# การสร้างแผนภูมิความคลาดเคลื่อน : งานขยายต่อเติม

นายคงเดช คำศรี

มหาวิทยาลัยเทคโนโลยีสุรนารี

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมอุตสาหการ
มหาวิทยาลัยเทคโนโลยีสุรนารี
ปีการศึกษา 2560

# TOLERANCE CHARTING : A FURTHER EXTENSION

**Khongdet Khumsri**

มหาวิทยาลัยเทคโนโลยีสุรนารี

**A Thesis Submitted in Partial Fulfillment of the Requirements for the**

**Degree of Master of Engineering in Industrial Engineering**

**Suranaree University of Technology**

**Academic Year 2017**

**การสร้างแผนภูมิความคลาดเคลื่อน : งานขยายต่อเติม**

มหาวิทยาลัยเทคโนโลยีสุรนารี อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

คณะกรรมการสอบวิทยานิพนธ์

(ผศ. ดร.ปวีร์ ศิริรักษ์)
ประธานกรรมการ

(รศ. ดร.พรศิริ จงกล)
กรรมการ (อาจารย์ที่ปรึกษาวิทยานิพนธ์)

(อ. ดร.จงกล ศรีธร)
กรรมการ

(ศ. ดร.สันติ แม้นศิริ)
รองอธิการบดีฝ่ายวิชาการ
และพัฒนาความเป็นสากล

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

มหาวิทยาลัยเทคโนโลยีสุรนารี

คงเดช คำศรี : การสร้างแผนภูมิความคลาดเคลื่อน : งานขยายต่อเติม (TOLERANCE CHARTING : A FURTHER EXTENSION) อาจารย์ที่ปรึกษา : รองศาสตราจารย์ ดร.พรศิริ จงกล, 197 หน้า

ในอุตสาหกรรมการผลิตชิ้นงานด้วยกรรมวิธีตัดโลหะ (machining) การควบคุมกระบวนการผลิตมีส่วนสำคัญอย่างมาก เพื่อให้เป็นไปเป้าหมาย จึงต้องมีการวางแผนและควบคุมคุณภาพของชิ้นงาน โดยใช้แผนภูมิความคลาดเคลื่อน (tolerance chart) ตรวจสอบความเหมาะสมของขั้นตอนการตัดโลหะ ในการตัดโลหะที่ซับซ้อนให้มีประสิทธิผล แผนภูมิความคลาดเคลื่อนจึงเป็นเครื่องมือตรวจสอบความเหมาะสมของขั้นตอนการตัดโลหะให้เป็นไปตามแบบกำหนดก่อนทำการผลิตชิ้นงานจริง

วัตถุประสงค์ของงานวิจัยนี้คือ สร้างแผนภูมิความคลาดเคลื่อนโดยใช้โปรแกรม Visual Basic ในการประมวลผลและวิเคราะห์ความคลาดเคลื่อน ที่เกิดขึ้นในแต่ละขั้นตอนการตัดโลหะ โดยโปรแกรมจะแสดงผลออกมาเป็นขนาด และความคลาดเคลื่อนสะสมของแต่ละขั้นตอนการตัดโลหะ เพิ่มความสามารถของโปรแกรมแผนภูมิฯใน 3 ส่วน ของงานวิจัยที่กำลังดำเนินงานอยู่ที่สาขาวิศวกรรมอุตสาหการ มหาวิทยาลัยเทคโนโลยีสุรนารี คือ

(1) สามารถเปลี่ยนแปลงลำดับขั้นตอนการตัดโลหะ

(2) สามารถเพิ่มหรือลดขั้นตอนการตัดโลหะโดยวิธีปฏิสัมพันธ์กับผู้ใช้

(3) สามารถตรวจสอบความเหมาะสมของแผนภูมิความคลาดเคลื่อนของทั้งสองส่วนแรก

โปรแกรมแผนภูมิความคลาดเคลื่อนที่พัฒนาขึ้นนี้ สามารถให้ผลลัพธ์เป็นที่น่าพอใจไม่ว่าจะเป็นการคำนวณค่าความคลาดเคลื่อนสะสมการปรับเปลี่ยนขั้นตอนในการผลิต หรือ การปรับปรุงเงื่อนไขของการตัดโลหะ ตลอดจนการติดต่อระหว่างโปรแกรมและผู้ใช้ (interaction) อย่างไรก็ตามยังคงมีงานที่ต้องปรับปรุงโปรแกรมในรายละเอียดอยู่อีกหลายอย่าง เพื่อเพิ่มขีดความสามารถของโปรแกรมให้ดียิ่งขึ้นไป

สาขาวิชาวิศวกรรมอุตสาหการ ลายมือชื่อนักศึกษา_____

ปีการศึกษา 2560 ลายมือชื่ออาจารย์ที่ปรึกษา_____

KHONGDET KHUMSRI : TOLERANCE CHARTING : A FURTHER EXTENSION. THESIS ADVISOR : ASSOC. PROF. PORNSIRI JONGKOL, Ph.D., 197 PP.

TOLERANCE CHART/TOLERANCE ANALYSIS/ ROOTED - TREE GRAPH/ MACHINING

This thesis describes the development of a computer package for constructing a tolerance chart for a rotational machined part. It is an extension of the work developed previously, Tolerance Charting with Excel, in the School of Industrial Engineering at Suranaree University of Technology. The package developed, however, is written in Microsoft Visual Basic and all the algorithms have been newly created without the use of Excel. The method of modular programming development has been implemented throughout the package.

All data for charting are interactively input into the program. These include: the sketch of the machined part, a tentative sequence of machining, the face cut and located, and machine used at each machining step, lengthwise dimensions and tolerances of the part blueprint. The program is able to provide the tolerance for each machining cut on the request of the user, and this can also be modified to suit the user's machine capacity.

The calculations of tolerance stacks are based on the concept of Rooted-Tree Graph. However, the program does not construct the Rooted-Tree Graph separately from the machining sequence as normally done in manual charting. Instead, it

recognizes the machining sequence as a tree with each cut face being the end node of a branch which can be traced back to the root of the tree. Using this concept, a path containing the tolerances of the cuts accumulated at any pair of faces can be identified; then, the tolerance stack can be calculated accordingly.

The output from the program is a complete tolerance chart. In such the case that the resultant tolerances in the chart are outside their corresponding blueprint specifications, the user can interactively modify either the machining sequence or the related processing tolerances. This feature has been greatly enhanced beyond the previous Excel version.

This computer program has been verified by comparing its results with those obtained from the manual charting. More than 10 complicated machined parts from the papers and books have been used in testing the program. All the results agree perfectly with the manual calculations.

In comparison to the Excel package, the present program is not only less in the number of source codes but the executing time is also much less. The part with as many as 24 machining cuts took only about 2 seconds of the executing time, while the Excel version, 20 seconds.

มหาวิทยาลัยเทคโนโลยีสุรนารี

School of Industrial Engineering Student's Signatur Khongdot

Academic Year 2017 Advisor's Signature

# กิตติกรรมประกาศ

วิทยานิพนธ์นี้สำเร็จลุล่วงไปได้ด้วยดี ผู้วิจัยขอขอบคุณ บุคคลที่ให้ความความกรุณาและอนุเคราะห์แนะนำช่วยเหลือเป็นอย่างดี ทั้งในด้านวิชาการ และด้านการดำเนินงานวิจัยอาทิเช่น อาจารย์ รองศาสตราจารย์ ดร.พรศิริ จงกล อาจารย์ที่ปรึกษาวิทยานิพนธ์ ที่ให้การสนับสนุนช่วยเหลือ ให้คำปรึกษาที่มีคุณค่าด้วยดีเสมอมา รวมทั้งสละเวลาแก้ไขตรวจทานเนื้อหาวิทยานิพนธ์จนเสร็จสมบูรณ์ และขอขอบคุณอาจารย์ ผู้ช่วยศาสตราจารย์ ดร.ยงยุทธ เสริมสุธีอนุวัฒน์ ที่ให้คำปรึกษาช่วยเหลือแนะนำในด้านต่างๆ และให้กำลังใจแก่ผู้วิจัยมาโดยตลอด

สุดท้ายนี้ขอขอบคุณกรรมการสอบวิทยานิพนธ์ทุกท่าน และผู้ที่มีส่วนเกี่ยวข้องทุกท่านที่มีส่วนร่วมในการให้ข้อมูลเป็นที่ปรึกษาในการทำวิทยานิพนธ์ฉบับนี้จนเสร็จสมบูรณ์ ข้าพเจ้าขอขอบคุณไว้ ณ ที่นี้ด้วย

คงเดช คำศรี

มหาวิทยาลัยเทคโนโลยีสุรนารี

# สารบัญ

หน้า


บทคัดย่อ (ภาษาไทย)..... ก
บทคัดย่อ (ภาษาอังกฤษ)..... ข
กิตติกรรมประกาศ..... ง
สารบัญ..... จ
สารบัญตาราง..... ช
สารบัญรูป..... ฌ
บทที่
1 บทนำ
1.1 ความเป็นมาและความสำคัญของปัญหา.....1
1.2 การสร้างแผนภูมิความคลาดเคลื่อน..... 3
1.2.1 การสร้างแผนภูมิความคลาดเคลื่อนด้วยเทคนิคแกะรอย (Method of Traces)..... 7
1.2.2 การสร้างแผนภูมิความคลาดเคลื่อนโดยใช้ Balance dimensions..... 9
1.2.3 การสร้างแผนภูมิความคลาดเคลื่อนโดยเทคนิคกราฟต้นไม้.....12
1.3 ทบทวนวรรณกรรม..... 20
1.4 วัตถุประสงค์ของงานวิจัย.....21
1.5 สมมติฐานและขอบเขตของการวิจัย.....22
1.6 สรุป.....22
2 โปรแกรมแผนภูมิความคลาดเคลื่อน
2.1 โปรแกรม Visual Basic.....23
2.2 โครงสร้างโปรแกรมแผนภูมิความคลาดเคลื่อน..... 24
2.3 ขั้นตอนการทำงานของโปรแกรมแผนภูมิความคลาดเคลื่อน..... 26
2.4 สรุป.....27


มหาวิทยาลัยเทคโนโลยีสุรนารี

# สารบัญ (ต่อ)

**หน้า**


3 หน้าที่และการทำงานของโมดูล

3.1 กลุ่มโมดูลรับข้อมูล (Input Modules).....28

3.1.1 โมดูลรับภาพสเกตซ์ชิ้นงาน 2 มิติ..... 28

3.1.2 โมดูลกำหนดขนาดตามแบบ Blueprint.....32

3.1.3 โมดูลกำหนดจำนวนขั้นตอนในการตัด.....36

3.1.4 โมดูลรับค่าข้อมูลการสร้างแผนภูมความคลาดเคลื่อน..... 37

3.2 กลุ่มโมดูลประมวลผล (Processing Modules).....41

3.2.1 โมดูลสำหรับประมวลผลรอยตัดของชิ้นงาน..... 41

3.2.2 โมดูลกำหนดรหัสชื่อผิวชิ้นงานตามลำดับขั้นในการตัดโลหะ.....48

3.2.3 โมดูลคำนวณหาขนาดมูลฐานของชิ้นงาน..... 53

3.2.4 โมดูลประมวลผลค่า tolerance ในแต่ละขั้นตอนการผลิต..... 61

3.2.5 โมดูลเปรียบเทียบ Path เพื่อคำนวณค่าความคลาดเคลื่อนของ stock removal และ ค่า basic stock removal..... 64

3.2.6 โมดูลเปรียบเทียบ Path เพื่อคำนวณค่าความคลาดเคลื่อนของ Resultant tolerance..... 73

3.2.7 โมดูลคำนวณหาขนาดมูลฐานของชิ้นงานในแต่ละขั้นตอนการผลิต..... 82

3.2.8 Function CreaPath..... 85

3.2.9 Function PathOne..... 88

3.3 กลุ่มโมดูลแสดงผลลัพธ์ (Output Modules)..... 88

3.4 กลุ่มข้อมูลย้อนกลับ (Feedback Modules)..... 88

3.5 สรุป.....88

4 ตัวอย่างการใช้โปรแกรมแผนภูมิความคลาดเคลื่อน

4.1 ส่วนของการรับภาพสเกตซ์ชิ้นงาน 2 มิติ.....90

4.2 ส่วนของการกำหนดขนาด Blueprint.....94

4.3 ส่วนของการกำหนดขั้นตอนในการตัด.....95

4.4 ส่วนของการรับข้อมูลและแสดงผลลัพธ์ของแผนภูมิความคลาดเคลื่อน.....95


มหาวิทยาลัยเทคโนโลยีสุรนารี

# สารบัญ (ต่อ)

หน้า

4.5 ส่วนปรับปรุงแก้ไขแผนภูมิความคลาดเคลื่อน ..... 102
4.6 สรุป ..... 102
5 สรุปผลงานวิจัย
5.1 สรุป ..... 103
5.2 วิจารณ์โปรแกรม ..... 103
5.3 ข้อจำกัดของโปรแกรม ..... 104
5.4 ข้อเสนอแนะในการพัฒนาต่อไป ..... 104
รายการอ้างอิง ..... 105
ภาคผนวก
ภาคผนวก ก. คู่มือการใช้โปรแกรมแผนภูมิความคลาดเคลื่อน ..... 106
ภาคผนวก ข. ความคลาดเคลื่อน(Tolerance of machined) ..... 134
ภาคผนวก ค. การทดสอบโปรแกรม ..... 142
ภาคผนวก ง. รายงานโปรแกรม (Program Listing) ..... 149
ภาคผนวก จ. บทความวิชาการที่ได้รับการตีพิมพ์เผยแพร่ ..... 194
ประวัติผู้เขียน ..... 205

มหาวิทยาลัยเทคโนโลยีสุรนารี

# สารบัญตาราง

**ตารางที่** **หน้า**


1.1 แสดงสัญลักษณ์ที่ใช้ในแผนภูมิความคลาดเคลื่อน....4
1.2 แสดงขั้นตอนการตัดโลหะของชิ้นงาน รูปที่ 1.2....7
3.1 แสดงข้อมูลของตัวแปรสำคัญในโมดูลรับภาพสเกตซ์ชิ้นงาน 2 มิติ....29
3.2 แสดงข้อมูลของตัวแปรสำคัญในโมดูลกำหนดขนาดตามแบบ Blueprint....32
3.3 แสดงข้อมูลของตัวแปรสำคัญในโมดูลสำหรับประมวลผลรอยตัดของชิ้นงาน....41
3.4 แสดงข้อมูลของตัวแปรสำคัญในโมดูลกำหนดรหัสชื่อผิวชิ้นงานตามลำดับขั้นในการตัดโลหะ....48
3.5 แสดงผลลัพธ์ของผิวตัดชิ้นงาน....48
3.6 แสดงข้อมูลของตัวแปรสำคัญในโมดูลคำนวณหาขนาดมูลฐานของชิ้นงาน....53
3.7 สูตรคำนวณค่า IT....61
3.8 แสดงข้อมูลของตัวแปรสำคัญในโมดูลประมวลผลค่า tolerance ในแต่ละขั้นตอนการผลิต....61
3.9 แสดงข้อมูลของตัวแปรสำคัญในโมดูลเปรียบเทียบ Path เพื่อคำนวณค่าความคลาดเคลื่อนของ stock removal และค่า basic stock removal....64
3.10 แสดงข้อมูลของตัวแปรสำคัญในโมดูลเปรียบเทียบ Path เพื่อคำนวณค่าความคลาดเคลื่อนของ Resultant tolerance....73
3.11 แสดงข้อมูลของตัวแปรสำคัญในโมดูลคำนวณหาขนาดมูลฐานของชิ้นงานในแต่ละขั้นตอนการผลิต....82
3.12 แสดงขั้นตอนการตัดโลหะของรูปชิ้นงานตัวอย่างที่ 3.15 และแสดง Path ในแต่ละขั้นตอนการตัดโลหะ....85
4.1 แสดงขั้นตอนการตัดโลหะของชิ้นงาน รูปที่ 4.1....91
4.2 แสดงขั้นตอนการตัดโลหะของชิ้นงาน รูปที่ 1.2 แบบปรับปรุงใหม่....100


มหาวิทยาลัยเทคโนโลยีสุรนารี

# สารบัญรูป

**รูปที่** **หน้า**


1.1 แสดงลักษณะของแผนภูมิความคลาดเคลื่อน (tolerance chart)ในหนังสืออ้างอิง[1].....2
1.2 แสดงแบบของชิ้นงาน.....5
1.3 แสดงตารางเริ่มต้นสำหรับสร้างแผนภูมิความคลาดเคลื่อน.....5
1.4 แสดงการคำนวณ resultant tolerance โดยเทคนิค Method of Traces..... 8
1.5 ประกอบขั้นตอนการสร้างแผนภูมิความคลาดเคลื่อน.....10
1.6 ไดอะแกรมสำหรับคำนวณความคลาดเคลื่อนสะสมของ balance dimension..... 11
1.7 แสดงแผนภาพต้นไม้.....12
1.8 ประกอบขั้นตอนการสร้างแผนภูมิความคลาดเคลื่อน.....13
1.9 ประกอบขั้นตอนการสร้างแผนภูมิความคลาดเคลื่อน.....14
1.10 ประกอบขั้นตอนการสร้างแผนภูมิความคลาดเคลื่อน.....16
2.1 แสดงลักษณะของโปรแกรม Visual Basic..... 24
2.2 โครงสร้างโปรแกรมแผนภูมิความคลาดเคลื่อน.....24
2.3 แสดงขั้นตอนการทำงานของโปรแกรมแผนภูมิแผนภูมิความคลาดเคลื่อน.....26
3.1 แสดงขั้นตอนการทำงานของแผนภูมิแผนภูมิความคลาดเคลื่อน.....29
3.2 Flow chart แสดงลำดับขั้นตอนการทำงานของโมดูลรับภาพสเกตซ์ชิ้นงาน 2 มิติ.....31
3.3 กำหนดขนาดตามแบบ Blueprint .....33
3.4 Flow chart แสดงลำดับขั้นตอนการทำงานโมดูลสร้างกล่องรับข้อมูล.....34
3.5 Flow chart แสดงลำดับขั้นตอนการรับข้อมูลเกี่ยวกับขนาดตามแบบ Blueprint.....35
3.6 แสดงการกำหนดจำนวนขั้นตอนในการตัดของชิ้นงาน.....36
3.7 แสดงการรับข้อมูลการสร้างแผนภูมความคลาดเคลื่อน.....37
3.8 Flow chart แสดงลำดับขั้นตอนการทำงานรับค่าข้อมูลการสร้าง
แผนภูมิความคลาดเคลื่อน ส่วนที่ 1 ..... 38
3.9 Flow chart แสดงลำดับขั้นตอนการทำงานรับค่าข้อมูลการสร้าง
แผนภูมิความคลาดเคลื่อน ส่วนที่ 2 ..... 40

# สารบัญรูป (ต่อ)

**รูปที่** **หน้า**


3.10 Flow chart แสดงลำดับการทำงานเพื่อนำค่าไปเก็บในตัวแปร S1 ..... 43
3.11 Flow chart แสดงลำดับการทำงานเพื่อนำค่าไปเก็บในตัวแปร S2 ..... 45
3.12 Flow chart แสดงลำดับการทำงานเพื่อนำค่าไปเก็บในตัวแปร NumberOfCut ..... 47
3.13 Flow chart แสดงขั้นตอนการทำงานของโมดูลประมวลผลผิวตัด ..... 50
3.14 Flow chart แสดงขั้นตอนการทำงานของโมดูลประมวลผลผิวอ้างอิง ..... 52
3.15 แสดงแบบของชิ้นงานเพื่อประกอบการคำนวณ ..... 53
3.16 แสดงตัวอย่างการคำนวณส่วนที่ 1 ..... 54
3.17 Flow chart แสดงขั้นตอนการทำงานของโมดูลคำนวณหาขนาดมูลฐานของ ส่วนที่ 1 ..... 55
3.18 แสดงตัวอย่างการคำนวณส่วนที่ 2 ..... 56
3.19 Flow chart แสดงขั้นตอนการทำงานของโมดูลคำนวณหาขนาดมูลฐานของ ส่วนที่ 2 ..... 57
3.20 แสดงตัวอย่างการคำนวณส่วนที่ 3 ..... 58
3.21 Flow chart แสดงขั้นตอนการทำงานของโมดูลคำนวณหาขนาดมูลฐานของ ส่วนที่ 3 ..... 60
3.22 Flow chart แสดงขั้นตอนการทำงานโมดูลประมวลผลค่า tolerance ..... 63
3.23 Flow chart แสดงขั้นตอนการทำงานโมดูลเปรียบเทียบ Path เพื่อคำนวณค่าความ
คลาดเคลื่อนของ stock removal และค่า basic stockremoval ..... 65
3.24 Flow chart แสดงลำดับขั้นตอนการทำงานของส่วนย่อยที่ 1 ..... 67
3.25 Flow chart แสดงลำดับขั้นตอนการทำงานของส่วนย่อยที่ 2 ..... 70
3.26 Flow chart แสดงลำดับขั้นตอนการทำงานของส่วนย่อยที่ 3 ..... 72
3.27 Flow chart แสดงขั้นตอนการทำงานโมดูลเปรียบเทียบ Path เพื่อคำนวณค่าความ
คลาดเคลื่อนของ Resultant tolerance ..... 74
3.28 Flow chart แสดงลำดับขั้นตอนการทำงานของส่วนย่อยที่ 1 ..... 76
3.29 Flow chart แสดงลำดับขั้นตอนการทำงานของส่วนย่อยที่ 2 ..... 79
3.30 Flow chart แสดงลำดับขั้นตอนการทำงานของส่วนย่อยที่ 3 ..... 81
3.31 Flow chart แสดงลำดับขั้นตอนการทำงานโมดูลคำนวณหาขนาดมูลฐานของชิ้นงาน
ในแต่ละขั้นตอนการผลิต ..... 74
3.32 กราฟต้นไม้แสดงความสัมพันธ์ของรอยตัด ..... 86

# สารบัญรูป (ต่อ)

**รูปที่** **หน้า**


3.33 Flow chart แสดงลำดับขั้นตอนการทำงานของ Function CreaPath..... 86
4.1 แสดงชิ้นงานตัวอย่างประกอบการคำนวณแผนภูมิฯ..... 90
4.2 แสดงการสเกตซ์ภาพชิ้นงาน 2 มิติ..... 92
4.3 แสดงผลจำนวนรอยตัดของชิ้นงาน..... 93
4.4 แสดงการรับข้อมูลจากแบบ Blueprint (มิติทุกมิติหน่วยเป็นมิลลิเมตร)..... 94
4.5 แสดงการกำหนดจำนวนขั้นตอนการตัดโลหะจำนวน 26 ขั้นตอน..... 95
4.6 แสดงส่วนของการรับข้อมูลในการตัดโลหะ..... 96
4.7 แสดงการรับข้อมูลในการตัดโลหะ และคำนวณความคลาดเคลื่อนสะสม..... 97
4.8 แสดงผลลัพธ์ของแผนภูมิความคลาดเคลื่อน..... 99
4.9 แสดงผลลัพธ์ของแผนภูมิความคลาดเคลื่อน..... 101


มหาวิทยาลัยเทคโนโลยีสุรนารี

# บทที่ 1
# บทนำ

## 1.1 ความเป็นมาและความสำคัญของปัญหา

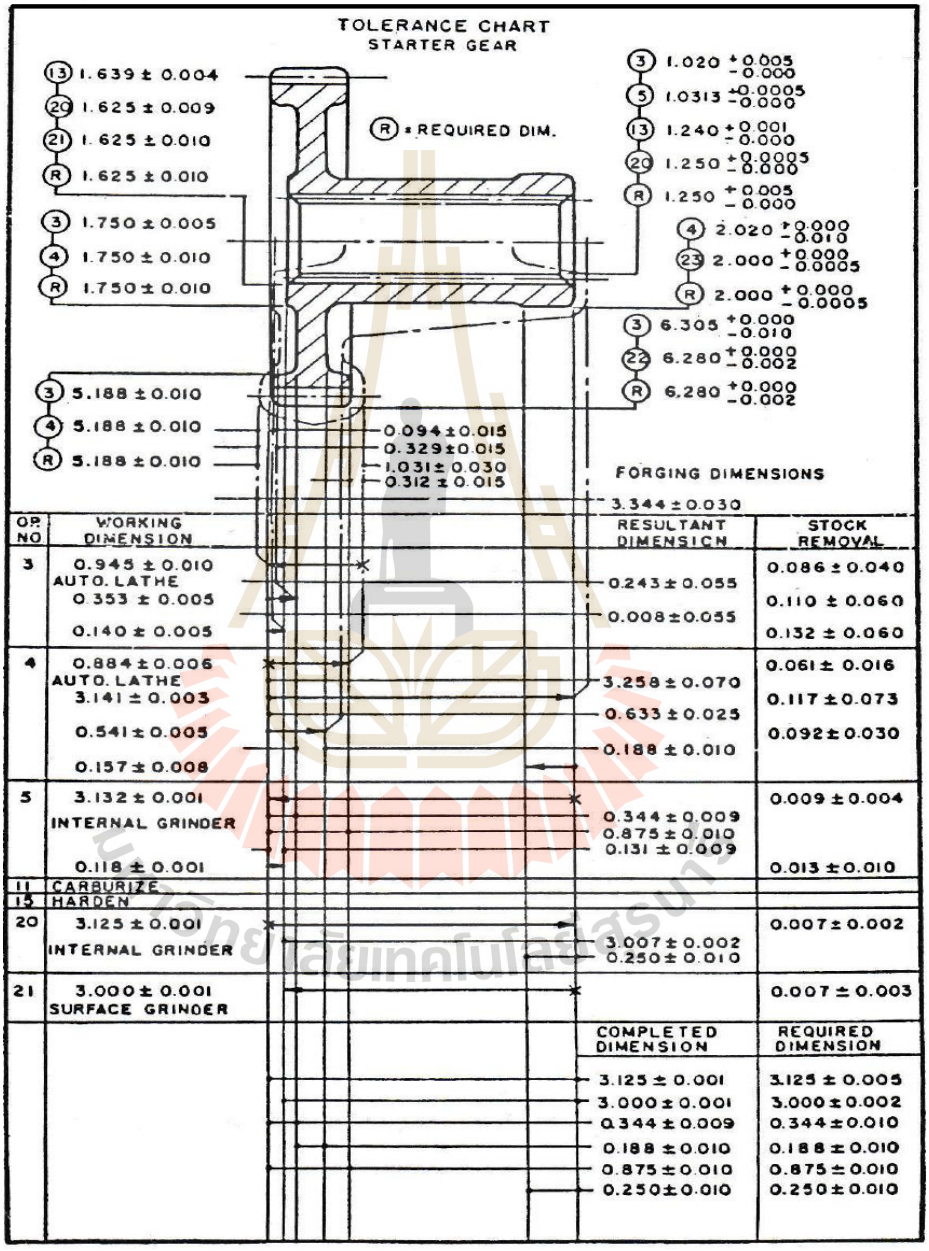
ในปัจจุบันอุตสาหกรรมการผลิตสินค้าด้วยกรรมวิธีตัดโลหะ (metal machining industry) ทวีการแข่งขันอย่างรุนแรง ทางด้านคุณภาพและราคาของสินค้า ซึ่งทั้งสองส่วนนี้มีผลกระทบโดยตรงต่อความสำเร็จของการลงทุน ทั้งนี้เพราะความเสียหายใดๆที่เกิดจากการผลิตจะเพิ่มต้นทุนของสินค้าและลดกำไร ด้วยเหตุนี้เองการวางแผนกระบวนการผลิตจึงมีบทบาทที่สำคัญต่อการผลิตสินค้าที่เป็นชิ้นงานโลหะอย่างมาก

แต่งานวางแผนเช่นนี้ต้องอาศัยผู้ที่มีความรู้และความชำนาญสูง จึงจะได้แผนกระบวนการผลิต (process plan) ที่มีคุณภาพ ซึ่งปัจจุบันบุคลากรชนิดนี้จะลดน้อยลงไปเรื่อยๆ ข้อเท็จจริงนี้เป็นที่ตระหนักกันดีในอุตสาหกรรมการตัดโลหะตั้งแต่ปลายศตวรรษที่ 20 ซึ่งมีการนำเอาความรู้ของผู้ชำนาญการมาใช้ในรูปของ ระบบผู้ชำนาญการ (Expert system) ต่างๆอย่างแพร่หลายตั้งแต่ปี 1980 ที่ผ่านมา แต่เป็นที่น่าเสียดายที่ระบบเหล่านี้ส่วนใหญ่ หรือเกือบทั้งหมดขาดกลไก หรือเครื่องมือตรวจสอบแผนกระบวนการผลิต (process plan) ว่าสามารถจะให้ผลตามข้อกำหนดหรือไม่

อันที่จริงวิศวกรเครื่องมือ (tool engineers) มีเครื่องมือสำหรับตรวจสอบประสิทธิผลของแผนกระบวนการผลิตเรียกว่าแผนภูมิความคลาดเคลื่อน (tolerance chart) และปรากฏหลักฐาน [1] ว่ามีการใช้กันตั้งแต่ก่อนปี 1950 เพราะปรากฏในหนังสือของ Doyle (รูปที่1.1) กล่าวได้ว่าเทคนิคชนิดนี้เป็นเทคนิคของการวิเคราะห์ความคลาดเคลื่อน (tolerance analysis) ที่เป็นระบบและมีการประยุกต์ใช้ในหลากหลายวัตถุประสงค์

แผนภูมิความคลาดเคลื่อน (tolerance charts) เป็นแผนภาพที่แสดงขั้นตอนการตัดโลหะ ที่มีลำดับขั้นตอนอย่างเป็นระบบ โดยแผนภาพจะแสดงให้เห็นถึงการเปลี่ยนแปลงของขนาด (dimension) ส่วนของโลหะที่ถูกตัดออกไปในแต่ละขั้นตอนการตัดและความคลาดเคลื่อนของมิติความคลาดเคลื่อนของมิติเป็นผลมาจากการตัดโลหะ, และความคลาดเคลื่อนสะสมตั้งแต่การตัดโลหะครั้งแรกจนสิ้นสุดการตัดโลหะครั้งสุดท้าย ส่วนมากแผนภูมิความคลาดเคลื่อนใช้กับงานที่

ซับซ้อน เพื่อให้สามารถตรวจสอบขั้นตอนการตัดโลหะว่ามีความเหมาะสมหรือไม่ และต้องควบคุมความคลาดเคลื่อนสะสมของกระบวนการตัดโลหะอย่างเคร่งครัดเพื่อให้ขนาดของชิ้นงานสำเร็จตามแบบกำหนด




รูปที่ 1.1 แสดงลักษณะของแผนภูมิความคลาดเคลื่อน (tolerance chart)ในหนังสืออ้างอิง Doyle

เนื่องจากการสร้างแผนภูมิฯต้องอาศัยพื้นฐานความรู้ทางปฏิบัติเกี่ยวกับความคลาดเคลื่อนสะสม และรูปแบบของแผนภูมิฯอาจจะแตกต่างกันตามวัตถุประสงค์ของการใช้งานทำให้เข้าใจหลักการของแผนภูมิฯได้ยาก นอกจากนั้นการคำนวณความคลาดเคลื่อนในแผนภูมิฯยังมีโอกาสผิดพลาดได้ง่าย ด้วยเหตุนี้ จึงมีการใช้คอมพิวเตอร์ช่วยสร้างแผนภูมิความคลาดเคลื่อน โดยเริ่มจากปี 1984 Ahaluwalia and Karolin ได้พัฒนาโปรแกรมคอมพิวเตอร์สำหรับสร้างแผนภูมิความคลาดเคลื่อน และถือได้ว่าเป็นจุดเริ่มต้นของการใช้คอมพิวเตอร์เพื่อคำนวณและวิเคราะห์ความคลาดเคลื่อนในแนวทางของแผนภูมิความคลาดเคลื่อน

สำหรับในประเทศไทย การวิเคราะห์ความคลาดเคลื่อน และการใช้แผนภูมิความคลาดเคลื่อนยังไม่เป็นที่รู้จักแพร่หลายเท่าที่ควร งานวิจัยทางด้านนี้ยังอยู่ในขอบเขตจำกัด ในปี 2016 ชลาลัย ได้สร้างแผนภูมิความคลาดเคลื่อนขึ้นด้วยโปรแกรมตารางคำนวณ ซึ่งโปรแกรมดังกล่าว สามารถแสดงผลลัพธ์เป็นแผนภูมิความคลาดเคลื่อนและสามารถแก้ไขเปลี่ยนแปลงความคาดเคลื่อนที่เกี่ยวข้องได้ ส่วนดีของโปรแกรมตารางคำนวณคือ การรับค่า Input และแสดงผล Output นั้นสามารถอาศัย graphics ของโปรแกรมตารางคำนวณแสดงผลได้โดยง่าย แต่ยังไม่ได้แสดงความสามารถของแผนภูมิความคาดเคลื่อนอย่างสมบูรณ์ได้ เนื่องจากเซลล์ในตารางคำนวณนั้นถูกจำกัดด้วยฟังก์ชั่น ทำให้ความสามารถในการเปลี่ยนแปลงขั้นตอนในการตัดโลหะและการเพิ่มหรือลดงานตัดโลหะเป็นไปได้ยาก ซึ่งทั้งสองส่วนนี้เป็นข้อจำกัดของโปรแกรมแผนภูมิความคลาดเคลื่อนบนโปรแกรมตารางคำนวณ

วิทยานิพนธ์เล่มนี้รายงานผลงานวิจัย ซึ่งเป็นส่วนต่อเติมงานของชลาลัย แต่ไม่ได้อาศัยโปรแกรมตารางคำนวณ (spread sheet) โดยมุ่งเน้นพัฒนาความสามารถในการเปลี่ยนแปลงของขั้นตอนการตัดโลหะให้สมบูรณ์ยิ่งขึ้น

## 1.2 การสร้างแผนภูมิความคลาดเคลื่อน

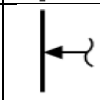
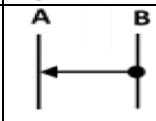
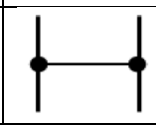
การสร้างแผนภูมิความคลาดเคลื่อนนอกจากวิธีของ Doyle แล้ว ยังมีวิธีที่ได้รับความนิยมอีกหลายวิธี ในช่วงต้นศตวรรษที่ 20 งานวิจัยที่เด่นชัดของ Gadzala(1959) โดย Method of traces และของ O.R.Wade(1967) โดยใช้ Balance dimensions นอกจากนั้นยังมีวิธีของนักวิจัยอื่นๆอีก เช่น ผลงานของ Davie(1988) ใช้กราฟในการช่วยวิเคราะห์ความคลาดเคลื่อน, Iran, Mittal and Lehtihet (1989) ใช้กราฟเช่นกันในรูปแบบ Path tracing และล่าสุด Whybrew และคณะ (1990) ได้เสนอวิธีประยุกต์ใช้กราฟต้นไม้ (rooted-tree graph) ช่วยในการสร้างแผนภูมิความคลาดเคลื่อน

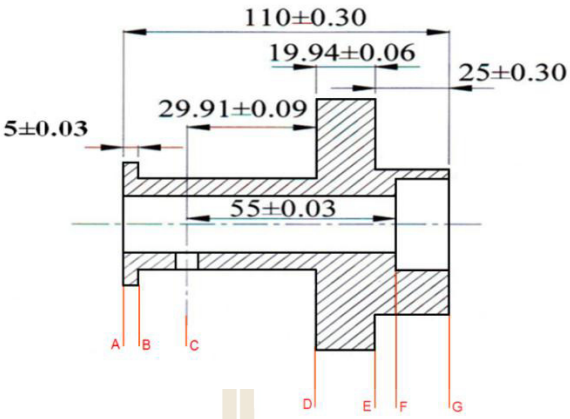
ในที่นี้จะกล่าวถึงรายละเอียดโดยสังเขปเพียง 3 วิธี คือ การใช้ Method of traces, การใช้ Balance dimensions และการใช้กราฟต้นไม้ (rooted-tree graph) เพราะทั้ง 3 วิธีนี้ มีรายละเอียดมากเพียงพอที่สามารถทำความเข้าใจได้ง่ายกว่าวิธีอื่นๆ

คำอธิบายวิธีการสร้างแผนภูมิความคลาดเคลื่อนในทั้ง 3 วิธีจะอาศัยชิ้นงานในรูปที่ 1.2 เป็นหลักในการอธิบาย ถึงแม้ว่าจะเป็นชิ้นงานที่ไม่ซับซ้อน และมีงานที่เกี่ยวข้องเพียง กลึงปลอก, กลึงคว้าน, กัดโลหะ และเจาะรู เท่านั้น ซึ่งไม่จำเป็นต้องใช้แผนภูมิความคลาดเคลื่อนวิเคราะห์ แต่ที่นำมาใช้ประกอบการอธิบายก็เพื่อให้ง่ายต่อการทำความเข้าใจกลไกของเทคนิคแต่ละอย่าง

สำหรับรูปที่ 1.3 แสดงแบบฟอร์มของแผนภูมิความคลาดเคลื่อนทั่วๆไป ซึ่งประกอบด้วย (1) พื้นที่ตอนบนสำหรับเขียนรูปชิ้นงาน; (2) รายละเอียดของการตัดอยู่ทางด้านซ้าย; (3) ด้านขวามือของตารางแสดงผลการตัด; (4) ตรงกลางเป็นพื้นที่สำหรับสัญลักษณ์แสดงผิวที่ถูกตัดแต่ละครั้ง และผิวที่ใช้ในการกำหนดตำแหน่งงาน หรือตำแหน่งอ้างอิงในการวัดดังตารางที่ 1.1; และ (5) ตอนล่างของแผนภูมิฯเป็นขนาดของชิ้นงานตามข้อกำหนดของแบบ และผลของการตัดโลหะตามขั้นตอนในแผนภูมิฯ

ตารางที่ 1.1 แสดงสัญลักษณ์ที่ใช้ในแผนภูมิความคลาดเคลื่อน

| ลำดับ | สัญลักษณ์ | ความหมาย |
|---|---|---|
| 1 | | สัญลักษณ์ระบุระนาบแนวดิ่งของชิ้นงาน |
| 2 |  | สัญลักษณ์ระบุผิวที่ถูกปาดหน้าครั้งแรก หรือระบุผิวกำหนดตำแหน่งเริ่มต้น |
| 3 | | สัญลักษณ์ระบุผิวที่ถูกตัด |
| 4 | | สัญลักษณ์ระบุผิวอ้างอิง หรือจุดเริ่มต้นของการวัดขนาด |
| 5 |  | สัญลักษณ์แสดงการตัดลูกศรชี้ไปยังผิว หรือระนาบ A ซึ่งเป็นผิวที่ถูกตัด และจุดวงกลมสีดำระบุระนาบอ้างอิงของการวัดขนาดความยาว A-B |
| 6 |  | สัญลักษณ์ระบุ Balance dimension คือ dimension ที่เกิดจากการบวก หรือระหว่าง machining cut กับ balance dimension |




รูปที่ 1.2 แสดงแบบของชิ้นงาน

| Op. | Face | M/C | Working Dim | | | Balance Dim. | | Stock rmoval | |
|---|---|---|---|---|---|---|---|---|---|
| NO. | cut | used | Basic | ±Tol./2 | | Basic | ±Tol./2 | Basic | ±Tol./2 |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | Blue print Dim. | | | Resultants | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |

รูปที่ 1.3 แสดงตารางเริ่มต้นสำหรับสร้างแผนภูมิความคลาดเคลื่อน

การสร้างแผนภูมิความคลาดเคลื่อนไม่ว่าจะใช้วิธีใดก็ตาม มีหลักการสร้างพื้นฐานเหมือนกัน ส่วนที่แตกต่างกันคือ วิธีการคำนวณความคลาดเคลื่อนสะสม ซึ่งทำให้รูปแบบของแผนภูมิ หรือองค์ประกอบขอบแผนภูมิฯแตกต่างกันไป ถ้าใช้ผิวของชิ้นงานในรูป 1.2 และขั้นตอนการตัดโลหะในตารางที่ 1.2 เป็นตัวอย่างของการสร้างแผนภูมิฯ เราสามารถดำเนินขั้นตอนในการสร้างได้ดังนี้:

(1) สเกตซ์ภาพชิ้นงานตอนบนของแผนภูมิความคลาดเคลื่อน

(2) ลากเส้นดิ่ง A, B, C, D, E, F และ G จากผิวรอยตัดของชิ้นงานในแนวดิ่ง

(3) เขียนสัญลักษณ์การตัดผิว (machining cut) จากจุดอ้างอิงไปยังผิวที่ถูกตัด (หัวลูกศร) ตามขั้นตอนการตัดที่วางแผนไว้

(4) เติมหมายเลขของงานแต่ละลำดับขั้นตอนการตัด (cut) และใส่เครื่องจักรหรือกระบวนการผลิตที่ใช้ในแต่ละขั้นตอน ลงในแผนภูมิความคลาดเคลื่อน

(5) ใส่ข้อกำหนดมิติมูลฐานความยาวตามแบบลงในสดมภ์ของ BLUEPRINT

(6) ใส่ค่าความคลาดเคลื่อนให้กับการตัดของแต่ละ cut ลงในสดมภ์ของ WORKING DIM.

