()
            Add_clearance(3, Clearance_cal(3, ArrPos(2), ArrPos(3 + VarArrpos(1)), ArrPos(0), ArrPos(1)))
            A = A + 1
            Calsize_path(Path_Result)
            PrintOutput_path(A)
            Print_Path()
        End If

        If Form1.RadioButton5.Checked = True Then
            Set_Po()
            ReDim ArrPos(Max - 1)
            ArrPos = sortstr()
            Set_Po()
            Add_clearance(1, Clearance_cal(1, ArrPos(0 + VarArrpos(0)), ArrPos(4 + VarArrpos(0)), ArrPos(0 +
VarArrpos(0)), ArrPos(4 + VarArrpos(0))))
            A = A + 1
            Calsize_path(Path_Result)
            PrintOutput_path(A)
            Print_Path()
            Rec_Data()
            Add_clearance(2, Clearance_cal(2, ArrPos(0 + VarArrpos(0)), ArrPos(4 + VarArrpos(0)), ArrPos(0 +
VarArrpos(0)), ArrPos(4 + VarArrpos(0))))
            A = A + 1
            Calsize_path(Path_Result)
            PrintOutput_path(A)
            Print_Path()
            Rec_Data()
            Add_clearance(3, Clearance_cal(3, ArrPos(0 + VarArrpos(0)), ArrPos(4 + VarArrpos(0)), ArrPos(0 +
VarArrpos(0)), ArrPos(4 + VarArrpos(0))))
            A = A + 1
            Calsize_path(Path_Result)
            PrintOutput_path(A)
            Print_Path()
        End If

        If Form1.RadioButton6.Checked = True Then
            Set_Po()
            ReDim ArrPos(Max - 1)
            ArrPos = sortstr()
            Set_Po()
            Add_clearance(1, Clearance_cal(1, ArrPos(0 + VarArrpos(0)), ArrPos(4 + VarArrpos(0)), ArrPos(0 +
VarArrpos(0)), ArrPos(4 + VarArrpos(0))))
            A = A + 1
            Calsize_path(Path_Result)
            PrintOutput_path(A)
            Print_Path()
            Rec_Data()
            Add_clearance(2, Clearance_cal(2, ArrPos(0 + VarArrpos(0)), ArrPos(4 + VarArrpos(0)), ArrPos(0 +
VarArrpos(0)), ArrPos(4 + VarArrpos(0))))
            A = A + 1
            Calsize_path(Path_Result)
            PrintOutput_path(A)
            Print_Path()
            Rec_Data()
```

```
            Add_clearance(3, Clearance_cal(3, ArrPos(0 + VarArrpos(0)), ArrPos(4 + VarArrpos(0)), ArrPos(0 +
VarArrpos(0)), ArrPos(4 + VarArrpos(0))))
            A = A + 1
            Calsize_path(Path_Result)
            PrintOutput_path(A)
            Print_Path()
         End If

         If Form1.RadioButton7.Checked = True Then
            Set_Po()
            ReDim ArrPos(Max - 1)
            ArrPos = sortstr()
            Set_Po()
            Add_clearance(1, Clearance_cal(1, ArrPos(1), ArrPos(4 + VarArrpos(0)), ArrPos(1), ArrPos(4 +
VarArrpos(0))))
            A = A + 1
            Calsize_path(Path_Result)
            PrintOutput_path(A)
            Print_Path()
            Rec_Data()
            Add_clearance(2, Clearance_cal(2, ArrPos(1), ArrPos(4 + VarArrpos(0)), ArrPos(1), ArrPos(4 +
VarArrpos(0))))
            A = A + 1
            Calsize_path(Path_Result)
            PrintOutput_path(A)
            Print_Path()
            Rec_Data()
            Add_clearance(3, Clearance_cal(3, ArrPos(1), ArrPos(4 + VarArrpos(0)), ArrPos(1), ArrPos(4 +
VarArrpos(0))))
            A = A + 1
            Calsize_path(Path_Result)
            PrintOutput_path(A)
            Print_Path()
         End If
      End Select

   ElseIf Changeaxis = 1 Then
      Select Case Shift_to
         Case "Left"
            If Form1.RadioButton23.Checked = True Then
               Set_Po()
               ReDim ArrPos(Max - 1)
               ArrPos = sortstr()
               Set_Po()
               Add_clearance(1, Clearance_cal(1, ArrPos(0), ArrPos(1 + VarArrpos(1)), ArrPos(0), ArrPos(1 +
VarArrpos(1))))
               A = A + 1
               Calsize_path(Path_Result)
               PrintOutput_path(A)
               Print_Path()
               Rec_Data()
               Add_clearance(2, Clearance_cal(2, ArrPos(0), ArrPos(1 + VarArrpos(1)), ArrPos(0), ArrPos(1 +
VarArrpos(1))))
               A = A + 1
               Calsize_path(Path_Result)
               PrintOutput_path(A)
               Print_Path()
               Rec_Data()
               Add_clearance(3, Clearance_cal(3, ArrPos(0), ArrPos(1 + VarArrpos(1)), ArrPos(0), ArrPos(1 +
VarArrpos(1))))
               A = A + 1
               Calsize_path(Path_Result)
               PrintOutput_path(A)
               Print_Path()
            End If

            If Form1.RadioButton24.Checked = True Then
               Set_Po()
               ReDim ArrPos(Max - 1)
               ArrPos = sortstr()
               Set_Po()
               Add_clearance(1, Clearance_cal(1, ArrPos(1), ArrPos(2 + VarArrpos(1)), ArrPos(0), ArrPos(1)))
```

```
        A = A + 1
        Calsize_path(Path_Result)
        PrintOutput_path(A)
        Print_Path()
        Rec_Data()
        Add_clearance(2, Clearance_cal(2, ArrPos(1), ArrPos(2 + VarArrpos(1)), ArrPos(0), ArrPos(1)))
        A = A + 1
        Calsize_path(Path_Result)
        PrintOutput_path(A)
        Print_Path()
        Rec_Data()
        Add_clearance(3, Clearance_cal(3, ArrPos(1), ArrPos(2 + VarArrpos(1)), ArrPos(0), ArrPos(1)))
        A = A + 1
        Calsize_path(Path_Result)
        PrintOutput_path(A)
        Print_Path()
    End If

    If Form1.RadioButton25.Checked = True Then
        Set_Po()
        ReDim ArrPos(Max - 1)
        ArrPos = sortstr()
        Set_Po()
        Add_clearance(1, Clearance_cal(1, ArrPos(0), ArrPos(5), ArrPos(0), ArrPos(5)))
        A = A + 1
        Calsize_path(Path_Result)
        PrintOutput_path(A)
        Print_Path()
        Rec_Data()
        Add_clearance(2, Clearance_cal(2, ArrPos(0), ArrPos(5), ArrPos(0), ArrPos(5)))
        A = A + 1
        Calsize_path(Path_Result)
        PrintOutput_path(A)
        Print_Path()
        Rec_Data()
        Add_clearance(3, Clearance_cal(3, ArrPos(0), ArrPos(5), ArrPos(0), ArrPos(5)))
        A = A + 1
        Calsize_path(Path_Result)
        PrintOutput_path(A)
        Print_Path()
    End If

    If Form1.RadioButton26.Checked = True Then
        Set_Po()
        ReDim ArrPos(Max - 1)
        ArrPos = sortstr()
        Set_Po()
        Add_clearance(1, Clearance_cal(1, ArrPos(0), ArrPos(4), ArrPos(0), ArrPos(4)))
        A = A + 1
        Calsize_path(Path_Result)
        PrintOutput_path(A)
        Print_Path()
        Rec_Data()
        Add_clearance(2, Clearance_cal(2, ArrPos(0), ArrPos(4), ArrPos(0), ArrPos(4)))
        A = A + 1
        Calsize_path(Path_Result)
        PrintOutput_path(A)
        Print_Path()
        Rec_Data()
        Add_clearance(3, Clearance_cal(3, ArrPos(0), ArrPos(4), ArrPos(0), ArrPos(4)))
        A = A + 1
        Calsize_path(Path_Result)
        PrintOutput_path(A)
        Print_Path()
    End If

    If Form1.RadioButton27.Checked = True Then
        Set_Po()
        ReDim ArrPos(Max - 1)
        ArrPos = sortstr()
        Set_Po()
        Add_clearance(1, Clearance_cal(1, ArrPos(0), ArrPos(4), ArrPos(0), ArrPos(4)))
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
                A = A + 1
                Calsize_path(Path_Result)
                PrintOutput_path(A)
                Print_Path()
                Rec_Data()
                Add_clearance(2, Clearance_cal(1, ArrPos(0), ArrPos(4), ArrPos(0), ArrPos(4)))
                A = A + 1
                Calsize_path(Path_Result)
                PrintOutput_path(A)
                Print_Path()
                Rec_Data()
                Add_clearance(3, Clearance_cal(1, ArrPos(0), ArrPos(4), ArrPos(0), ArrPos(4)))
                A = A + 1
                Calsize_path(Path_Result)
                PrintOutput_path(A)
                Print_Path()
            End If
        Case "Right"
            If Form1.RadioButton23.Checked = True Then
                Set_Po()
                ReDim ArrPos(Max - 1)
                ArrPos = sortstr()
                Set_Po()
                Add_clearance(1, Clearance_cal(1, ArrPos(1), ArrPos(2 + VarArrpos(1)), ArrPos(1), ArrPos(2 +
VarArrpos(1))))
                A = A + 1
                Calsize_path(Path_Result)
                PrintOutput_path(A)
                Print_Path()
                Rec_Data()
                Add_clearance(2, Clearance_cal(2, ArrPos(1), ArrPos(2 + VarArrpos(1)), ArrPos(1), ArrPos(2 +
VarArrpos(1))))
                A = A + 1
                Calsize_path(Path_Result)
                PrintOutput_path(A)
                Print_Path()
                Rec_Data()
                Add_clearance(3, Clearance_cal(3, ArrPos(1), ArrPos(2 + VarArrpos(1)), ArrPos(1), ArrPos(2 +
VarArrpos(1))))
                A = A + 1
                Calsize_path(Path_Result)
                PrintOutput_path(A)
                Print_Path()
            End If

            If Form1.RadioButton24.Checked = True Then
                Set_Po()
                ReDim ArrPos(Max - 1)
                ArrPos = sortstr()
                Set_Po()
                Add_clearance(1, Clearance_cal(1, ArrPos(2), ArrPos(3 + VarArrpos(1)), ArrPos(0), ArrPos(1)))
                A = A + 1
                Calsize_path(Path_Result)
                PrintOutput_path(A)
                Print_Path()
                Rec_Data()
                Add_clearance(2, Clearance_cal(2, ArrPos(2), ArrPos(3 + VarArrpos(1)), ArrPos(0), ArrPos(1)))
                A = A + 1
                Calsize_path(Path_Result)
                PrintOutput_path(A)
                Print_Path()
                Rec_Data()
                Add_clearance(3, Clearance_cal(3, ArrPos(2), ArrPos(3 + VarArrpos(1)), ArrPos(0), ArrPos(1)))
                A = A + 1
                Calsize_path(Path_Result)
                PrintOutput_path(A)
                Print_Path()
            End If

            If Form1.RadioButton25.Checked = True Then
                Set_Po()
                ReDim ArrPos(Max - 1)
```

```
            ArrPos = sortstr()
            Set_Po()
            Add_clearance(1, Clearance_cal(1, ArrPos(0), ArrPos(5), ArrPos(0), ArrPos(5)))
            A = A + 1
            Calsize_path(Path_Result)
            PrintOutput_path(A)
            Print_Path()
            Rec_Data()
            Add_clearance(2, Clearance_cal(2, ArrPos(0), ArrPos(5), ArrPos(0), ArrPos(5)))
            A = A + 1
            Calsize_path(Path_Result)
            PrintOutput_path(A)
            Print_Path()
            Rec_Data()
            Add_clearance(3, Clearance_cal(3, ArrPos(0), ArrPos(5), ArrPos(0), ArrPos(5)))
            A = A + 1
            Calsize_path(Path_Result)
            PrintOutput_path(A)
            Print_Path()
        End If

        If Form1.RadioButton26.Checked = True Then
            Set_Po()
            ReDim ArrPos(Max - 1)
            ArrPos = sortstr()
            Set_Po
            Add_clearance(1, Clearance_cal(1, ArrPos(1), ArrPos(5), ArrPos(1), ArrPos(5)))
            A = A + 1
            Calsize_path(Path_Result)
            PrintOutput_path(A)
            Print_Path()
            Rec_Data()
            Add_clearance(2, Clearance_cal(2, ArrPos(1), ArrPos(5), ArrPos(1), ArrPos(5)))
            A = A + 1
            Calsize_path(Path_Result)
            PrintOutput_path(A)
            Print_Path()
            Rec_Data()
            Add_clearance(3, Clearance_cal(3, ArrPos(1), ArrPos(5), ArrPos(1), ArrPos(5)))
            A = A + 1
            Calsize_path(Path_Result)
            PrintOutput_path(A)
            Print_Path()
        End If

        If Form1.RadioButton27.Checked = True Then
            Set_Po()
            ReDim ArrPos(Max - 1)
            ArrPos = sortstr()
            Set_Po()
            Add_clearance(1, Clearance_cal(1, ArrPos(1), ArrPos(5), ArrPos(1), ArrPos(5)))
            A = A + 1
            Calsize_path(Path_Result)
            PrintOutput_path(A)
            Print_Path()
            Rec_Data()
            Add_clearance(2, Clearance_cal(1, ArrPos(1), ArrPos(5), ArrPos(1), ArrPos(5)))
            A = A + 1
            Calsize_path(Path_Result)
            PrintOutput_path(A)
            Print_Path()
            Rec_Data()
            Add_clearance(3, Clearance_cal(1, ArrPos(1), ArrPos(5), ArrPos(1), ArrPos(5)))
            A = A + 1
            Calsize_path(Path_Result)
            PrintOutput_path(A)
            Print_Path()
        End If

    End Select
End If

Form4.Show()
```

```
    If Form1.CheckBox1.Checked = True Then
        Changeaxis = Changeaxis + 1
    End If

    If Changeaxis = 1 Then
       Form4.B = 0
       Rec_Req()
    End If

End Sub
```

Output Module

```
  Dim Hole As Boolean
  Public Sub PrintOutput_path(ByVal A As Integer)
    Dim B As String

    If A = 1 Then
       B = "C8_h8"
       If Changeaxis <> 1 Then
          If A = 1 Then
             Form4.Outbox.AppendText(vbNewLine & ">>>>>>>>> HORIZONTAL AXIS <<<<<<<<<")
          End If
          If Form1.RadioButton4.Checked = True Then
             Form4.Outbox.AppendText(vbNewLine & "งานเหลี่ยมโดยใช้หมุดเป็นรัง" & vbNewLine)
             Jigtype = 2
          ElseIf Form1.RadioButton3.Checked = True Then
             Form4.Outbox.AppendText(vbNewLine & "งานเหลี่ยมโดยใช้ราง" & vbNewLine)
             Jigtype = 1
          ElseIf Form1.RadioButton5.Checked = True Then
             Form4.Outbox.AppendText(vbNewLine & "งานกลมโดยใช้หมุดเป็นตัวกำหนดตำแหน่ง" & vbNewLine)
             Jigtype = 3
          ElseIf Form1.RadioButton6.Checked = True Then
             Form4.Outbox.AppendText(vbNewLine & "งานกลมโดยใช้รูเป็นตัวกำหนดตำแหน่ง" & vbNewLine)
             Jigtype = 4
          ElseIf Form1.RadioButton7.Checked = True Then
             Form4.Outbox.AppendText(vbNewLine & "งานกลมโดยใช้รัง" & vbNewLine)
             Jigtype = 5
          End If