(7) คำนวณค่าความคลาดเคลื่อนให้กับส่วนของโลหะที่ถูกตัดออกไป (stock removal) แล้วใส่ค่าลงในสดมภ์ STOCK REM.

(8) กำหนดขนาดของ stock removal น้อยที่สุดที่ยอมให้เกิดขึ้นกับรอยตัดแต่ละรอย ขึ้นอยู่กับชนิดของเครื่องจักรแต่ละ operation เช่นงานกลึงกำหนดให้มากกว่างานเจียรนัย เป็นต้น(9) หาขนาดมูลฐาน (basic size) ที่เกิดจากการตัดแต่ละ cut (คือค่า BASIC ของ WORKING DIM.) ซึ่งมีหลักการอยู่ว่าให้พิจารณาจากด้านล่างของแผนภูมิขึ้นไปทางด้านบน โดยอาศัยกฎเกณฑ์อย่างใดอย่างหนึ่งดังนี้

(a) ยกมาจาก blueprint dimensions ถ้าหากเป็น working dimension ของ cut ที่ทำให้เกิด blueprint dimension

(b) คำนวณจาก blueprint dimension

(c) คำนวณจาก basic working dimension ที่ทราบค่า

(d) คำนวณจากมิติที่เป็นผลจากการบวกหรือลบ blueprint dimensions

มหาวิทยาลัยเทคโนโลยีสุรนารี

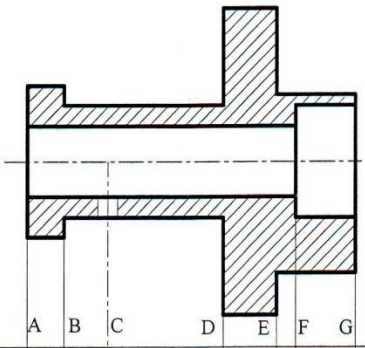
ตารางที่ 1.2 แสดงขั้นตอนการตัดโลหะของชิ้นงาน รูปที่ 1.2

| | | | | |
|---|---|---|---|---|
| 1 | Operation 00 | กลึงปาดหน้าผิวอ้างอิง | | ตัดผิว G |
| 2 | Operation 10 | กลึงคว้านปากรูขนาด Ø40 mm. | อ้างอิงผิว G | ตัดผิว F |
| 3 | Operation 20 | กลึงปลอกขนาด Ø110 mm. | อ้างอิงผิว G | ตัดผิว D |
| 4 | Operation 30 | กลึงปลอกขนาด Ø75 mm. | อ้างอิงผิว G | ตัดผิว E |
| 5 | Operation 40 | กลึงเซาะร่อง ขนาด Ø40 mm. | อ้างอิงผิว D | ตัดผิว B |
| 6 | Operation 50 | กลึงปลอกผิว Ø60 mm. | อ้างอิงผิว B | ตัดผิว A |
| 7 | Operation 60 | เจาะรู Ø10 mm. | อ้างอิงผิว F | ตัดผิว C |
| 8 | Operation 70 | กลึงปาดหน้า | อ้างอิงผิว F | ตัดผิว G |
| 9 | Operation 80 | กลึงปาดหน้า | อ้างอิงผิว G | ตัดผิว E |
| 10 | Operation 90 | กลึงปาดหน้า | อ้างอิงผิว G | ตัดผิว D |
| 11 | Operation 100 | กลึงปาดหน้า | อ้างอิงผิว D | ตัดผิว B |
| 12 | Operation 110 | กลึงปาดหน้า | อ้างอิงผิว G | ตัดผิว A |

### 1.2.1 การสร้างแผนภูมิความคลาดเคลื่อนด้วยเทคนิคแกะรอย (Method of Traces)

ในปี 1959 Gadzala ได้เสนอเทคนิค Method of Traces เพื่อคำนวณหาความคลาดเคลื่อนสะสมของมิติระหว่างผิวตัด 2 ผิว โดยมีวิธีดังนี้ (1) พิจารณาผิวสองผิวว่าหากลากเส้นในแนวดิ่งจากผิวทั้งสองพร้อมๆกันขึ้นไปด้านบน ถ้าหากเส้นใดพบหัวลูกศรก่อนให้ลากเส้นแนวระดับจากหัวลูกศรไปยังปลายลูกศร (2) ดำเนินการข้อ (1) จากเส้นดิ่งที่ลากไปยังปลายลูกศรและเส้นดิ่งเดิม (3) ถ้าหากเส้นในแนวดิ่งทั้งสองถูกลากพร้อมๆกันขึ้นไปบนผิวเดียวกันแสดงว่าผิวนั้นเป็นจุดสิ้นสุดของวงรอบกลุ่มรอยตัดที่มีผลต่อความคลาดเคลื่อนระหว่างผิวสองผิวที่พิจารณา

ตัวอย่างเช่น ต้องการหาความคลาดเคลื่อนสะสม (resultant tolerance) สำหรับมิติ 19.94 mm.ซึ่งเป็นผลลัพธ์จากการตัดโลหะตามขั้นตอนในรูปที่ 1.4 ใช้วิธีลากเส้นดังนี้:-




| Op. no. | Face cut | M/C used | Working Dim. Basic | Working Dim. ±Tol./2 | Balance Dim. Basic | Balance Dim. ±Tol./2 | Stock removal Basic | Stock removal ±Tol./2 |
|---|---|---|---|---|---|---|---|---|
| 10 | | | 20.15 | 0.03 | | | Solid | cut |
| 20 | | | 45.54 | 0.03 | | | Solid | cut |
| 30 | | | 25 | 0.03 | | | Solid | cut |
| 40 | | | 59.96 | 0.10 | | | Solid | cut |
| 50 | | | 5.10 | 0.03 | | | Solid | cut |
| 60 | | | 55 | 0.03 | | | Solid | cut |
| 70 | | | 19.85 | 0.005 | | | 0.30 | 0.035 |
| 80 | | | 25 | 0.01 | | | 0.30 | 0.075 |
| 90 | | | 44.94 | 0.01 | | | 0.30 | 0.075 |
| 100 | | | 60.06 | 0.01 | | | 0.40 | 0.185 |
| 110 | | | 110 | 0.01 | | | 0.50 | 0.205 |
| | | | Blue print Dim. | | Resultants | | | |
| | | | 19.94 | 0.06 | 19.94 | 0.02 | | |
| | | | 29.91 | 0.09 | 29.91 | | | |
| | | | 25 | 0.30 | 25 | | | |
| | | | 55 | 0.03 | 55 | | | |
| | | | 5 | 0.03 | 5 | | | |
| | | | 110 | 0.30 | 110 | | | |

รูปที่ 1.4 แสดงการคำนวณ resultant tolerance โดยเทคนิค Method of Traces

(1) จากปลายทั้งสองของมิติ 19.94 ลากเส้นในแนวดิ่งตามผิว D และ E ขึ้นไปทางด้านบนของแผนภูมิฯในกรณีนี้เส้นที่ลากตามผิว D พบหัวลูกศรก่อนคือหัวลูกศรก่อน คือ หัวลูกศร cut ที่ 9 จึงลากเส้นไปตามแนวระดับจนถึงปลายลูกศรที่ผิว G

(2) จากผิว G และเส้นดิ่งเดิม ( เส้นที่ลากจากผิว E) ลากเส้นในแนวดิ่งขึ้นไปอีก ตอนนี้จะเห็นได้ว่าเส้นที่ลากตามผิว E พบหัวลูกศรก่อนดังนั้นจึงลากเส้นในแนวระดับไปยังปลายลูกศรที่พบ คือผิว G ซึ่งเป็น cut ที่ 8

(3) ถ้าหากว่าเหนือ cut ที่8 และ 9 ตรงจุดที่รอยลากเส้นสิ้นสุดยังมีหัวลูกศรอยู่ทางด้านบนอีกก็ให้ลากเป็นเส้นแนวดิ่งขึ้นไปหาหัวลูกศรนั้น ซึ่งจะเห็นได้ว่าเส้นดิ่ง 2 เส้นจะมา

รวมกันที่ผิว G และถูกลากขึ้นไปพร้อมๆกัน แสดงว่าผิว G เป็นผิวสิ้นสุดวงรอบของกลุ่มรอยตัดที่ทำให้เกิดความคลาดเคลื่อนสะสมของมิติ 19.94 เพราะฉะนั้นมิตินี้จึงมีความสัมพันธ์กับ cut ที่ 8 และ cut ที่9 และเมื่อนำค่า tolerance ของ cut ทั้งสองมารวมกันได้ดังนี้ (0.01+0.01= ± 0.02) จะได้เป็น tolerance ของมิติ 19.94

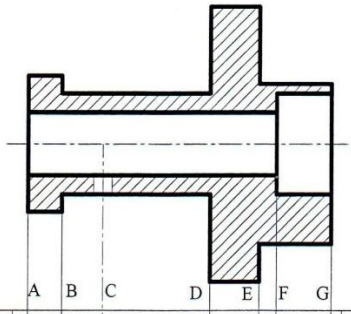
การคำนวณอื่นๆ เช่น ความคลาดเคลื่อนของ Stock Removal หรือการคำนวณหาขนาดมูลฐานของ Working Dimension ก็อาศัยวงรอบรอยตัดที่เกิดจาก Method of Traces นี้ เช่นกัน

จะสังเกตเห็นได้ว่าจากรูป 1.4 มีบางสดมภ์ของแบบฟอร์มไม่ได้ถูกใช้งาน เช่น สดมภ์ Face cut และ Balance Dimension ทั้งนี้เพราะ วิธีการคำนวณความคลาดเคลื่อนสะสมแบบนี้ไม่ต้องอาศัยสดมภ์ดังกล่าวในการคำนวณ

### 1.2.2 การสร้างแผนภูมิความคลาดเคลื่อนโดยใช้ Balance dimensions

O.R. Wade (1976) ใช้ Balance Dimensions ช่วยในการคำนวณหาขนาดของมิติในแต่ละขั้นตอนการผลิต , ใช้ในการคำนวณหาค่าความคลาดเคลื่อนของความคลาดเคลื่อนสะสม หรือคำนวณค่าความคลาดเคลื่อนของผลลัพธ์ ในรูปที่ 1.10 จะเห็นสัญลักษณ์ dot - dot เป็นมิติที่สร้างจำลองขึ้นมาใหม่เรียกว่า Balance Dimensions (มิติที่เกิดจากการตัดโลหะที่ผ่านมา)

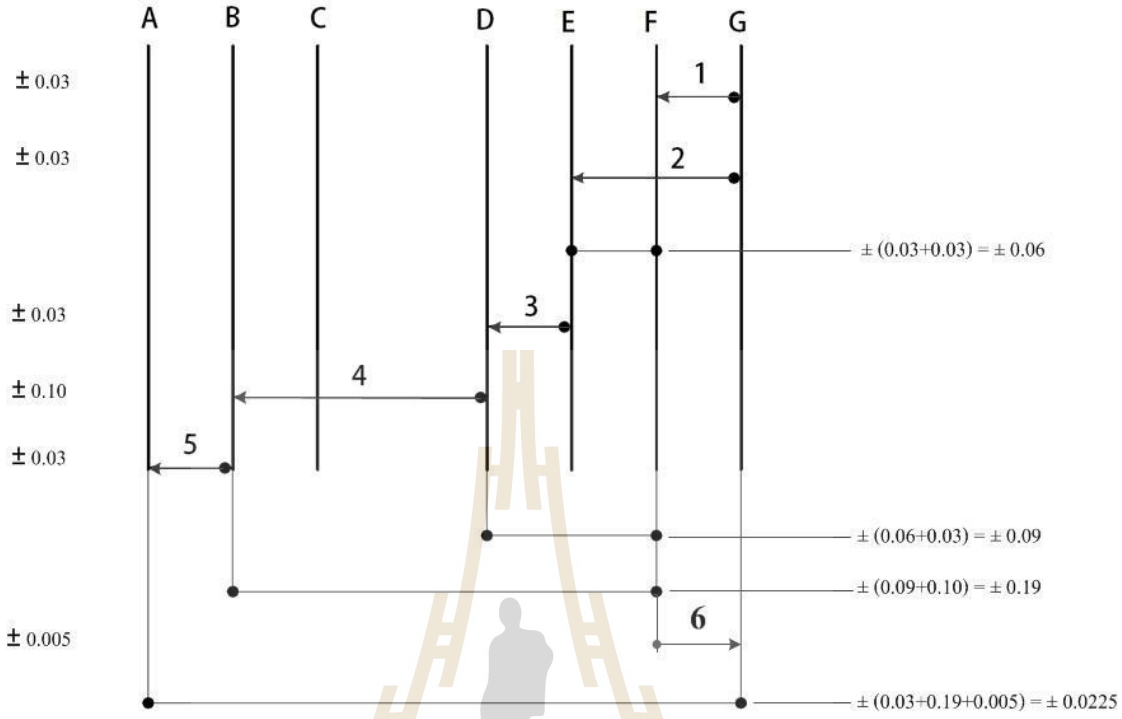
มหาวิทยาลัยเทคโนโลยีสุรนารี




| Op. no. | Face cut | M/C used | Working Dim. Basic | Working Dim. ±Tol./2 | Balance Dim. Basic | Balance Dim. ±Tol./2 | Stock removal Basic | Stock removal ±Tol./2 |
|---|---|---|---|---|---|---|---|---|
| 10 | | | 20.15 | 0.03 | | | Solid cut | |
| 20 | | | 25.30 | 0.03 | | | Solid cut | |
| | | | | | 5.15 | 0.06 | | |
| 30 | | | 20.34 | 0.03 | | | Solid cut | |
| 40 | | | 59.86 | 0.10 | | | Solid cut | |
| | | | | | 25.49 | 0.09 | | |
| 50 | | | 5 | 0.03 | | | Solid cut | |
| 60 | | | 55 | 0.03 | | | Solid cut | |
| 70 | | | 19.85 | 0.005 | | | 0.30 | 0.035 |
| | | | | | 74.85 | 0.035 | | |
| | | | | | 45.34 | 0.095 | | |
| | | | | | 104.40 | 0.195 | | |
| 80 | | | 44.94 | 0.01 | | | 0.40 | 0.105 |
| | | | | | 59.46 | 0.205 | | |
| 90 | | | 60.06 | 0.01 | | | 0.60 | 0.215 |
| | | | | | 105 | 0.02 | | |
| | | | | | 110.60 | 0.225 | | |
| 100 | | | 110 | 0.01 | | | 0.60 | 0.235 |
| | | | Blue print Dim. | | Resultants | | | |
| | | | 19.94 | 0.06 | 19.94 | 0.075 | | |
| | | | 29.91 | 0.09 | 29.91 | 0.045 | | |
| | | | 25 | 0.30 | 25 | 0.065 | | |
| | | | 55 | 0.03 | 55 | 0.03 | | |
| | | | 5 | 0.03 | 5 | 0.03 | | |
| | | | 110 | 0.30 | 110 | 0.01 | | |

รูปที่ 1.5 ประกอบขั้นตอนการสร้างแผนภูมิความคลาดเคลื่อน

ตัวอย่างของการคำนวณความคลาดเคลื่อนสะสมระหว่างผิว A และ G ในรูปที่ 1.5 ซึ่งเป็นผลจากการตัดโลหะ 9 ครั้งที่ผ่านมาแต่จะมีเพียงบาง cut เท่านั้นที่มีผลต่อความคลาดเคลื่อนของระยะ A – G นั่นคือ cut ที่ 5 และ 7 ซึ่งตัดปลายผิว A และ G ล่าสุด และ cut อื่นๆที่มีผลก็คือ cut ที่ทำให้ตำแหน่งอ้างอิงของ cut ที่ 5 และ 7 เปลี่ยนแปลงได้แก่ cut ที่ 1, 2, 3 และ 4 เพราะฉะนั้นเราสามารถเขียน diagram เพื่อรวบรวม cut เหล่านี้ได้ดังรูปที่ 1.6 ซึ่งสามารถอธิบายได้ดังนี้:-




รูปที่ 1.6 ไดอะแกรมสำหรับคำนวณความคลาดเคลื่อนสะสมของ balance dimension

ความคลาดเคลื่อนสะสมระหว่างผิว A และ G ที่พิจารณาเกิดจากผลรวมของความคลาดเคลื่อนของ cut ที่5, cut ที่7 และความคลาดเคลื่อนสะสมระหว่างผิว B กับ F ที่ถูกตัดมาก่อน หรือพูดว่า balance dimension B – F; ซึ่งความคลาดเคลื่อนสะสมที่ balance dimension นี้ก็คือผลรวมของความคลาดเคลื่อนของ cut 4 กับความคลาดเคลื่อนสะสมที่ balance dimension D – F; และความคลาดเคลื่อนสะสมที่ balance dimension D – F ก็คือผลรวมระหว่างความคลาดเคลื่อนสะสมที่ balance dimension E – F กับความคลาดเคลื่อนของ cut ที่ 3; ซึ่งความคลาดเคลื่อนสะสมที่ balance dimension E – F จะเท่ากับผลรวมของความคลาดเคลื่อนของ cut ที่ 1 และ 2

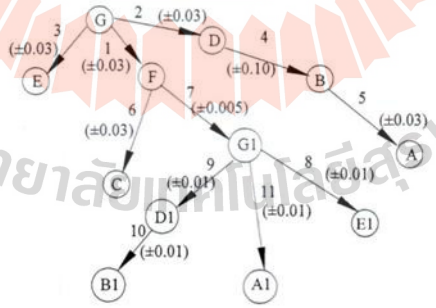
วิธีการสร้างแผนภูมิฯ โดยใช้ balance dimension จำเป็นจะต้องอาศัยไดอะแกรมชนิดนี้ (รูปที่1.6) ประกอบการคำนวณความคลาดเคลื่อนสะสมของ STOCK REM. และใช้ในการหา basic size ของแต่ละ cut

### 1.2.3 การสร้างแผนภูมิความคลาดเคลื่อนโดยเทคนิคกราฟต้นไม้

ในรูปที่ 1.7 แสดงแผนภาพต้นไม้เพื่อใช้ประกอบการคำนวณด้วยมือ ซึ่งรายละเอียดของวิธีการสร้างแผนภาพชนิดนี้ปรากฏอยู่ในบทความของ Whybrew และคณะ (1990) แล้ว ในที่นี้จะสรุปวิธีการดังกล่าวและวิธีใช้งานเพียงสั้นๆดังนี้:

เราสามารถเขียนแผนภาพต้นไม้จากขั้นตอนของการตัดโลหะ สำหรับชิ้นงานตัวอย่างใช้ตาราง 1.2 โดยเริ่มต้นด้วยการเขียนโหนด (node) แรกของผิวที่ถูกตัดครั้งแรก; จากนั้นลากลูกศร อ้างอิงจากผิวแรกนี้ ไปยังผิวที่ถูกตัด; ถ้าผิวที่ถูกตัดใหม่ เป็นผิวอ้างอิงของการตัดต่อๆไป, การลากเส้นลูกศร จะดำเนินต่อไป จนครบรอยตัดทั้งหมด ในการเขียนกราฟชนิดนี้ มีหลักที่สำคัญอย่างหนึ่งคือ ผิวที่เกิดจากการตัดซ้ำ จะถูกระบุด้วยตัวเลขห้อยท้าย เช่น G1 เกิดจากการตัดซ้ำครั้งที่ 1 เป็นต้น

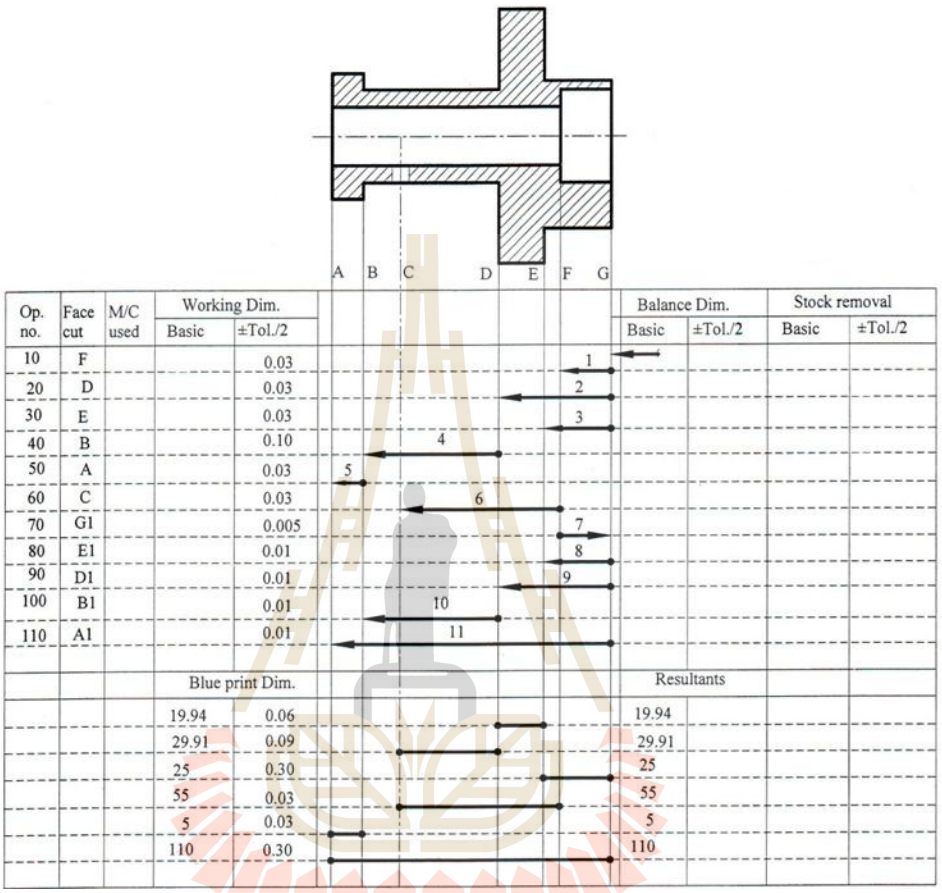
เราสามารถนำแผนภาพต้นไม้ (rooted-tree graph) ไปใช้เป็นพื้นฐานในการคำนวณหลายอย่างในแผนภูมิฯ ตัวอย่างเช่น การคำนวณค่าความคลาดเคลื่อนสะสมของ Stock removal ที่เกิดจากการตัดครั้งที่ 7, คือผลรวมของค่าความคลาดเคลื่อนในเส้นของกราฟต้นไม้ระหว่างโหนด G และ G1 = ±(0.03 + 0.005) = ±0.035 (ดูรูปที่1.7 ประกอบ); หรือ ค่าคลาดเคลื่อนสะสมของ Resultants ผิว D กับ E จะมีค่าเท่ากับ ผลรวมของความคลาดเคลื่อนในเส้นกราฟระหว่างโหนด D1 และ E1 = ±(0.01 + 0.01) = ± 0.02 เป็นต้น.




รูปที่ 1.7 แสดงแผนภาพต้นไม้

ในที่นี้จะกล่าวถึงการสร้างแผนภูมิความคลาดเคลื่อนและใช้ rooted- tree graph อย่างละเอียดเพราะเป็นวิธีที่ใช้กับงานพัฒนาโปรแกรมในวิทยานิพนธ์นี้ ซึ่งขั้นตอนการสร้างพื้นฐานจะเหมือนกับที่กล่าวไว้ในหน้า 6 และ 7 จะมีส่วนที่แตกต่างกันก็คือ การใช้สัญลักษณ์แทน

ผิวงานที่ถูกตัด การคำนวณความคลาดเคลื่อนสะสมและการคำนวณหาขนาดมูลฐาน (basic size) ของการตัดแต่ละครั้ง




| Op. no. | Face cut | M/C used | Working Dim. Basic | Working Dim. ±Tol./2 | Balance Dim. Basic | Balance Dim. ±Tol./2 | Stock removal Basic | Stock removal ±Tol./2 |
|---|---|---|---|---|---|---|---|---|
| 10 | F | | | 0.03 | | | | |
| 20 | D | | | 0.03 | | | | |
| 30 | E | | | 0.03 | | | | |
| 40 | B | | | 0.10 | | | | |
| 50 | A | | | 0.03 | | | | |
| 60 | C | | | 0.03 | | | | |
| 70 | G1 | | | 0.005 | | | | |
| 80 | E1 | | | 0.01 | | | | |
| 90 | D1 | | | 0.01 | | | | |
| 100 | B1 | | | 0.01 | | | | |
| 110 | A1 | | | 0.01 | | | | |
| | | | Blue print Dim. | | Resultants | | | |
| | | | 19.94 | 0.06 | 19.94 | | | |
| | | | 29.91 | 0.09 | 29.91 | | | |
| | | | 25 | 0.30 | 25 | | | |
| | | | 55 | 0.03 | 55 | | | |
| | | | 5 | 0.03 | 5 | | | |
| | | | 110 | 0.30 | 110 | | | |

รูปที่ 1.8 ประกอบขั้นตอนการสร้างแผนภูมิความคลาดเคลื่อน

การสร้างแผนภูมิความคลาดเคลื่อนมีขั้นตอนดังนี้:-

(1) สเกตซ์ภาพชิ้นงานตอนบนของแผนภูมความคลาดเคลื่อน

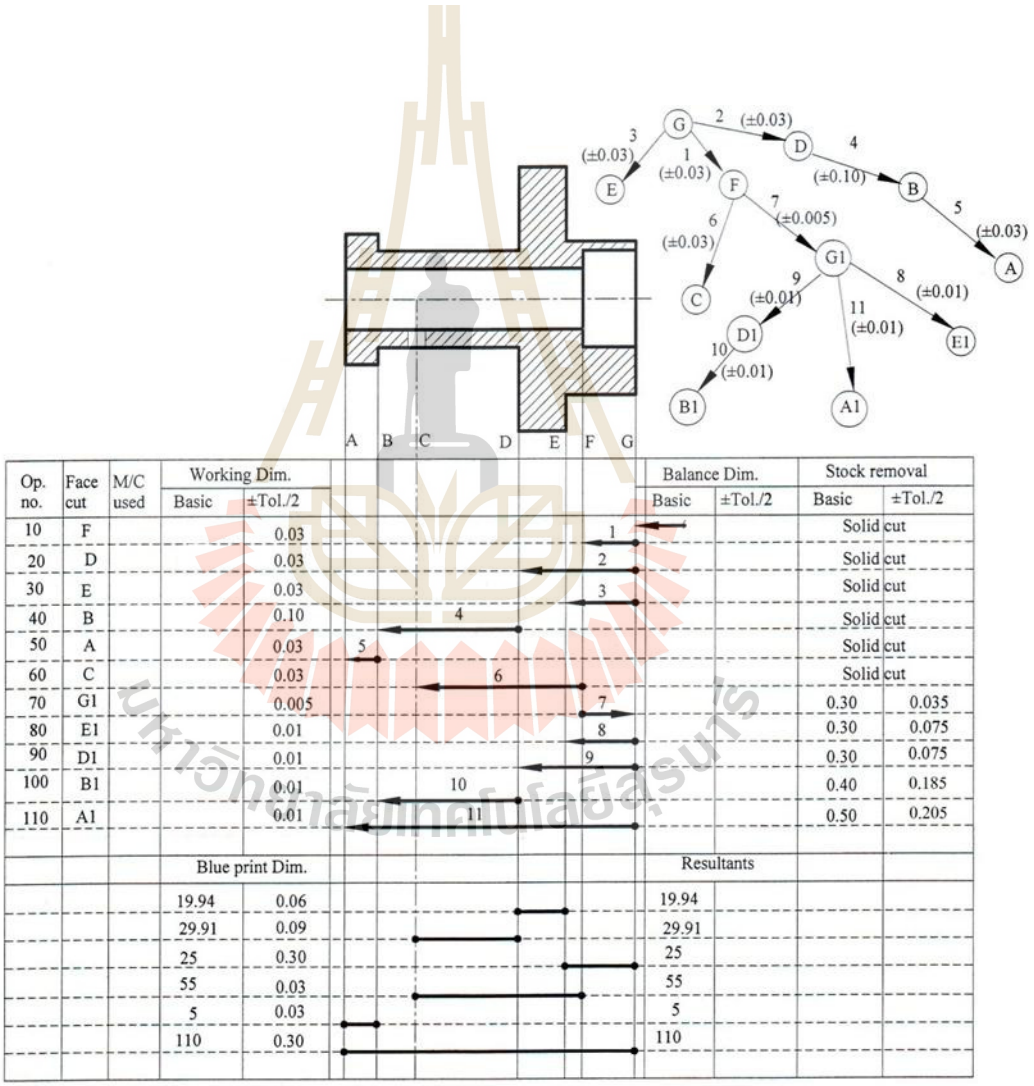
(2) ลากเส้นดิ่ง A, B, C, D, E, F, และ G จากผิวรอยตัดของชิ้นงานในแนวดิ่ง

(3) เขียนสัญลักษณ์การตัดผิว (machining cut) จากจุดอ้างอิงไปยังผิวที่ถูกตัด (หัวลูกศร) ตามขั้นตอนการตัดที่วางแผนไว้ ตัวอย่าง ดังรูปที่ 1.8

(4) เติมหมายเลขของงานแต่ละลำดับขั้นตอนการตัด (cut) และใส่เครื่องจักรหรือกระบวนการผลิตที่ใช้ในแต่ละขั้นตอน ลงในแผนภูมิความคลาดเคลื่อน ดังรูปที่ 1.8

(5) ใส่ข้อกำหนดมิติความยาวตามแบบลงในสดมภ์ของ BLUEPRINT : 19.94±0.06, 29.91±0.09 , 25.00 ± 0.30, 55.00 ± 0.03, 5.00 ± 0 .03 และ 110.00 ± 0.30 แล้วเขียนสัญลักษณ์ dot-to-dot ระหว่างผิว D-E, C-D, E-G, C-F, A-B, และ A-G สำหรับแต่ละมิติตามลำดับ ดังรูปที่ 1.8

(6) ใส่ค่าความคลาดเคลื่อนให้กับการตัดของแต่ละ cut ลงในสดมภ์ของ WORKING DIM.เช่น cut 1 = ±0.03, cut 2 = ±0.03,…, และ cut 11 = ±0.01 เป็นต้น ดังรูป 1.8




| Op. no. | Face cut | M/C used | Working Dim. Basic | ±Tol./2 | Balance Dim. Basic | ±Tol./2 | Stock removal Basic | ±Tol./2 |
|---|---|---|---|---|---|---|---|---|
| 10 | F | | | 0.03 | | | Solid cut | |
| 20 | D | | | 0.03 | | | Solid cut | |
| 30 | E | | | 0.03 | | | Solid cut | |
| 40 | B | | | 0.10 | | | Solid cut | |
| 50 | A | | | 0.03 | | | Solid cut | |
| 60 | C | | | 0.03 | | | Solid cut | |
| 70 | G1 | | | 0.005 | | | 0.30 | 0.035 |
| 80 | E1 | | | 0.01 | | | 0.30 | 0.075 |
| 90 | D1 | | | 0.01 | | | 0.30 | 0.075 |
| 100 | B1 | | | 0.01 | | | 0.40 | 0.185 |
| 110 | A1 | | | 0.01 | | | 0.50 | 0.205 |
| | | | Blue print Dim. | | Resultants | | | |
| | | | 19.94 | 0.06 | 19.94 | | | |
| | | | 29.91 | 0.09 | 29.91 | | | |
| | | | 25 | 0.30 | 25 | | | |
| | | | 55 | 0.03 | 55 | | | |
| | | | 5 | 0.03 | 5 | | | |
| | | | 110 | 0.30 | 110 | | | |

รูปที่ 1.9 ประกอบขั้นตอนการสร้างแผนภูมิความคลาดเคลื่อน

(7) สร้างแผนภาพต้นไม้ดังรูปที่ 1.9 เริ่มต้นด้วยการเขียนโหนด (node) แรกของผิวที่ถูกตัดครั้งแรก; จากนั้นลากลูกศร อ้างอิงจากผิวแรกนี้ ไปยังผิวที่ถูกตัด (เขียนโหนดผิวที่ถูกตัด) การลากเส้นลูกศร จะดำเนินต่อไป จนครบรอยตัดทั้งหมด ในการเขียนกราฟชนิดนี้ มีหลักที่สำคัญอย่างหนึ่งคือ ผิวที่เกิดจากการตัดซ้ำ จะถูกระบุด้วยตัวเลขห้อยท้าย เช่น G1 เกิดจากการตัดซ้ำครั้งที่ 1 จากเดิมคือผิวถูกตัดครั้งแรกให้เป็น G เป็นต้น

(8) คำนวณค่าความคลาดเคลื่อนให้กับส่วนของโลหะที่ถูกตัดออกไป (stock removal) แล้วใส่ค่าลงในสดมภ์ STOCK REM. ดังนี้

บรรทัด (1)-(6): มีรอยตัด 6 รอยคือ cut 1-cut 6 และความคลาดเคลื่อนของ stock removal ของแต่ละ cut เขียนได้ดังนี้ ผิวทั้งหกผิวเหล่านี้แต่ละผิวยังไม่ถูกตัดมาก่อน สังเกตได้จากตัวอักษรแสดงรอยตัดไม่มีตัวเลขต่อท้าย เช่นผิวตัด G, D, E, B, A และ C จึงมีเนื้อโลหะมากจนไม่ต้องคำนึงถึงค่าความคลาดเคลื่อนของโลหะส่วนที่ถูกตัดออกไป ดังนั้นจึงเขียนคำว่า solid cut ที่สดมภ์ STOCK REM. ของแต่ละ cut

บรรทัดที่ (7): ความคลาดเคลื่อนของ stock removal สำหรับ cut 7 เราสามารถนำแผนภาพต้นไม้ (rooted-tree graph) ไปใช้เป็นพื้นฐานในการคำนวณค่า stock removal คือผลรวมของค่าความคลาดเคลื่อนในเส้นของกราฟต้นไม้ ระหว่างโหนด G และ G1 = ±(0.03 + 0.005) = ±0.035 (ดูรูปที่ 1.9 ประกอบ)

บรรทัดที่(8): ความคลาดเคลื่อนของ stock removal สำหรับ cut 8 คือผลรวมของค่าความคลาดเคลื่อนในเส้นของกราฟต้นไม้ ระหว่างโหนด E และ E1 = ±(0.03 + 0.03 + 0.005 + 0.01) = ±0.075 (ดูรูปที่ 1.9 ประกอบ)

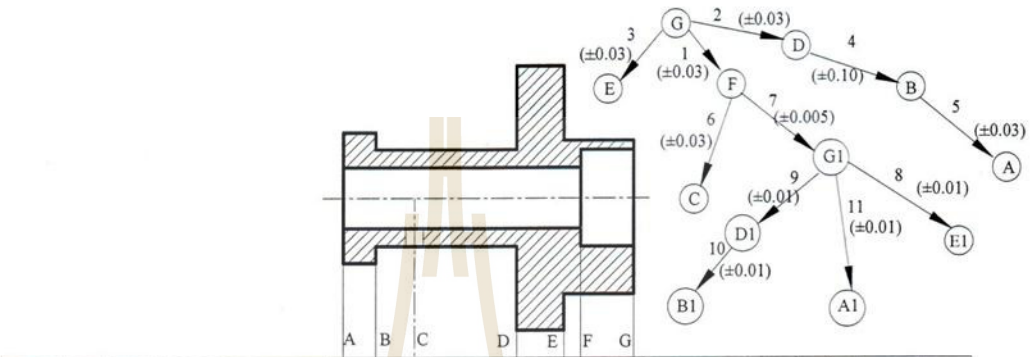
บรรทัดที่(9): ความคลาดเคลื่อนของ stock removal สำหรับ cut 9 คือผลรวมของค่าความคลาดเคลื่อนในเส้นของกราฟต้นไม้ ระหว่างโหนด D และ D1 = ±(0.03 + 0.03 + 0.005 + 0.01) = ±0.075 (ดูรูปที่ 1.9 ประกอบ)

บรรทัดที่(10): ความคลาดเคลื่อนของ stock removal สำหรับ cut 10 คือผลรวมของค่าความคลาดเคลื่อนในเส้นของกราฟต้นไม้ ระหว่างโหนด B และ B1 = ±(0.10 + 0.03 + 0.03 + 0.005 + 0.01 + 0.01) = ±0.185 (ดูรูปที่ 1.9 ประกอบ)

บรรทัดที่(11): ความคลาดเคลื่อนของ stock removal สำหรับ cut 11 คือผลรวมของค่าความคลาดเคลื่อนในเส้นของกราฟต้นไม้ ระหว่างโหนด A และ A1 = ±(0.03 + 0.10 + 0.03 + 0.03 + 0.005 + 0.01) = ±0.205 (ดูรูปที่ 1.9 ประกอบ)

(9) กำหนดขนาดของ stock removal น้อยที่สุดที่ยอมให้เกิดขึ้นกับรอยตัดแต่ละรอย ขึ้นอยู่กับชนิดของเครื่องจักรแต่ละ operation เช่นงานกลึงกำหนดให้มากกว่างานเจียรนัย เป็นต้น

ดังนั้น สำหรับ cut 7, ถ้ากำหนดให้ stock removal น้อยที่สุด = 0.2 mm. จะได้ค่า BASIC ของ STOCK REM. ของ cut 7 = 0.2 + 0.035 = 0.235 mm. หรือใส่ค่าประมาณโดยปัดค่าทศนิยมของ 0.035 เป็น 0.10 จะได้ค่า BASIC ของ STOCK REM. ของ cut 7 เท่ากับ .030 และด้วยวิธีเดียวกัน ค่าอื่นๆสำหรับ cut 8 – cut 1 จะเป็น 0.30, 0.30, 0.40, และ 0.50 ตามลำดับ




| Op. no. | Face cut | M/C used | Working Dim. Basic | Working Dim. ±Tol./2 | Balance Dim. Basic | Balance Dim. ±Tol./2 | Stock removal Basic | Stock removal ±Tol./2 |
|---|---|---|---|---|---|---|---|---|
| 10 | F | | 20.15 | 0.03 | | | Solid cut | |
| 20 | D | | 45.54 | 0.03 | | | Solid cut | |
| 30 | E | | 25 | 0.03 | | | Solid cut | |
| 40 | B | | 59.96 | 0.10 | | | Solid cut | |
| 50 | A | | 5.10 | 0.03 | | | Solid cut | |
| 60 | C | | 55 | 0.03 | | | Solid cut | |
| 70 | G1 | | 19.85 | 0.005 | | | 0.30 | 0.035 |
| 80 | E1 | | 25 | 0.01 | | | 0.30 | 0.075 |
| 90 | D1 | | 44.94 | 0.01 | | | 0.30 | 0.075 |
| 100 | B1 | | 60.06 | 0.01 | | | 0.40 | 0.185 |
| 110 | A1 | | 110 | 0.01 | | | 0.50 | 0.205 |
| | | | Blue print Dim. | | | Resultants | | |
| | | | 19.94 | 0.06 | 19.94 | 0.02 | | |
| | | | 29.91 | 0.09 | 29.91 | 0.045 | | |
| | | | 25 | 0.30 | 25 | 0.01 | | |
| | | | 55 | 0.03 | 55 | 0.03 | | |
| | | | 5 | 0.03 | 5 | 0.03 | | |
| | | | 110 | 0.30 | 110 | 0.01 | | |

รูปที่ 1.10 ประกอบขั้นตอนการสร้างแผนภูมิความคลาดเคลื่อน

(10) หาขนาดมูลฐาน (basic size) ที่เกิดจากการตัดแต่ละ cut (คือค่า BASIC ของ WORKING DIM.) ซึ่งมีหลักการอยู่ว่าให้พิจารณาจากด้านล่างของแผนภูมิขึ้นไปทางด้านบน โดยอาศัยกฎเกณฑ์อย่างใดอย่างหนึ่งดังนี้

(a.) ยกมาจาก blueprint dimensions ถ้าหากเป็น working dimension ของ cut ที่ทำให้เกิด blueprint dimension

(b.) คำนวณจาก blueprint dimension หนึ่ง

(c.)คำนวณจาก basic working dimensionที่ทราบค่า

(d.) คำนวณจากมิติที่เป็นผลจากการบวกหรือลบ blueprint dimensions

ซึ่งการวิเคราะห์หาขนาดมูลฐานของแต่ละ cut เป็นดังนี้:

Cut 11: Basic working dimension ของ cut 11 ทำให้เกิดมิติตามแบบ เพราะฉะนั้น cut 11 มีขนาดมูลฐานเท่ากับ 110 พิจารณาว่ามีขั้นตอนการตัดต่อจาก cut 11 หรือไม่ในที่นี้ไม่มีขั้นตอนการตัดต่อจาก cut 11 จึงนำค่าขนาดมูลฐานคือ 110 ยกไปใส่บรรทัดที่ 11 ในสดมภ์ BASIC (ดูรูปที่ 1.10 ประกอบ)

Cut 10: Basic working dimension ของ cut 10 เกิดจากการคำนวณขนาดจาก blueprint dimension ได้ค่าเท่ากับ 60.06, พิจารณาว่ามีขั้นตอนการตัดต่อจาก cut 10 หรือไม่ ในที่นี้มีขั้นตอนการตัดต่อจาก cut 10 พิจารณาต่อว่ารอยตัดถัดไปมีการตัดซ้ำตำแหน่งจุดอ้างอิงหรือจุดตัดของ cut 10 หรือไม่ ในที่นี้ไม่พบรอยตัดซ้ำ จึงนำค่าขนาดมูลฐานคือ 60.06 ยกไปใส่บรรทัดที่ 10 ในสดมภ์ BASIC (ดูรูปที่ 1.10 ประกอบ)

Cut 9: Basic working dimension ของ cut 9 เกิดจากการคำนวณขนาดจาก blueprint dimension ได้ค่าเท่ากับ 44.94, พิจารณาว่ามีขั้นตอนการตัดต่อจาก cut 9 หรือไม่ ในที่นี้มีขั้นตอนการตัดต่อจาก cut 9 พิจารณาต่อว่ารอยตัดถัดไปมีการตัดซ้ำตำแหน่งจุดอ้างอิงหรือจุดตัดของ cut 9 หรือไม่ ในที่นี้ไม่พบรอยตัดซ้ำ จึงนำค่าขนาดมูลฐานคือ 44.94 ยกไปใส่บรรทัดที่ 9 ในสดมภ์ BASIC (ดูรูปที่ 1.10 ประกอบ)

Cut 8: Basic working dimension ของ cut 8 เกิดจากการคำนวณขนาดจาก blueprint dimension ได้ค่าเท่ากับ 25.00, พิจารณาว่ามีขั้นตอนการตัดต่อจาก cut 8 หรือไม่ ในที่นี้มีขั้นตอนการตัดต่อจาก cut 8 พิจารณาต่อว่ารอยตัดถัดไปมีการตัดซ้ำตำแหน่งจุดอ้างอิงหรือจุดตัดของ cut 8 หรือไม่ ในที่นี้ไม่พบรอยตัดซ้ำ จึงนำค่าขนาดมูลฐานคือ 25.00 ยกไปใส่บรรทัดที่ 8 ในสดมภ์ BASIC (ดูรูปที่ 1.10 ประกอบ)

Cut 7: Basic working dimension ของ cut 7 เกิดจากการคำนวณขนาดจาก blueprint dimension ได้ค่าเท่ากับ 19.85, พิจารณาว่ามีขั้นตอนการตัดต่อจาก cut 7 หรือไม่ ในที่นี้มีขั้นตอนการตัดต่อจาก cut 7 พิจารณาต่อว่ารอยตัดถัดไปมีการตัดซ้ำตำแหน่งจุดอ้างอิงหรือจุดตัดของ cut 7 หรือไม่ ในที่นี้ไม่พบรอยตัดซ้ำ จึงนำค่าขนาดมูลฐานคือ 19.85 ยกไปใส่บรรทัดที่ 7 ในสดมภ์ BASIC (ดูรูปที่ 1.10 ประกอบ)

Cut 6: Basic working dimension ของ cut 6 เกิดจากการคำนวณขนาดจาก blueprint dimension ได้ค่าเท่ากับ 55.00, พิจารณาว่ามีขั้นตอนการตัดต่อจาก cut 6 หรือไม่ ในที่นี้มี

ขั้นตอนการตัดต่อจาก cut 6 พิจารณาต่อว่ารอยตัดถัดไปมีการตัดซ้ำตำแหน่งจุดอ้างอิงหรือจุดตัดของ cut 6 หรือไม่ ในที่นี้ไม่พบรอยตัดซ้ำ จึงนำค่าขนาดมูลฐานคือ 55.00 ยกไปใส่บรรทัดที่ 6 ในสคมภ์ BASIC (ดูรูปที่ 1.10 ประกอบ)

Cut 5: Basic working dimension ของ cut 5 ทำให้เกิดมิติตามแบบ เพราะฉะนั้น cut 5 มีขนาดมูลฐานเท่ากับ 5.00, พิจารณาว่ามีขั้นตอนการตัดต่อจาก cut 5 หรือไม่ ในที่นี้มีขั้นตอนการตัดต่อจาก cut 5 พิจารณาต่อว่ารอยตัดถัดไปมีการตัดซ้ำตำแหน่งจุดอ้างอิงหรือจุดตัดของ cut 5 หรือไม่ ในที่นี้พบรอยตัดซ้ำ cut 5 ได้แก่การตัดของ cut 10 และ 11 จากนั้นพิจารณาว่าหากมีการตัดผิวซ้ำ cut 10 และ cut 11 ขนาดของรอยตัดจะยาวขึ้นหรือสั้นลง หากยาวขึ้นนำค่า basic stock removal ไปลบค่าขนาดมูลฐาน หรือหากสั้นลงนำค่า basic stock removal ไปบวกค่าขนาดมูลฐาน จะได้ค่า Basic working dimension ของ cut 5 เท่ากับ 5.0-0.4+0.5= 5.10 ยกไปใส่บรรทัดที่ 5 ในสคมภ์ BASIC (ดูรูปที่ 1.10 ประกอบ)

Cut 4: Basic working dimension ของ cut 4 เกิดจากการคำนวณขนาดจาก blueprint dimension ได้ค่าเท่ากับ 60.06, พิจารณาว่ามีขั้นตอนการตัดต่อจาก cut 4 หรือไม่ ในที่นี้มีขั้นตอนการตัดต่อจาก cut 4 พิจารณาต่อว่ารอยตัดถัดไปมีการตัดซ้ำตำแหน่งจุดอ้างอิงหรือจุดตัดของ cut 4 หรือไม่ ในที่นี้พบรอยตัดซ้ำ cut 4 ได้แก่การตัดของ cut 9 และ 10 จากนั้นพิจารณาว่าหากมีการตัดผิวซ้ำ cut 9 และ cut 10 ขนาดของรอยตัดจะยาวขึ้นหรือสั้นลง หากยาวขึ้นนำค่า basic stock removal ไปลบค่าขนาดมูลฐาน หรือหากสั้นลงนำค่า basic stock removal ไปบวกค่าขนาดมูลฐาน จะได้ค่า Basic working dimension ของ cut 4 เท่ากับ 60.06 + 0.3 - 0.4 = 59.96 ยกไปใส่บรรทัดที่ 4 ในสคมภ์ BASIC (ดูรูปที่ 1.10 ประกอบ)

Cut 3: Basic working dimension ของ cut 3 ทำให้เกิดมิติตามแบบ เพราะฉะนั้น cut5 มีขนาดมูลฐานเท่ากับ 25.00, พิจารณาว่ามีขั้นตอนการตัดต่อจาก cut 3 หรือไม่ ในที่นี้มีขั้นตอนการตัดต่อจาก cut 3 พิจารณาต่อว่ารอยตัดถัดไปมีการตัดซ้ำตำแหน่งจุดอ้างอิงหรือจุดตัดของ cut 3 หรือไม่ ในที่นี้พบรอยตัดซ้ำ cut 3 ได้แก่การตัดของ cut 7 และ 8 จากนั้นพิจารณาว่าหากมีการตัดผิวซ้ำ cut 7 และ cut 8 ขนาดของรอยตัดจะยาวขึ้นหรือสั้นลง หากยาวขึ้นนำค่า basic stock removal ไปลบค่าขนาดมูลฐาน หรือหากสั้นลงนำค่า basic stock removal ไปบวกค่าขนาดมูลฐาน จะได้ค่า Basic working dimension ของ cut 3 เท่ากับ 25.00 + 0.3 – 0.3 = 25.00 ยกไปใส่บรรทัดที่ 3 ในสคมภ์ BASIC (ดูรูปที่ 1.10 ประกอบ)

Cut 2: Basic working dimension ของ cut 2 เกิดจากการคำนวณขนาดจาก blueprint dimension ได้ค่าเท่ากับ 44.94, พิจารณาว่ามีขั้นตอนการตัดต่อจาก cut 2 หรือไม่ ในที่นี้มีขั้นตอนการตัดต่อจาก cut 2 พิจารณาต่อว่ารอยตัดถัดไปมีการตัดซ้ำตำแหน่งจุดอ้างอิงหรือจุดตัด

ของ cut 2 หรือไม่ ในที่นี้พบรอยตัดซ้ำ cut 2 ได้แก่การตัดของ cut 7 และ 9 จากนั้นพิจารณาว่าหากมีการตัดผิวซ้ำ cut 7 และ cut 9 ขนาดของรอยตัดจะยาวขึ้นหรือสั้นลง หากยาวขึ้นนำค่า basic stock removal ไปลบค่าขนาดมูลฐาน หรือหากสั้นลงนำค่า basic stock removal ไปบวกค่าขนาดมูลฐาน จะได้ค่า Basic working dimension ของ cut 2 เท่ากับ 44.94 + 0.3 + 0.3 = 45.94 ยกไปใส่บรรทัดที่ 2 ในสดมภ์ BASIC (ดูรูปที่ 1.10 ประกอบ)

Cut 1: Basic working dimension ของ cut 1 เกิดจากการคำนวณขนาดจาก blueprint dimension ได้ค่าเท่ากับ 19.85, พิจารณาว่ามีขั้นตอนการตัดต่อจาก cut 1 หรือไม่ ในที่นี้มีขั้นตอนการตัดต่อจาก cut 1 พิจารณาต่อว่ารอยตัดถัดไปมีการตัดซ้ำตำแหน่งจุดอ้างอิงหรือจุดตัดของ cut 1 หรือไม่ ในที่นี้พบรอยตัดซ้ำ cut 2 ได้แก่การตัดของ cut จากนั้นพิจารณาว่าหากมีการตัดผิวซ้ำ cut 7 ขนาดของรอยตัดจะยาวขึ้นหรือสั้นลง หากยาวขึ้นนำค่า basic stock removal ไปลบค่าขนาดมูลฐาน หรือหากสั้นลงนำค่า basic stock removal ไปบวกค่าขนาดมูลฐาน จะได้ค่า Basic working dimension ของ cut 2 เท่ากับ 19.85 + 0.3 = 20.15 ยกไปใส่บรรทัดที่ 2 ในสดมภ์ BASIC (ดูรูปที่ 1.10 ประกอบ)

(11) หาค่าความคลาดเคลื่อนของมิติที่เป็นผลลัพธ์ของการตัดโลหะทั้งหมด (ในสดมภ์ RESULTANTS.) (ดูรูปที่ 1.10 ประกอบ)

มิติระหว่างผิว D และ E: ค่าคลาดเคลื่อนสะสมของ Resultants นำแผนภาพต้นไม้ (rooted-tree graph) ใช้ในการคำนวณค่าคลาดเคลื่อนสะสมของ Resultants ได้ดังนี้ ค่าคลาดเคลื่อนสะสมของ Resultants ผิว D กับ E จะมีค่าเท่ากับ ผลรวมของความคลาดเคลื่อนในเส้นกราฟระหว่างโหนด D1 และ E1 = ±(0.01 + 0.01) = ± 0.02 (ดูรูปที่ 1.10 ประกอบ)

มิติระหว่างผิว C และ D: ค่าคลาดเคลื่อนสะสมของ Resultants นำแผนภาพต้นไม้ (rooted-tree graph) ใช้ในการคำนวณค่าคลาดเคลื่อนสะสมของ Resultants ได้ดังนี้ ค่าคลาดเคลื่อนสะสมของ Resultants ผิว C กับ D จะมีค่าเท่ากับ ผลรวมของความคลาดเคลื่อนในเส้นกราฟระหว่างโหนด C และ D1 = ±(0.03 + 0.005 + 0.01) = ± 0.045 (ดูรูปที่ 1.10 ประกอบ)

มิติระหว่างผิว E และ G: ค่าคลาดเคลื่อนสะสมของ Resultants นำแผนภาพต้นไม้ (rooted-tree graph) ใช้ในการคำนวณค่าคลาดเคลื่อนสะสมของ Resultants ได้ดังนี้ ค่าคลาดเคลื่อนสะสมของ Resultants ผิว E กับ G จะมีค่าเท่ากับ ผลรวมของความคลาดเคลื่อนในเส้นกราฟระหว่างโหนด E1 และ G1 = ±(0.01) = ± 0.01 (ดูรูปที่ 1.10 ประกอบ)

มิติระหว่างผิว C และ F: ค่าคลาดเคลื่อนสะสมของ Resultants นำแผนภาพต้นไม้ (rooted-tree graph) ใช้ในการคำนวณค่าคลาดเคลื่อนสะสมของ Resultants ได้ดังนี้ ค่าคลาดเคลื่อน

สะสมของ Resultants ผิว C กับ F จะมีค่าเท่ากับ ผลรวมของความคลาดเคลื่อนในเส้นกราฟระหว่างโหนด C และ F = ±(0.03) = ± 0.03 (ดูรูปที่ 1.10 ประกอบ)

มิติระหว่างผิว A และ B: ค่าคลาดเคลื่อนสะสมของ Resultants นำแผนภาพต้นไม้ (rooted-tree graph) ใช้ในการคำนวณค่าคลาดเคลื่อนสะสมของ Resultantsได้ดังนี้ ค่าคลาดเคลื่อนสะสมของ Resultants ผิว A กับ B จะมีค่าเท่ากับ ผลรวมของความคลาดเคลื่อนในเส้นกราฟระหว่างโหนด A1 และ B1 = ±(0.01 + 0.01 + 0.01 ) = ± 0.03 (ดูรูปที่ 1.10 ประกอบ)

มิติระหว่างผิว A และ G: ค่าคลาดเคลื่อนสะสมของ Resultants นำแผนภาพต้นไม้ (rooted-tree graph) ใช้ในการคำนวณค่าคลาดเคลื่อนสะสมของ Resultantsได้ดังนี้ ค่าคลาดเคลื่อนสะสมของ Resultants ผิว A กับ G จะมีค่าเท่ากับ ผลรวมของความคลาดเคลื่อนในเส้นกราฟระหว่าง โหนด A1 และ G1 = ±(0.01) = ± 0.01 (ดูรูปที่ 1.10 ประกอบ)

## 1.3 ทบทวนวรรณกรรม

การวิเคราะห์ความคลาดเคลื่อน (tolerance analysis) คือการแยกองค์ประกอบของความคลาดเคลื่อนที่พิจารณาออกเป็นส่วนย่อยๆ และอาจมีการสร้างความสัมพันธ์ระหว่างความคลาดเคลื่อนย่อยๆ เพื่อสะท้อนถึงความคลาดเคลื่อนรวม

วิธีการวิเคราะห์ความคลาดเคลื่อน มักจะได้รับการพัฒนามาจากการออกแบบชิ้นงานประกอบ (assembly) ซึ่งเป็นการวิเคราะห์เพื่อหาความสัมพันธ์ระหว่างความต้องการใช้งาน (functional requirement) กับความคลาดเคลื่อนของมิติในชิ้นงานประกอบแต่ละชิ้น

แผนภูมิความคลาดเคลื่อน เป็นการวิเคราะห์ความคลาดเคลื่อนอีกชนิดหนึ่ง ที่ใช้กับความคลาดเคลื่อนของมิติต่างๆที่เกิดจากกระบวนการผลิตในแต่ละขั้นตอน ของการผลิตที่เกิดขึ้นกับชิ้นงานชิ้นเดียว

แม้ว่าแผนภูมิความคลาดเคลื่อนจะมีประโยชน์มาก เพราะทำให้แน่ใจว่า ขั้นตอนการผลิตที่กำหนดนั้นสามารถบรรลุเป้าหมายตามข้อกำหนดในแบบได้หรือไม่ แต่เนื่องจากเครื่องมือชนิดนี้เข้าใจได้ยาก และ เกิดความผิดพลาดในการคำนวณได้ง่าย ด้วยเหตุนี้เอง บทความรายงานวิจัย จึงปรากฏหลายบทความในช่วง 1950-1960 เศษๆหลังจากนั้นดูเหมือนว่า งานวิจัยเกี่ยวกับแผนภูมิความคลาดเคลื่อนจะลดน้อยลง ช่วงนี้มีหนังสือที่กล่าวถึงแผนภูมิความคลาดเคลื่อนอยู่ 2 เล่ม ที่สำคัญ 2 เล่มคือ Gadzala (1959) และ O.R.Wade (1966) ที่ได้กล่าวมาแล้วข้างต้น

การวิเคราะห์ความคลาดเคลื่อนด้วยคอมพิวเตอร์เริ่มเป็นงานวิจัยแพร่หลาย เมื่อ Bjork (1978) รายงานการสร้างโปรแกรมคำนวณความคลาดเคลื่อนสะสม (tolerance stacks) ด้วยคอมพิวเตอร์ขนาดใหญ่ (mainframe computer) Bjork ใช้ Tolerance Chain สร้างความสัมพันธ์

ระหว่างความคลาดเคลื่อนของมิติ แต่ในช่วงเวลานั้นคอมพิวเตอร์ยังมีขีดความสามารถจำกัด ต่อมาในปี 1984 บทความของ Ahaluwalia และ Karolin รายงานการพัฒนาโปรแกรมคอมพิวเตอร์สำหรับสร้างแผนภูมิความคลาดเคลื่อน ซึ่งถือได้ว่าเป็นจุดเริ่มต้นของการใช้คอมพิวเตอร์วิเคราะห์ความคลาดเคลื่อนด้วยหลักการของแผนภูมิความคลาดเคลื่อน

Davie (1988) ใช้กราฟร่วมกับกฎเกณฑ์ต่างๆในการวิเคราะห์ความคลาดเคลื่อนในแผนภูมิถัดมา 1989 Irani, Mittal and Lehtihet ได้เสนอเทคนิค path-tracing เพื่อสร้างความสัมพันธ์ระหว่างรอยตัดของแผนภูมิความคลาดเคลื่อน แต่ด้วยเทคนิคนี้มีความซับซ้อนมากจึงไม่เป็นที่นิยมใช้

ล่าสุดในปี 1990 Whybrew และคณะ ได้พัฒนาวิธีการใช้กราฟต้นไม้ (Rooted-Tree Graph) ช่วยสร้างแผนภูมิความคลาดเคลื่อน ซึ่งเป็นวิธีที่นอกจากจะง่ายต่อการทำความเข้าใจและไม่ยุ่งยากซับซ้อนแล้ว ยังมีลักษณะเป็นระบบที่ใช้ตรวจสอบความผิดพลาดได้ วิธีนี้ยังสามารถนำไปประยุกต์ใช้เขียนโปรแกรมคอมพิวเตอร์ช่วยในการวิเคราะห์ความคลาดเคลื่อนได้อีกด้วย

นอกจากงานวิจัยทางด้านพัฒนา algorithm สำหรับคำนวณความคลาดเคลื่อนสะสมในแผนภูมิฯ แล้วยังมีงานวิจัยหลายชิ้นที่เกี่ยวข้องกับการทำ tolerance optimization ในแผนภูมิความคลาดเคลื่อน เช่นงานของ B.K.A. Noi และ Ji

และล่าสุดชลาลัย (2016) ได้ใช้เทคนิคกราฟต้นไม้ (rooted-tree graph) เป็นพื้นฐานในการสร้างแผนภูมิแผนภูมิความคลาดเคลื่อนบนโปรแกรมตารางคำนวณ (spreadsheet) ซึ่งโปรแกรมดังกล่าว สามารถแสดงผลลัพธ์เป็นแผนภูมิความคลาดเคลื่อน และแก้ไขเปลี่ยนแปลงความคลาดเคลื่อนที่เกี่ยวข้องได้ นอกจากนี้ยังใช้ graphics functions ของโปรแกรมตารางคำนวณในการรับค่า Input และแสดงผล Output เป็นแผนภูมิความคลาดเคลื่อน; แต่เนื่องจากเซลล์ในตารางคำนวณนั้นถูกจำกัดด้วยฟังก์ชั่น ทำให้ความสามารถในการเปลี่ยนแปลงขั้นตอนในการตัดโลหะ เช่นการเพิ่มหรือลดงานตัดโลหะเป็นไปได้อย่างยากลำบาก บทความนี้จึงรายงานการพัฒนาโปรแกรมสร้างแผนภูมิความคลาดเคลื่อนให้สามารถมีปฏิสัมพันธ์ กับผู้ใช้ได้มากขึ้น เพื่อปรับเปลี่ยนลำดับของการตัดโลหะและ/หรือ เงื่อนไขของการตัดโลหะได้อย่างรวดเร็ว

## 1.4 วัตถุประสงค์ของงานวิจัย

วัตถุประสงค์ของงานวิจัยนี้คือ สร้างแผนภูมิความคลาดเคลื่อนโดยใช้โปรแกรม Visual Basic ในการประมวลผลและวิเคราะห์ความคลาดเคลื่อน ที่เกิดขึ้นในแต่ละขั้นตอนการตัดโลหะ โดยโปรแกรมจะแสดงผลออกมาเป็นขนาดและความคลาดเคลื่อนสะสมของแต่ละขั้นตอนการตัดโลหะ เพิ่มความสามารถของโปรแกรม 3 ส่วนคือ

(1) สามารถเปลี่ยนแปลงลำดับขั้นตอนการตัดโลหะ

(2) สามารถเพิ่มหรือลดขั้นตอนการตัดโลหะโดยวิธีปฏิสัมพันธ์กับผู้ใช้

(3) สามารถตรวจสอบความเหมาะสมของแผนภูมิความคลาดเคลื่อนของทั้งสองส่วนแรก

## 1.5 สมมติฐานและขอบเขตของการวิจัย

การพัฒนาโปรแกรมแผนภูมิความคลาดเคลื่อนนี้เขียนด้วย โปรแกรม Visual basic โดยมีสมมติฐานและขอบเขตของการวิจัยดังนี้

(1) ใช้เฉพาะชิ้นงานตัดโลหะสองมิติที่มีการผลิตแบบไม่ต่อเนื่อง (rotational discrete machined part)

(2) ใช้กับกระบวนการตัดโลหะพื้นฐาน (basic machining processes) เช่นการกลึง (turning operation) และการเจียรนัย (grinding operation)

(3) ไม่มี Geometric tolerance มาเกี่ยวข้อง

(4) โดยสมมุติความสามารถของเครื่องมือกลอยู่ในสภาวะเฉลี่ยของอายุและการใช้งาน ซึ่งสามารถให้ค่าความคลาดเคลื่อนตามมาตรฐาน ISO 286

## 1.6 สรุป

ในบทนี้กล่าวถึงที่มาและความสำคัญของปัญหา ทบทวนวรรณกรรมเกี่ยวกับแผนภูมิความคลาดเคลื่อน วัตถุประสงค์ของงานวิจัย สมมติฐานและขอบเขตของการวิจัย และได้กล่าวถึงวิธีการสร้างแผนภูมิความคลาดเคลื่อนด้วยเทคนิคกราฟต้นไม้โดยแสดงรายละเอียดขั้นตอนการสร้างอย่างชัดเจน ตามแนวทางการสร้างแผนภูมิความคลาดเคลื่อน

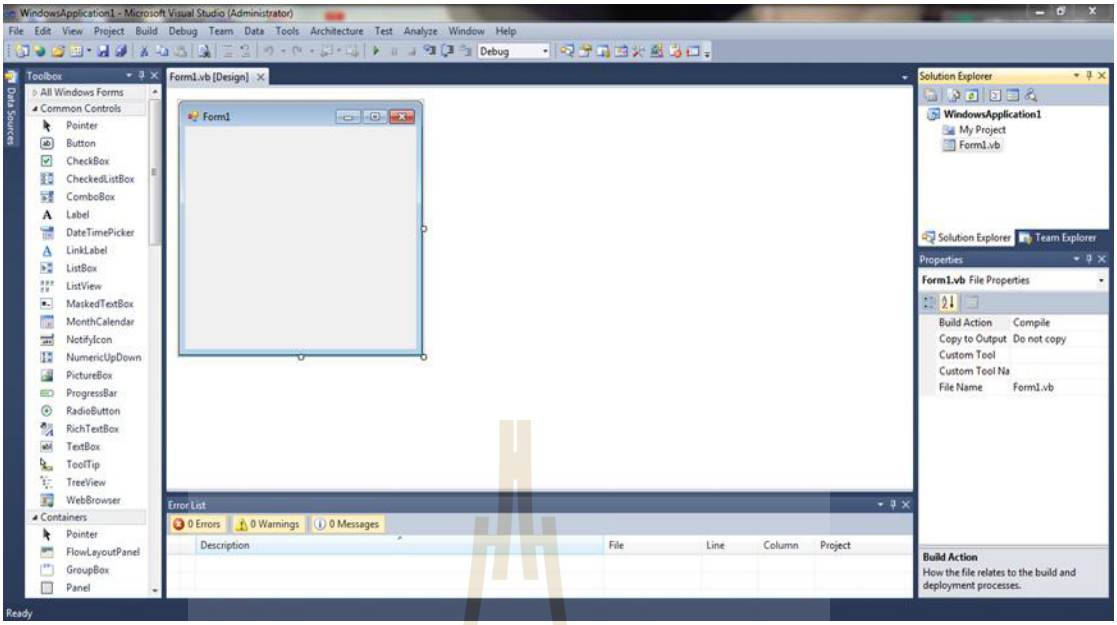
มหาวิทยาลัยเทคโนโลยีสุรนารี

# บทที่ 2
# โปรแกรมแผนภูมิความคลาดเคลื่อน

โปรแกรมแผนภูมิความคลาดเคลื่อน (Tolerance Charting Program) เขียนด้วยชุดคำสั่งของโปรแกรม Microsoft Visual Studio ซึ่งเป็นชุดพัฒนาโปรแกรม ที่สะดวกต่อการใช้สร้างโปรแกรมสำเร็จรูปบนระบบปฏิบัติการวินโดวส์ ในบทนี้กล่าวถึงโปรแกรม Visual basic และโครงสร้างของโปรแกรมแผนภูมิความคลาดเคลื่อนที่มีส่วนประกอบหลัก 4 ส่วน และขั้นตอนการทำงานของโปรแกรมโดยแบ่งลักษณะการทำงานเป็น 8 ขั้นตอน

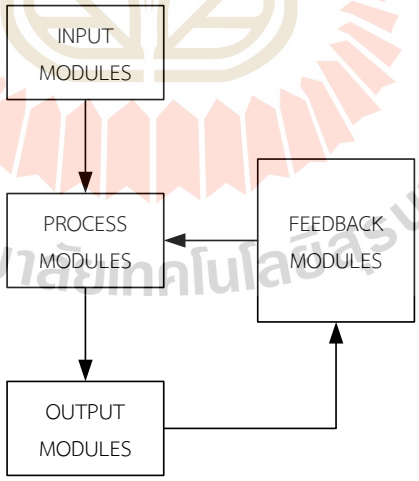
## 2.1 โปรแกรม Visual Basic

Visual Basic เป็นภาษาคอมพิวเตอร์ (Programming Language) ที่พัฒนาโดยบริษัทไมโครซอฟท์ ซึ่งเป็นบริษัทที่สร้างระบบปฏิบัติการ Windows ที่ใช้กันอย่างแพร่หลายในปัจจุบัน โดยตัวภาษาเองมีรากฐานมาจากภาษา Basic ซึ่งย่อ มา จาก Beginner's All Purpose Symbolic Instruction ซึ่งหมายถึง ชุดคำ สั่งหรือภาษาคอมพิวเตอร์ สำหรับผู้เริ่มต้น ภาษา Basic มีจุดเด่นคือผู้เขียนโปรแกรมไม่จำเป็นต้องมีพื้นฐานการเขียนโปรแกรมมาก่อน ก็สามารถเรียนรู้และนำไปใช้งานได้ ง่ายสะดวกรวดเร็ว Visual basic เวอร์ชันแรกคือเวอร์ชัน 1.0 ออกสู่ตลาด ปี1991 โดยในช่วงแรกยังไม่มีความสามารถต่างจากภาษา QBASIC มากนัก แต่จะเน้นเรื่องเครื่องมือที่ช่วยในการเขียนโปรแกรมบนวินโดวส์ ซึ่งปรากฏว่า Visual Basic ได้รับความนิยมและประสบความสำเร็จเป็นอย่างดีไมโครซอฟท์จึงพัฒนา Visual Basic ให้ดีขึ้นเรื่อยๆ ทั้งในด้านประสิทธิภาพ ความสามารถและเครื่องมือต่าง ๆ เช่น เครื่องมือตรวจสอบแก้ไขโปรแกรม สภาพแวดล้อมของการพัฒนาโปรแกรม การเขียนโปรแกรมแบบหลายวินโดวส์ย่อย (MDI) และอื่น ๆ อีกมากมาย Microsoft Visual Basic เป็นเครื่องมือในการสร้างโปรแกรมบนระบบปฏิบัติการ windows ที่ใช้งานง่าย โดยการเลือกเครื่องมือต่าง ๆ มาออกแบบหน้าจอของโปรแกรมที่จะสร้าง การเขียน โปรแกรมลักษณะนี้เรียกว่า Visual Programming ซึ่งไม่จำ เป็นต้องเขียนคำสั่งต่าง ๆ มากนักและ สามารถสร้างโปรแกรมได้อย่างรวดเร็ว




รูปที่ 2.1 แสดงลักษณะของโปรแกรม Visual Basic

## 2.2 โครงสร้างโปรแกรมแผนภูมิความคลาดเคลื่อน




รูปที่ 2.2 โครงสร้างโปรแกรมแผนภูมิความคลาดเคลื่อน

โปรแกรมแผนภูมิความคลาดเคลื่อนที่ได้รับการพัฒนา มีส่วนประกอบหลัก 4 ส่วนคือ:

(1) โมดูลรับข้อมูล (Input Modules) ทำหน้าที่รับข้อมูล สำหรับสร้างแผนภูมิความคลาดเคลื่อน และนำค่าไปเก็บไว้ในตัวแปรต่างๆเพื่อประมวลผลในแต่ละขั้นตอน

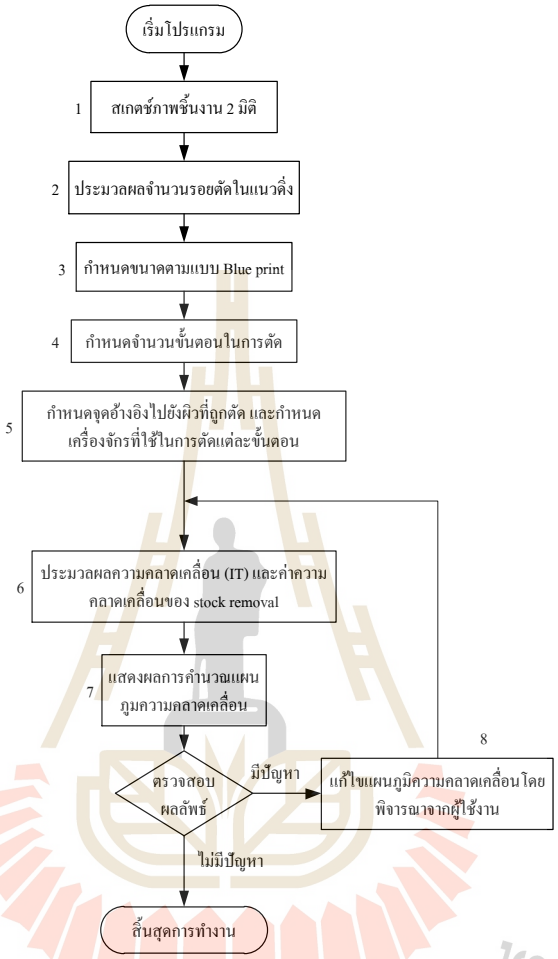
(2) โมดูลประมวลผล (Processing Modules) ทำหน้าที่ประมวลผลตามขั้นตอนของการสร้างแผนภูมิความคลาดเคลื่อน เช่น การกำหนดรหัสผิวชิ้นงานใหม่จากข้อมูล input, สร้างเส้นทางของ rooted tree สำหรับความคลาดเคลื่อนของ stock removal และของ resultant dimensions, และการหาขนาดมูลฐานของ working dimensions

(3) โมดูลแสดงผลลัพธ์ (Output Modules) ทำหน้าที่แสดงผลลัพธ์ของโปรแกรมตามแบบขั้นตอนการคำนวณของแผนภูมิความคลาดเคลื่อน

(4) โมดูลข้อมูลย้อนกลับ (Feedback Modules) ทำหน้าที่อำนวยความสะดวกต่อผู้ใช้โปรแกรมในการปรับเปลี่ยนข้อมูลในการตัดโลหะ โดยไม่ต้องเริ่มใส่ข้อมูลเบื้องต้นใหม่

มหาวิทยาลัยเทคโนโลยีสุรนารี

## 2.3 ขั้นตอนการทำงานของโปรแกรมแผนภูมิความคลาดเคลื่อน




รูปที่ 2.3 แสดงขั้นตอนการทำงานของโปรแกรมแผนภูมิแผนภูมิความคลาดเคลื่อน

Flow chart ในรูปที่ 2.3 แสดงลำดับการทำงานโดยทั่วไปของโปรแกรมแผนภูมิความคลาดเคลื่อน เริ่มต้นด้วยการรับข้อมูลซึ่งอ้างอิงถึงภาพสเกตช์ของชิ้นงาน และรายละเอียดจากแบบวิศวกรรม (กรอบหมายเลข 1 ถึง 5); จากนั้นนำข้อมูลมาประมวลหาค่าความคลาดเคลื่อนสะสมต่างๆ และ ขนาดมูลฐานของ Basic working dim. (กรอบหมายเลข 6); ผลจากการประมวลในขั้นตอนประมวลผลจะถูกส่งไปแสดงออกเป็นแผนภูมิฯ โดย Outputs Modules (กรอบหมายเลข 7); และ ถ้าเกิดปัญหาความคลาดเคลื่อนสะสม (tolerance-stack problem) ผู้ใช้โปรแกรมสามารถแก้ไขข้อมูลของกระบวนการตัดโลหะได้ (กรอบหมายเลข 8) จากนั้นโปรแกรมจะประมวลผลใหม่ (ทำซ้ำ กรอบหมายเลข 6 และ 7).