       ElseIf Changeaxis = 1 Then
          If A = 1 Then
             Form4.Outbox.AppendText(vbNewLine & ">>>>>>>>> VERTICAL AXIS <<<<<<<<<")
          End If
          If Form1.RadioButton24.Checked = True Then
             Form4.Outbox.AppendText(vbNewLine & "งานเหลี่ยมโดยใช้หมุดเป็นรัง" & vbNewLine)
             Jigtype = 2
          ElseIf Form1.RadioButton23.Checked = True Then
             Form4.Outbox.AppendText(vbNewLine & "งานเหลี่ยมโดยใช้ราง" & vbNewLine)
             Jigtype = 1
          ElseIf Form1.RadioButton25.Checked = True Then
             Form4.Outbox.AppendText(vbNewLine & "งานกลมโดยใช้หมุดเป็นตัวกำหนดตำแหน่ง" & vbNewLine)
             Jigtype = 3
          ElseIf Form1.RadioButton26.Checked = True Then
             Form4.Outbox.AppendText(vbNewLine & "งานกลมโดยใช้รูเป็นตัวกำหนดตำแหน่ง" & vbNewLine)
             Jigtype = 4
          ElseIf Form1.RadioButton27.Checked = True Then
             Form4.Outbox.AppendText(vbNewLine & "งานกลมโดยใช้รัง" & vbNewLine)
             Jigtype = 5
          End If
       End If
    ElseIf A = 2 Then
       B = "G8_h8"
    ElseIf A = 3 Then
       B = "H8_h8"
    End If

If Changeaxis <> 1 Then
       Check_BushPoH()
    ElseIf Changeaxis = 1 Then
```

```
    Check_BushPoV()
End If
If Hole = True Then
    Form4.Outbox.AppendText(vbNewLine & "Preferred Fit " & " " & B & vbNewLine & _
                "Bush position along with axis " & "(" & Req(0) & "-" & Req(1) & ")" & " is")
Else
    Form4.Outbox.AppendText(vbNewLine & "Preferred Fit " & " " & B & vbNewLine & _
                "Locator size/position along with axis " & "(" & Req(0) & "-" & Req(1) & ")" & " is")
End If

If Tol_result > 0 Then
    Form4.Outbox.AppendText("Tolerance stack problem occured")
ElseIf Jigtype = 1 Then
    If Po(Req(0)) = ArrPos(0) And Po(Req(1)) = ArrPos(3) Then
        Form4.Outbox.AppendText(vbNewLine & _
                vbTab & Abs((Abs(Basic_result) - Abs(Tol_result))).ToString("###.0000") & _
                vbTab & " + " & CStr(Abs(Tol_result * 2).ToString("###.0000")) & _
                 " / - " & "0 " & vbNewLine)
    ElseIf Po(Req(1)) = ArrPos(0) And Po(Req(0)) = ArrPos(3) Then
        Form4.Outbox.AppendText(vbNewLine & _
                vbTab & Abs((Abs(Basic_result) - Abs(Tol_result))).ToString("###.0000") & _
                vbTab & " + " & CStr(Abs(Tol_result * 2).ToString("###.0000")) & _
                 " / - " & "0 " & vbNewLine)
    Else
        Form4.Outbox.AppendText(vbNewLine & _
                vbTab & (Abs(Basic_result)).ToString("###.0000") & _
                vbTab & " + " & CStr(Abs(Tol_result).ToString("###.0000")) & _
                 " / - " & CStr(Abs(Tol_result).ToString("###.0000")) & vbNewLine)
    End If

ElseIf Jigtype = 3 Then
    If Po(Req(0)) = ArrPos(3) And Po(Req(1)) = ArrPos(0) Then
        Form4.Outbox.AppendText(vbNewLine & _
                vbTab & Abs((Abs(Basic_result) + Abs(Tol_result))).ToString("###.0000") & _
                vbTab & " + " & "0 " & _
                 " / - " & CStr(Abs(Tol_result * 2).ToString("###.0000")) & vbNewLine)
    ElseIf Po(Req(1)) = ArrPos(3) And Po(Req(0)) = ArrPos(0) Then
        Form4.Outbox.AppendText(vbNewLine & _
                vbTab & Abs((Abs(Basic_result) + Abs(Tol_result))).ToString("###.0000") & _
                vbTab & " + " & "0 " & _
                 " / - " & CStr(Abs(Tol_result * 2).ToString("###.0000")) & vbNewLine)
    Else
        Form4.Outbox.AppendText(vbNewLine & _
               vbTab & (Abs(Basic_result)).ToString("###.0000") & _
               vbTab & " + " & CStr(Abs(Tol_result).ToString("###.0000")) & _
                " / - " & CStr(Abs(Tol_result).ToString("###.0000")) & vbNewLine)
    End If
ElseIf Jigtype = 4 Then
    If Po(Req(0)) = ArrPos(0) And Po(Req(1)) = ArrPos(4) Then
        Form4.Outbox.AppendText(vbNewLine & _
                vbTab & Abs((Abs(Basic_result) - Abs(Tol_result))).ToString("###.0000") & _
                vbTab & " + " & CStr(Abs(Tol_result * 2).ToString("###.0000")) & _
                 " / - " & "0 " & vbNewLine)

    ElseIf Po(Req(1)) = ArrPos(0) And Po(Req(0)) = ArrPos(4) Then
        Form4.Outbox.AppendText(vbNewLine & _
                vbTab & Abs((Abs(Basic_result) - Abs(Tol_result))).ToString("###.0000") & _
                vbTab & " + " & CStr(Abs(Tol_result * 2).ToString("###.0000")) & _
                 " / - " & "0 " & vbNewLine)
    Else
        Form4.Outbox.AppendText(vbNewLine & _
               vbTab & (Abs(Basic_result)).ToString("###.0000") & _
               vbTab & " + " & CStr(Abs(Tol_result).ToString("###.0000")) & _
                " / - " & CStr(Abs(Tol_result).ToString("###.0000")) & vbNewLine)
    End If
ElseIf Jigtype = 5 Then
    If Po(Req(0)) = ArrPos(0) And Po(Req(1)) = ArrPos(5) Then
        Form4.Outbox.AppendText(vbNewLine & _
                vbTab & Abs((Abs(Basic_result) - Abs(Tol_result))).ToString("###.0000") & _
                vbTab & " + " & CStr(Abs(Tol_result * 2).ToString("###.0000")) & _
                 " / - " & "0 " & vbNewLine)

    ElseIf Po(Req(1)) = ArrPos(0) And Po(Req(0)) = ArrPos(5) Then
```

```
            Form4.Outbox.AppendText(vbNewLine & _
                    vbTab & Abs((Abs(Basic_result) - Abs(Tol_result))).ToString("###.0000") & _
                    vbTab & " + " & CStr(Abs(Tol_result * 2).ToString("###.0000")) & _
                     " / - " & "0 " & vbNewLine)
        Else
            Form4.Outbox.AppendText(vbNewLine & _
                    vbTab & (Abs(Basic_result)).ToString("###.0000") & _
                    vbTab & " + " & CStr(Abs(Tol_result).ToString("###.0000")) & _
                     " / - " & CStr(Abs(Tol_result).ToString("###.0000")) & vbNewLine)
        End If
    Else
        Form4.Outbox.AppendText(vbNewLine & _
                     vbTab & Abs(Basic_result).ToString("###.0000") & _
                     vbTab & " + " & CStr(Abs(Tol_result).ToString("###.0000")) & _
                      " - " & CStr(Abs(Tol_result).ToString("###.0000")) & vbNewLine)
    End If
    Form4.BringToFront()
    Form4.Show()
End Sub

Public Sub Print_Path()
    Dim icount As Integer
    Dim jcount As Integer

     Form4.Outbox.AppendText(vbNewLine & "From" & vbTab & "To" & vbTab & "Basic sizes" & vbTab &
"Tolerances")
    For icount = 0 To Path_Result.Length - 2
        For jcount = 0 To inputx.Length - 1
            If inputx(jcount).F = Path_Result(icount) And inputx(jcount).T = Path_Result(icount + 1) Then
                Form4.Outbox.AppendText(vbNewLine & Path_Result(icount) & vbTab & Path_Result(icount + 1) & vbTab &
Abs(inputx(jcount).Dims) & vbTab & vbTab & Abs(inputx(jcount).Dimtol))
            ElseIf inputx(jcount).T = Path_Result(icount) And inputx(jcount).F = Path_Result(icount + 1) Then
                Form4.Outbox.AppendText(vbNewLine & Path_Result(icount) & vbTab & Path_Result(icount + 1) & vbTab &
Abs(inputx(jcount).Dims) & vbTab & vbTab & Abs(inputx(jcount).Dimtol))
            End If
        Next
    Next
    Form4.Outbox.AppendText(vbNewLine)
End Sub
```

Utilities Module