## 2.4 สรุป

ในบทนี้ได้กล่าวถึงความเป็นมาของโปรแกรม Visual basic, โครงสร้างโปรแกรมแผนภูมิความคลาดเคลื่อน มีส่วนประกอบหลัก 4 โมดูล ประกอบด้วย;โมดูลรับข้อมูล (Input Modules); โมดูลประมวลผล (Processing Modules); โมดูลแสดงผลลัพธ์ (Output Modules); และโมดูลข้อมูลย้อนกลับ (Feedback Modules) และกล่าวถึงขั้นตอนการทำงานของโปรแกรมแผนภูมิแผนภูมิความคลาดเคลื่อน

มหาวิทยาลัยเทคโนโลยีสุรนารี

# บทที่ 3
# หน้าที่และการทำงานของโมดูล

ในบทนี้กล่าวถึงรายละเอียดการทำหน้าที่ของโมดูลหลักทั้ง 4 โมดูลประกอบด้วย; โมดูลรับข้อมูล (Input Modules); โมดูลประมวลผล (Processing Modules); โมดูลแสดงผลลัพธ์ (Output Modules); และโมดูลข้อมูลย้อนกลับ (Feedback Modules) และอธิบายขั้นตอนการทำงานของโมดูลย่อยต่างๆภายในโปรแกรมแผนภูมิความคลาดเคลื่อน โดยแสดงจนวิธีการ (algorithm) เป็นผังการทำงาน (flow chart) ของโมดูลย่อย

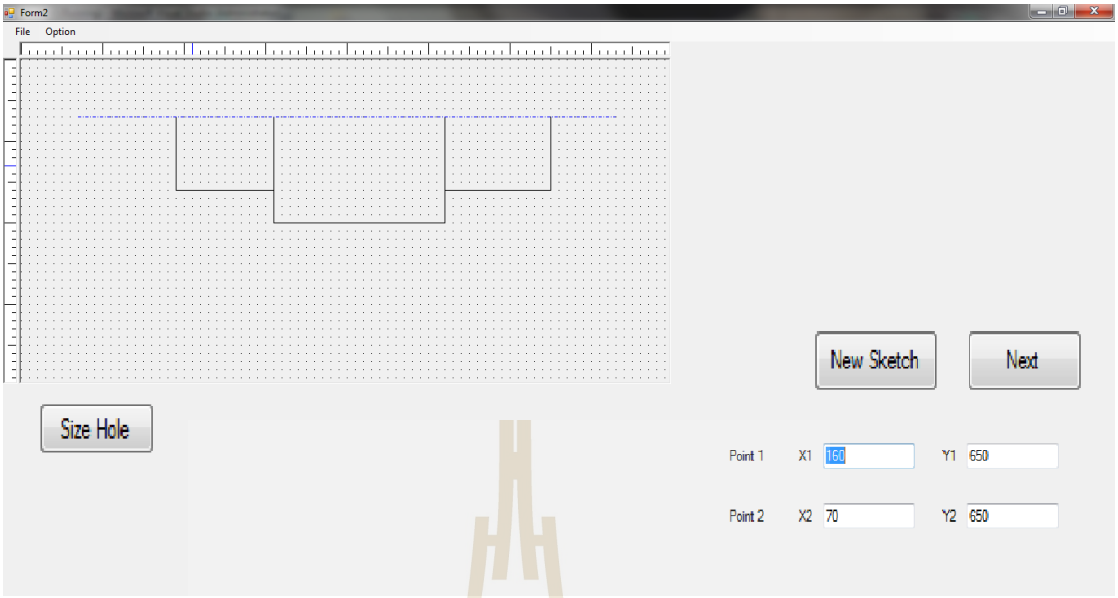
## 3.1 กลุ่มโมดูลรับข้อมูล (Input Modules)

กลุ่มโมดูลรับข้อมูล (Input Modules): ทำหน้าที่เกี่ยวกับการรับข้อมูล สำหรับการสร้างแผนภูมิฯ และเก็บไว้ในตัวแปรต่างๆเพื่อประมวลผลในแต่ละขั้นตอน ต่อไป ในกลุ่มโมดูลนี้แบ่งเป็นกลุ่มโมดูลย่อย 2 กลุ่มคือ: กลุ่มแรกรับค่าต่างๆจากภาพสเกตซ์ชิ้นงาน ได้แก่: รหัสตัวอักษรกำกับผิวงานในแนวดิ่ง, ขนาดของมิติตามแบบของชิ้นงาน (ขนาดมูลฐานและความคลาดเคลื่อน), และ ผิวที่กำหนดขนาดของแต่ละมิติ; ส่วนกลุ่มโมดูลย่อยที่ 2 ทำหน้าที่รับค่าข้อมูลที่เกี่ยวข้องกับการสร้างแผนภูมิความคลาดเคลื่อนเช่น จำนวนขั้นตอนในการตัด เครื่องมือกลที่ใช้ในการผลิตของแต่ละขั้นตอน และผิวอ้างอิงและผิวที่ถูกตัดในแต่ละครั้งของการตัด โมดูลย่อยของโมดูลรับข้อมูลมีจำนวน 4 โมดูล

### 3.1.1 โมดูลรับภาพสเกตซ์ชิ้นงาน 2 มิติ

โมดูลนี้จะอยู่ในขั้นตอนการทำงานที่ 1 ในรูปที่ 2.1 ผู้ใช้งานจะต้องสเกตซ์ภาพชิ้นงานตามแบบกำหนดในพื้นเขียนภาพโดยลักษณะการเขียนภาพชิ้นงานจะเป็นแบบ half sketch ผู้ใช้งานเขียนเส้นตรงบนพื้นที่เขียนภาพด้วยวิธีการกดปุ่มเมาส์ด้านซ้ายค้างจะปรากฏจุดเริ่มต้นของเส้นตรงผู้ใช้งานลากเมาส์ไปในตำแหน่งที่ต้องการแล้วปล่อยปุ่มเมาส์ด้านซ้ายจะเกิดเส้นตรง 1 เส้น

ในโมดูลนี้มีเครื่องมือที่ช่วยสำหรับการลากเส้นตรงให้ง่ายขึ้นคือ การแสดงจุด Grid หากผู้ใช้งานเลือกทำเครื่องหมายถูกในคำสั่ง Show Grid โปรแกรมจะแสดงจุด Grid เพื่อให้ผู้ใช้งานประมาณตำแหน่งในการลากเส้นได้แม่นยำยิ่งขึ้น สำหรับตัวแปรสำคัญที่ใช้งานในโมดูลนี้แสดงดังตารางที่ 3.1




รูปที่ 3.1 แสดงขั้นตอนการทำงานของแผนภูมิแผนภูมิความคลาดเคลื่อน

ตารางที่ 3.1 แสดงข้อมูลของตัวแปรสำคัญในโมดูลรับภาพสเกตซ์ชิ้นงาน 2 มิติ

| ชื่อตัวแปร | ชนิด | ขนาด | หน้าที่ของตัวแปร |
|---|---|---|---|
| NumbersX1( ) | Integer | Array 1 มิติ | เก็บค่าตำแหน่ง X ตอนกดเมาส์ |
| NumbersY1( ) | Integer | Array 1 มิติ | เก็บค่าตำแหน่ง Y ตอนกดเมาส์ |
| NumbersX2( ) | Integer | Array 1 มิติ | เก็บค่าตำแหน่ง X ตอนปล่อยเมาส์ |
| NumbersY2( ) | Integer | Array 1 มิติ | เก็บค่าตำแหน่ง Y ตอนปล่อยเมาส์ |
| Hole( ) | SizeHole | Array 1 มิติ | เก็บข้อมูลของรูเจาะทั้งหมด |

Flow chart ในรูปที่ 3.2 แสดงลำดับขั้นตอนการทำงานของโมดูลรับภาพสเกตซ์ชิ้นงาน 2 มิติ โมดูลนี้โปรแกรมจะทำการบันทึกตำแหน่งของเส้น โดยจะทำการบันทึกตำแหน่งจุดเริ่มต้นลงในตัวแปร NumberX1 และ NumberY1 จากนั้นทำการบันทึกจุดสิ้นสุดของเส้นตรงลงในตัวแปร NumberX2 และ NumberY2 ซึ่งภาพชิ้นงานจะปรากฏเป็นลักษณะในรูปที่ 3.1และรายละเอียดการทำงานของโมเดลเป็นดังนี้:-

(1) กรอบที่ 1 ทำการเขียนเส้นตรงโดยการกดเมาส์ด้านซ้าย โปรแกรมจะทำการบันทึกผลคือ นำตัวแปร searchValue1 และ searchValue2 ไปบวก 1 จากนั้นทำการประกาศตัวแปรโดยเก็บค่าเดิมของ NumberX1() และ NumberY1() เท่ากับค่า searchValue1 และ searchValue2

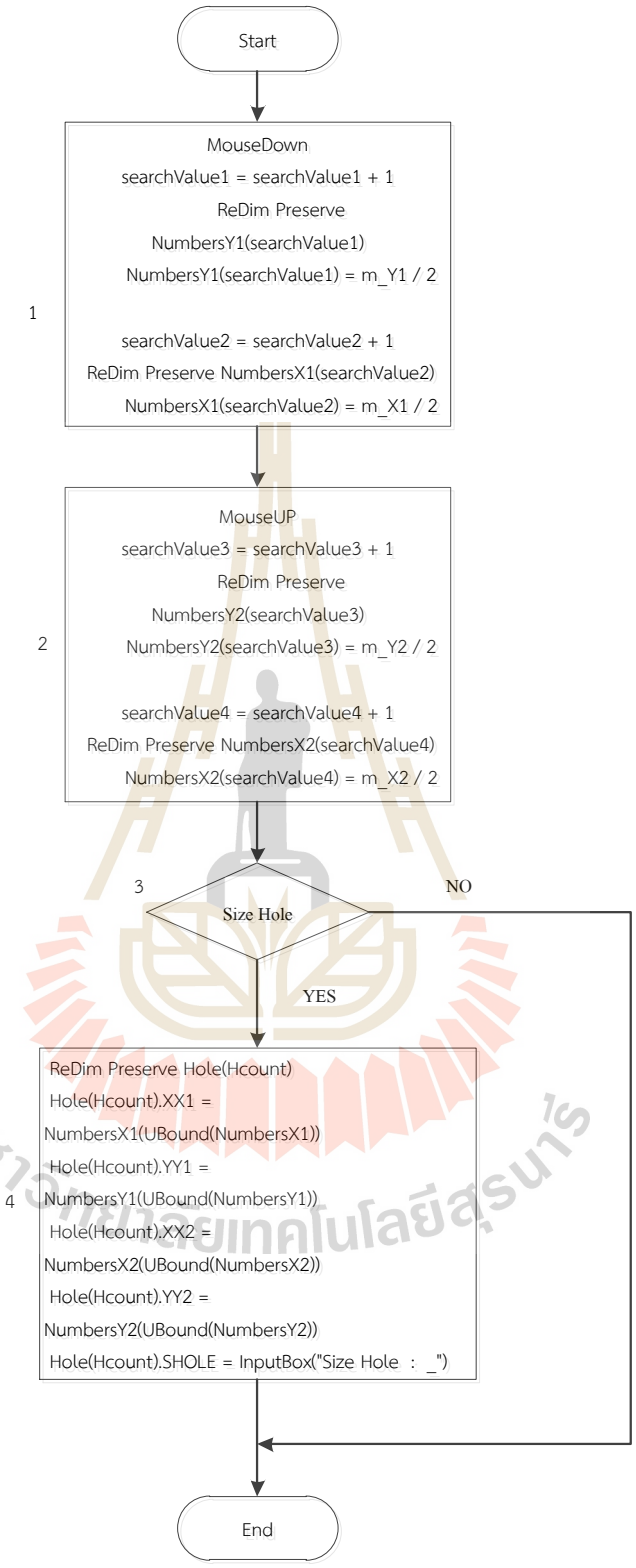
สุดท้ายนำค่าตำแหน่งของ X และ Y บันทึกลงในตัวแปร NumberX1() และ NumberY1() จากนั้นทำในกรอบที่ 2

(2) กรอบที่ 2 เมื่อผู้ใช้งานปล่อยเมาส์ด้านซ้าย โปรแกรมจะทำการบันทึกผลคือ นำตัวแปร searchValue3 และ searchValue3 ไปบวก 1 จากนั้นทำการประกาศตัวแปรโดยเก็บค่าเดิมของ NumberX2() และNumberY2() เท่ากับค่า searchValue3 และ searchValue4 สุดท้ายนำค่าตำแหน่งของ X และ Y บันทึกลงในตัวแปร NumberX2() และ NumberY2() จากนั้นทำในกรอบที่ 2

(3) กรอบที่ 3 เงื่อนไขสำหรับ สร้างรูเจาะบนชิ้นงาน หากผู้ใช้งานต้องการสร้างรูเจาะจะต้องกดปุ่ม SizeHole ที่หน้าจอ จากนั้นทำในกรอบที่ 4 หากไม่ต้องการเจาะรูจบการทำงาน

(4) กรอบที่ 4 โปรแกรมจะทำการบันทึกผลคือ นำตัวแปรนับจำนวน Hcount ไปบวก 1 จากนั้นทำการประกาศตัวแปรโดยเก็บค่าเดิมของ Hole() เท่ากับค่า Hcount สุดท้ายนำค่าตำแหน่งของ X และ Y ของตัวแปร NumberX1, NumberY1, NumberX2, และ NumberY2ตำแหน่งล่าสุด บันทึกลงใน ตัวแปร Hole.XX1(), Hole.YY1(), Hole().XX2 และ Hole().YY2 จากนั้นทำการรับขนาดของรูจาก InputBox นำไปเก็บในตัวแปร Hole().SHOLE

มหาวิทยาลัยเทคโนโลยีสุรนารี




รูปที่ 3.2 Flow chart แสดงลำดับขั้นตอนการทำงานของโมดูลรับภาพสเกตซ์ชิ้นงาน 2 มิติ

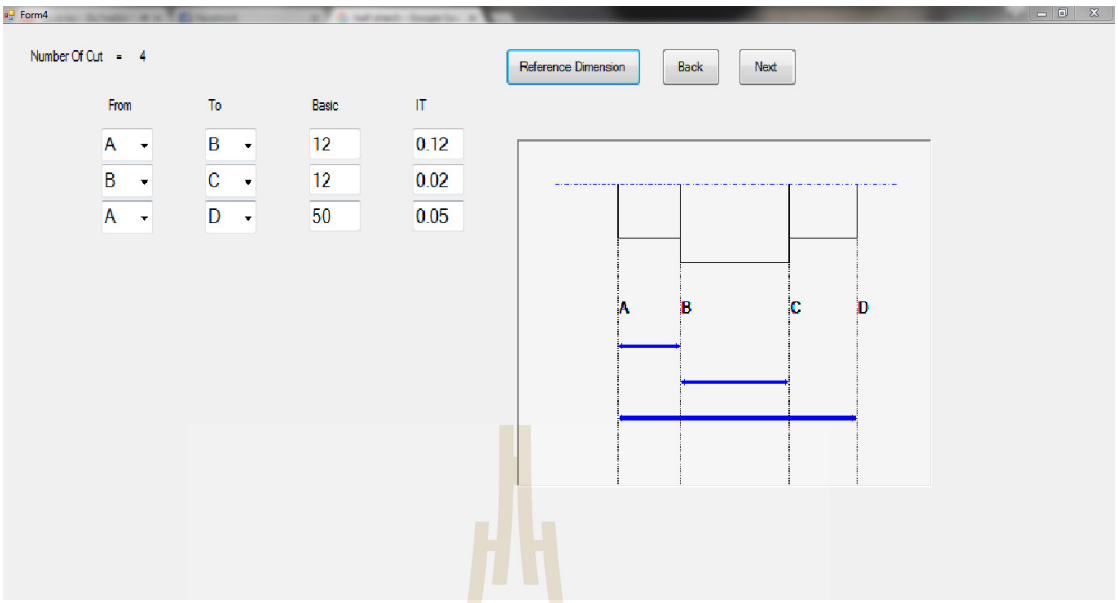
### 3.1.2 โมดูลกำหนดขนาดตามแบบ Blueprint

โมดูลนี้อยู่ในขั้นตอนการทำงานที่ 3 ในรูปที่ 2.1 โปรแกรมจะนำภาพสเกตช์ชิ้นงานมาหาจำนวนผิวในแนวดิ่งแล้วจึงแสดง textbox สำหรับรับข้อมูลขนาดของชิ้นงาน และความคลาดเคลื่อนของขนาด โดยกำหนดจุดเริ่มต้นและจุดสิ้นสุดของของขนาดเลือกตัวหนังสือภาษาอังกฤษพิมพ์ใหญ่ตามตำแหน่งที่ต้องการ(โปรแกรมจะทำการตั้งชื่อผิวในแนวดิ่งเป็นตัวอักษรภาษาอังกฤษโดยกำหนดรอยตัดแรกด้านซ้ายมือของผู้ใช้งานเป็นตัวอักษร A) วิธีประมวลผลจำนวนมิติของชิ้นงานคือ นับจำนวนผิวในแนวดิ่ง -1 จะเป็นจำนวนมิติที่โปรแกรมจะแสดงออกมาเพื่อรับข้อมูลจากแบบชิ้นงาน ดังรูปที่ 3.3 โมดูลนี้มีหน้าที่ 2 ส่วน คือ ส่วนแรกจะเกี่ยวข้องกับการสร้าง box รับข้อมูลทางหน้าจอ และส่วนที่สองจะเป็นการรับข้อมูลค่าของตัวแปรที่เป็นข้อกำหนดในแบบของชิ้นงานโมดูลนี้มีตัวแปรที่สำคัญดังตารางที่ 3.2

ตารางที่ 3.2 แสดงข้อมูลของตัวแปรสำคัญในโมดูลกำหนดขนาดตามแบบ Blueprint

| ชื่อตัวแปร | ชนิด | ขนาด | หน้าที่ของตัวแปร |
|---|---|---|---|
| BluePrint( ).Basic | Integer | Array 1 มิติ | เก็บค่าขนาดของแบบกำหนด |
| BluePrint( )Tolerance | Integer | Array 1 มิติ | เก็บค่าความคลาดเคลื่อนของแบบกำหนด |
| BluePrint( ).FromType | String | Array 1 มิติ | เก็บตำแหน่งจุดเริ่มของขนาดเป็นตัวอักษรภาษาอังกฤษพิมพ์ใหญ่ |
| BluePrint( ).ToType | String | Array 1 มิติ | เก็บตำแหน่งจุดสิ้นสุดของขนาดเป็นตัวอักษรภาษาอังกฤษพิมพ์ใหญ่ |

มหาวิทยาลัยเทคโนโลยีสุรนารี




รูปที่ 3.3 กำหนดขนาดตามแบบ Blueprint

**ส่วนที่ 1** โปรแกรมแกรมทำการสร้าง ComboBox และ textbox รับข้อมูลขนาดของชิ้นงาน และความคลาดเคลื่อนของขนาด โดยนำค่าของตัวแปร NumberofCut มาลบด้วย 2 แล้วนำไปเก็บในค่า k จากนั้นทำการวนลูปเพื่อสร้างคอนโทรล ให้เท่ากับจำนวนค่า k แสดงดังรูป 3.4

Flow chart รูปที่ 3.4 อธิบายขั้นตอนการทำงานของโมดูลดังนี้:-

(1) กรอบที่ 1 นำค่าจากตัวแปร NumberOfCut -2 ไปเก็บในตัวแปร k จากนั้นกำหนดค่า i = 0 และทำการประกาศจำนวนคอนโทรลใหม่ให้เท่ากับค่า k จากนั้นทำในกรอบที่ 2

(2) กรอบที่ 2 ตรวจสอบเงื่อนไขว่า i มากกว่า k หรือไม่ หากเป็นจริงไปทำกรอบที่ 5 หากไม่เป็นจริงทำในกรอบที่ 3

(3) กรอบที่ 3 สร้างคอนโทรลใหม่ในบรรทัดที่ i จากนั้นทำในกรอบที่ 4

(4) กรอบที่ 4 กำหนดค่า i เท่ากับ i + 1 จากนั้นทำในกรอบที่ 2

(5) กรอบที่ 5 กำหนดค่าตัวแปร addnext = 0 จากนั้นทำในกรอบที่ 6

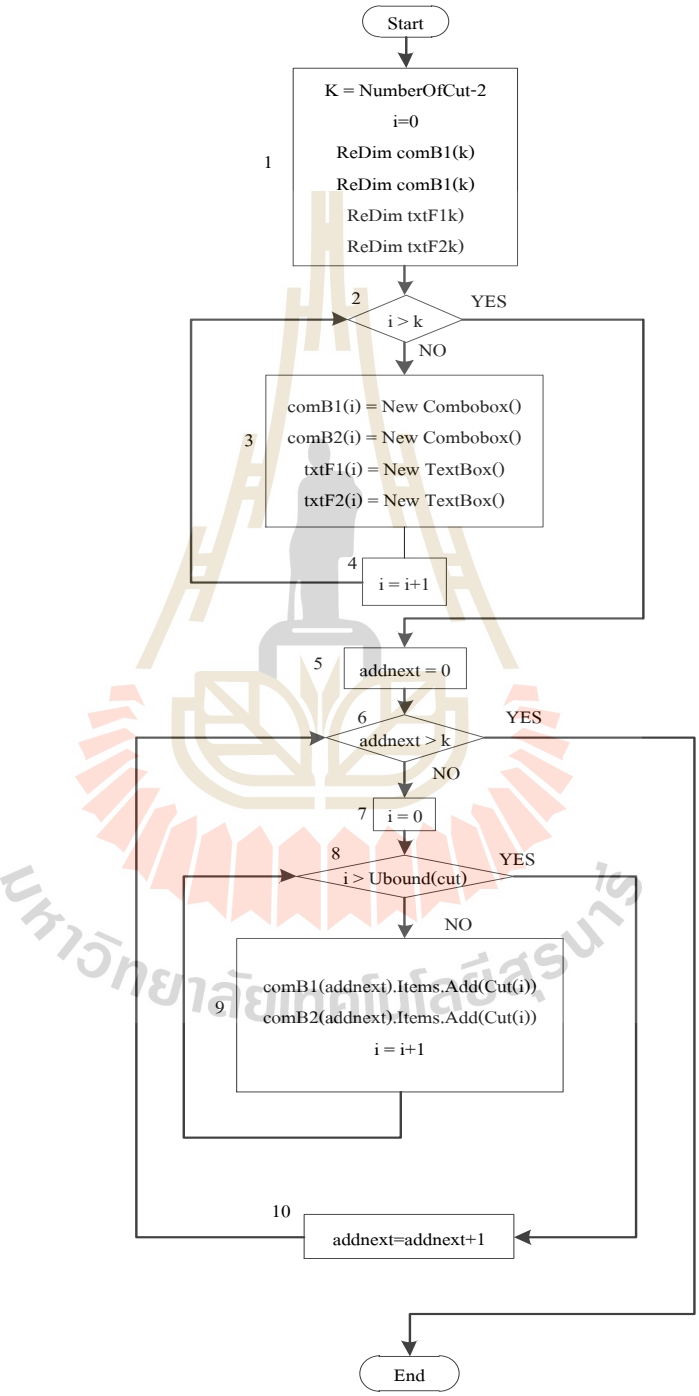
(6) กรอบที่ 6 ตรวจสอบเงื่อนไขว่า addnext มากกว่า k หรือไม่ หากเป็นจริงจบการทำงาน หากไม่เป็นจริงทำในกรอบที่ 7

(7) กรอบที่ 7 กำหนดค่า i = 0 จากนั้นทำในกรอบที่ 8

(8) กรอบที่ 8 ตรวจสอบเงื่อนไขว่า i มากกว่า Ubound(cut) หรือไม่หากเป็นจริงไปทำกรอบที่ 10 ไม่เป็นจริงทำในกรอบที่ 9

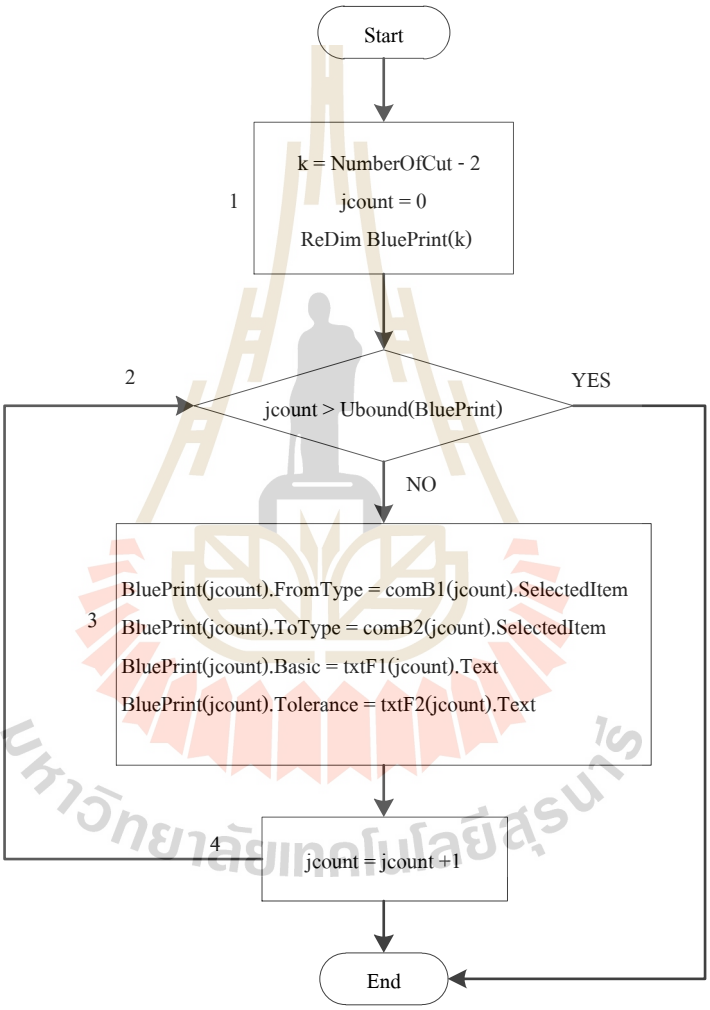
(9) กรอบที่ 9 ใส่ชื่อรอยตัดในแนวดิ่งที่เป็นตัวอักษรภาษาอังกฤษของตัวแปร cut(i) เข้าไปในคอน ComboBox(addnext) กำหนดค่า i = i+1 จากนั้นทำในกรอบที่ 8

(10) กรอบที่ 10 กำหนดค่า addnext เท่ากับ addnext + 1 จากนั้นทำในกรอบที่ 6




รูปที่ 3.4 Flow chart แสดงลำดับขั้นตอนการทำงานโมดูลสร้างกล่องรับข้อมูล

**ส่วนที่ 2** โปรแกรมรับข้อมูลขนาดของชิ้นงานจากแบบเก็บในตัวแปร BluePrint(jcount).Basic; ความคลาดเคลื่อนของขนาดเก็บในตัวแปร BluePrint(jcount).Tolerance; โดยกำหนดจุดเริ่มต้นและจุดสิ้นสุดของของขนาดเลือกตัวอักษรภาษาอังกฤษพิมพ์ใหญ่ตามตำแหน่งที่ต้องการ จุดเริ่มต้นเก็บในตัวแปร BluePrint(jcount).FromType; จุดสิ้นสุดเก็บในตัวแปร BluePrint(jcount).ToType ดัง แสดงได้เป็น flow chart ในรูปที่ 3.5 และมีลำดับขั้นตอนการทำดังนี้




รูปที่ 3.5 Flow chart แสดงลำดับขั้นตอนการรับข้อมูลเกี่ยวกับขนาดตามแบบ Blueprint

(1) กรอบที่ 1 นำค่าจากตัวแปร NumberOfCut -2 ไปเก็บในตัวแปร k จากนั้นกำหนดค่าjcount = 0 และทำการประกาศตัวแปร Blueprint ให้เท่ากับจำนวน k จากนั้นทำในกรอบที่ 2

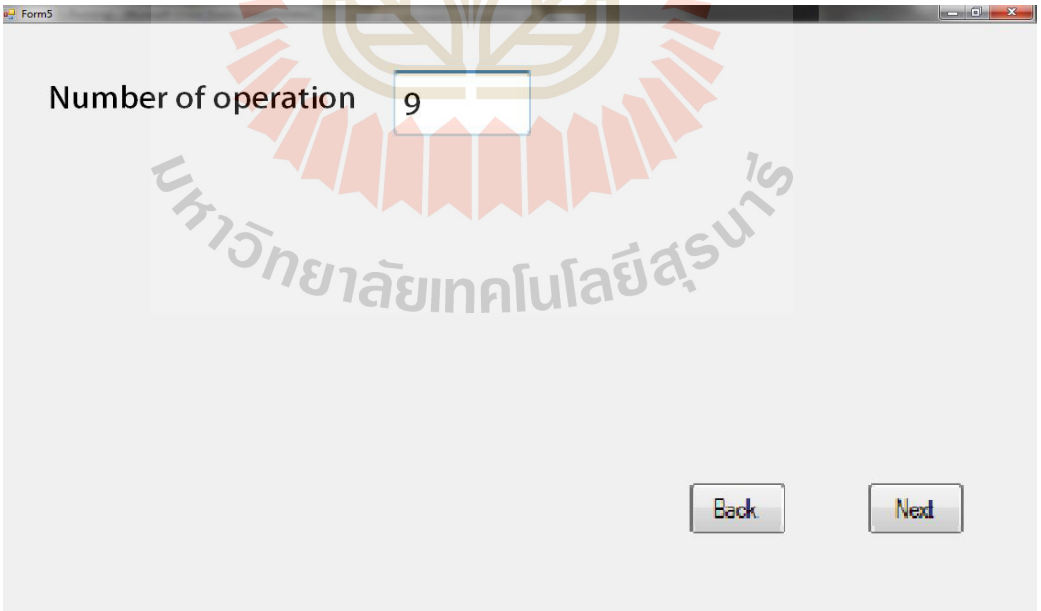
(2) กรอบที่ 2 ตรวจสอบเงื่อนไขว่า jcount มากกว่า Ubound(BluePrint) หรือไม่ หากเป็นจริงจบการทำงาน หากไม่เป็นจริงทำในกรอบที่ 3

(3) กรอบที่ 3 ทำการบันทึกค่าตัวแปร BluePrint(jcount).FromType จากคอนโทรล comB1(jcount).SelectedItem; ทำการบันทึกค่าตัวแปร BluePrint(jcount).ToType จากคอนโทรล comB2(jcount).SelectedItem; ทำการบันทึกค่าตัวแปร BluePrint(jcount).Basic จากคอนโทรล txtF1(jcount).Text; ทำการบันทึกค่าตัวแปร BluePrint(jcount). Tolerance จากคอนโทรล txtF2(jcount).Text จากนั้นทำในกรอบที่ 4

(4) กรอบที่ 4 กำหนดค่า jcount เท่ากับ jcount + 1 จากนั้นทำในกรอบที่ 2

### 3.1.3 โมดูลกำหนดจำนวนขั้นตอนในการตัด

โมดูลนี้อยู่ในขั้นการการทำงานที่ 4 ในรูปที่ 2.1 โปรแกรมจะแสดง textbox ขึ้นมาเพื่อให้ผู้ใช้งานกำหนดจำนวนขั้นตอนในการตัดดังรูป 3.6 ที่กำหนดขั้นตอนในการตัดเป็น 9 ขั้นตอน

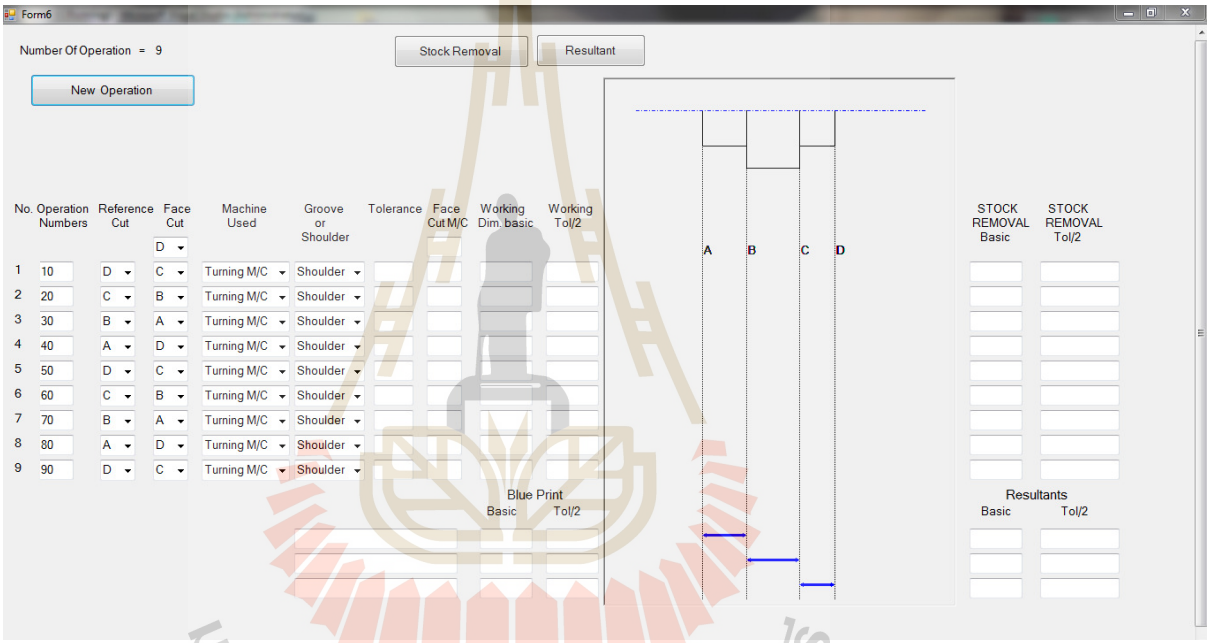



รูปที่ 3.6 แสดงการกำหนดจำนวนขั้นตอนในการตัดของชิ้นงาน

ขั้นตอนการทำงานของโมดูลนี้คือผู้ใช้งานเป็นผู้กำหนดจำนวนขั้นตอนในการตัดลงใน คอนโทรล TextBox1.Text จากนั้นนำค่าที่ได้ไปบันทึกลงในตัวแปร NumberOfOperation