```
Public Function Fit_Gh(Phii As Single) As Single
    Dim Mincl As Single
    If (Phii > 0) And (Phii <= 3) Then
        Mincl = 0.002
    ElseIf (Phii > 3) And (Phii <= 6) Then
        Mincl = 0.004
    ElseIf (Phii > 6) And (Phii <= 10) Then
        Mincl = 0.005
    ElseIf (Phii > 10) And (Phii <= 18) Then
        Mincl = 0.006
    ElseIf (Phii > 18) And (Phii <= 30) Then
        Mincl = 0.007
    ElseIf (Phii > 30) And (Phii <= 50) Then
        Mincl = 0.009
    ElseIf (Phii > 50) And (Phii <= 80) Then
        Mincl = 0.01
    ElseIf (Phii > 80) And (Phii <= 120) Then
        Mincl = 0.012
    ElseIf (Phii > 120) And (Phii <= 180) Then
        Mincl = 0.014
    ElseIf (Phii > 180) And (Phii <= 250) Then
        Mincl = 0.015
    ElseIf (Phii > 250) And (Phii <= 315) Then
        Mincl = 0.017
    ElseIf (Phii > 315) And (Phii <= 400) Then
        Mincl = 0.018
    ElseIf (Phii > 400) And (Phii <= 500) Then
        Mincl = 0.02
    ElseIf (Phii > 500) And (Phii <= 630) Then
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        Mincl = 0.022
    ElseIf (Phii > 630) And (Phii <= 800) Then
        Mincl = 0.024
    ElseIf (Phii > 800) And (Phii <= 1000) Then
        Mincl = 0.026
    End If
    Return Mincl
End Function

 Public Function Fit_Hh(Phii As Single) As Single
    Dim Mincl As Double
    If (Phii > 0) And (Phii <= 3) Then
        Mincl = 0
    ElseIf (Phii > 3) And (Phii <= 6) Then
        Mincl = 0
    ElseIf (Phii > 6) And (Phii <= 10) Then
        Mincl = 0
    ElseIf (Phii > 10) And (Phii <= 18) Then
        Mincl = 0
    ElseIf (Phii > 18) And (Phii <= 30) Then
        Mincl = 0
    ElseIf (Phii > 30) And (Phii <= 50) Then
        Mincl = 0
    ElseIf (Phii > 50) And (Phii <= 80) Then
        Mincl = 0
    ElseIf (Phii > 80) And (Phii <= 120) Then
        Mincl = 0
    ElseIf (Phii > 120) And (Phii <= 180) Then
        Mincl = 0
    ElseIf (Phii > 180) And (Phii <= 250) Then
        Mincl = 0
    ElseIf (Phii > 250) And (Phii <= 315) Then
        Mincl = 0
    ElseIf (Phii > 315) And (Phii <= 400) Then
        Mincl = 0
    ElseIf (Phii > 400) And (Phii <= 500) Then
        Mincl = 0
    ElseIf (Phii > 500) And (Phii <= 630) Then
        Mincl = 0
    ElseIf (Phii > 630) And (Phii <= 800) Then
        Mincl = 0
    ElseIf (Phii > 800) And (Phii <= 1000) Then
        Mincl = 0
    End If
    Return Mincl
End Function

Public Function Fit_Ch(Phii As Single) As Single
    Dim Mincl As Double
    If (Phii > 0) And (Phii <= 3) Then
        Mincl = 0.06
    ElseIf (Phii > 3) And (Phii <= 6) Then
        Mincl = 0.07
    ElseIf (Phii > 6) And (Phii <= 10) Then
        Mincl = 0.08
    ElseIf (Phii > 10) And (Phii <= 18) Then
        Mincl = 0.095
    ElseIf (Phii > 18) And (Phii <= 30) Then
        Mincl = 0.11
    ElseIf (Phii > 30) And (Phii <= 40) Then
        Mincl = 0.12
    ElseIf (Phii > 40) And (Phii <= 50) Then
        Mincl = 0.13
    ElseIf (Phii > 50) And (Phii <= 65) Then
        Mincl = 0.14
    ElseIf (Phii > 65) And (Phii <= 80) Then
        Mincl = 0.15
    ElseIf (Phii > 80) And (Phii <= 100) Then
        Mincl = 0.17
    ElseIf (Phii > 100) And (Phii <= 120) Then
        Mincl = 0.18
    ElseIf (Phii > 120) And (Phii <= 140) Then
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        Mincl = 0.2
    ElseIf (Phii > 140) And (Phii <= 160) Then
        Mincl = 0.21
    ElseIf (Phii > 160) And (Phii <= 180) Then
        Mincl = 0.23
    ElseIf (Phii > 180) And (Phii <= 200) Then
        Mincl = 0.24
    ElseIf (Phii > 200) And (Phii <= 225) Then
        Mincl = 0.26
    ElseIf (Phii > 225) And (Phii <= 250) Then
        Mincl = 0.28
    ElseIf (Phii > 250) And (Phii <= 280) Then
        Mincl = 0.3
    ElseIf (Phii > 280) And (Phii <= 315) Then
        Mincl = 0.33
    ElseIf (Phii > 315) And (Phii <= 355) Then
        Mincl = 0.36
    ElseIf (Phii > 355) And (Phii <= 400) Then
        Mincl = 0.4
    ElseIf (Phii > 400) And (Phii <= 450) Then
        Mincl = 0.44
    ElseIf (Phii > 450) And (Phii <= 500) Then
        Mincl = 0.48
    End If
    Return Mincl
End Function

Public Function IT6(Phii As Single) As Single
    Dim IT As Single
    If (Phii > 0) And (Phii <= 3) Then
        IT = 0.006
    ElseIf (Phii > 3) And (Phii <= 6) Then
        IT = 0.008
    ElseIf (Phii > 6) And (Phii <= 10) Then
        IT = 0.009
    ElseIf (Phii > 10) And (Phii <= 18) Then
        IT = 0.011
    ElseIf (Phii > 18) And (Phii <= 30) Then
        IT = 0.013
    ElseIf (Phii > 30) And (Phii <= 50) Then
        IT = 0.016
    ElseIf (Phii > 50) And (Phii <= 80) Then
        IT = 0.019
    ElseIf (Phii > 80) And (Phii <= 120) Then
        IT = 0.022
    ElseIf (Phii > 120) And (Phii <= 180) Then
        IT = 0.025
    ElseIf (Phii > 180) And (Phii <= 250) Then
        IT = 0.029
    ElseIf (Phii > 250) And (Phii <= 315) Then
        IT = 0.032
    ElseIf (Phii > 315) And (Phii <= 400) Then
        IT = 0.036
    ElseIf (Phii > 400) And (Phii <= 500) Then
        IT = 0.04
    ElseIf (Phii > 500) And (Phii <= 630) Then
        IT = 0.044
    ElseIf (Phii > 630) And (Phii <= 800) Then
        IT = 0.05
    ElseIf (Phii > 800) And (Phii <= 1000) Then
        IT = 0.056
    End If
    Return IT
End Function

Public Function IT8(Phii As Single) As Single
    Dim IT As Single
    If (Phii > 0) And (Phii <= 3) Then
        IT = 0.014
    ElseIf (Phii > 3) And (Phii <= 6) Then
        IT = 0.018
    ElseIf (Phii > 6) And (Phii <= 10) Then
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        IT = 0.022
    ElseIf (Phii > 10) And (Phii <= 18) Then
        IT = 0.027
    ElseIf (Phii > 18) And (Phii <= 30) Then
        IT = 0.033
    ElseIf (Phii > 30) And (Phii <= 50) Then
        IT = 0.039
    ElseIf (Phii > 50) And (Phii <= 80) Then
        IT = 0.046
    ElseIf (Phii > 80) And (Phii <= 120) Then
        IT = 0.054
    ElseIf (Phii > 120) And (Phii <= 180) Then
        IT = 0.063
    ElseIf (Phii > 180) And (Phii <= 250) Then
        IT = 0.072
    ElseIf (Phii > 250) And (Phii <= 315) Then
        IT = 0.081
    ElseIf (Phii > 315) And (Phii <= 400) Then
        IT = 0.089
    ElseIf (Phii > 400) And (Phii <= 500) Then
        IT = 0.097
    ElseIf (Phii > 500) And (Phii <= 630) Then
        IT = 0.11
    ElseIf (Phii > 630) And (Phii <= 800) Then
        IT = 0.125
    ElseIf (Phii > 800) And (Phii <= 1000) Then
        IT = 0.14
    End If
    Return IT
End Function
```

Support Module

Savefile :

```
Public Sub PicConPos()
    Dim Astr As String
    Dim Bstr() As String
    Dim Dstr() As String
    Dim Namesearch As String
    Astr = File.ReadAllText("D:\study\MD\thesis\ตัวโปรแกรม\determination of bush position\determination of bush position
2\determination of bush position 2\Picture\Pic-type.txt")
    Bstr = Split(Astr, vbNewLine)

    If Form1.CheckBox1.Checked = False Then
        Namesearch = "OnlyHor"
        If Form1.RadioButton3.Checked = True Then
            Namesearch = Namesearch & "Spouse"
            If CInt(Form1.txtDim.Text) = 3 Then
                Namesearch = Namesearch & "1"
            ElseIf CInt(Form1.txtDim.Text) > 3 And CInt(Form1.txtDim.Text) <= 7 Then
                Namesearch = Namesearch & CStr(CInt(Form1.txtDim.Text) - 2)
            Else
                Namesearch = Namesearch & "1"
            End If

        ElseIf Form1.RadioButton4.Checked = True Then
            Namesearch = Namesearch & "NP"
            If CInt(Form1.txtDim.Text) = 5 Then
                Namesearch = Namesearch & "1"
            ElseIf CInt(Form1.txtDim.Text) > 5 And CInt(Form1.txtDim.Text) <= 9 Then
                Namesearch = Namesearch & CStr(CInt(Form1.txtDim.Text) - 4)
            Else
                Namesearch = Namesearch & "1"
            End If

        ElseIf Form1.RadioButton5.Checked = True Then
            Namesearch = Namesearch & "Pinlocator"
            If CInt(Form1.txtDim.Text) = 3 Then
                Namesearch = Namesearch & "1"
            ElseIf CInt(Form1.txtDim.Text) > 3 And CInt(Form1.txtDim.Text) <= 4 Then
                Namesearch = Namesearch & CStr(CInt(Form1.txtDim.Text) - 2)
```

```
            Else
               Namesearch = Namesearch & "1"
            End If

         ElseIf Form1.RadioButton6.Checked = True Then
            Namesearch = Namesearch & "Holelocator"
            If CInt(Form1.txtDim.Text) = 3 Then
               Namesearch = Namesearch & "1"
            ElseIf CInt(Form1.txtDim.Text) > 3 And CInt(Form1.txtDim.Text) <= 4 Then
               Namesearch = Namesearch & CStr(CInt(Form1.txtDim.Text) - 2)
             Else
               Namesearch = Namesearch & "1"
            End If

         ElseIf Form1.RadioButton7.Checked = True Then
            Namesearch = Namesearch & "Nest"
            If CInt(Form1.txtDim.Text) = 3 Then
               Namesearch = Namesearch & "1"
            ElseIf CInt(Form1.txtDim.Text) > 3 And CInt(Form1.txtDim.Text) <= 4 Then
               Namesearch = Namesearch & CStr(CInt(Form1.txtDim.Text) - 2)
            Else
               Namesearch = Namesearch & "1"
            End If
         End If

      ElseIf Form1.CheckBox1.Checked = True Then
         Namesearch = "HorVer"
         If Form1.RadioButton3.Checked = True Then
            Namesearch = Namesearch & "Spouse"
            If Form1.RadioButton23.Checked = True Then
               Namesearch = Namesearch & "Spouse"
            ElseIf Form1.RadioButton24.Checked = True Then
               Namesearch = Namesearch & "NP"
            End If

            If CInt(Form1.txtDim.Text) = 3 Then
               Namesearch = Namesearch & "1"
            ElseIf CInt(Form1.txtDim.Text) > 3 And CInt(Form1.txtDim.Text) <= 7 Then
               Namesearch = Namesearch & CStr(CInt(Form1.txtDim.Text) - 2)
            Else
               Namesearch = Namesearch & "1"
            End If

         ElseIf Form1.RadioButton4.Checked = True Then
            Namesearch = Namesearch & "NP"
            If Form1.RadioButton23.Checked = True Then
               Namesearch = Namesearch & "Spouse"
            ElseIf Form1.RadioButton24.Checked = True Then
               Namesearch = Namesearch & "NP"
            End If

            If CInt(Form1.txtDim.Text) = 5 Then
               Namesearch = Namesearch & "1"
            ElseIf CInt(Form1.txtDim.Text) > 5 And CInt(Form1.txtDim.Text) <= 9 Then
               Namesearch = Namesearch & CStr(CInt(Form1.txtDim.Text) - 4)
            Else
               Namesearch = Namesearch & "1"
            End If

         ElseIf Form1.RadioButton5.Checked = True Then
            Namesearch = Namesearch & "Pinlocator"
            If CInt(Form1.txtDim.Text) = 3 Then
               Namesearch = Namesearch & "1"
            ElseIf CInt(Form1.txtDim.Text) > 3 And CInt(Form1.txtDim.Text) <= 4 Then
               Namesearch = Namesearch & CStr(CInt(Form1.txtDim.Text) - 2)
            Else
               Namesearch = Namesearch & "1"
            End If

         ElseIf Form1.RadioButton6.Checked = True Then
            Namesearch = Namesearch & "Holelocator"
            If CInt(Form1.txtDim.Text) = 3 Then
```

```
            Namesearch = Namesearch & "1"
          ElseIf CInt(Form1.txtDim.Text) > 3 And CInt(Form1.txtDim.Text) <= 4 Then
            Namesearch = Namesearch & CStr(CInt(Form1.txtDim.Text) - 2)
          Else
            Namesearch = Namesearch & "1"
          End If

        ElseIf Form1.RadioButton7.Checked = True Then
          Namesearch = Namesearch & "Nest"
          If CInt(Form1.txtDim.Text) = 3 Then
            Namesearch = Namesearch & "1"
          ElseIf CInt(Form1.txtDim.Text) > 3 And CInt(Form1.txtDim.Text) <= 4 Then
            Namesearch = Namesearch & CStr(CInt(Form1.txtDim.Text) - 2)
          Else
            Namesearch = Namesearch & "1"
          End If
        End If
      End If

      For i As Integer = 0 To Bstr.Length - 1
        Dstr = Split(Bstr(i), ",")
        If String.Compare(Dstr(0), Namesearch) = 0 Then
          PicFileName = Dstr(1)
          Exit For
        End If
      Next
    End Sub

      SaveFileDialog1.Filter = "Text files (*.txt) | *.txt"
      If SaveFileDialog1.ShowDialog() = DialogResult.OK Then
        My.Computer.FileSystem.WriteAllText(SaveFileDialog1.FileName, str.ToString(), False)
      End If

    Private Sub SaveFileToolStripMenuItem_Click(sender As Object, e As EventArgs) Handles
    SaveFileToolStripMenuItem.Click
      Dim str As String = ""
      Dim jig_type As Integer

      str = Form1.txtDim.Text & vbNewLine
      'str = str & Form1.txtW.Text & vbNewLine
      If Form1.RadioButton4.Checked Then
        jig_type = 4
      ElseIf Form1.RadioButton3.Checked Then
        jig_type = 3
      ElseIf Form1.RadioButton5.Checked Then
        jig_type = 1
      ElseIf Form1.RadioButton6.Checked Then
        jig_type = 2
      End If
      str = str & CStr(jig_type) & vbNewLine
      For icount As Integer = 0 To CInt(Form1.txtDim.Text) - 1
        str = str & txtF(icount).Text & " "
      Next
      str = str & vbNewLine
      For icount As Integer = 0 To CInt(Form1.txtDim.Text) - 1
        str = str & txtT(icount).Text & " "
      Next
      str = str & vbNewLine
      For icount As Integer = 0 To CInt(Form1.txtDim.Text) - 1
        str = str & Form1.NodePo(CInt(txtF(icount).Text - 1)) & " "
      Next
      str = str & vbNewLine
      For icount As Integer = 0 To CInt(Form1.txtDim.Text) - 1
        str = str & Form1.NodePo(CInt(txtT(icount).Text) - 1) & " "
      Next
      str = str & vbNewLine
      For icount As Integer = 0 To CInt(Form1.txtDim.Text) - 1
        str = str & txtdimX(icount).Text & " "
      Next
      str = str & vbNewLine
      For icount As Integer = 0 To CInt(Form1.txtDim.Text) - 1
        str = str & txtTolXUp(icount).Text & " "
      Next
```