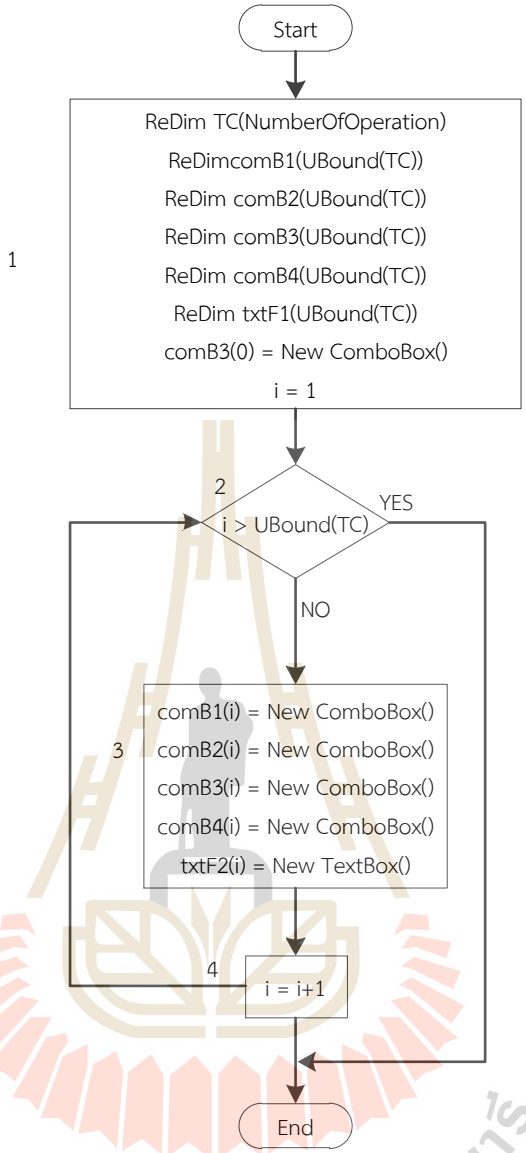
### 3.1.4 โมดูลรับค่าข้อมูลการสร้างแผนภูมความคลาดเคลื่อน

โมดูลนี้ทำหน้าที่รับข้อมูลที่เกี่ยวข้องกับการสร้างแผนภูมิความคลาดเคลื่อนเช่น หมายเลขขั้นตอนการตัด เครื่องมือกลที่ใช้ในการผลิตของแต่ละขั้นตอน ตำแหน่งผิวอ้างอิง ผิวที่ถูกตัดในแต่ละขั้นตอนการตัด ลักษณะของรอยตัด และค่าความคลาดเคลื่อนของรอยตัด โดยโมดูลนี้จะประกอบด้วย 2 ส่วน คล้ายกับโมดูลรับข้อมูล Blueprint ของชิ้นงานคือ




รูปที่ 3.7 แสดงการรับข้อมูลการสร้างแผนภูมความคลาดเคลื่อน

**ส่วนที่ 1** โปรแกรมแกรมทำการสร้าง ComboBox และ textbox รับข้อมูลที่เกี่ยวข้องกับการสร้างแผนภูมิความคลาดเคลื่อนเช่น หมายเลขขั้นตอนการตัด เครื่องมือกลที่ใช้ในการผลิตของแต่ละขั้นตอน ตำแหน่งผิวอ้างอิง ผิวที่ถูกตัดในแต่ละขั้นตอนการตัด ลักษณะของรอยตัด และค่าความคลาดเคลื่อนของรอยตัด ให้เท่ากับจำนวนค่าตัวแปร NumberOfOperation แสดงดังรูป 3.8 และมีขั้นตอนการทำงานสรุปได้ดังนี้:-




รูปที่ 3.8 Flow chart แสดงลำดับขั้นตอนการทำงานรับค่าข้อมูลการสร้างแผนภูมิความคลาดเคลื่อน ส่วนที่ 1

(1) กรอบที่ 1 ประกาศตัวแปร TC เท่ากับค่าตัวแปร NumberOfOperation ประกาศคอนโทรลที่ใช้สำหรับรับค่าข้อมูลการสร้างแผนภูมิความคลาดเคลื่อน สร้างคอนโทรล comB3(0) ในบรรทัดแรก และกำหนดค่า i =1 จากนั้นทำในกรอบที่ 2

(2) กรอบที่ 2 ตรวจสอบเงื่อนไขว่า i มากกว่า Ubound(TC) หรือไม่ หากเป็นจริงจบการทำงาน หากไม่เป็นจริงทำในกรอบที่ 3

(3) กรอบที่ 3 สร้างคอนโทรลใหม่ในบรรทัดที่ i จากนั้นทำในกรอบที่ 4

(4) กรอบที่ 4 กำหนดค่า i เท่ากับ i + 1 จากนั้นทำในกรอบที่ 2

**ส่วนที่ 2** โปรแกรมรับข้อมูลชนิดของเครื่องจักรเก็บในตัวแปร TC(i).Machine; ตำแหน่งจุดอ้างอิงเก็บในตัวแปร TC(i).FromType; ตำแหน่งจุดตัดเก็บในตัวแปร TC(i).ToType; ลักษณะของรอยตัดเก็บในตัวแปร DirectionCut(i); และค่าความคลาดเคลื่อนของรอยตัดเก็บในตัวแปร TC(i).Tolerance แสดงดังรูป 3.9 และมีขั้นตอนการทำงานดังนี้

(1) กรอบที่ 1 ประกาศตัวแปร DirectionCut เท่ากับค่าขอบเขตสูงสุดของตัวแปร TC นำค่าจากcomB3(0).SelectedItem เก็บในตัวแปร TC(0).ToType จากนั้นกำหนด i เท่ากับ 1 จากนั้นทำในกรอบที่ 2

(2) กรอบที่ 2 ตรวจสอบเงื่อนไขว่า i มากกว่า Ubound(TC) หรือไม่ หากเป็นจริงจบการทำงาน หากไม่เป็นจริงทำในกรอบที่ 3

(3) กรอบที่ 3 ทำการบันทึกค่าตัวแปร TC(i).Machine จากคอนโทร comB1(i).SelectedItem; ทำการบันทึกค่าตัวแปร TC(i).FromType จากคอนโทรล comB2(i).SelectedItem; ทำการบันทึกค่าตัวแปร TC(i).ToType จากคอนโทรล comB3(i).SelectedItem; และทำการบันทึกค่าตัวแปร TC(i).Tolerance จากคอนโทรล txtF2(i)

(4) กรอบที่ 4 ตรวจสอบเงื่อนไขว่า comB4 เท่ากับ “Shoulder”(shoulder คือลักษณะการตัดผิวบ่า พิจารณาจากจุดอ้างอิงถึงผิวตัดหากมีการตัดซ้ำตำแหน่งเดิมรอยตัดนั้นจะสั้นลง) หรือไม่ หากเป็นจริงทำในกรอบที่ 5 หากไม่เป็นจริงให้ทำในกรอบที่ 6

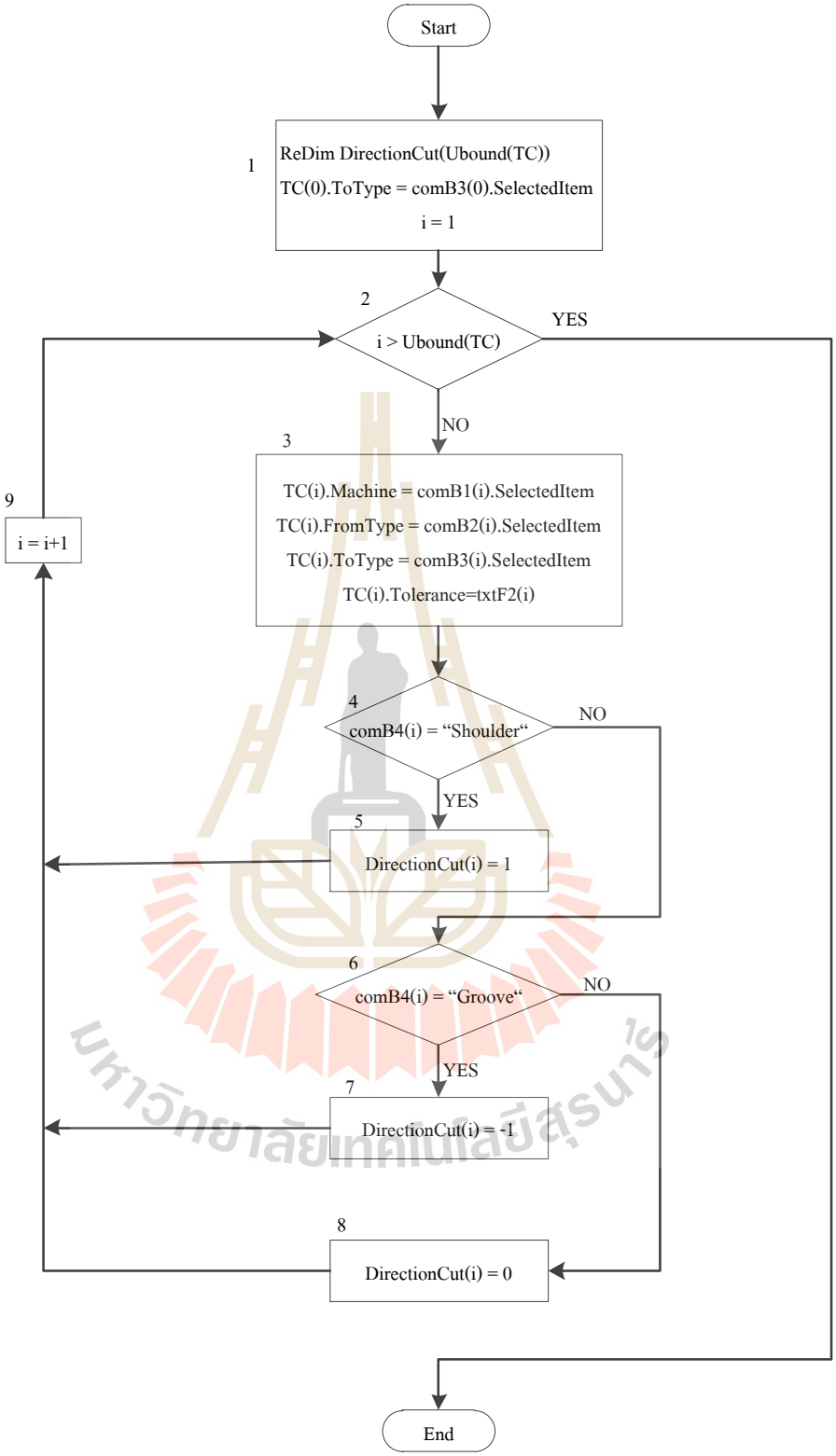
(5) กรอบที่ 5 ทำการบันทึกค่าตัวแปร DirectionCut (i) เท่ากับ 1 จากนั้น ไปทำในกรอบที่ 9

(6) กรอบที่ 6 ตรวจสอบเงื่อนไขว่า comB4 เท่ากับ “Groove” (groove คือลักษณะการตัดผิวร่อง พิจารณาจากจุดอ้างอิงถึงผิวตัดหากมีการตัดซ้ำตำแหน่งเดิมรอยตัดนั้นจะยาวขึ้น) หรือไม่ หากเป็นจริงทำในกรอบที่ 7 หากไม่เป็นจริงให้ทำในกรอบที่ 8

(7) กรอบที่ 7 ทำการบันทึกค่าตัวแปร DirectionCut (i) เท่ากับ -1 จากนั้นทำในกรอบที่ 9

(8) กรอบที่ 8 ทำการบันทึกค่าตัวแปร DirectionCut (i) เท่ากับ 0 จากนั้นทำในกรอบที่ 9

(9) กรอบที่ 9 กำหนดค่า i เท่ากับ i + 1 จากนั้นทำในกรอบที่ 2




รูปที่ 3.9 Flow chart แสดงลำดับขั้นตอนการทำงานรับค่าข้อมูลการสร้างแผนภูมความคลาดเคลื่อน ส่วนที่ 2

## 3.2 กลุ่มโมดูลประมวลผล (Processing Modules)

กลุ่มโมดูลประมวลผล (Processing Modules) มีหน้าที่ประมวลผลตามขั้นตอนของการสร้างแผนภูมิฯ และเป็นกลุ่มโมดูลหลักของโปรแกรม ซึ่งประกอบไปด้วยโมดูลย่อยๆที่ทำหน้าที่ต่างๆกัน นับตั้งแต่การเรียกตัวแปรมาใช้, การกำหนดรหัสผิวชิ้นงานใหม่จากข้อมูล input, สร้างเส้นทางของ rooted tree สำหรับความคลาดเคลื่อนของ stock removal และของ resultant dimensions, จนไปถึงการหาขนาดมูลฐานของ working dimensions. นอกจากนี้ยังประกอบด้วย utility modules อื่นๆ เช่น กำหนดค่าความคลาดเคลื่อนสำหรับการตัดแต่ละครั้ง เป็น IT Grade Number (ในกรณีที่ไม่มีข้อมูลส่วนนี้จากผู้ใช้ หรือผู้ใช้ต้องการให้โปรแกรมกำหนดให้) และโมดูลแสดงความสัมพันธ์ของรอยตัด (cut) เพื่อให้ผู้ใช้ปรับความคลาดเคลื่อน เป็นต้น

### 3.2.1 โมดูลสำหรับประมวลผลรอยตัดของชิ้นงาน

โมดูลนี้มีหน้าที่หาจำนวนรอยตัดของชิ้นงานจากภาพสเกตซ์ชิ้นงาน 2 มิติ โมดูลนี้โปรแกรมจะนำค่าที่ได้จากตัวแปร NumberX1 และ ตัวแปร NumberX2 มาประมวลผลหาจำนวนรอยตัดของรูปชิ้นงาน แล้วนำไปเก็บในตัวแปร NumberOfCut การทำงานของโมดูลนี้จะประกอบด้วย 3 ส่วน และมีตัวแปรที่สำคัญ ดังในตารางที่ 3.3

ตารางที่ 3.3 แสดงข้อมูลของตัวแปรสำคัญในโมดูลสำหรับประมวลผลรอยตัดของชิ้นงาน

| ชื่อตัวแปร | ชนิด | ขนาด | หน้าที่ของตัวแปร |
|---|---|---|---|
| AR1 | ArrayList | Array 1 มิติ | เก็บค่าจากตัวแปร NumberX1() โดยไม่เก็บค่าที่ซ้ำกัน |
| ARY1 | ArrayList | Array 1 มิติ | เก็บค่าจากตัวแปร NumberY1() |
| AR2 | ArrayList | Array 1 มิติ | เก็บค่าจากตัวแปร NumberX2() โดยไม่เก็บค่าที่ซ้ำกัน |
| ARY2 | ArrayList | Array 1 มิติ | เก็บค่าจากตัวแปร NumberY2() |
| S1 | Integer | Array 1 มิติ | นำค่าในตัวแปร AR1 ทั้งหมดมาเก็บในตัวแปร S1 |
| S2 | Integer | Array 1 มิติ | นำค่าในตัวแปร AR2 ทั้งหมดมาเก็บในตัวแปร S2 |
| positionX | Integer | Array 1 มิติ | นำค่าจาก ตัวแปร S1 และ S2 มาเปรียบเทียบกัน หากมีค่าที่เท่ากัน นำมาเก็บในตัวแปร positionX |
| NumberOfCut | Integer | | เก็บจำนวนรอยตัดของชิ้นงาน |

**ส่วนที่ 1** มีหน้าที่นำตำแหน่งแกน X ในการกดเมาส์ด้านซ้ายมาคัดแยกเก็บไว้ในตัวแปร S1 โดยโปรแกรมนำค่าจากตัวแปร NumberX1 มารวมเก็บไว้ในตัวแปร AR1 โดยคัดแยกนำค่าที่ซ้ำกันออก จากนั้นทำการเรียงค่าใหม่จากมากไปน้อย แล้วนำไปเก็บในตัวแปร S1 ดังแสดงใน flow chart ของรูป 3.10 และมีรายละเอียดในการทำงานดังนี้ :-

(1) กรอบที่ 1 กำหนดค่า ndx1 เท่ากับ 0 จากนั้นทำกรอบที่ 2

(2) กรอบที่ 2 ตรวจสอบเงื่อนไขว่า ndx1 มากกว่า Ubound(NumberX1) หรือไม่ หากเป็นจริงให้ทำกรอบที่ 7 หากไม่เป็นจริงทำกรอบที่ 3

(3) กรอบที่ 3 ทำการบันทึกค่าในตัวแปร ARY1 เท่ากับค่าตัวแปร NumberY1

(4) กรอบที่ 4 ตรวจสอบเงื่อนไขว่าตัวแปร NumberX1 มีค่าเหมือนค่าใดค่าหนึ่งที่เก็บในตัวแปร AR1 หรือไม่ หากไม่ไม่ซ้ำค่าใดๆในตัวแปร AR1 ทำกรอบที่ 5 หากมีค่าซ้ำในตัวแปร AR1 ทำกรอบที่ 6

(5) กรอบที่ 5 ทำการบันทึกค่าในตัวแปร AR1 เท่ากับค่าตัวแปร NumberX1

(6) กรอบที่ 6 กำหนดค่า ndx1 เท่ากับ ndx1 + 1 จากนั้นทำกรอบที่ 2

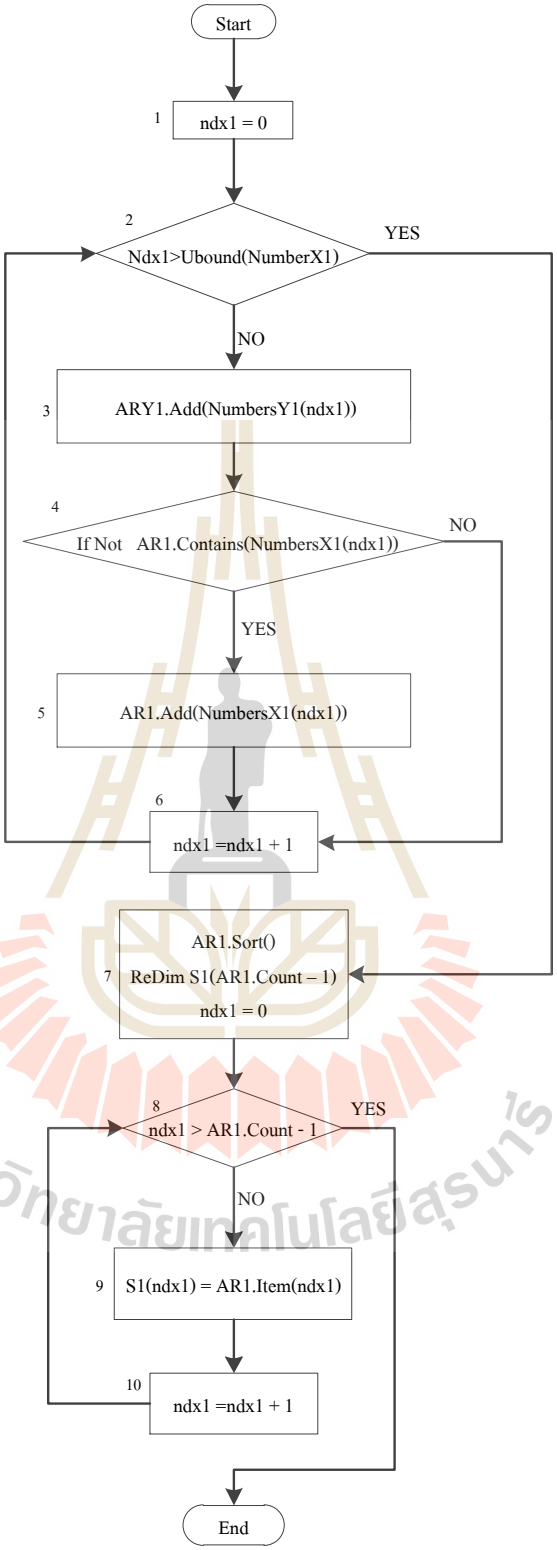
(7) กรอบที่ 7 นำค่าในตัวแปร AR1 เรียงลำดับจากน้อยไปหามาก จากนั้นประกาศตัวแปร S1 เท่ากับค่า AR1.Count-1 และกำหนดค่า ndx1 = 0

(8) กรอบที่ 8 ตรวจสอบเงื่อนไขว่า ndx1 มากกว่า AR1.Count-1 หรือไม่ หากเป็นจริงให้จบการทำงาน หากไม่เป็นจริงให้ทำในกรอบที่ 9

(9) กรอบที่ 9 ทำการบันทึกค่าในตัวแปร S1() เท่ากับค่าตัวแปร AR1.Item()

(10) กรอบที่ 10 กำหนดค่า ndx1 เท่ากับ ndx1 + 1 จากนั้นทำกรอบที่ 8

มหาวิทยาลัยเทคโนโลยีสุรนารี




รูปที่ 3.10 Flow chart แสดงลำดับการทำงานเพื่อนำค่าไปเก็บในตัวแปร S1

**ส่วนที่ 2** มีหน้าที่นำตำแหน่งแกน Y ในการปล่อยเมาส์ด้านซ้ายมาคัดแยกเก็บไว้ในตัวแปร S2 โดยโปรแกรมนำค่าจากตัวแปร NumberX2 มารวมเก็บไว้ในตัวแปร AR2 โดยคัดแยกนำค่าที่ซ้ำกันออก จากนั้นทำการเรียงค่าใหม่จากมากไปน้อย แล้วนำไปเก็บในตัวแปร S2 ดังแสดงในรูป 3.11 โดยมีขั้นตอนการทำงานดังนี้ :-

(1) กรอบที่ 1 กำหนดค่า ndx2 เท่ากับ 0 จากนั้นทำกรอบที่ 2

(2) กรอบที่ 2 ตรวจสอบเงื่อนไขว่า ndx2 มากกว่า Ubound(NumberX2) หรือไม่ หากเป็นจริงให้ทำกรอบที่ 7 หากไม่เป็นจริงทำกรอบที่ 3

(3) กรอบที่ 3 ทำการบันทึกค่าในตัวแปร ARY2 เท่ากับค่าตัวแปร NumberY2

(4) กรอบที่ 4 ตรวจสอบเงื่อนไขว่าตัวแปร NumberX2 มีค่าเหมือนค่าใดค่าหนึ่งที่เก็บในตัวแปร AR2 หรือไม่ หากไม่ไม่ซ้ำค่าใดๆในตัวแปร AR2 ทำกรอบที่ 5 หากมีค่าซ้ำในตัวแปร AR2 ทำกรอบที่ 6

(5) กรอบที่ 5 ทำการบันทึกค่าในตัวแปร AR2 เท่ากับค่าตัวแปร NumberX2

(6) กรอบที่ 6 กำหนดค่า ndx2 เท่ากับ ndx2 + 1 จากนั้นทำกรอบที่ 2

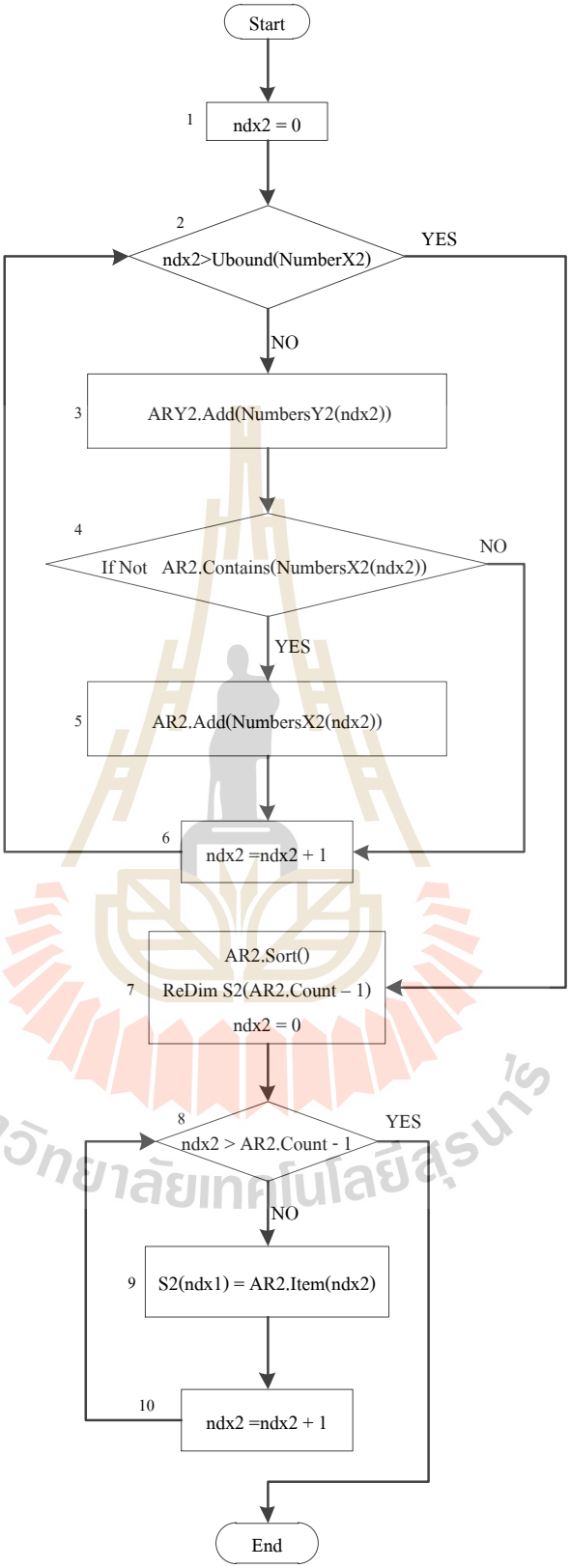
(7) กรอบที่ 7 นำค่าในตัวแปร AR2 เรียงลำดับจากน้อยไปหามาก จากนั้นประกาศตัวแปร S1 เท่ากับค่า AR2.Count-1 และกำหนดค่า ndx1 = 0

(8) กรอบที่ 8 ตรวจสอบเงื่อนไขว่า ndx2 มากกว่า AR2.Count-1 หรือไม่ หากเป็นจริงให้จบการทำงาน หากไม่เป็นจริงให้ทำในกรอบที่ 9

(9) กรอบที่ 9 ทำการบันทึกค่าในตัวแปร S2() เท่ากับค่าตัวแปร AR2.Item()

(10) กรอบที่ 10 กำหนดค่า ndx2 เท่ากับ ndx2 + 1 จากนั้นทำกรอบที่ 8

มหาวิทยาลัยเทคโนโลยีสุรนารี




รูปที่ 3.11 Flow chart แสดงลำดับการทำงานเพื่อนำค่าไปเก็บในตัวแปร S2

**ส่วนที่ 3** มีหน้าที่เปรียบเทียบตัวแปรS1 และ S2 เพื่อหาจำนวนรอยตัดของชิ้นงาน ไปเก็บในตัวแปร NumberOfCut โดยโปรแกรมนำค่าจากตัวแปร S1 และตัวแปร S2 มาเปรียบเทียบกันทีละตัวหากเหมือนกัน k = k+1 จากนั้นเก็บตำแหน่งที่ตัวแปร S1 และตัวแปร S2 ที่เหมือนกันในตัวแปร positionX และสุดท้ายนำค่า k มาเก็บในตัวแปร NumberOfCut ดังในรูป 3.12 และมีลำดับการทำงานดังนี้:-

(1) กรอบที่ 1 กำหนดค่า k เท่ากับ 0 และ a เท่ากับ 0 จากนั้นทำกรอบที่ 2

(2) กรอบที่ 2 ตรวจสอบเงื่อนไขว่า a มากกว่า Ubound(S1) หรือไม่ หากเป็นจริงให้ทำกรอบที่ 9 หากไม่เป็นจริงทำกรอบที่ 3

(3) กรอบที่ 3 กำหนดค่า r เท่ากับ 0 จากนั้นทำกรอบที่ 4

(4) กรอบที่ 4 ตรวจสอบเงื่อนไขว่า r มากกว่า Ubound(S2) หรือไม่ หากเป็นจริงให้ทำกรอบที่ 8 หากไม่เป็นจริงทำกรอบที่ 5

(5) กรอบที่ 5 ตรวจสอบเงื่อนไขว่า S1(a) เท่ากับ S2(r)หรือไม่ หากเป็นจริงให้ทำกรอบที่ 7 หากไม่เป็นจริงทำกรอบที่ 6

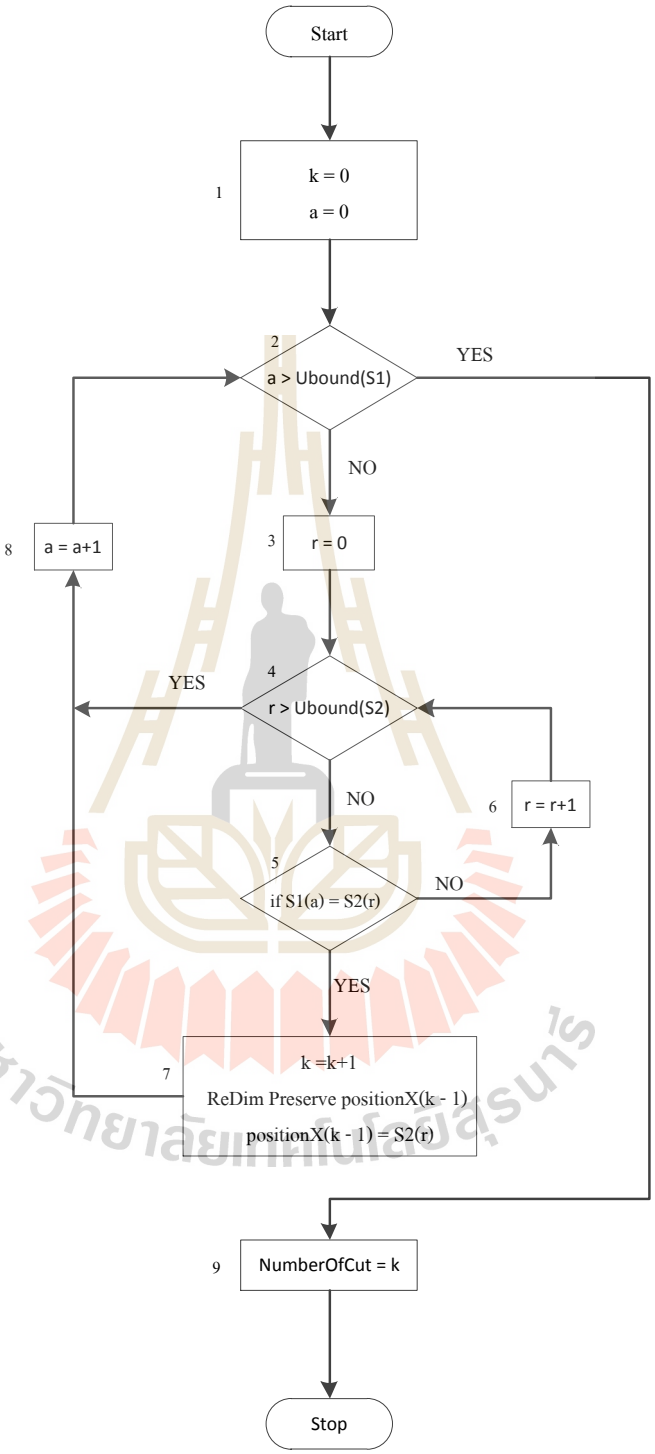
(6) กรอบที่ 6 กำหนดค่า r เท่ากับ r + 1 จากนั้นทำกรอบที่ 4

(7) กรอบที่ 7 กำหนดค่า k เท่ากับ k + 1 จากนั้นทำการประกาศตัวแปร positionX เท่ากับตัวแปร k-1 และสุดท้ายทำการบันทึกค่าในตัวแปร positionX(k-1) เท่ากับค่าตัวแปร S2(r) จากนั้นทำกรอบที่ 8

(8) กรอบที่ 8 กำหนดค่า a เท่ากับ a + 1 จากนั้นทำกรอบที่ 2

(9) กรอบที่ 9 ทำการบันทึกค่าในตัวแปร NumberOfCut เท่ากับค่าตัวแปร k จบการทำงาน

มหาวิทยาลัยเทคโนโลยีสุรนารี




รูปที่ 3.12 Flow chart แสดงลำดับการทำงานเพื่อนำค่าไปเก็บในตัวแปร NumberOfCut

### 3.2.2 โมดูลกำหนดรหัสชื่อผิวชิ้นงานตามลำดับขั้นในการตัดโลหะ

โมดูลนี้มีความสำคัญต่อการสร้างเส้นทาง (path) ของ rooted tree ที่ประกอบด้วยความคลาดเคลื่อนสะสม ระหว่างผิวที่ถูกตัด เพราะว่าในการเขียนโปรแกรม จะไม่มีการสร้าง rooted tree เพื่อหา path ดังเช่นการคำนวณด้วยมือ แต่ถ้าชื่อผิวของชิ้นถูกกำหนดให้เป็นไปตามลำดับตัดโลหะ, rooted tree. แต่ละแขนงก็คือขั้นตอนการตัดโลหะของแต่ละผิว โมดูลนี้จะประกอบด้วย 2 ส่วน และมีตัวแปรที่สำคัญ ดังในตารางที่ 3.4

ตารางที่ 3.4 แสดงข้อมูลของตัวแปรสำคัญในโมดูลกำหนดรหัสชื่อผิวชิ้นงานตามลำดับขั้นในการตัดโลหะ

| ชื่อตัวแปร | ชนิด | ขนาด | หน้าที่ของตัวแปร |
|---|---|---|---|
| TC().ToType | Dimension | Array 1 มิติ | เก็บตัวอักษรภาษาอังกฤษของผิวที่ถูกตัด |
| TC().FromType | Dimension | Array 1 มิติ | เก็บตัวอักษรภาษาอังกฤษของผิวอ้างอิง |
| count1 | Integer | Array 1 มิติ | เก็บค่าตัวเลขตามหลังตัวอักษรของผิวที่ถูกตัด |
| count2 | Integer | Array 1 มิติ | เก็บค่าตัวเลขตามหลังตัวอักษรของผิวที่อ้างอิง |
| TC().ReferenceCut | Dimension | Array 1 มิติ | เก็บรหัสชื่อผิวอ้างอิงตามลำดับขั้นในการตัดโลหะ |
| TC().Facecut | Dimension | Array 1 มิติ | เก็บรหัสชื่อผิวที่ถูกตัดตามลำดับขั้นในการตัดโลหะ |

ตารางที่ 3.5 แสดงผลลัพธ์ของผิวตัดชิ้นงาน

| Operation | Input | | Output | |
|---|---|---|---|---|
| | reference cut | face cut | reference cut | face cut |
| 10 | G | F | G0 | F0 |
| 20 | G | D | G0 | D0 |
| 30 | D | E | D0 | E0 |
| 40 | D | B | D0 | B0 |
| 50 | B | A | B0 | A0 |
| 60 | F | C | F0 | C0 |
| 70 | F | G | F0 | G1 |
| 80 | G | D | G1 | D1 |
| 90 | D | B | D1 | B1 |
| 100 | G | A | G1 | A1 |

มหาวิทยาลัยเทคโนโลยีสุรนารี

โมดูล input จะรับค่าผิวอ้างอิงและผิวที่ถูกตัดจากผู้ใช้โปรแกรมเป็นรหัสอักษรตัวเดียว ซึ่งจะถูกเปลี่ยนให้เป็นโหนดของ rooted tree (รหัสตัวอักษรตามด้วยตัวเลข) ด้วยเงื่อนไขดังนี้: ผิวที่ไม่ถูกตัดซ้ำ ตามด้วยเลข 0; ผิวที่ถูกตัดซ้ำครั้งที่หนึ่ง ตามด้วยเลข 1; ถูกตัดซ้ำครั้งที่สอง ตามด้วยเลข 2, เป็นต้น ในตารางที่ 3.5 แสดงชื่อผิวบนชิ้นงานก่อน และหลัง การเปลี่ยนแปลง สำหรับหน้าที่ของแต่ละส่วนของโมดูล เราสามารถแยกอธิบายได้ดังนี้:-

**ส่วนที่1** มีหน้าที่สร้างผิวชื่อที่ถูกตัด (face cut) ดังตารางที่ 3.5 โดยนำตัวแปร TC().FromType รวมกับตัวแปร count1 ไปเก็บในตัวแปร TC().Facecut แสดงดังในรูป 3.13 และมีขั้นตอนดังนี้:-

อธิบายขั้นตอนการทำงานของโมดูลย่อย แสดงใน Flow chart รูปที่ 3.11

(1) กรอบที่ 1 กำหนดค่า m เท่ากับ Ubound(TC) และนั้นทำการประกาศตัวแปร count1เท่ากับตัวแปร m จากนั้นทำกรอบที่ 2

(2) กรอบที่ 2 กำหนดค่า i เท่ากับ 0 จากนั้นทำกรอบที่ 3

(3) กรอบที่ 3 ตรวจสอบเงื่อนไขว่า i มากกว่า m หรือไม่ หากเป็นจริงจบการทำงาน หากไม่เป็นจริงทำกรอบที่ 4

(4) กรอบที่ 4 กำหนดค่า a เท่ากับ -1 จากนั้นทำกรอบที่ 5

(5) กรอบที่ 5 กำหนดค่า j เท่ากับ 0 จากนั้นทำกรอบที่ 6

(6) กรอบที่ 6 ตรวจสอบเงื่อนไขว่า j มากกว่า i หรือไม่ หากเป็นจริงให้ทำกรอบที่ 10 หากไม่เป็นจริงทำกรอบที่ 7

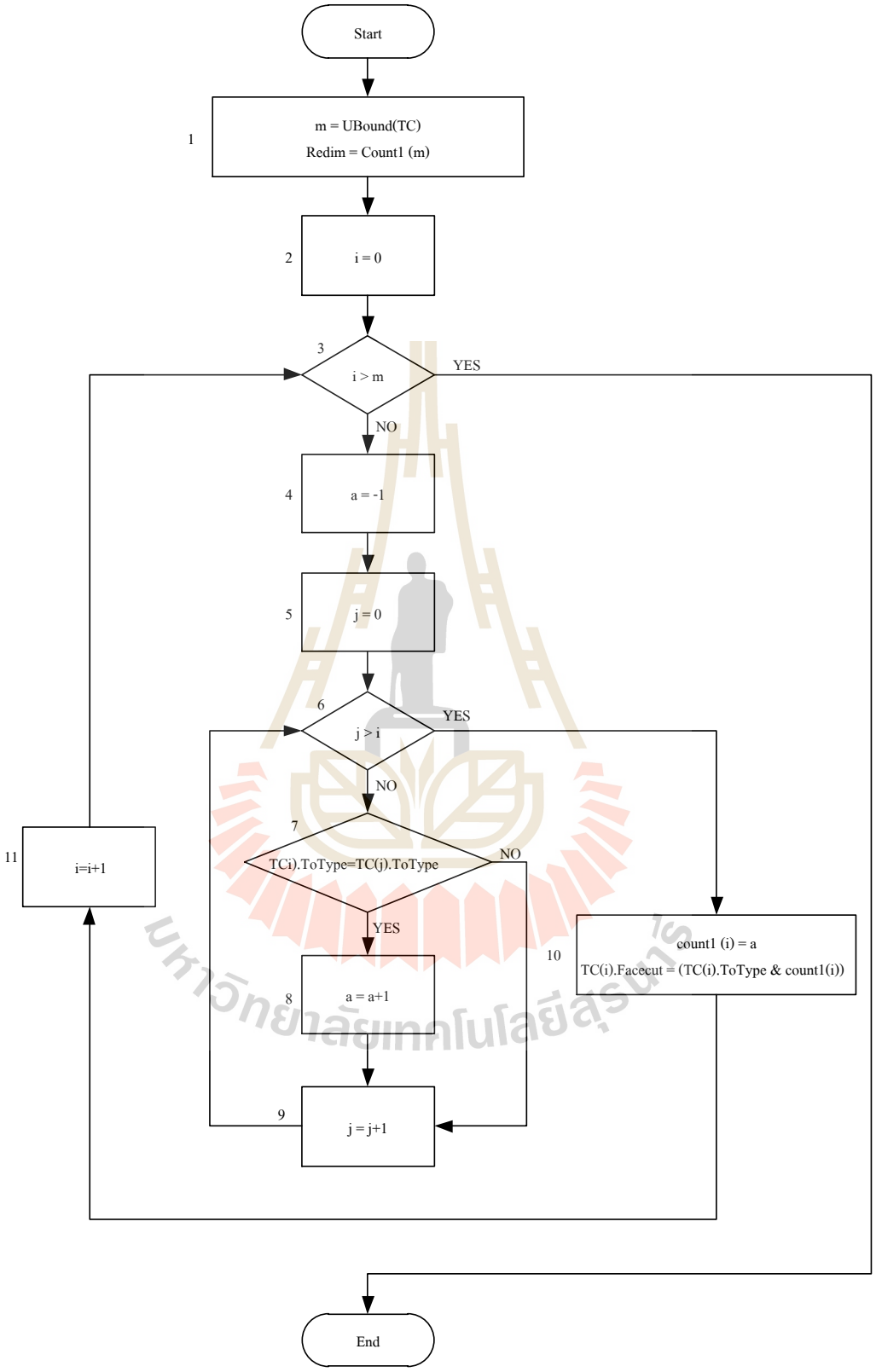
(7) กรอบที่ 7 ตรวจสอบเงื่อนไขว่า TCi).ToType เท่ากับ TC(j).ToType หรือไม่ หากเป็นจริงให้ทำกรอบที่ 8 หากไม่เป็นจริงทำกรอบที่ 9

(8) กรอบที่ 8 กำหนดค่า a เท่ากับ a + 1 จากนั้นทำกรอบที่ 9

(9) กรอบที่ 9 กำหนดค่า j เท่ากับ j + 1 จากนั้นทำกรอบที่ 6

(10) กรอบที่ 10 ทำการบันทึกค่าในตัวแปร count1(i) และนำตัวแปร TC(i).FromType รวมกับตัวแปร count1(i) ไปเก็บในตัวแปร TC(i).Facecut จากนั้นทำกรอบที่ 11

(11) กรอบที่ 11 กำหนดค่า i เท่ากับ i + 1 จากนั้นทำกรอบที่ 3




รูปที่ 3.13 Flow chart แสดงขั้นตอนการทำงานของโมดูลประมวลผลผิวตัด

**ส่วนที่ 2** มีหน้าที่สร้างชื่อผิวที่อ้างอิง (reference cut) ดังตารางที่ 3.5 โดยนำตัวแปร TC().ToType รวมกับตัวแปร coun2ไปเก็บในตัวแปร TC(i).ReferenceCut แสดงดังในรูป 3.14 และมีรายละเอียดดังนี้ :-

(1) กรอบที่ 1 กำหนดค่า m เท่ากับ Ubound(TC) และนั้นทำการประกาศตัวแปร count2เท่ากับตัวแปร m จากนั้นทำกรอบที่ 2

(2) กรอบที่ 2 กำหนดค่า i เท่ากับ 0 จากนั้นทำกรอบที่ 3

(3) กรอบที่ 3 ตรวจสอบเงื่อนไขว่า i มากกว่า m หรือไม่ หากเป็นจริงจบการทำงาน หากไม่เป็นจริงทำกรอบที่ 4

(4) กรอบที่ 4 กำหนดค่า a เท่ากับ -1 จากนั้นทำกรอบที่ 5

(5) กรอบที่ 5 กำหนดค่า j เท่ากับ 0 จากนั้นทำกรอบที่ 6

(6) กรอบที่ 6 ตรวจสอบเงื่อนไขว่า j มากกว่า i หรือไม่ หากเป็นจริงให้ทำกรอบที่ 10 หากไม่เป็นจริงทำกรอบที่ 7

(7) กรอบที่ 7 ตรวจสอบเงื่อนไขว่า TCi).FromType เท่ากับ TC(j).ToType หรือไม่ หากเป็นจริงให้ทำกรอบที่ 8 หากไม่เป็นจริงทำกรอบที่ 9

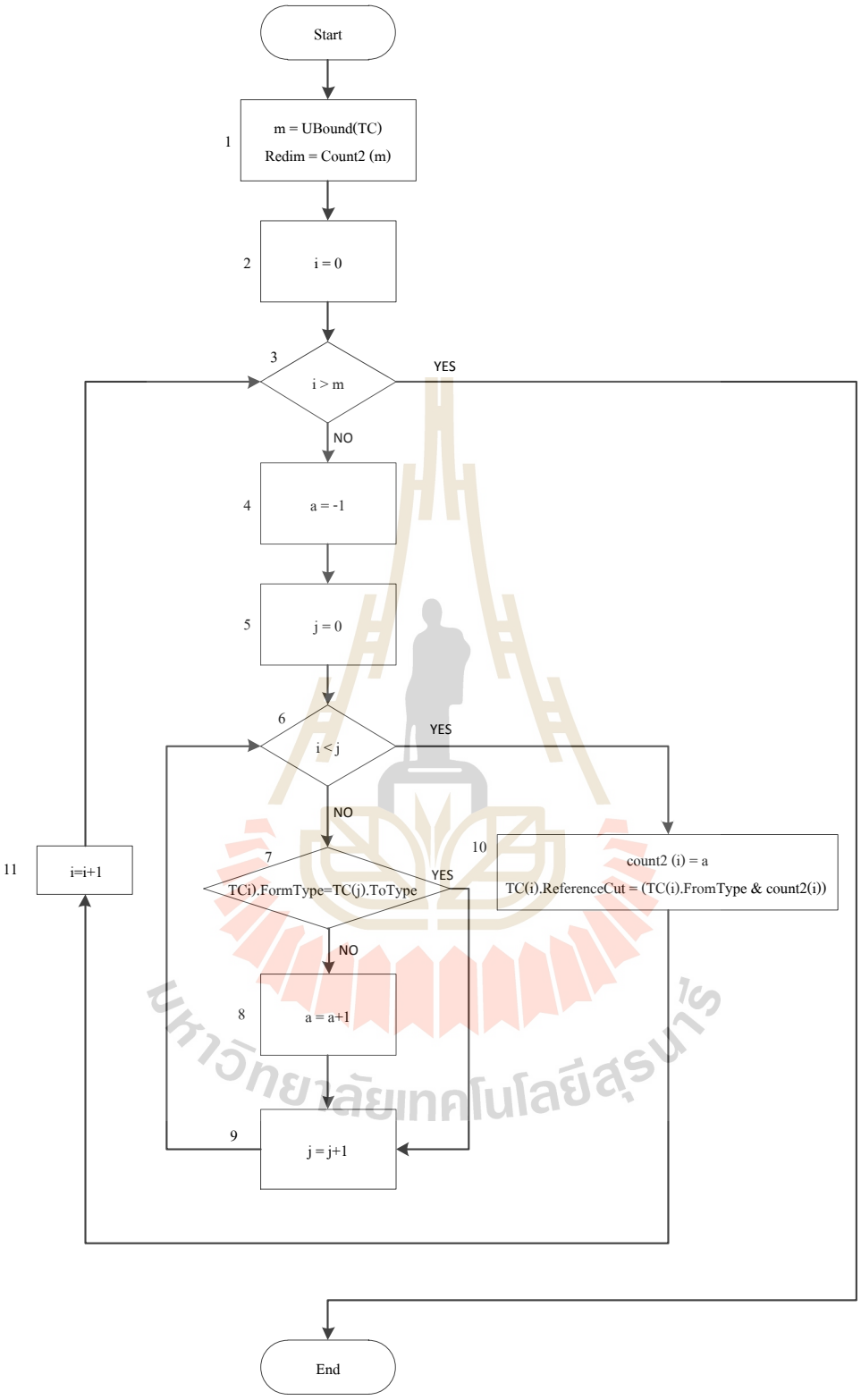
(8) กรอบที่ 8 กำหนดค่า a เท่ากับ a + 1 จากนั้นทำกรอบที่ 9

(9) กรอบที่ 9 กำหนดค่า j เท่ากับ j + 1 จากนั้นทำกรอบที่ 6

(10) กรอบที่ 10 ทำการบันทึกค่าในตัวแปร count2(i) และนำตัวแปร TC(i).FromType รวมกับตัวแปร count2(i) ไปเก็บในตัวแปร TC(i).ReferenceCut จากนั้นทำกรอบที่ 11

(11) กรอบที่ 11 กำหนดค่า i เท่ากับ i + 1 จากนั้นทำกรอบที่ 3

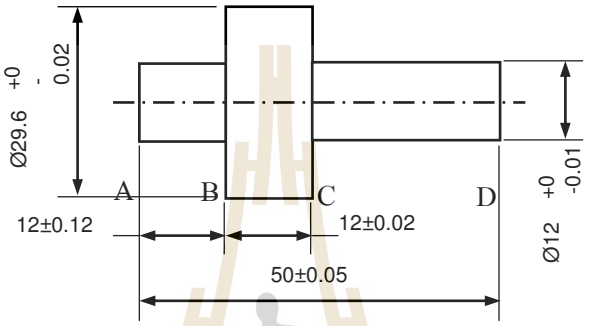
มหาวิทยาลัยเทคโนโลยีสุรนารี




รูปที่ 3.14 Flow chart แสดงขั้นตอนการทำงานของโมดูลประมวลผลผิวอ้างอิง

### 3.2.3 โมดูลคำนวณหาขนาดมูลฐานของชิ้นงาน

โมดูลนี้มีหน้าที่คำนวณหาขนาดมูลฐานเพื่อนำค่ามาใส่ในตัวแปร GenBP(),() มีขนาด 2 มิติ แต่ละมิติมีค่าเท่ากับ NoSurf โดยคำนวณค่าจากแบบกำหนดที่เก็บในตัวแปร blueprint รูปที่ 3.15 แสดงแบบตัวอย่างประกอบการคำนวณ โมดูลนี้จะประกอบด้วย 3 ส่วน และมีตัวแปรที่สำคัญ ดังแสดงในตารางที่ 3.6




รูปที่ 3.15 แสดงแบบของชิ้นงานเพื่อประกอบการคำนวณ

ตารางที่ 3.6 แสดงข้อมูลของตัวแปรสำคัญในโมดูลคำนวณหาขนาดมูลฐานของชิ้นงาน

| ชื่อตัวแปร | ชนิด | ขนาด | หน้าที่ของตัวแปร |
|---|---|---|---|
| NoSurf | Integer | | เก็บค่าจากตัวแปร NumberOfCut |
| BluePrint().FromType | Dimension | Array 1 มิติ | เก็บตำแหน่งจุดเริ่มต้นของขนาดเป็นตัวอักษร |
| BluePrint().ToType | Dimension | Array 1 มิติ | เก็บตำแหน่งจุดสิ้นสุดของขนาดเป็นตัวอักษร |
| F | Integer | | เก็บตำแหน่งจุดเริ่มต้นของขนาดเป็นตัวเลข |
| T | Integer | | เก็บตำแหน่งจุดสิ้นสุดของขนาดเป็นตัวเลข |
| BluePrint().Basic | Integer | Array 1 มิติ | เก็บขนาดจากแบบ |
| GenBP(),() | single | Array 2 มิติ | เก็บขนาดจากแบบรับค่าจาก BluePrint().Basic |

**ส่วนที่1** มีหน้าที่นำค่าของขนาด (basic) จากแบบ (blue print) มาเก็บในตัวแปร GenBP(),() โดยเก็บค่าของขนาดติดลบแบบสลับตำแหน่งระหว่าง ตัวแปร F และ T แสดงผลลัพธ์ส่วนที่แรเงาดังรูปที่ 3.16 ส่วนรูปที่ 3.17 แสดงขั้นตอนการทำงาน และมีรายละเอียดดังนี้ :-

GenBP(4,4)

| | A=1 | B=2 | C=3 | D=4 |
|---|---|---|---|---|
| A=1 | | 12 | | 50 |
| B=2 | -12 | | 12 | |
| C=3 | | -12 | | |
| D=4 | -50 | | | |

รูปที่ 3.16 แสดงตัวอย่างการคำนวณส่วนที่ 1

(1) กรอบที่ 1 กำหนดค่า NoSurf เท่ากับ NumberOfCut; ประกาศตัวแปรj, n, F, T, a, และ b; ประกาศตัวแปร GenBP(NoSurf, NoSurf) และกำหนดค่า i เท่ากับ 0 จากนั้นทำกรอบที่ 2

(2) กรอบที่ 2 ตรวจสอบเงื่อนไขว่า i มากกว่า NoSurf – 2 หรือไม่ หากเป็นจริงจบการทำงาน หากไม่เป็นจริงทำกรอบที่ 3

(3) กรอบที่ 3 แปลงตัวอักษรเป็นตัวเลขโดยใช้คำสั่ง Asc(BluePrint(i).FromType) – 64 เก็บในตัวแปร F เป็นตำแหน่งจุดเริ่มต้นของขนาด; แปลงตัวอักษรเป็นตัวเลขโดยใช้คำสั่ง Asc(BluePrint(i).FromType) – 64 เก็บในตัวแปร T เป็นตำแหน่งจุดสิ้นสุดของขนาด จากนั้นทำกรอบที่ 4

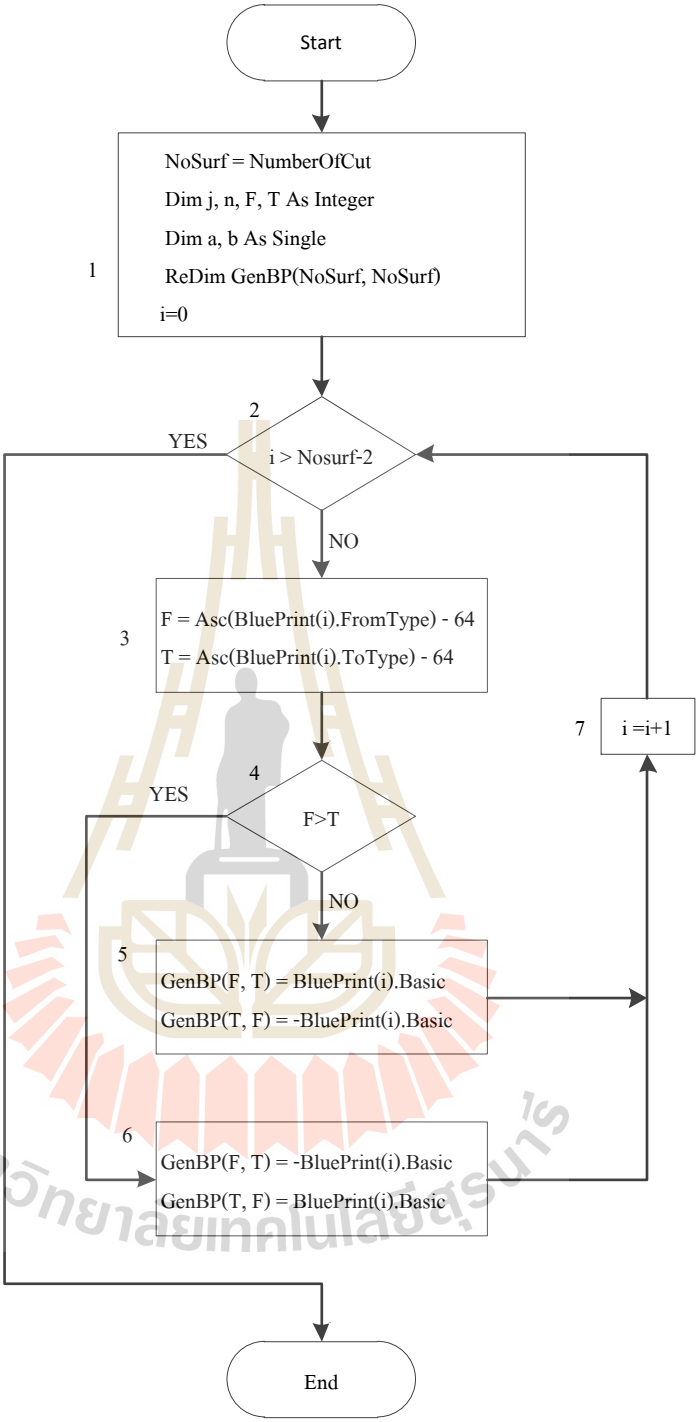
(4) กรอบที่ 4 ตรวจสอบเงื่อนไขว่า F มากกว่า T หรือไม่ หากเป็นจริงให้ทำกรอบที่ 6 หากไม่เป็นจริงทำกรอบที่ 5

(5) กรอบที่ 5 กำหนดค่า GenBP(F, T) เท่ากับ BluePrint(i).Basic และกำหนดค่า GenBP(T, F) เท่ากับ - BluePrint(i).Basic จากนั้นทำกรอบที่ 7

(6) กรอบที่ 6 กำหนดค่า GenBP(F, T) เท่ากับ -BluePrint(i).Basic และกำหนดค่า GenBP(T, F) เท่ากับ BluePrint(i).Basic จากนั้นทำกรอบที่ 7

(7) กรอบที่ 7 กำหนดค่า i เท่ากับ i + 1 จากนั้นทำกรอบที่ 2

มหาวิทยาลัยเทคโนโลยีสุรนารี




รูปที่ 3.17 Flow chart แสดงขั้นตอนการทำงานของโมดูลคำนวณหาขนาดมูลฐานของ ส่วนที่ 1 ชิ้นงาน

**ส่วนที่ 2** มีหน้าที่นำค่าของขนาด (basic) จากแบบ (blue print) มาคำนวณเก็บในตัวแปร GenBP(),()แสดงผลลัพธ์ส่วนที่แรเงาดังรูปที่ 3.18 ส่วนรูปที่ 3.19 แสดงขั้นตอนการทำงานและมีรายละเอียดดังนี้ :-

GenBP(4,4)

| | A=1 | B=2 | C=3 | D=4 |
|---|---|---|---|---|
| A=1 | 0 | 12 | 24 | 50 |
| B=2 | -12 | 0 | 12 | |
| C=3 | -24 | -12 | | |
| D=4 | -50 | | | |

รูปที่ 3.18 แสดงตัวอย่างการคำนวณส่วนที่ 2

(1) กรอบที่ 1 กำหนดค่า i เท่ากับ 0 จากนั้นทำกรอบที่ 2

(2) กรอบที่ 2 ตรวจสอบเงื่อนไขว่า i มากกว่า NoSurf – 2 หรือไม่ หากเป็นจริงจบการทำงาน หากไม่เป็นจริงทำกรอบที่ 3

(3) กรอบที่ 3 แปลงตัวอักษรเป็นตัวเลขโดยใช้คำสั่ง Asc(BluePrint(i).FromType) – 64 เก็บในตัวแปร F เป็นตำแหน่งจุดเริ่มต้นของขนาด; แปลงตัวอักษรเป็นตัวเลขโดยใช้คำสั่ง Asc(BluePrint(i).FromType) – 64 เก็บในตัวแปร T เป็นตำแหน่งจุดสิ้นสุดของขนาด และกำหนดค่า r เท่ากับ 0 จากนั้นทำกรอบที่ 4

(4) กรอบที่ 4 ตรวจสอบเงื่อนไขว่า r มากกว่า NoSurf หรือไม่ หากเป็นจริงให้ทำกรอบที่ 9 หากไม่เป็นจริงทำกรอบที่ 5

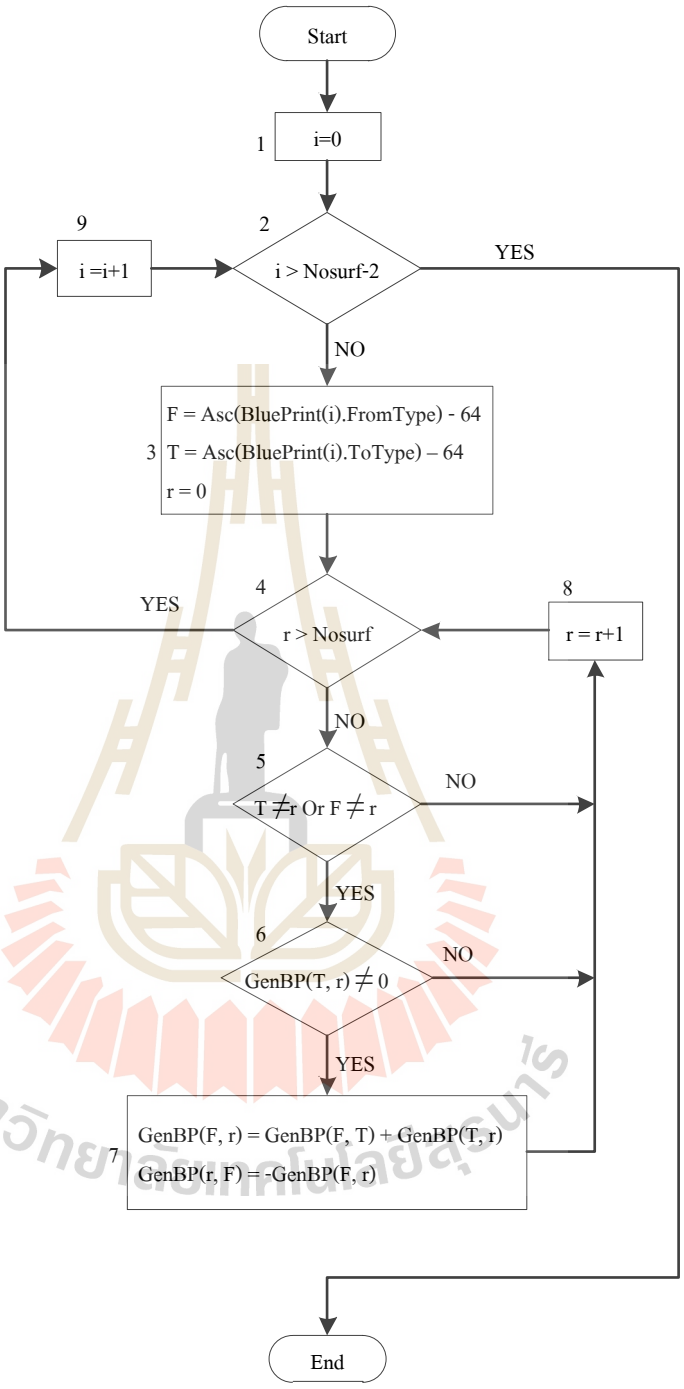
(5) กรอบที่ 5 ตรวจสอบเงื่อนไขว่า T ไม่เท่ากับ r หรือ F ไม่เท่ากับ r หากเป็นจริงให้ทำกรอบที่ 6 หากไม่เป็นจริงทำกรอบที่ 8

(6) กรอบที่ 6 ตรวจสอบเงื่อนไขว่า GenBP(T,r) ไม่เท่ากับ 0 หากเป็นจริงให้ทำกรอบที่ 67หากไม่เป็นจริงทำกรอบที่ 8

(7) กรอบที่ 7 กำหนดค่า GenBP(F, r) เท่ากับ GenBP(F, T)+ GenBP(T, r) และกำหนดค่า GenBP(r, F) เท่ากับ - GenBP(F, r) จากนั้นทำกรอบที่ 8

(8) กรอบที่ 8 กำหนดค่า r เท่ากับ r + 1 จากนั้นทำกรอบที่ 4

(9) กรอบที่ 9 กำหนดค่า i เท่ากับ i + 1 จากนั้นทำกรอบที่ 2




รูปที่ 3.19 Flow chart แสดงขั้นตอนการทำงานของโมดูลคำนวณหาขนาดมูลฐานของ ส่วนที่ 2 ชิ้นงาน

**ส่วนที่ 3** มีหน้าที่นำค่าของขนาด (basic) จากแบบ (blue print) มาคำนวณเก็บในตัวแปร GenBP(,)แสดงผลลัพธ์ส่วนที่แรเงาดังรูปที่ 3.20 ส่วนรูปที่ 3.21 แสดงขั้นตอนการทำงานโดยมีลำดับขั้นดังนี้:-

GenBP(4,4)

| | A=1 | B=2 | C=3 | D=4 |
|---|---|---|---|---|
| A=1 | 0 | 12 | 24 | 50 |
| B=2 | -12 | 0 | 12 | 38 |
| C=3 | -24 | -12 | 0 | 26 |
| D=4 | -50 | -38 | -26 | 0 |

รูปที่ 3.20 แสดงตัวอย่างการคำนวณส่วนที่ 3

(1) กรอบที่ 1 กำหนดค่า i เท่ากับ 1 และ กำหนดค่า j เท่ากับ 1 จากนั้นทำกรอบที่ 2

(2) กรอบที่ 2 ตรวจสอบเงื่อนไขว่า i มากกว่า NoSurf หรือไม่ หากเป็นจริงจบการทำงาน หากไม่เป็นจริงทำกรอบที่ 3

(3) กรอบที่ 3 ตรวจสอบเงื่อนไขว่า j มากกว่า NoSurf หรือไม่ หากเป็นจริงทำกรอบที่ 4 หากไม่เป็นจริงทำกรอบที่ 5

(4) กรอบที่ 4 กำหนดค่า i เท่ากับ i + 1 จากนั้นทำกรอบที่ 2

(5) กรอบที่ 5 ตรวจสอบเงื่อนไขว่า i เท่ากับ j หรือไม่ หากเป็นจริงให้ทำกรอบที่ 7 หากไม่เป็นจริงทำกรอบที่ 6

(6) กรอบที่ 6 ตรวจสอบเงื่อนไขว่า GenBP(i,j) เท่ากับ 0 หรือไม่ หากเป็นจริงให้ทำกรอบที่ 18 หากไม่เป็นจริงทำกรอบที่ 8

(7) กรอบที่ 7 กำหนดค่า GenBP(i,j) เท่ากับ 0 จากนั้นทำกรอบที่ 18

(8) กรอบที่ 8 กำหนดค่า n เท่ากับ 1; a เท่ากับ 0; และ b เท่ากับ 0 จากนั้นทำกรอบที่ 9

(9) กรอบที่ 9 ตรวจสอบเงื่อนไขว่า GenBP(i,n) ต้องไม่เท่ากับ 0 หรือไม่ หากเป็นจริงให้ทำกรอบที่ 10 หากไม่เป็นจริงทำกรอบที่ 13

(10) กรอบที่ 10 กำหนดค่า a เท่ากับ GenBP(i,n) จากนั้นทำกรอบที่ 11

(11) กรอบที่ 11 ตรวจสอบเงื่อนไขว่า GenBP(n,j) ต้องไม่เท่ากับ 0 หรือไม่ หากเป็นจริงให้ทำกรอบที่ 12 หากไม่เป็นจริงทำกรอบที่ 13

(12) กรอบที่ 12 กำหนดค่า b เท่ากับ GenBP(n,j) จากนั้นทำกรอบที่ 13

(13) กรอบที่ 13 กำหนดค่า n เท่ากับ n + 1 จากนั้นทำกรอบที่ 14

(14) กรอบที่ 14 ตรวจสอบเงื่อนไขว่า a ไม่เท่ากับ 0 และ b ไม่เท่ากับ 0 หากเป็นจริงให้ทำกรอบที่ 15 หากไม่เป็นจริงทำกรอบที่ 9

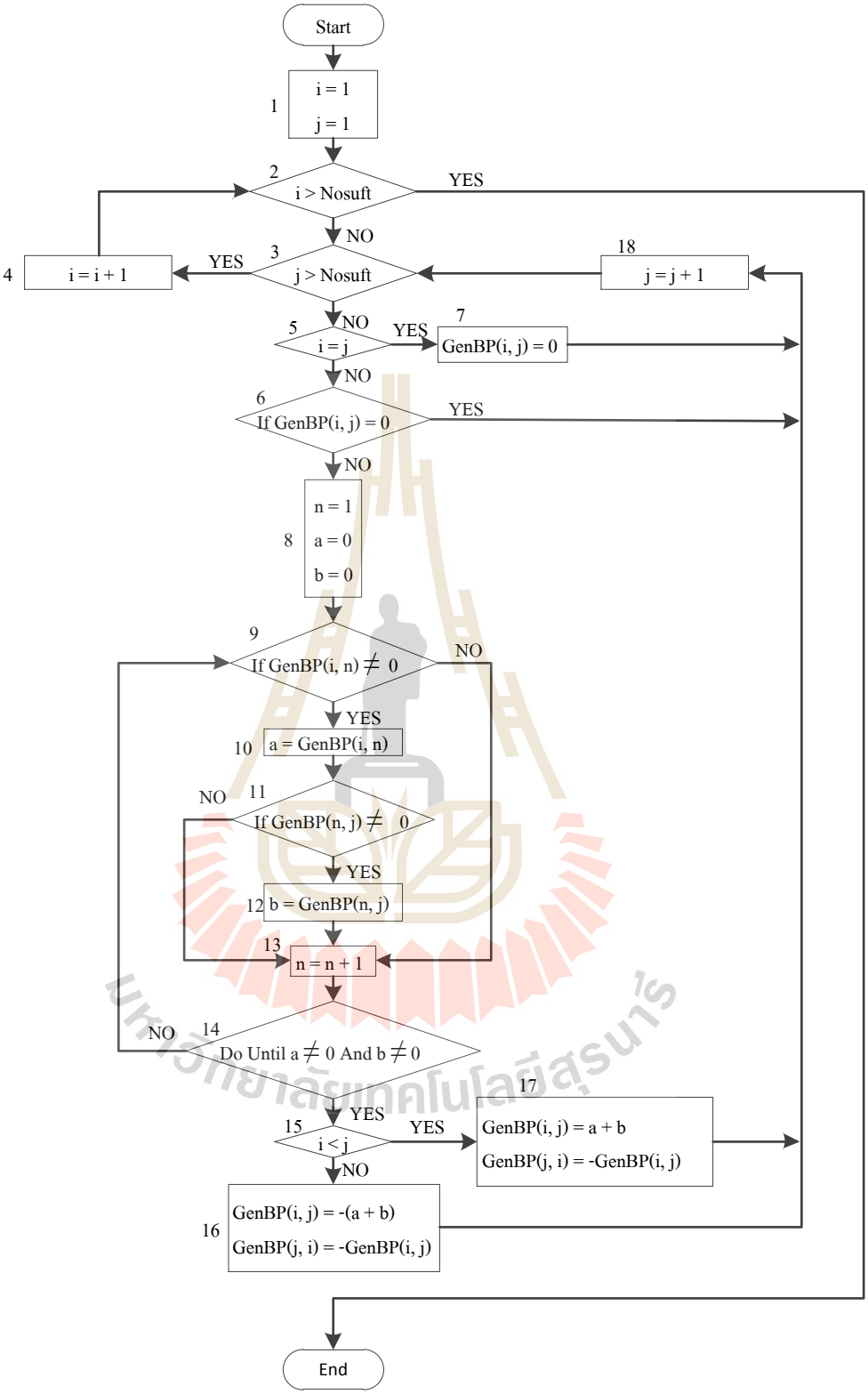
(15) กรอบที่ 15 ตรวจสอบเงื่อนไขว่า i น้อยกว่า j หรือไม่ หากเป็นจริงให้ทำกรอบที่ 17 หากไม่เป็นจริงทำกรอบที่ 16

(16) กรอบที่ 16 กำหนดค่า GenBP(i, j) เท่ากับ –(a+b) และกำหนดค่า GenBP(j, i) เท่ากับ - GenBP(i, j) จากนั้นทำกรอบที่ 18

(17) กรอบที่ 17 กำหนดค่า GenBP(i, j) เท่ากับ (a+b) และกำหนดค่า GenBP(j, i) เท่ากับ - GenBP(i, j) จากนั้นทำกรอบที่ 18

(18) กรอบที่ 18 กำหนดค่า j เท่ากับ j + 1 จากนั้นทำกรอบที่ 3

มหาวิทยาลัยเทคโนโลยีสุรนารี




รูปที่ 3.21 Flow chart แสดงขั้นตอนการทำงานของโมดูลคำนวณหาขนาดมูลฐานของ ส่วนที่ 3 ชิ้นงาน

### 3.2.4 โมดูลประมวลผลค่า tolerance ในแต่ละขั้นตอนการผลิต

โมดูลนี้ทำหน้าที่ประมวลผลค่า tolerance เพื่อมาใส่ในแต่ละขั้นตอนการผลิต โดยโปรแกรมจะนำค่าขนาดมูลฐาน และเครื่องจักรที่ใช้ในการผลิตแต่ละขั้นตอนไปหาค่า IT จากสูตรในตารางที่ 3.7 แล้วนำค่าที่ได้ไปคำนวณหาค่า tolerance ในโมดูลมีตัวแปรที่สำคัญ ดังตารางที่ 3.8

ตารางที่ 3.7 สูตรคำนวณค่า IT

| สูตรคำนวณค่า IT (international tolerance grade) |
|---|
| $T=10^{\frac{ITG-16}{5}}(0.45\times D^{\frac{1}{3}}+(0.001\times D)$ |

ตารางที่ 3.8 แสดงข้อมูลของตัวแปรสำคัญในโมดูลประมวลผลค่า tolerance ในแต่ละขั้นตอนการผลิต

| ชื่อตัวแปร | ชนิด | ขนาด | หน้าที่ของตัวแปร |
|---|---|---|---|
| TC().Machine | Dimension | Array 1 มิติ | เก็บเครื่องจักรที่ใช้งานขั้นตอนการตัดโลหะ |
| TC().Tolerance | Dimension | Array 1 มิติ | เก็บค่า tolerance ในแต่ละขั้นตอนการตัดโลหะ |

Flow chart ในรูปที่ 3.22 แสดงลำดับขั้นตอนการทำงานของประมวลผลค่า tolerance ในแต่ละขั้นตอนการผลิต โมดูลนี้โปรแกรมจะทำการคำนวณค่า tolerance จากเครื่องจักรที่ใช้ในกระบวนการผลิตแต่ละขั้นตอน หรือจากผู้ใช้งานโปรแกรมเป็นคนกำหนดเองก็ได้ หลังจากโปรแกรมได้ค่า tolerance แล้วจะนำไปเก็บไว้ในตัวแปร TC().Tolerance ซึ่งลำดับขั้นการทำงานสามารถสรุปได้ดังนี้

(1) กรอบที่ 1 กำหนดค่า jcount เท่ากับ 1 จากนั้นทำกรอบที่ 2

(2) กรอบที่ 2 ตรวจสอบเงื่อนไขว่า jcount มากกว่า Ubound(TC) หรือไม่ หากเป็นจริงจบการทำงาน หากไม่เป็นจริงทำกรอบที่ 3

(3) กรอบที่ 3 ตรวจสอบเงื่อนไขว่า txtF2(jcount).Text = "" มีการกำหนดค่า tolerance จากผู้ใช้งานหรือไม่ หากเป็นจริงให้ทำกรอบที่ 4 หากไม่เป็นจริงทำกรอบที่ 5