```
    str = str & vbNewLine
    For icount As Integer = 0 To CInt(Form1.txtDim.Text) - 1
       str = str & txtTolXLo(icount).Text & " "
    Next
    str = str & vbNewLine
    If Form1.ComboBox1.SelectedIndex = 0 Then
       str = str & "Diameter" & vbNewLine
    ElseIf Form1.ComboBox1.SelectedIndex = 1 Then
       str = str & "Length" & vbNewLine
    End If
    For icount As Integer = 0 To CInt(Form1.NodePo.Length - 1)
       str = str & Form1.NodePo(icount) & " "
    Next
    str = str & vbNewLine
    For icount As Integer = 0 To CheckHF.Length - 1
       If CheckHF(icount).Checked = True Then
          str = str & icount & " "
       End If
    Next
    str = str & vbNewLine
    If Form1.CheckBox1.Checked = True Then
       str = str & Form1.txtDimV.Text & vbNewLine
       If Form1.RadioButton23.Checked Then
          jig_type = 3
       ElseIf Form1.RadioButton24.Checked Then
          jig_type = 4
       ElseIf Form1.RadioButton25.Checked Then
          jig_type = 1
       ElseIf Form1.RadioButton26.Checked Then
          jig_type = 2
       End If
       str = str & jig_type & vbNewLine
       For icount As Integer = 0 To CInt(Form1.txtDimV.Text) - 1
          str = str & txtFV(icount).Text & " "
       Next
       str = str & vbNewLine
       For icount As Integer = 0 To CInt(Form1.txtDimV.Text) - 1
          str = str & txtTV(icount).Text & " "
       Next
       str = str & vbNewLine
       For icount As Integer = 0 To CInt(Form1.txtDimV.Text) - 1
          str = str & Form1.NodePoV(CInt(txtFV(icount).Text) - 1) & " "
       Next
       str = str & vbNewLine
       For icount As Integer = 0 To CInt(Form1.txtDimV.Text) - 1
          str = str & Form1.NodePoV(CInt(txtTV(icount).Text) - 1) & " "
       Next
       str = str & vbNewLine
       For icount As Integer = 0 To CInt(Form1.txtDimV.Text) - 1
          str = str & txtdimXV(icount).Text & " "
       Next
       str = str & vbNewLine
       For icount As Integer = 0 To CInt(Form1.txtDimV.Text) - 1
          str = str & txtTolXUpV(icount).Text & " "
       Next
       str = str & vbNewLine
       For icount As Integer = 0 To CInt(Form1.txtDimV.Text) - 1
          str = str & txtTolXLoV(icount).Text & " "
       Next
       str = str & vbNewLine
       If Form1.ComboBox2.SelectedIndex = 0 Then
          str = str & "Diameter" & vbNewLine
       ElseIf Form1.ComboBox2.SelectedIndex = 1 Then
          str = str & "Length" & vbNewLine
       End If
       For icount As Integer = 0 To CInt(Form1.NodePoV.Length - 1)
          str = str & Form1.NodePoV(icount) & " "
       Next
       str = str & vbNewLine
       For icount As Integer = 0 To CheckVF.Length - 1
          If CheckVF(icount).Checked = True Then
```

```
            str = str & icount & " "
          End If
        Next
      End If

      SaveFileDialog1.Filter = "Text files (*.txt) | *.txt"
      If SaveFileDialog1.ShowDialog() = DialogResult.OK Then
        My.Computer.FileSystem.WriteAllText(SaveFileDialog1.FileName, str.ToString(), False)
       End If
    End Sub
```

Open file :

```
    Private Sub OpenFileToolStripMenuItem_Click(sender As Object, e As EventArgs) Handles
    OpenFileToolStripMenuItem.Click
      Dim str As String = ""
      Dim strtotal() As String

       fromfile = True
      OpenFileDialog1.Filter = "Text file (*.txt|*txt"
      If OpenFileDialog1.ShowDialog() = DialogResult.OK Then
        Try
          str = My.Computer.FileSystem.ReadAllText(OpenFileDialog1.FileName)
          strtotal = str.Split(vbCrLf)

        Catch ex As Exception
          MsgBox("An error occurred " & vbCrLf & ex.Message)
        End Try
      End If
      If strtotal.Length > 12 Then
        GroupBox2.Visible = True
        CheckBox1.Checked = True
        Form3.GroupBox2.Visible = True
      End If

       Dim strFrom() As String
      Dim strTo() As String
      Dim strDim() As String
      Dim strTolUp() As String
      Dim strTolLo() As String
      Dim jigform As Integer
      Dim strFreqcheck() As String

      txtDim.Text = strtotal(0)
      jigform = CInt(strtotal(1))

       Add_textbox()

      strFrom = strtotal(2).Split(" ")
      strTo = strtotal(3).Split(" ")
      PosFrom = strtotal(4).Split(" ")
      PosTo = strtotal(5).Split(" ")
      strDim = strtotal(6).Split(" ")
      strTolUp = strtotal(7).Split(" ")
      strTolLo = strtotal(8).Split(" ")
      NodePo = strtotal(10).Split(" ")
      If strtotal(9).EndsWith("Diameter") = True Then
        ComboBox1.SelectedIndex = 0
      Else
        ComboBox1.SelectedIndex = 1
      End If

      'เลือกประเภทจิกของงานเหลี่ยม
      If jigform = 1 And ComboBox1.SelectedIndex = 1 Then
      ElseIf jigform = 2 And ComboBox1.SelectedIndex = 1 Then
      ElseIf jigform = 3 And ComboBox1.SelectedIndex = 1 Then
        RadioButton3.Checked = True
        RadioButton4.Checked = False
      ElseIf jigform = 4 And ComboBox1.SelectedIndex = 1 Then
        RadioButton3.Checked = False
        RadioButton4.Checked = True
      End If

      'เลือกประเภทจิกของงานกลม
      If jigform = 1 And ComboBox1.SelectedIndex = 0 Then
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        RadioButton5.Checked = True
        RadioButton6.Checked = False
        RadioButton7.Checked = False
    ElseIf jigform = 2 And ComboBox1.SelectedIndex = 0 Then
        RadioButton5.Checked = False
        RadioButton6.Checked = True
        RadioButton7.Checked = False
    ElseIf jigform = 3 And ComboBox1.SelectedIndex = 0 Then
        RadioButton5.Checked = False
        RadioButton6.Checked = False
        RadioButton7.Checked = True
    ElseIf jigform = 4 And ComboBox1.SelectedIndex = 0 Then
        RadioButton5.Checked = False
        RadioButton6.Checked = False
        RadioButton7.Checked = False
    End If

    strtotal(11) = Trim(strtotal(11))
    FReqStr = strtotal(11).Split(" ")
    For icount As Integer = 0 To FReqStr.Length - 1
        CheckHF(CInt(FReqStr(icount))).Checked = True
    Next
    For i As Integer = 0 To CInt(txtDim.Text) - 1
        txtF(i).Text = strFrom(i)
        txtT(i).Text = strTo(i)
        txtdimX(i).Text = strDim(i)
        txtTolXUp(i).Text = strTolUp(i)
        txtTolXLo(i).Text = strTolLo(i)
    Next
    If strtotal.Length = 12 Then
        GoTo Vertical_Ignor
    End If
    txtDimV.Text = strtotal(12)
    jigform = CInt(strtotal(13))

    Add_textboxV()

     strFrom = strtotal(14).Split(" ")
    strTo = strtotal(15).Split(" ")
    PosFromV = strtotal(16).Split(" ")
    PosToV = strtotal(17).Split(" ")
    strDim = strtotal(18).Split(" ")
    strTolUp = strtotal(19).Split(" ")
    strTolLo = strtotal(20).Split(" ")
    NodePoV = strtotal(22).Split(" ")
    If strtotal(21).EndsWith("Diameter") = True Then
        ComboBox2.SelectedIndex = 0
    Else
        ComboBox2.SelectedIndex = 1
    End If

    'เลือกประเภทจิกของงานเหลี่ยม
    If jigform = 1 And ComboBox2.SelectedIndex = 1 Then
    ElseIf jigform = 2 And ComboBox2.SelectedIndex = 1 Then
    ElseIf jigform = 3 And ComboBox2.SelectedIndex = 1 Then
        RadioButton23.Checked = True
        RadioButton24.Checked = False
    ElseIf jigform = 4 And ComboBox2.SelectedIndex = 1 Then
        RadioButton23.Checked = False
        RadioButton24.Checked = True
    End If