(4) กรอบที่ 4 ทำการบันทึกค่าในตัวแปร TC(jcount).Tolerance เท่ากับ txtF2(jcount).Text จากนั้นทำกรอบที่ 24

(5) กรอบที่ 5 ตรวจสอบเงื่อนไขว่า TC(jcount).Machine มีค่าเท่ากับ "-" หากเป็นจริงให้ทำกรอบที่ 6 หากไม่เป็นจริงทำกรอบที่ 7

(6) กรอบที่ 6 ทำการบันทึกค่าในตัวแปร TC(jcount).Tolerance เท่ากับ 0 จากนั้นทำกรอบที่ 24

(7) กรอบที่ 7 ตรวจสอบเงื่อนไขว่า TC(jcount).Machine มีค่าเท่ากับ "Lapping M/C" หากเป็นจริงให้ทำกรอบที่ 8 หากไม่เป็นจริงทำกรอบที่ 9

(8) กรอบที่ 8 กำหนดค่า it(jcount) เท่ากับ 2 จากนั้นทำกรอบที่ 23

(9) กรอบที่ 9 ตรวจสอบเงื่อนไขว่า TC(jcount).Machine มีค่าเท่ากับ "Superfinishing M/C" หากเป็นจริงให้ทำกรอบที่ 10 หากไม่เป็นจริงทำกรอบที่ 11

(10) กรอบที่ 10 กำหนดค่า it(jcount) เท่ากับ 4 จากนั้นทำกรอบที่ 23

(11) กรอบที่ 11 ตรวจสอบเงื่อนไขว่า TC(jcount).Machine มีค่าเท่ากับ "Diamond turning M/C" หากเป็นจริงให้ทำกรอบที่ 12 หากไม่เป็นจริงทำกรอบที่ 13

(12) กรอบที่ 12 กำหนดค่า it(jcount) เท่ากับ 4 จากนั้นทำกรอบที่ 23

(13) กรอบที่ 13 ตรวจสอบเงื่อนไขว่า TC(jcount).Machine มีค่าเท่ากับ "Grinding M/C" หากเป็นจริงให้ทำกรอบที่ 14 หากไม่เป็นจริงทำกรอบที่ 15

(14) กรอบที่ 14 กำหนดค่า it(jcount) เท่ากับ 6 จากนั้นทำกรอบที่ 23

(15) กรอบที่ 15 ตรวจสอบเงื่อนไขว่า TC(jcount).Machine มีค่าเท่ากับ "Turning M/C" หากเป็นจริงให้ทำกรอบที่ 16 หากไม่เป็นจริงทำกรอบที่ 17

(16) กรอบที่ 16 กำหนดค่า it(jcount) เท่ากับ 9 จากนั้นทำกรอบที่ 23

(17) กรอบที่ 17 ตรวจสอบเงื่อนไขว่า TC(jcount).Machine มีค่าเท่ากับ "Milling M/C" หากเป็นจริงให้ทำกรอบที่ 18 หากไม่เป็นจริงทำกรอบที่ 19

(18) กรอบที่ 18 กำหนดค่า it(jcount) เท่ากับ 9 จากนั้นทำกรอบที่ 23

(19) กรอบที่ 19 ตรวจสอบเงื่อนไขว่า TC(jcount).Machine มีค่าเท่ากับ "Shaping M/C" หากเป็นจริงให้ทำกรอบที่ 20 หากไม่เป็นจริงทำกรอบที่ 21

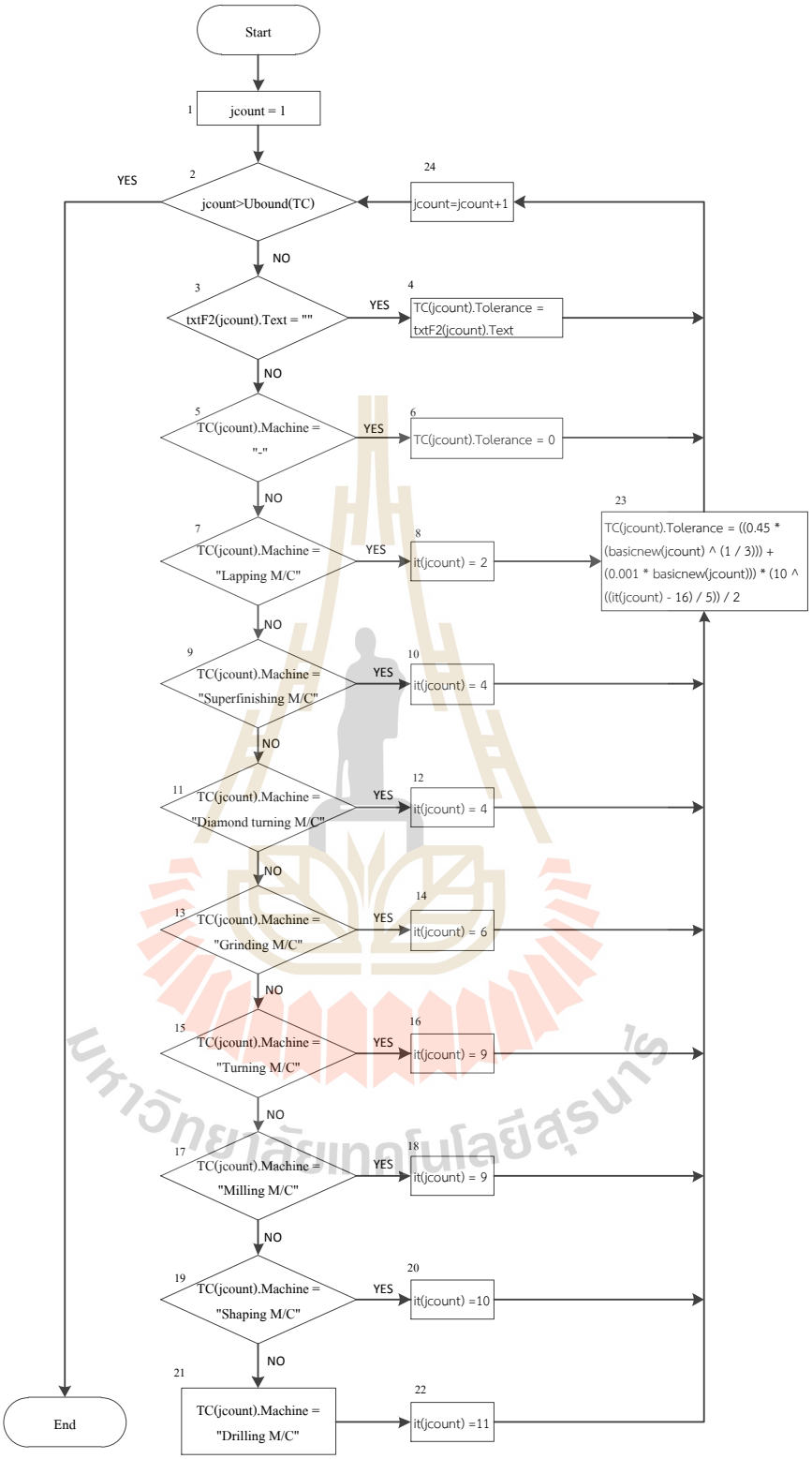
(20) กรอบที่ 20 กำหนดค่า it(jcount) เท่ากับ 10 จากนั้นทำกรอบที่ 23

(21) กรอบที่ 21 TC(jcount).Machine มีค่าเท่ากับ "Drilling M/C" จากนั้นทำกรอบที่ 22

(22) กรอบที่ 22 กำหนดค่า it(jcount) เท่ากับ 11 จากนั้นทำกรอบที่ 23

(23) กรอบที่ 23 ทำการบันทึกค่าในตัวแปร TC(jcount).Tolerance จากการคำนวณตามสูตร จากนั้นทำกรอบที่ 24

(24) กรอบที่ 24 กำหนดค่า jcount เท่ากับ jcount + 1 จากนั้นทำกรอบที่ 2




รูปที่ 3.22 Flow chart แสดงขั้นตอนการทำงานโมดูลประมวลผลค่า tolerance

### 3.2.5 โมดูลเปรียบเทียบ Path เพื่อคำนวณค่าความคลาดเคลื่อนของ stock removal และค่า basic stock removal

โดยโมดูลนี้จะนำชื่อตำแหน่งของรอยตัดใน cut นั้น และจุดตัดตำแหน่งเดิมก่อนหน้า (ต่างกันตรงตัวเลขลงท้ายตัวอักษรโดยเอาตัวเลขตามหลังตัวอักษรของรอยตัดปัจจุบัน - 1) สองตำแหน่งที่เป็นตัวอักษรรวมตัวเลขตามหลังไปค้นหา Path ของตัวเองมาเปรียบเทียบกันเพื่อหาเส้นทางความสัมพันธ์ จากนั้นนำค่าของ tolerance ในเส้นทางความสัมพันธ์ที่เกี่ยวของมารวมกันก็จะได้ค่าความคลาดเคลื่อนของ stock removal จากนั้นคำนวณค่า basic stock removal โดยนำ 2 มาคูณ stock removal เพราะในที่นี้ต้องการให้มีเนื้อโลหะตัดออกไปทุกครั้งที่ตัดแต่ก็มีโอกาสที่เกิดการตัด “ลม” คือเนื้อโลหะที่เหลือให้ตัดเท่ากับศูนย์ ซึ่งโอกาสเช่นนี้เกิดได้น้อยมากโมดูลนี้จะประกอบด้วย 3 ส่วนย่อย; 2 function และมีตัวแปรที่สำคัญ ดังตารางที่ 3.9

(1) ส่วนย่อยที่ 1 มีหน้าที่นำ Path ที่สร้างมาเรียงลำดับใหม่จากหลังไปหน้า

(2) ส่วนย่อยที่ 1 มีหน้าที่เปรียบเทียบ Path

(3) ส่วนย่อยที่ 3 มีหน้าที่นำชื่อของรอยตัดในตัวแปร finalP มาค้นหาค่า tolerance

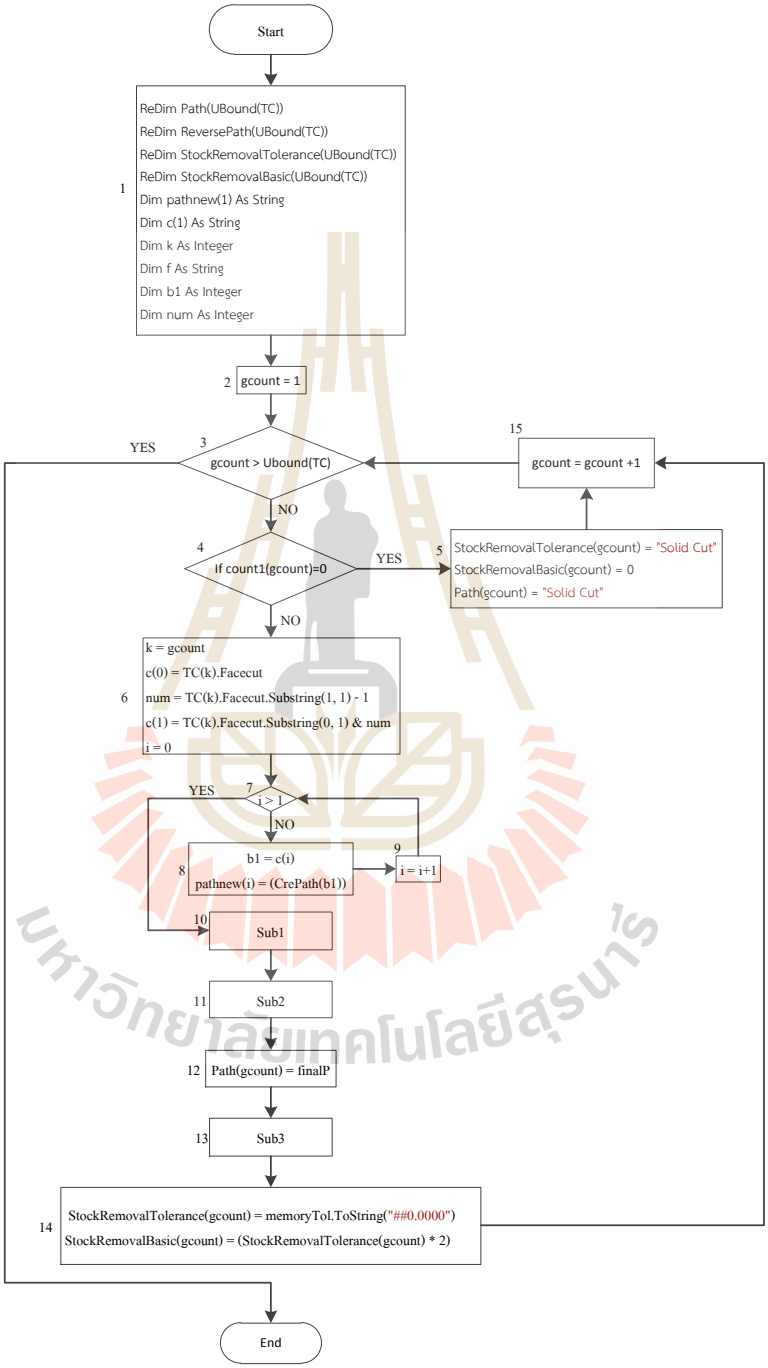
(4) function 1 **Function CreaPath** มีหน้าที่สร้าง Path

(5) function 2 **Function PathOne** มีหน้าที่เรียงลำดับความสัมพันธ์ใน Path

ตารางที่ 3.9 แสดงข้อมูลของตัวแปรสำคัญในโมดูลเปรียบเทียบ Path เพื่อคำนวณค่าความคลาดเคลื่อนของ stock removal และค่า basic stock removal

| ชื่อตัวแปร | ชนิด | ขนาด | หน้าที่ของตัวแปร |
|---|---|---|---|
| Path() | String | Array 1 มิติ | เก็บตัวอักษรแสดงความสัมพันธ์ของรอยตัด |
| ReversePath() | String | Array 1 มิติ | เก็บตัวอักษรแสดงความสัมพันธ์ของรอยตัดแบบเรียงลำดับใหม่จากหลังไปหน้า |
| StockRemovalTolerance() | String | Array 1 มิติ | เก็บค่าความคลาดเคลื่อนสะสมที่คำนวณจากการเปรียบเทียบ path |
| StockRemovalBasic() | String | Array 1 มิติ | เก็บค่าความคลาดเคลื่อนสะสมคูณสองของแต่ละcut |

Flow chart ในรูปที่ 3.23 แสดงลำดับขั้นตอนการทำงานโมดูลเปรียบเทียบ Path เพื่อคำนวณค่าความคลาดเคลื่อนของ stock removal และค่า basic stock removal ในภาพรวมส่วนลำดับการทำงานสรุปได้ดังนี้:-




รูปที่ 3.23 Flow chart แสดงขั้นตอนการทำงานโมดูลเปรียบเทียบ Path เพื่อคำนวณค่าความคลาดเคลื่อนของ stock removal และค่า basic stock removal

(1) กรอบที่ 1 ประกาศตัวแปร StockRemovalTolerance, StockRemovalToleranceBasic, Path, ReversePath เท่ากับ Ubound(TC); ประกาศตัวแปร pathnew, c, k, f, b1 และ num จากนั้นทำกรอบที่ 2

(2) กรอบที่ 2 กำหนดค่า gcount เท่ากับ 1 จากนั้นทำกรอบที่ 3

(3) กรอบที่ 3 ตรวจสอบเงื่อนไขว่า gcount มากกว่า Ubound(TC) หรือไม่ หากเป็นจริงจบการทำงาน หากไม่เป็นจริงทำกรอบที่ 4

(4) กรอบที่ 4 ตรวจสอบเงื่อนไขว่า count1(gcount) มีค่าเท่ากับ 0 หรือไม่ หากเป็นจริงให้ทำกรอบที่ 5 หากไม่เป็นจริงทำกรอบที่ 6

(5) กรอบที่ 5 ทำการบันทึกค่าในตัวแปร stockRemovalTolerance(gcount) เท่ากับ “Solid Cut”; ทำการบันทึกค่าในตัวแปร stockRemovalbasic(gcount) เท่ากับ 0; และทำการบันทึกค่าในตัวแปร Path(gcount) เท่ากับ 0 จากนั้นทำกรอบที่ 15

(6) กรอบที่ 6 กำหนดค่า k เท่ากับ gcount; กำหนดค่า c(0) เท่ากับ TC(k).Facecut; กำหนดค่า num เท่ากับ TC(k).Facecut.Substring(1,1)-1; กำหนดค่า c(1) เท่ากับ TC(k).Facecut.Substring(0,1)# และกำหนดค่า i เท่ากับ 0 จากนั้นทำกรอบที่ 7

(7) กรอบที่ 7 ตรวจสอบเงื่อนไขว่า i มากกว่า 1 หรือไม่ หากเป็นจริงทำกรอบที่ 10 หากไม่เป็นจริงทำกรอบที่ 8

(8) กรอบที่ 8 กำหนดค่า b1 เท่ากับ c(i) และกำหนดค่า pathnew(i) เท่ากับ Function (CrePath(b1)) จากนั้นทำกรอบที่ 9

(9) กรอบที่ 9 กำหนดค่า i เท่ากับ i + 1 จากนั้นทำกรอบที่ 7

(10) กรอบที่ 10 ทำในส่วนย่อยที่ 1 (Sub1) จากนั้นทำกรอบที่ 11

(11) กรอบที่ 11 ทำในส่วนย่อยที่ 2 (Sub2) จากนั้นทำกรอบที่ 12

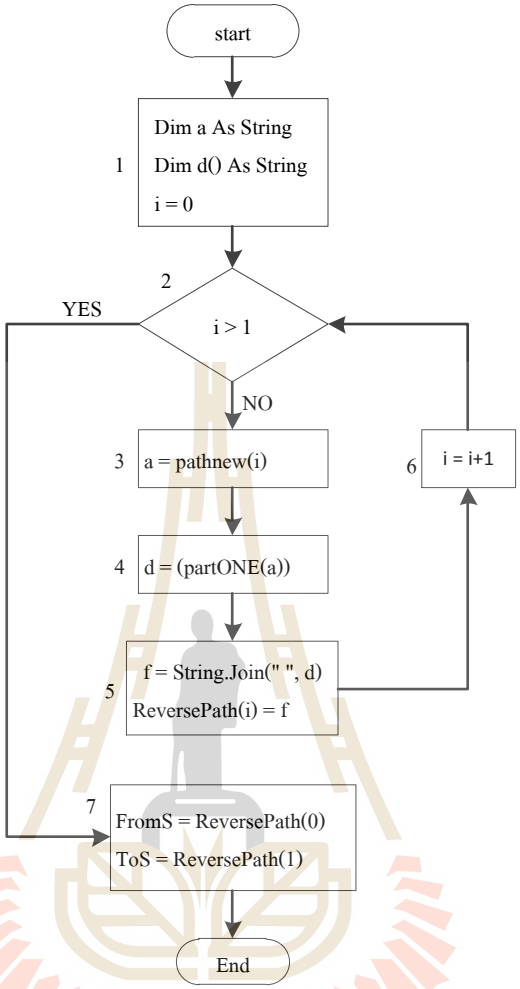
(12) กรอบที่ 12 กำหนดค่า Path(gcount) เท่ากับ finalP จากนั้นทำกรอบที่ 13

(13) กรอบที่ 13 ทำในส่วนย่อยที่ 3 (Sub3) จากนั้นทำกรอบที่ 14

(14) กรอบที่ 5 ทำการบันทึกค่าในตัวแปร stockRemovalTolerance(gcount) เท่ากับ memoryTol.ToString("##0.0000"); และ ทำการบันทึกค่าในตัวแปร stockRemovalbasic(gcount) เท่ากับ (StockRemovalTolerance(gcount) * 2) จากนั้นทำกรอบที่ 15

(15) กรอบที่ 15 กำหนดค่า gcount เท่ากับ gcount + 1 จากนั้นทำกรอบที่ 3

ส่วนย่อยที่ 1 มีหน้าที่นำ Path ที่สร้างมาเรียงลำดับใหม่จากหลังไปหน้า จากนั้นนำไปเก็บในตัวแปร FormS และ Tos แสดงในรูปที่ Flow Chart ที่ 3.24




รูปที่ 3.24 Flow chart แสดงลำดับขั้นตอนการทำงานของส่วนย่อยที่ 1

อธิบายขั้นตอนการทำงานของโมดูลย่อย แสดงใน Flow chart ในรูปที่ 3.24เป็นดังนี้

(1) กรอบที่ 1 ประกาศตัวแปร a และ d ถัดมากำหนดค่า i เท่ากับ 0 จากนั้นทำกรอบที่ 2

(2) กรอบที่ 2 ตรวจสอบเงื่อนไขว่า i มากกว่า 1 หรือไม่ หากเป็นจริงทำกรอบที่ 7 หากไม่เป็นจริงทำกรอบที่ 3

(3) กรอบที่ 3 กำหนดค่า a เท่ากับ pathnew(i) จากนั้นทำกรอบที่ 4

(4) กรอบที่ 4 กำหนดค่า d เท่ากับ Function (pathONE(a)) จากนั้นทำกรอบที่ 5

(5) กรอบที่ 5 กำหนดค่า f เท่ากับ String.Join(“ “,d) และกำหนดค่า f เท่ากับ ReversePath(i) จากนั้นทำกรอบที่ 6

(6) กรอบที่ 6 กำหนดค่า i เท่ากับ i + 1 จากนั้นทำกรอบที่ 2

(7) กรอบที่ 7 ทำการบันทึกค่าในตัวแปร FromS เท่ากับ ReversePath(0); และทำการบันทึกค่าในตัวแปร ToS เท่ากับ ReversePath(1) จากนั้นจบการทำงาน

ส่วนย่อยที่ 2 มีหน้าที่เปรียบเทียบ Path ที่นำมาเก็บในตัวแปร FormS และตัวแปร Tos หลังจากเปรียบเทียบ Path จะนำไปเก็บในตัวแปร finalP ดังในรูปที่ 3.25 และมีขั้นตอนดังนี้:-

(1) กรอบที่ 1 ประกาศตัวแปร strArrayF(), และ strArryT() จากนั้นทำกรอบที่ 2

(2) กรอบที่ 2 แยกค่าในตัวแปร FormS ด้วยสัญลักษณ์ " " นำไปเก็บในตัวแปร strArrayF และ2 แยกค่าในตัวแปร ToS ด้วยสัญลักษณ์ " " นำไปเก็บในตัวแปร strArrayT จากนั้นทำกรอบที่ 3

(3) กรอบที่ 3 ประกาศตัวแปร ii และ kk และกำหนดค่า ii เท่ากับ Ubound(strArrayF); กำหนดค่า kk เท่ากับ 1 จากนั้นทำกรอบที่ 4

(4) กรอบที่ 4 ตรวจสอบเงื่อนไขว่า kk ต้องไม่เท่าเท่ากับ 0 หากเป็นจริงให้ทำกรอบที่ 5 หากไม่เป็นจริงทำกรอบที่ 18

(5) กรอบที่ 5 กำหนดค่า j เท่ากับ Ubound(strArrayT) จากนั้นทำกรอบที่ 6

(6) กรอบที่ 6 ตรวจสอบเงื่อนไขว่า j น้อยกว่า Lbound(strArrayT) หรือไม่ หากเป็นจริงทำกรอบที่ 8 หากไม่เป็นจริงทำกรอบที่ 7

(7) กรอบที่ 7 ตรวจสอบเงื่อนไขว่า strArrayF(ii) เท่ากับ strArrayT(j) หรือไม่ หากเป็นจริงทำกรอบที่ 10 หากไม่เป็นจริงทำกรอบที่ 9

(8) กรอบที่ 8 กำหนดค่า ii เท่ากับ ii - 1 จากนั้นทำกรอบที่ 5

(9) กรอบที่ 9 กำหนดค่า j เท่ากับ j - 1 จากนั้นทำกรอบที่ 6

(10) กรอบที่ 10 กำหนดค่า final1 เท่ากับ ""; กำหนดค่า kcount เท่ากับ ii+1; และกำหนดค่า jcount เท่ากับ ii+1 จากนั้นทำกรอบที่ 11

(11) กรอบที่ 11 ตรวจสอบเงื่อนไขว่า kcount น้อยกว่า Ubound(strArrayF) หรือไม่ หากเป็นจริงทำกรอบที่ 14 หากไม่เป็นจริงทำกรอบที่ 12

(12) กรอบที่ 12 กำหนดค่า finalP เท่ากับ finalP &" "& strArrayF (kcount) จากนั้นทำกรอบที่ 13

(13) กรอบที่ 10 กำหนดค่า kcount เท่ากับ kcount +1 จากนั้นทำกรอบที่ 11

(14) กรอบที่ 14 ตรวจสอบเงื่อนไขว่า jcount น้อยกว่า Ubound(strArrayT) หรือไม่ หากเป็นจริงทำกรอบที่ 17 หากไม่เป็นจริงทำกรอบที่ 15

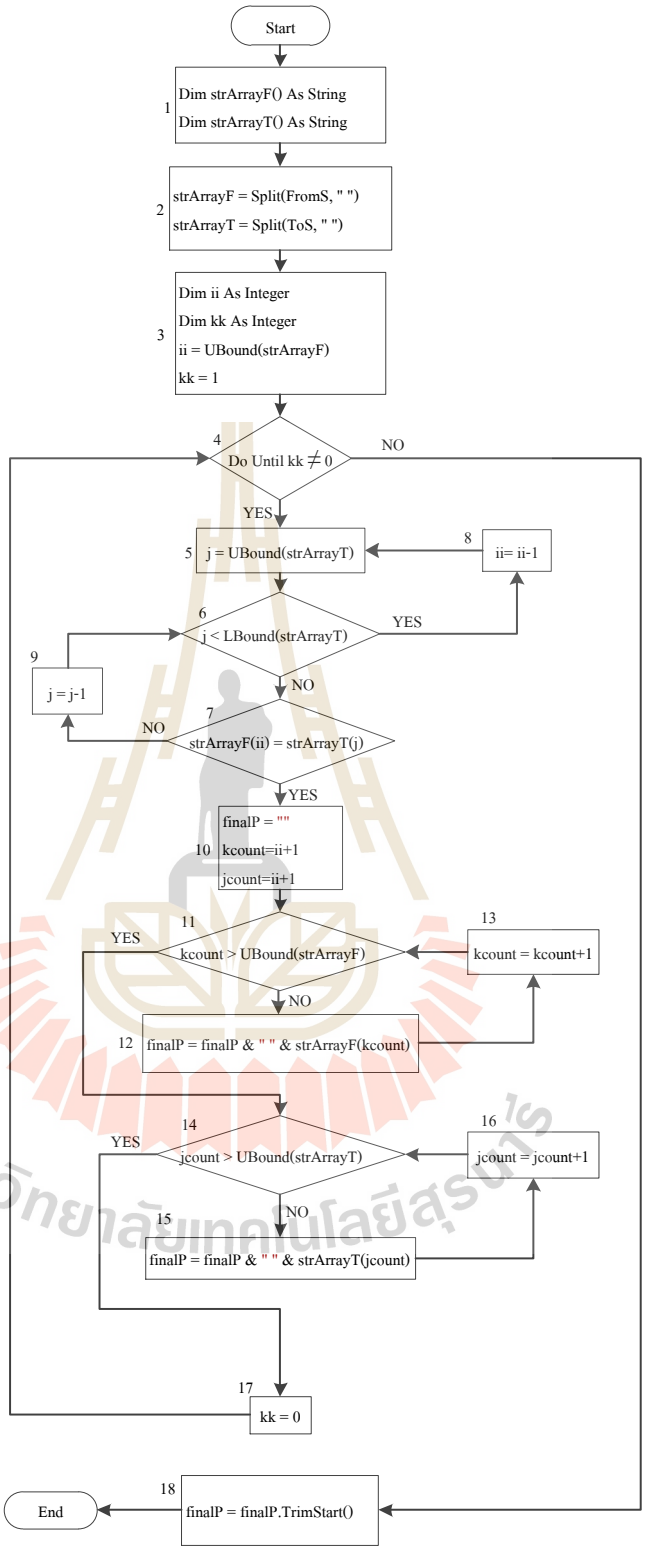
(15) กรอบที่ 15 กำหนดค่า finalP เท่ากับ finalP &" "& strArrayT (jcount) จากนั้นทำกรอบที่ 16

(16) กรอบที่ 16 กำหนดค่า jcount เท่ากับ jcount +1 จากนั้นทำกรอบที่ 14

(17) กรอบที่ 17 กำหนดค่า kk เท่ากับ 0 จากนั้นทำกรอบที่ 4

(18) กรอบที่ 18 ทำการบันทึกค่าในตัวแปร finalP เท่ากับ finalP.TrimStart() จากนั้นจบการทำงาน

มหาวิทยาลัยเทคโนโลยีสุรนารี




รูปที่ 3.25 Flow chart แสดงของลำดับขั้นตอนการทำงานของส่วนย่อยที่ 2

ส่วนย่อยที่ 3 มีหน้าที่นำชื่อของรอยตัดในตัวแปร finalP มาค้นหาค่า tolerance ของรอยตัด จากนั้นนำไปเก็บในตัวแปร memoryTol จนครบจำนวนรอยตัดที่มีความสัมพันธ์กัน ดังแสดงในรูปที่ Flow Chart ที่ 3.26 และมีลำดับการทำงานสรุปได้ดังนี้ :-

(1) กรอบที่ 1 ประกาศตัวแปร strArrayR(), และ memoryTol; กำหนดค่า memoryTol เท่ากับ 0 และแยกค่าในตัวแปร finalP ด้วยสัญลักษณ์ " " นำไปเก็บในตัวแปร strArrayRจากนั้นทำกรอบที่ 2

(2) กรอบที่ 2 กำหนดค่า i เท่ากับ Ubound(strArrayR) จากนั้นทำกรอบที่ 3

(3) กรอบที่ 3 ตรวจสอบเงื่อนไขว่า i มากกว่า Ubound(strArrayR) หรือไม่ หากเป็นจริงจบการทำงาน หากไม่เป็นจริงทำกรอบที่ 4

(4) กรอบที่ 4 กำหนดค่า j เท่ากับ 0 จากนั้นทำกรอบที่ 5

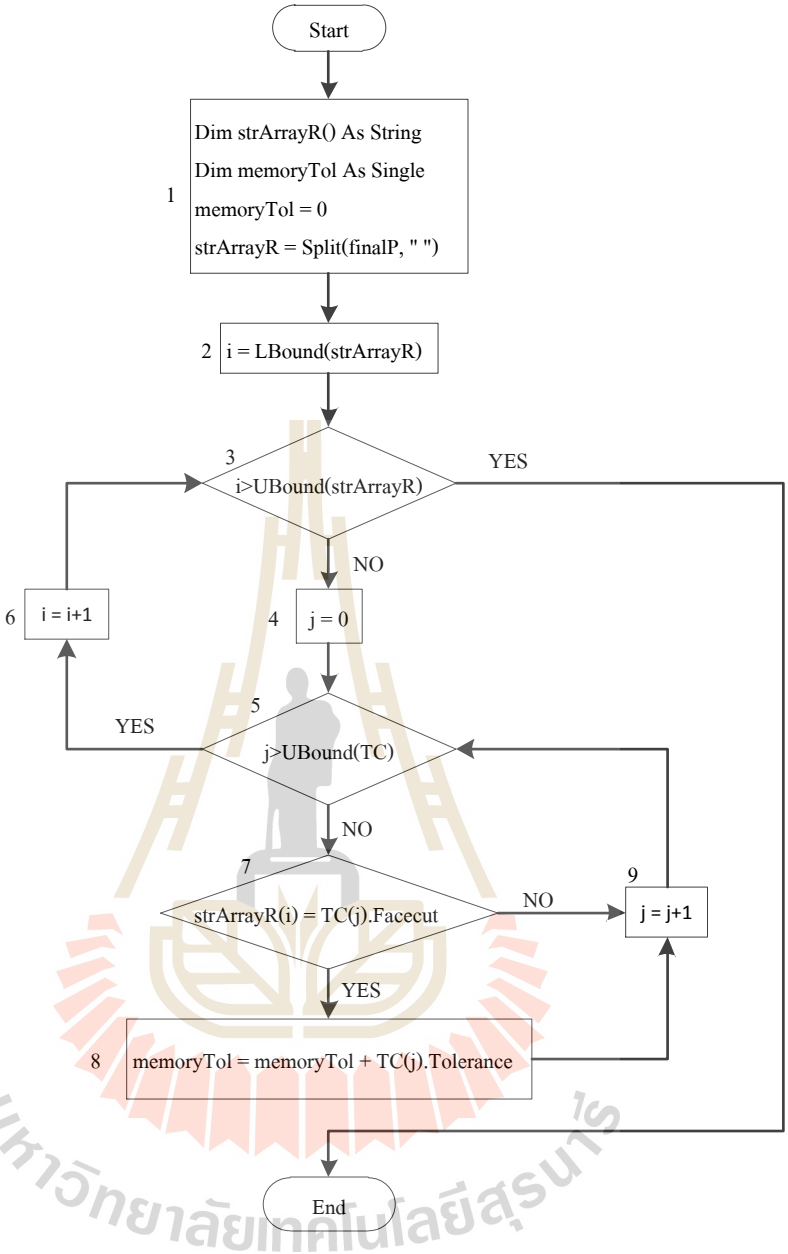
(5) กรอบที่ 5 ตรวจสอบเงื่อนไขว่า j น้อยกว่า Ubound(TC) หรือไม่ หากเป็นจริงทำกรอบที่ 6 หากไม่เป็นจริงทำกรอบที่ 7

(6) กรอบที่ 6 กำหนดค่า i เท่ากับ i + 1 จากนั้นทำกรอบที่ 3

(7) กรอบที่ 7 ตรวจสอบเงื่อนไขว่า strArrayT(i) เท่ากับ TC(j).Facecut หรือไม่ หากเป็นจริงทำกรอบที่ 8 หากไม่เป็นจริงทำกรอบที่ 9

(8) กรอบที่ 8 ทำการบันทึกค่าในตัวแปร memoryTol เท่ากับ memoryTol + TC(j).Tolerance จากนั้นทำกรอบที่ 9

มหาวิทยาลัยเทคโนโลยีสุรนารี




รูปที่ 3.26 Flow chart แสดงลำดับขั้นตอนการทำงานของส่วนย่อยที่ 3

### 3.2.6 โมดูลเปรียบเทียบ Path เพื่อคำนวณค่าความคลาดเคลื่อนของ Resultant tolerance

โมดูลนี้ จะนำชื่อตำแหน่งของรอยตัดล่าสุดที่ตัดรอยตัดการบอกขนาด (blue print) สองตำแหน่งที่เป็นตัวอักษรรวมตัวเลขตามหลังไปค้นหา Path ของตัวเองมาเปรียบเทียบกัน เพื่อหาเส้นทางความสัมพันธ์ จากนั้นนำค่าของ tolerance ของ working dimension มารวมกันตามความสัมพันธ์ที่ได้จากการเปรียบเทียบ Pathโดยโมดูลนี้จะประกอบด้วย 3 ส่วนย่อย; 2 function และมีตัวแปรที่สำคัญ ดังตารางที่ 3.10

(1) ส่วนย่อยที่ 1 มีหน้าที่นำ Path ที่สร้างมาเรียงลำดับใหม่จากหลังไปหน้า

(2) ส่วนย่อยที่ 2 มีหน้าที่เปรียบเทียบ Path

(3) ส่วนย่อยที่ 3 มีหน้าที่นำชื่อของรอยตัดในตัวแปร finalP มาค้นหาค่า tolerance

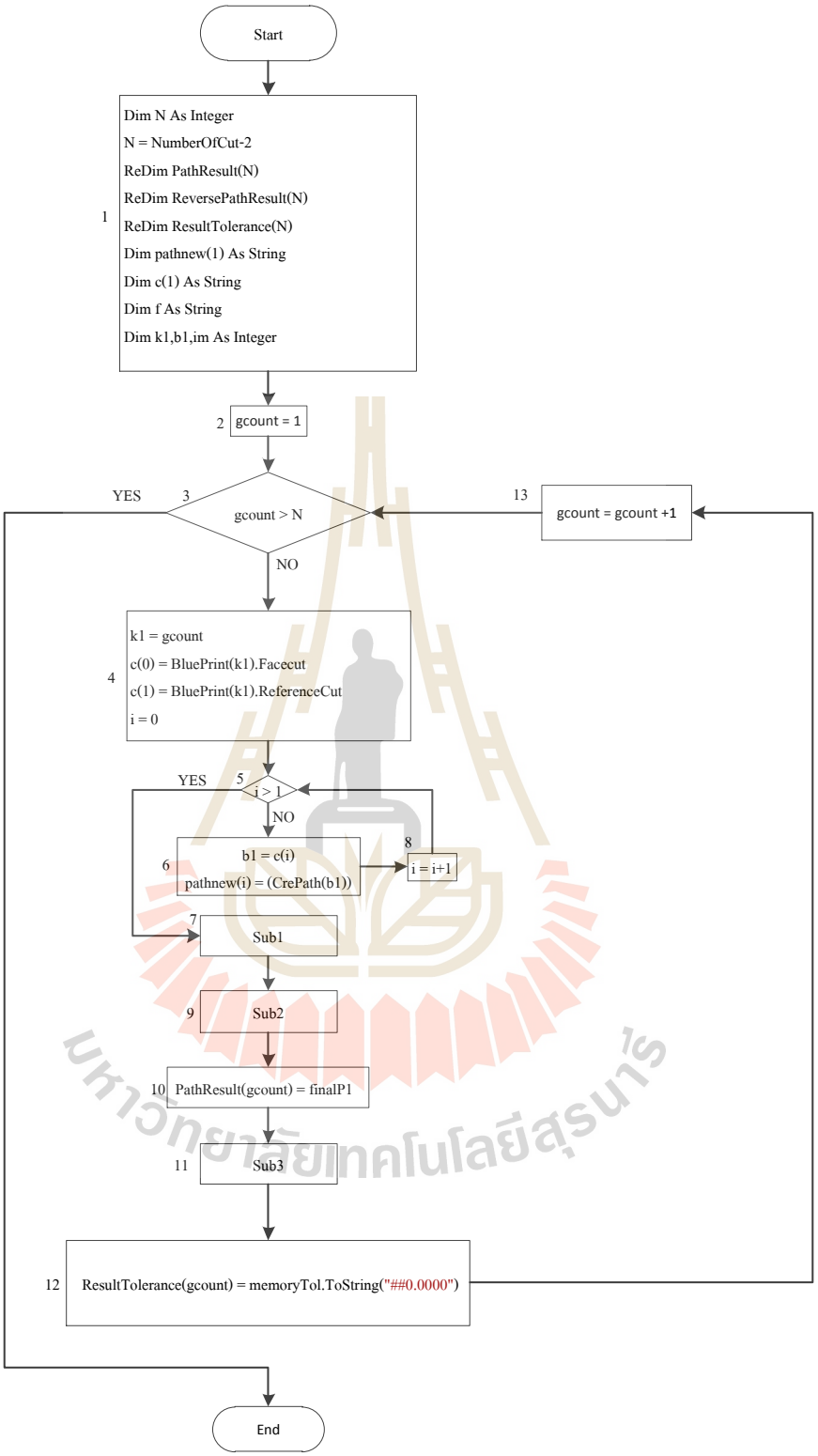
(4) function 1 **Function CreaPath** มีหน้าที่สร้าง Path