    'เลือกประเภทจิกของงานกลม
    If jigform = 1 And ComboBox2.SelectedIndex = 0 Then
        RadioButton25.Checked = True
        RadioButton26.Checked = False
        RadioButton27.Checked = False
    ElseIf jigform = 2 And ComboBox2.SelectedIndex = 0 Then
        RadioButton25.Checked = False
        RadioButton26.Checked = True
        RadioButton27.Checked = False
    ElseIf jigform = 3 And ComboBox1.SelectedIndex = 0 Then
        RadioButton25.Checked = False
```

```
            RadioButton26.Checked = False
            RadioButton27.Checked = True
        ElseIf jigform = 4 And ComboBox2.SelectedIndex = 0 Then
            RadioButton25.Checked = False
            RadioButton26.Checked = False
            RadioButton27.Checked = False
        End If
        strtotal(23) = Trim(strtotal(23))
        FReqStrV = strtotal(23).Split(" ")
        For icount As Integer = 0 To FReqStrV.Length - 1
            CheckVF(CInt(FReqStr(icount))).Checked = True
        Next
        For i As Integer = 0 To CInt(txtDimV.Text) - 1
            txtFV(i).Text = strFrom(i)
            txtTV(i).Text = strTo(i)
            txtdimXV(i).Text = strDim(i)
            txtTolXUpV(i).Text = strTolUp(i)
            txtTolXLoV(i).Text = strTolLo(i)
        Next

Vertical_Ignor:
        Form3.Show()

    End Sub
```

New window :

```
    Private Sub NewToolStripMenuItem_Click(sender As Object, e As EventArgs) Handles NewToolStripMenuItem.Click
        Me.Close()
        Form1.Refresh()
        Form1.ComboBox1.SelectedIndex = -1
        Form1.txtDim.Clear()        Form1.txtW.Clear()
        Form1.RadioButton3.Checked = False
        Form1.RadioButton4.Checked = False
        Form1.RadioButton5.Checked = False
        Form1.RadioButton6.Checked = False
        Form1.RadioButton7.Checked = False

        Form1.txtDimV.Clear()
        Form1.txtWV.Clear()

        Form1.RadioButton23.Checked = False
        Form1.RadioButton24.Checked = False
        Form1.RadioButton25.Checked = False
        Form1.RadioButton26.Checked = False
        Form1.RadioButton27.Checked = False
        Form1.GroupY.Visible = False
        Form1.Show()
    End Sub
```

Print :

```
    Private Sub OpenFileToolStripMenuItem_Click(sender As Object, e As EventArgs) Handles
    OpenFileToolStripMenuItem.Click
        If PrintDialog1.ShowDialog = Windows.Forms.DialogResult.OK Then
            PrintDocument1.Print()
        End If
    End Sub

    Private Sub PrintDocument1_PrintPage(sender As Object, e As Printing.PrintPageEventArgs) Handles
    PrintDocument1.PrintPage
        Dim myFont As New Font("Microsoft San Serif", 10)
        e.Graphics.DrawString(Outbox.Text, New Font(myFont, FontStyle.Regular), Brushes.Black, 90, 100)
    End Sub
```

Exit Program :

```
    Private Sub ExitToolStripMenuItem_Click(sender As Object, e As EventArgs) Handles ExitToolStripMenuItem.Click
        ExitProgram()
    End Sub


    Private Sub ExitToolStripMenuItem_Click(sender As Object, e As EventArgs) Handles ExitToolStripMenuItem.Click
        Form3.ExitProgram()
    End Sub
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
Public Sub ExitProgram()
    Dim mbr As MsgBoxResult

    mbr = MsgBox("ออกจากโปรแกรมหรือไม่?",
    MsgBoxStyle.OkCancel + _
    MsgBoxStyle.Question + _
    MsgBoxStyle.ApplicationModal, _
    "ออกจากโปรแกรม")
    If (mbr = MsgBoxResult.Ok) Then
        Me.Close()
        Form1.Close()
    End If
    Application.Exit()
End Sub
```

ภาคผนวก ค

บทความทางวิชาการที่ได้รับการตีพิมพ์เผยแพร่ในระหว่างศึกษา

มหาวิทยาลัยเทคโนโลยีสุรนารี

รายชื่อบทความที่ได้รับการตีพิมพ์เผยแพร่ในระหว่างศึกษา

Phornpan Thongpang, Yongyooth Sermsuti-Anuwat. (2015). **AN APPROACH TO ESTABLISHING THE RELATIONSHIP BETWEEN BUSH POSITION AND COMPONENT PARTS OF A DRILL JIG**. 9th South East Asian Technical University Consortium (SEATUC) Symposium. 27 – 30 July 2015. Suranaree University of Technology. Thailand.

AN APPROACH TO ESTABLISHING THE RELATIONSHIP BETWEEN BUSH POSITION AND COMPONENT PARTS OF A DRILL JIG

Phornpan Thongpang and Yongyooth Sermsuti-anuwat
School of Industrial Engineering, Suranaree University of Technology, Nakhon Ratchasima, Thailand.

ABSTRACT

This paper describes a practical approach basing on the loop diagram and the rooted-tree graph to establishing the tolerance relationship between drill bush position and all dimensions of jig components. The concepts of the two techniques are explained through a 3-part assembly. A simple plate jig with a pin locator and a locating slot is used for illustrating the approach.

1. INTRODUCTION

More than 3 decades since an early attempt to automate the design of workholding device for metal machining operations (Cutkosky et al., 1982), the success is still to be realized. Despite the fact that, at present, a great deal of advance has been achieved in computer technology, both hard- and soft-ware, the computerized fixturing systems developed so far still lack an essential ability to generate functional dimensions on the component parts of the workholding device. This could be because most of these systems neglect to address the fundamental aspect of the device: its function is to repeatedly impart the required dimensions to the workpieces. As a consequence, the existing systems stress only on the production of the pictorial representations or assembly drawings of the jig or fixture in question (Nee et al. (1985), Pham et al. (1990) and Vukelic et al. (2011)), which leaves all the details of determining a bush position or the sizes and position of a set-block to be worked out by tool designers or tool makers at a later stage. Therefore, only partially automatic fixture design can be achieved at this stage.

Sermsuti-anuwat (2009) proposed a tolerance analysis approach to fixture design basing on the loop diagram, a technique normally used in the analysis of part dimensions to be assembled together (Gladman, 1972). The procedure can be used for identifying the functional dimensions on the component parts affecting the functional requirement of a workholding device. Then, Orawan & Sermsuti-anuwat (2009) used the rooted tree-graph (Whybrew, et al., 1990) to re-allocate the tolerances to a new set of dimensions on a machined part when there was a change in the datum for dimensioning. These two techniques, the loop diagram and the rooted tree graph, can be combined and form a systematic procedure for establishing the tolerance relationship between a functional requirement of a workholding device and the dimensions of its components.

This paper is a result of a preliminary study made prior to the development of a computer program for assisting in the determination of a bush position in a drill jig. It describes the basic concepts of the loop diagram and the rooted tree graph and their applications in determining the position of a drill bush, which can be used as a basis for computerized generating dimensions for component parts of a jig or a fixture.

2. CONCEPTS OF LOOP DIAGRAM AND ROOTED TREE GRAPH

Fig. 1 shows the half view of a simple assembly, taken from Lissaman & Martin (1982), consisting of 3 components, numbered 1 to 3. The functional requirement of this assembly is width of a gap between parts 2 and 3 which is represented by Y± y.

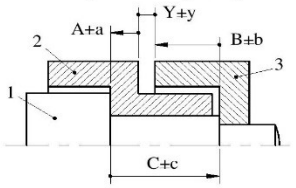


Fig. 1. The loop diagram of an assembly.

By using the method outlined in Sermsuti-anuwat (2009), the functional dimensions controlling the gap width can be identified as A±a, B±b and C±c as shown in Fig. 1.

From this diagram, the following 2 equations can be written, one for basic sizes and the other for tolerances:

$$C - A - B = Y, \text{ for basic sizes;}$$
$$\pm (a + b + c) \leq \pm y, \text{ for tolerances.}$$

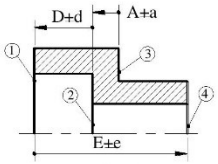
The tolerances of the dimensions on a particular part can be related by a rooted-tree graph. Assuming part numbered 2 with dimensions and tolerances as shown in Fig. 2a, following the steps suggested by Orawan & Sermsuti-anuwat (2009), a tree graph relating all the

length dimensions can be constructed as in Fig. 2b, where a node corresponds to each face dimensioned on the part.

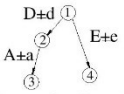
As an example of using this graph, let the Functional dimension A±a be replaced by dimension F±f, between faces 3 and 4, in Fig 2c, from which the tree graph is drawn in Fig. 2d. In order to maintain the original functional dimension (A±a) when the datum for dimensioning changes, the following equation for tolerances must be satisfied:

$$\pm (d + e + f) \leq \pm a$$

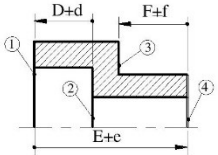
The tolerances on the left hand side of the inequality can be readily identified from the path in the tree whose nodes are the 2 faces of the omitted dimension A±a in Fig. 2d.



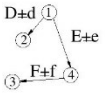
(a) Part 2 with dimensions.



(b) The tree for dimensions in (a).



(c) Part 2 with a modified dimension.

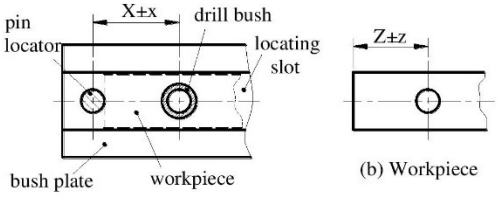


(d) The tree for dimensions in (c).

Fig. 2. An application of rooted-tree graph.

3. AN EXAMPLE

Consider the plan view of a drill jig in Fig.3a, which locates a narrow strip of workpiece by means of a slot. It is required to drill a hole at a distance of Z±z from an end of the workpiece (Fig. 3b). This is in fact the functional requirement of the jig assembly; however, it is not actually integrated with the assembly because a jig must be completely made before it can be used. Therefore, in the context of an assembly, a jig can be viewed as an incomplete assembly, and it will become a complete assembly when the hole is drilled.



(a) Plan view of a plate jig.

(b) Workpiece

Fig. 3. A plate jig and the workpiece.

To determine the bush position, first, the loop diagram must be constructed as shown in Fig.4. Then, 2 equations, for basic sizes and for tolerances, are formulated as follows:

$$A + Z = X, \text{ for basic sizes, and}$$

$$\pm(a + x) \leq \pm z, \text{ for tolerances.}$$

A±a is half the pin size which can be assigned corresponding to the interference fit to the mating hole in the bush plate. Hence, X±x can be calculated from these equations.

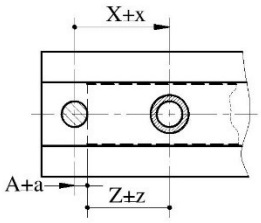
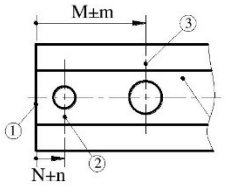


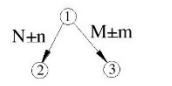
Fig. 4. The loop diagram.

In machining the bush plate, the dimensioning datum could be as shown in Fig. 5a, which requires re-allocation of tolerances, and can be accomplished using the rooted-graph as shown in Fig.5b. Then, the tolerances can be assigned to relevant dimensions according to the following inequality:

$$\pm (m + n) \leq \pm x$$



(a) New dimensions



(b) Rooted-tree graph

Fig. 5. New dimensions and tree.

This simple example demonstrates a procedure of tolerance analysis which can be applied to obtain practically meaningful results despite limited of working experience.

4. CONCLUSION

The rooted-tree graph is an essential technique for allocating tolerance to dimensions a part component, while the loop diagram is for the dimensions on different assembled parts. At present, in the School of Industrial Engineering at Suranaree University of Technology, Thailand, the research on automated fixture design has been conducted with a substantial progress in developing an algorithm to combine the two techniques by basing on the procedures outlined above. It is expected that the success of this ongoing research would bring another vital step closer to the fully automated fixture design.

References

Cutkosky, M.R., Eurokawa, E., and Wright, P.K., Programmable comformable clamps, AUTOFACT Conference Proceedings, SME, Dearbon, Michigan, pp. 11.51-11.58, 1982.

Gladman, C. A., Manual for geometric analysis of engineering designs, Sydney: Australian, trade publications, 1972.

Lissaman, A. J., & Martin, S. J., Principles of engineering production, 2nd ed., ELBS and Hodder and Stoughton, 1982.

Nee, A.Y.C., Bhattacharyya, N., and Poo, A.N., A knowledge-based CAD of jigs and fixtures, Tech. Paper No. TE 85-902, SME, USA, 1985.

Orawan, N., & Yongyooth, S., An optimization technique for tolerance allocation when datum for dimensioning changes, IE Network Conference 2009. (in Thai)

Pham, D.T., Natgh, M.J., and Lazaro, A.S., AUTOFIX-an expert CAD system for jigs and fixtures, Int. J. Mach. Tool. Des. Res., vol. 30, no. 3, pp. 403-411., 1990.

Sermsuti-anuwat, Y., Milling fixture design: a tolerance analysis approach. International Journal of Mechanical Engineering Education, vol. 37, no. 2, pp. 111-117, 2009.

Vukelic, D., Tadic, B., Luzanin, O., Budak, I., Krizan, P., and Hodolic, J., A rule-based system for fixture design, Scientific Research and Essays, vol. 6, no. 27, pp. 5787-5802, 2011

Whybrew, K., Britton, G. A., Robinson, D. F., & Sermsuti-anuwat, Y., A graph-theoretic approach to tolerance charting, The International Journal of Advanced Manufacturing Technology, vol. 5, no. 2, pp. 175-183, 1990.

Phornpan Thongpang is a post-graduate student in the School of Industrial Engineering, Suranaree University of technology, Thailand. The title of her masters thesis is " Computer-aided determination of bush position in drill jig "

Yongyooth Sermsuti-Anuwat received Ph. D. (Mechanical engineering) from The University of Canterbury, New Zealand.

He is an Assistant Professor in the School of Industrial Engineering, Suranaree University of Technology, Thailand.

His research interests are in the area of tolerance technology, process planning and fixture design.

ประวัติผู้เขียน

นางสาวพรพรรณ ทองแพง เกิดเมื่อวันที่ 27 พฤษภาคม พ.ศ. 2534 ภูมิลำเนาอยู่ที่บ้านเลขที่ 30/1 หมู่ 5 ตำบลนาป่า อำเภอเมือง จังหวัดชลบุรี จบการศึกษาระดับประถมศึกษาจาก โรงเรียนพงศ์สิริวิทยา ระดับมัธยมศึกษาตอนต้น และตอนปลายจากโรงเรียนชลบุรี “สุขบท” และเริ่มเข้าศึกษาระดับปริญญาตรีที่ สาขาวิชาวิศวกรรมอุตสาหการ สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี จังหวัดนครราชสีมา เมื่อปี พ.ศ. 2552 และสำเร็จการศึกษาเมื่อปี พ.ศ. 2555

จากนั้นได้เข้าศึกษาต่อในสาขาวิชาวิศวกรรมอุตสาหการ สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี ในปี พ.ศ. 2556 ในขณะที่ศึกษาต่อในระดับปริญญาโท ได้เป็นผู้ช่วยสอนรายวิชาปฏิบัติการกรรมวิธีการผลิต รายวิชาปฏิบัติการวิศวกรรมอุตสาหการขั้นพื้นฐาน รายวิชามาตรวิทยาทางวิศวกรรม และรายวิชาวิศวกรรมเครื่องมือกลของสาขาวิชาวิศวกรรมอุตสาหการ

ผลงานวิจัย : ได้เสนอบทความเข้าร่วมในการประชุมวิชาการ 9th SOUTH EAST ASIAN TECHNICAL UNIVERSITY CONSORTIUM (SEATUC) SYMPOSIUM ประจำปี 2558 เรื่อง AN APPROACH TO ESTABLISHING THE RELATIONSHIP BETWEEN BUSH POSITION AND COMPONENT PARTS OF A DRILL JIG

มหาวิทยาลัยเทคโนโลยีสุรนารี