(5) function 2 **Function PathOne** มีหน้าที่เรียงลำดับความสัมพันธ์ใน Path

ตารางที่ 3.10 แสดงข้อมูลของตัวแปรสำคัญในโมดูลเปรียบเทียบ Path เพื่อคำนวณค่าความคลาดเคลื่อนของ Resultant tolerance

| ชื่อตัวแปร | ชนิด | ขนาด | หน้าที่ของตัวแปร |
|---|---|---|---|
| PathResult() | String | Array 1 มิติ | เก็บตัวอักษรแสดงความสัมพันธ์ของรอยตัด |
| ReversePathResult() | String | Array 1 มิติ | เก็บตัวอักษรแสดงความสัมพันธ์ของรอยตัดแบบเรียงลำดับใหม่จากหลังไปหน้า |
| ResultTolerance() | String | Array 1 มิติ | เก็บค่าความคลาดเคลื่อนสะสมที่คำนวณจากการเปรียบเทียบ path |

Flow chart ในรูปที่ 3.27 แสดงลำดับขั้นตอนการทำงานโมดูลเปรียบเทียบ Path เพื่อคำนวณค่าความคลาดเคลื่อนของ Resultant tolerance และมีขั้นตอนสรุปได้ดังนี้:-




รูปที่ 3.27 Flow chart แสดงขั้นตอนการทำงานโมดูลเปรียบเทียบ Path เพื่อคำนวณค่าความคลาดเคลื่อนของ Resultant tolerance

อธิบายขั้นตอนการทำงานของโมดูลย่อย แสดงใน Flow chart รูปที่ 3.23

(1) กรอบที่ 1 ประกาศตัวแปร N กำหนดค่า N เท่ากับ NumberOfcut-2; ประกาศตัวแปร ResultlTolerance, PathResult, ReversePathResult เท่ากับ N ; ประกาศตัวแปร pathnew, c, f, k1, b1 และ im จากนั้นทำกรอบที่ 2

(2) กรอบที่ 2 กำหนดค่า gcount เท่ากับ 1 จากนั้นทำกรอบที่ 3

(3) กรอบที่ 3 ตรวจสอบเงื่อนไขว่า gcount มากกว่า N หรือไม่ หากเป็นจริงจบการทำงาน หากไม่เป็นจริงทำกรอบที่ 4

(4) กรอบที่ 4 กำหนดค่า k1 เท่ากับ gcount 1 ; กำหนดค่า i เท่ากับ 0 ; กำหนดค่า c(0) เท่ากับ BluePrint(k1).Facecut ; และกำหนดค่า c(1) เท่ากับ BluePrint(k1).Referencecut จากนั้นทำกรอบที่ 5

(5) กรอบที่ 5 ตรวจสอบเงื่อนไขว่า i มากกว่า 1 หรือไม่ หากเป็นจริงทำกรอบที่ 7 หากไม่เป็นจริงทำกรอบที่ 6

(6) กรอบที่ 6 กำหนดค่า b1 เท่ากับ c(i) และกำหนดค่า pathnew(i) เท่ากับ Function (CrePath(b1)) จากนั้นทำกรอบที่ 8

(7) กรอบที่ 7 ทำในส่วนย่อยที่ 1 (Sub1) จากนั้นทำกรอบที่ 9

(8) กรอบที่ 8 กำหนดค่า i เท่ากับ i + 1 จากนั้นทำกรอบที่ 5

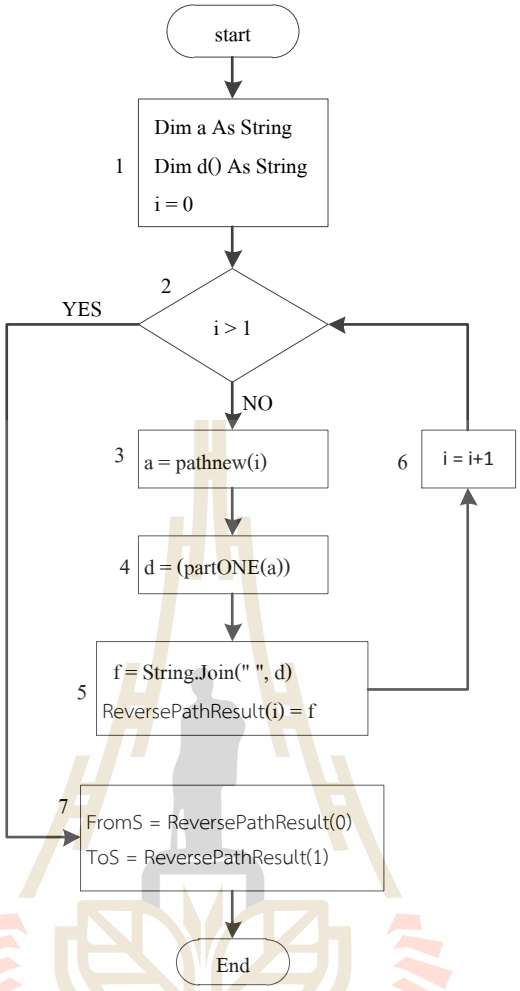
(9) กรอบที่ 9 ทำในส่วนย่อยที่ 2 (Sub2) จากนั้นทำกรอบที่ 10

(10) กรอบที่ 10 กำหนดค่า PathResult(gcount) เท่ากับ finalP1 จากนั้นทำกรอบที่ 11

(11) กรอบที่ 11 ทำในส่วนย่อยที่ 3 (Sub3) จากนั้นทำกรอบที่ 12

(12) กรอบที่ 12 ทำการบันทึกค่าในตัวแปร ResultlTolerance (gcount) เท่ากับ memoryTol.ToString("##0.0000") จากนั้นทำกรอบที่ 13

ส่วนย่อยที่ 1 มีหน้าที่นำ Path ที่สร้างมาเรียงลำดับใหม่จากหลังไปหน้า จากนั้นนำไปเก็บในตัวแปร FormS และ Tos แสดงดังรูปที่ Flow Chart ที่ 3.28




รูปที่ 3.28 แสดง Flow chart ลำดับขั้นตอนการทำงานของส่วนย่อยที่ 1

รูปที่ 3.24 แสดง Flow chart ลำดับขั้นตอนการทำงานและสามารถสรุปเป็นลำดับได้ดังนี้ :-

(1) กรอบที่ 1 ประกาศตัวแปร a และ d ถัดมากำหนดค่า i เท่ากับ 0 จากนั้นทำกรอบที่ 2

(2) กรอบที่ 2 ตรวจสอบเงื่อนไขว่า i มากกว่า 1 หรือไม่ หากเป็นจริงทำกรอบที่ 7 หากไม่เป็นจริงทำกรอบที่ 3

(3) กรอบที่ 3 กำหนดค่า a เท่ากับ pathnew(i) จากนั้นทำกรอบที่ 4

(4) กรอบที่ 4 กำหนดค่า d เท่ากับ Function (pathONE(a)) จากนั้นทำกรอบที่ 5

(5) กรอบที่ 5 กำหนดค่า f เท่ากับ String.Join(“ “,d) และกำหนดค่า f เท่ากับ ReversePath(i) จากนั้นทำกรอบที่ 6

(6) กรอบที่ 6 กำหนดค่า i เท่ากับ i + 1 จากนั้นทำกรอบที่ 2

(7) กรอบที่ 7 ทำการบันทึกค่าในตัวแปร FromS เท่ากับ ReversePath(0); และทำการบันทึกค่าในตัวแปร ToS เท่ากับ ReversePath(1) จากนั้นจบการทำงาน

ส่วนย่อยที่ 2 มีหน้าที่เปรียบเทียบ Path ที่นำมาเก็บในตัวแปร FormS และตัวแปร Tos หลังจากเปรียบเทียบ Path จะนำไปเก็บในตัวแปร finalP แสดงดังรูปที่ Flow Chart ที่ 3.29 และสรุปเป็นลำดับของการทำงานได้ดังนี้

(1) กรอบที่ 1 ประกาศตัวแปร strArrayF(), และ strArryT() จากนั้นทำกรอบที่ 2

(2) กรอบที่ 2 แยกค่าในตัวแปร FormS ด้วยสัญลักษณ์ " " นำไปเก็บในตัวแปร strArrayF และ2 แยกค่าในตัวแปร ToS ด้วยสัญลักษณ์ " " นำไปเก็บในตัวแปร strArrayT จากนั้นทำกรอบที่ 3

(3) กรอบที่ 3 ประกาศตัวแปร ii และ kk และกำหนดค่า ii เท่ากับ Ubound(strArrayF); กำหนดค่า kk เท่ากับ 1 จากนั้นทำกรอบที่ 4

(4) กรอบที่ 4 ตรวจสอบเงื่อนไขว่า kk ต้องไม่เท่าเท่ากับ 0 หากเป็นจริงให้ทำกรอบที่ 5 หากไม่เป็นจริงทำกรอบที่ 18

(5) กรอบที่ 5 กำหนดค่า j เท่ากับ Ubound(strArrayT) จากนั้นทำกรอบที่ 6

(6) กรอบที่ 6 ตรวจสอบเงื่อนไขว่า j น้อยกว่า Lbound(strArrayT) หรือไม่ หากเป็นจริงทำกรอบที่ 8 หากไม่เป็นจริงทำกรอบที่ 7

(7) กรอบที่ 7 ตรวจสอบเงื่อนไขว่า strArrayF(ii) เท่ากับ strArrayT(j) หรือไม่ หากเป็นจริงทำกรอบที่ 10 หากไม่เป็นจริงทำกรอบที่ 9

(8) กรอบที่ 8 กำหนดค่า ii เท่ากับ ii - 1 จากนั้นทำกรอบที่ 5

(9) กรอบที่ 9 กำหนดค่า j เท่ากับ j - 1 จากนั้นทำกรอบที่ 6

(10) กรอบที่ 10 กำหนดค่า final1 เท่ากับ ""; กำหนดค่า kcount เท่ากับ ii+1; และกำหนดค่า jcount เท่ากับ ii+1 จากนั้นทำกรอบที่ 11

(11) กรอบที่ 11 ตรวจสอบเงื่อนไขว่า kcount น้อยกว่า Ubound(strArrayF) หรือไม่ หากเป็นจริงทำกรอบที่ 14 หากไม่เป็นจริงทำกรอบที่ 12

(12) กรอบที่ 12 กำหนดค่า finalP เท่ากับ finalP &" "& strArrayF (kcount) จากนั้นทำกรอบที่ 13

(13) กรอบที่ 10 กำหนดค่า kcount เท่ากับ kcount +1 จากนั้นทำกรอบที่ 11

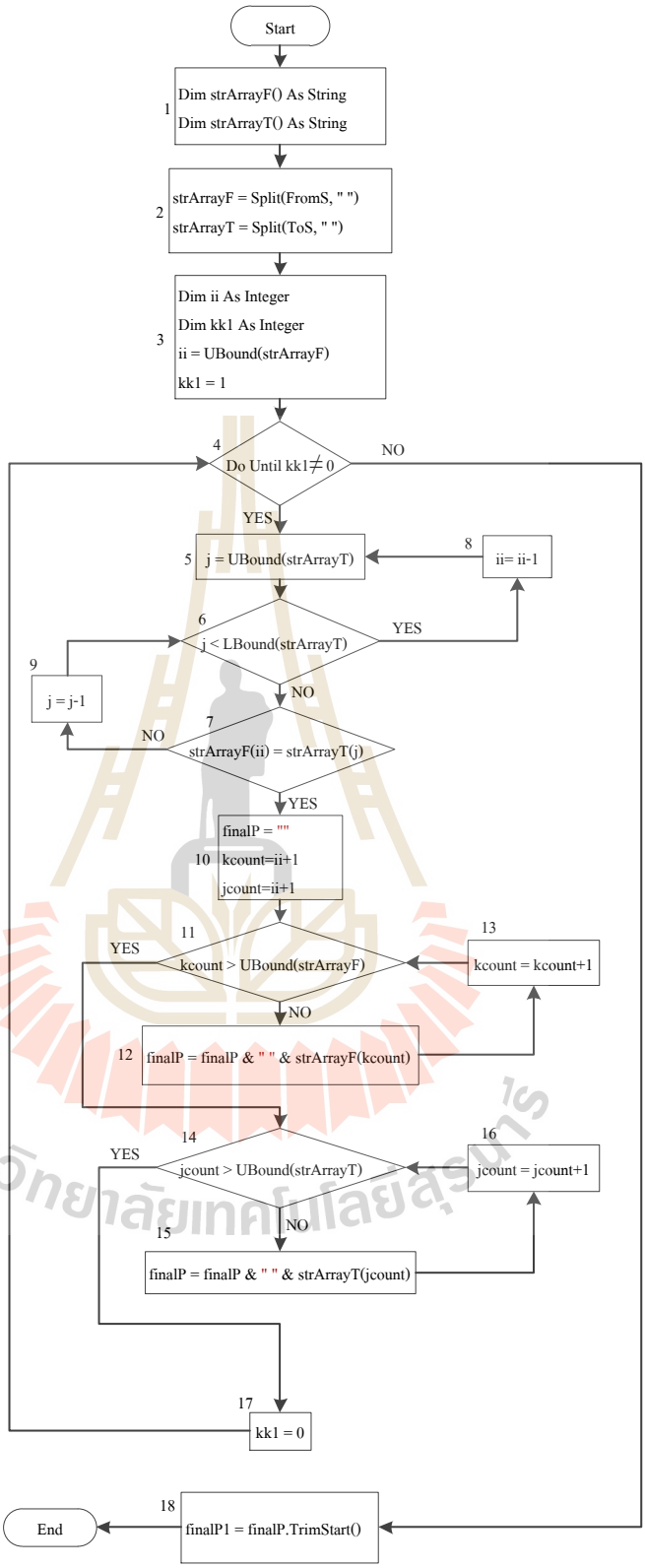
(14) กรอบที่ 14 ตรวจสอบเงื่อนไขว่า jcount น้อยกว่า Ubound(strArrayT) หรือไม่ หากเป็นจริงทำกรอบที่ 17 หากไม่เป็นจริงทำกรอบที่ 15

(15) กรอบที่ 15 กำหนดค่า finalP เท่ากับ finalP &" "& strArrayT (jcount) จากนั้นทำกรอบที่ 16

(16) กรอบที่ 16 กำหนดค่า jcount เท่ากับ jcount +1 จากนั้นทำกรอบที่ 14

(17) กรอบที่ 17 กำหนดค่า kk เท่ากับ 0 จากนั้นทำกรอบที่ 4

(18) กรอบที่ 18 ทำการบันทึกค่าในตัวแปร finalP1 เท่ากับ finalP.TrimStart() จากนั้นจบการทำงาน




รูปที่ 3.29 แสดง Flow chart ลำดับขั้นตอนการทำงานของส่วนย่อยที่ 2

ส่วนย่อยที่ 3 มีหน้าที่นำชื่อของรอยตัดในตัวแปร finalP มาค้นหาค่า tolerance ของรอยตัดจากนั้นนำไปเก็บในตัวแปร memoryTol จนครบจำนวนรอยตัดที่มีความสัมพันธ์กัน แสดงดังรูปที่ Flow Chart ที่ 3.30 และมีขั้นตอนคือ

(1) กรอบที่ 1 ประกาศตัวแปร strArrayR(), และ memoryTol; กำหนดค่า memoryTol เท่ากับ 0 และแยกค่าในตัวแปร finalP1 ด้วยสัญลักษณ์ " " นำไปเก็บในตัวแปร strArrayRจากนั้นทำกรอบที่ 2

(2) กรอบที่ 2 กำหนดค่า i เท่ากับ Ubound(strArrayR) จากนั้นทำกรอบที่ 3

(3) กรอบที่ 3 ตรวจสอบเงื่อนไขว่า i มากกว่า Ubound(strArrayR) หรือไม่ หากเป็นจริงจบการทำงาน หากไม่เป็นจริงทำกรอบที่ 4

(4) กรอบที่ 4 กำหนดค่า j เท่ากับ 0 จากนั้นทำกรอบที่ 5

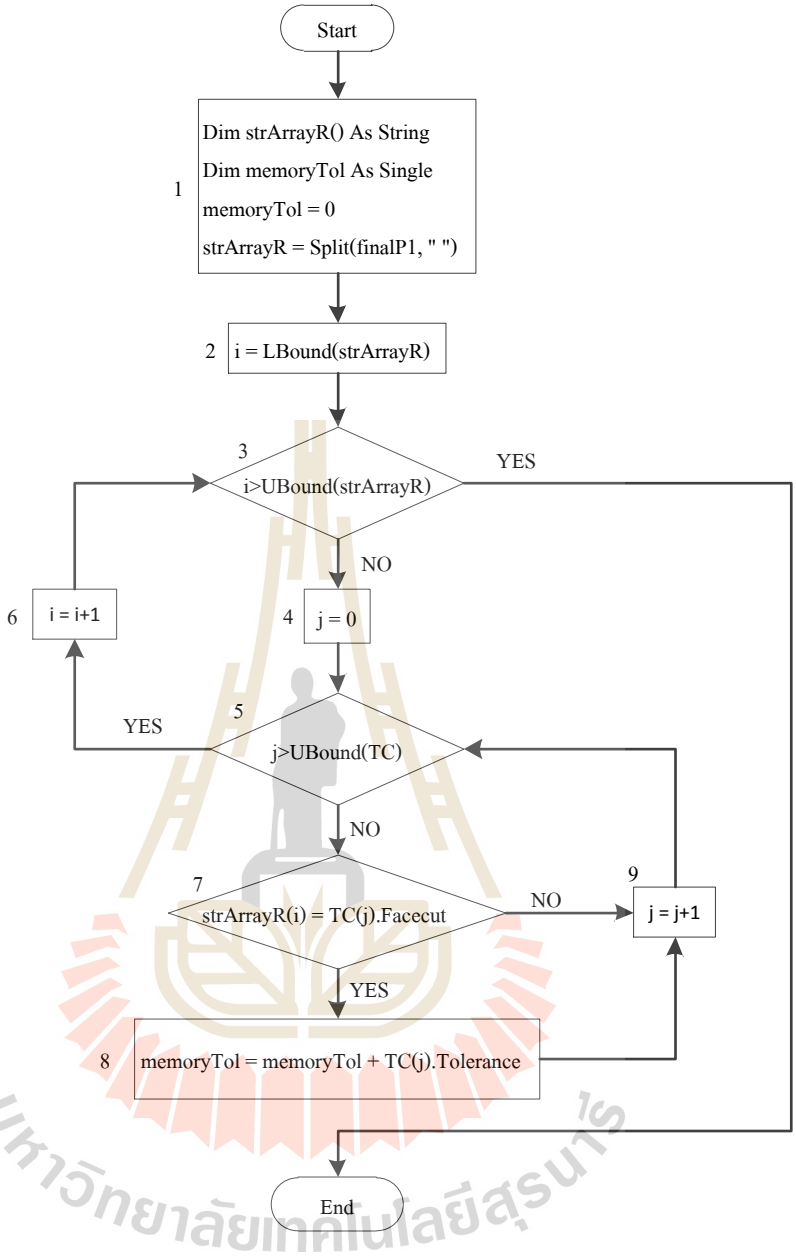
(5) กรอบที่ 5 ตรวจสอบเงื่อนไขว่า j น้อยกว่า Ubound(TC) หรือไม่ หากเป็นจริงทำกรอบที่ 6 หากไม่เป็นจริงทำกรอบที่ 7

(6) กรอบที่ 6 กำหนดค่า i เท่ากับ i + 1 จากนั้นทำกรอบที่ 3

(7) กรอบที่ 7 ตรวจสอบเงื่อนไขว่า strArrayT(i) เท่ากับ TC(j).Facecut หรือไม่ หากเป็นจริงทำกรอบที่ 8 หากไม่เป็นจริงทำกรอบที่ 9

(8) กรอบที่ 8 ทำการบันทึกค่าในตัวแปร memoryTol เท่ากับ memoryTol + TC(j).Tolerance จากนั้นทำกรอบที่ 9

มหาวิทยาลัยเทคโนโลยีสุรนารี




รูปที่ 3.30 แสดง Flow chart ลำดับขั้นตอนการทำงานของส่วนย่อยที่ 3

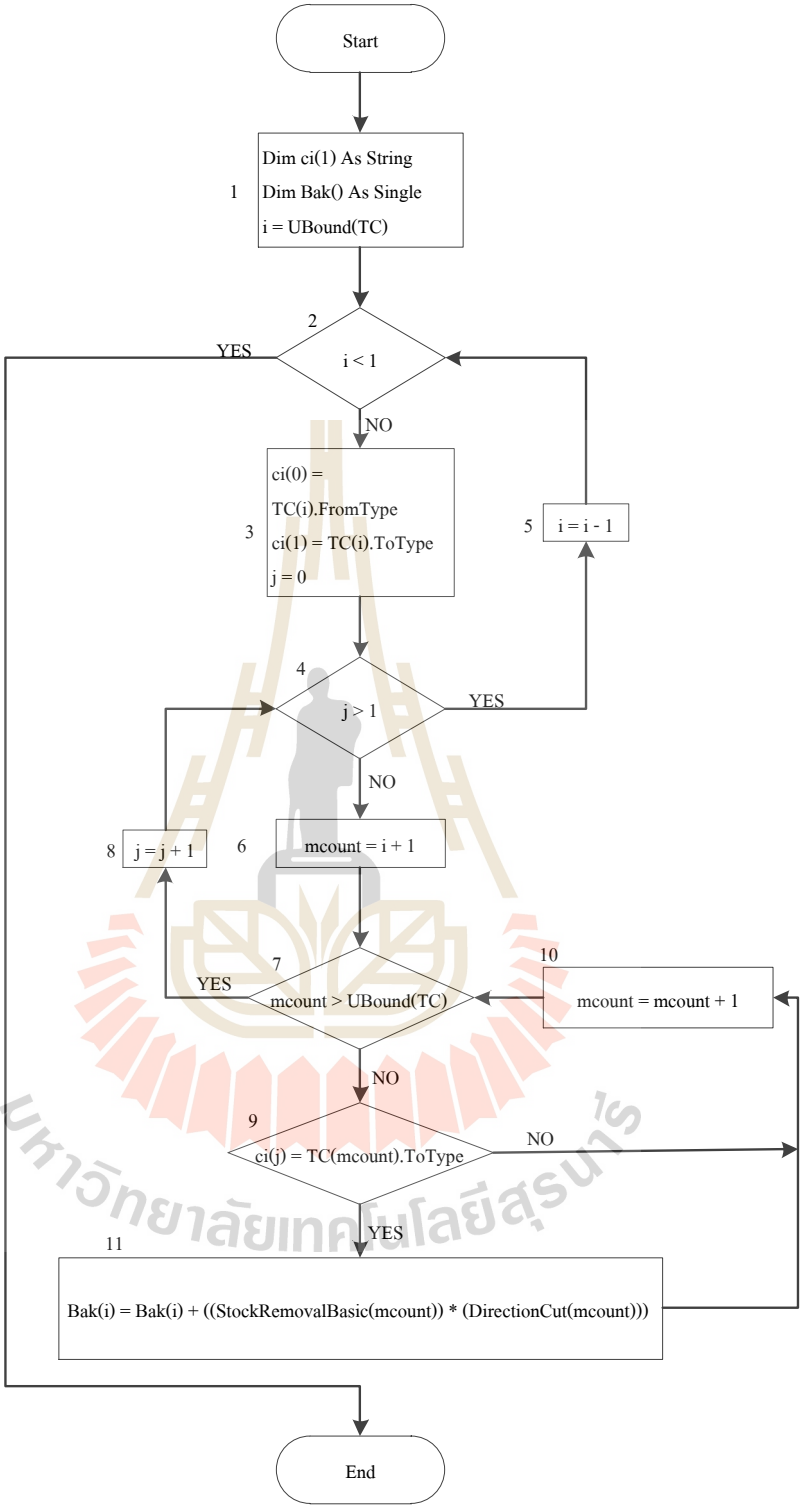
### 3.2.7 โมดูลคำนวณหาขนาดมูลฐานของชิ้นงานในแต่ละขั้นตอนการผลิต

โมดูลนี้มีหน้าที่คำนวณขนาดมูลฐานเพื่อนำค่ามาใส่ในแต่ละขั้นตอนการผลิต โดยในแต่ละขั้นตอนการผลิตจะมีการกำหนดจุดอ้างอิงและจุดตัด โดยค่านั้นจะเก็บในตัวแปร referenccut และ facecut จากนั้นนำค่าที่ได้ไปคำนวณหาขนาดมูลฐาน โดยคำนวณแบบหาความสัมพันธ์ของรอยตัดเริ่มคำนวณเริ่มจากขั้นตอนสุดท้ายไปหาขั้นตอนแรกโมดูลนี้มีตัวแปรที่สำคัญ ดังตารางที่ 3.11

ตารางที่ 3.11 แสดงข้อมูลของตัวแปรสำคัญในโมดูลคำนวณหาขนาดมูลฐานของชิ้นงานในแต่ละขั้นตอนการผลิต

| ชื่อตัวแปร | ชนิด | ขนาด | หน้าที่ของตัวแปร |
|---|---|---|---|
| Bak() | Single | Array 1 มิติ | เก็บค่าความคลาดเคลื่อนสะสมก่อนนำไปรวมค่าขนาดมูลฐานของรอยตัดนั้นๆ |
| DirectionCut() | Integer | Array 1 มิติ | เก็บตัวเลขบอกลักษณะรอยตัดว่าเป็นร่องหรือเป็นบ่า |
| StockRemovalBasic | String | Array 1 มิติ | เก็บเก็บค่าความคลาดเคลื่อนสะสมคูณสองของแต่ละcut |

Flow chart ในรูปที่ 3.31 แสดงลำดับขั้นตอนการทำงานของโมดูลคำนวณหาขนาดมูลฐานของชิ้นงานในแต่ละขั้นตอนการผลิต โปรแกรมจะทำการประมวลผลโดยนำค่า StockRemovalBasic ของ cut ที่เกี่ยวข้องนำไปคูณค่า DirectionCut หลังจากนั้นนำไปเก็บในตัวแปร Bak ซึ่งรายละเอียดของการทำงานสรุปเป็นขั้นตอนได้ดังนี้ :-




รูปที่ 3.31 Flow chart แสดงลำดับขั้นตอนการทำงานโมดูลคำนวณหาขนาดมูลฐานของชิ้นงานในแต่ละขั้นตอนการผลิต

(1) กรอบที่ 1 ประกาศตัวแปร ci(), และ Bak; กำหนดค่า i เท่ากับ Ubound(TC) จากนั้นทำกรอบที่ 2

(2) กรอบที่ 2 ตรวจสอบเงื่อนไขว่า i น้อยกว่า 1 หรือไม่ หากเป็นจริงจบการทำงาน หากไม่เป็นจริงทำกรอบที่ 3

(3) กรอบที่ 3 กำหนดค่า ci(0) เท่ากับ TC(i).FromType; กำหนดค่า ci(1) เท่ากับ TC(i).ToType และกำหนดค่า j เท่ากับ 0 จากนั้นทำกรอบที่ 4

(4) กรอบที่ 4 ตรวจสอบเงื่อนไขว่า j มากกว่า 1 หรือไม่ หากเป็นจริงทำกรอบที่ 5 หากไม่เป็นจริงทำกรอบที่ 6

(5) กรอบที่ 5 กำหนดค่า i เท่ากับ i - 1 จากนั้นทำกรอบที่ 2

(6) กรอบที่ 6 กำหนดค่า mcount เท่ากับ i + 1 จากนั้นทำกรอบที่ 7

(7) กรอบที่ 7 ตรวจสอบเงื่อนไขว่า mcount มากกว่า Ubound(TC) หรือไม่ หากเป็นจริงทำกรอบที่ 8 หากไม่เป็นจริงทำกรอบที่ 9

(8) กรอบที่ 8 กำหนดค่า j เท่ากับ j + 1 จากนั้นทำกรอบที่ 4

(9) กรอบที่ 9 กำหนดค่า ci(j) เท่ากับ TC(mcount).ToType จากนั้นทำกรอบที่ 11

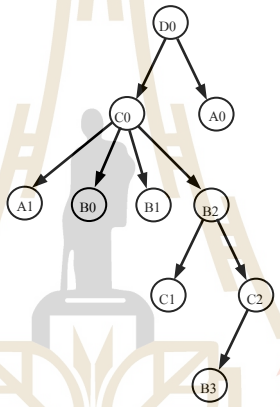
(10) กรอบที่ 6 กำหนดค่า mcount เท่ากับ mcount + 1 จากนั้นทำกรอบที่ 7

(11) กรอบที่ 11 ทำการบันทึกค่าในตัวแปร Bak(i) เท่ากับ ((StockRemovalBasic(mcount)) x (DirectionCut(mcount))) + Bak(i) จากนั้นทำกรอบที่ 10

มหาวิทยาลัยเทคโนโลยีสุรนารี

### 3.2.8 Function CreaPath

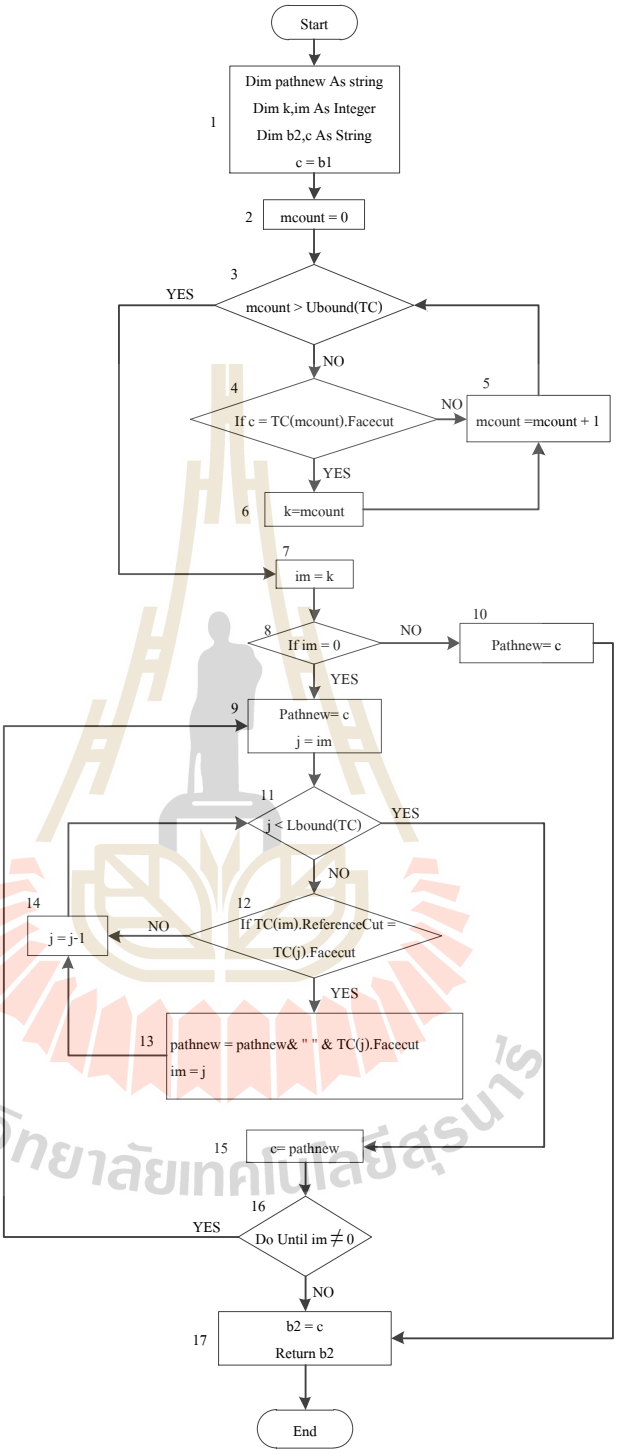
โดย Function มีหน้าที่สร้าง Path เพื่อคำนวณค่าความคลาดเคลื่อนสะสมในแผนภูมิฯโดยเส้นทาง (path) แทนความสัมพันธ์ของผิวตัดที่เป็นโหนดในกราฟต้นไม้ในรูปที่ 3.32 จากตารางที่ 3.12 แสดงเส้นความสัมพันธ์ของรอยตัดสุดท้าย จะได้ความสัมพันธ์ของรอยตัดดังนี้ ตัดผิวสุดท้ายที่ผิว B3 โดยอ้างอิงจากผิว C2 จากนั้นไปค้นหาผิว C2 ว่าอ้างอิงจากผิวใด ผลจากการค้นหาคือผิว B2 ทำการค้นหาไปแบบที่กล่าวมาจนถึงผิวอ้างอิงเริ่มต้นคือ D0 จึงจะสิ้นสุดความสัมพันธ์ในรอยตัดสุดท้าย รูปที่ 3.33 แสดง Flow Chart ของขั้นตอนการทำงานของ Function CreaPath และสรุปลำดับการทำงานได้ดังนี้ :-




รูปที่ 3.32กราฟต้นไม้แสดงความสัมพันธ์ของรอยตัด

ตารางที่ 3.12 แสดงขั้นตอนการตัดโลหะของรูปชิ้นงานตัวอย่างที่ 3.15 และแสดง Path ในแต่ละขั้นตอนการตัดโลหะ

| OP.No. | M/C USED | Referenced cut | Machined cut | Path |
|---|---|---|---|---|
| 10 | W&S turret lathe | D0 | C0 | C0 D0 |
| 10 | W&S turret lathe | C0 | B0 | B0 C0 D0 |
| 10 | W&S turret lathe | D0 | A0 | A0 D0 |
| 20 | Harding chucker | C0 | B1 | B1 C0 D0 |
| 20 | Harding chucker | C0 | A1 | A1 C0 D0 |
| 30 | Norton cylinder grinder | C0 | B2 | B2 C0 D0 |
| 40 | Norton cylinder grinder | B2 | C1 | C1 B2 C0 D0 |
| 50 | Rough grind 29.6 mm. dia. | - | - | |
| 60 | Harden | - | - | |
| 70 | Lap centers | - | - | |
| 80 | Norton cylinder grinder | B2 | C2 | C2 B2 C0 D0 |
| 90 | Norton cylinder grinder | C2 | B3 | B3 C2 B2 C0 D0 |
| 100 | Finish grind 29.6 mm. dia. | - | - | |




รูปที่ 3.33 Flow chart แสดงลำดับขั้นตอนการทำงานของ Function CreaPath

(1) กรอบที่ 1 ประกาศตัวแปร pathnew,k,b2,c และ im; กำหนดค่า c เท่ากับ b1 จากนั้นทำกรอบที่ 2

(2) กรอบที่ 2 กำหนดค่า mcount เท่ากับ 0 จากนั้นทำกรอบที่ 3

(3) กรอบที่ 3 ตรวจสอบเงื่อนไขว่า mcount มากกว่า Ubound(TC) หรือไม่ หากเป็นจริงทำกรอบที่ 7 หากไม่เป็นจริงทำกรอบที่ 4

(4) กรอบที่ 4 ตรวจสอบเงื่อนไขว่า c เท่ากับ TC(mcount).Facecut หรือไม่ หากเป็นจริงทำกรอบที่ 6 หากไม่เป็นจริงทำกรอบที่ 5

(5) กรอบที่ 5 กำหนดค่า mcount เท่ากับ mcount + 1 จากนั้นทำกรอบที่ 3

(6) กรอบที่ 6 กำหนดค่า k เท่ากับ mcount จากนั้นทำกรอบที่ 5

(7) กรอบที่ 7 กำหนดค่า im เท่ากับ k จากนั้นทำกรอบที่ 8

(8) กรอบที่ 8 ตรวจสอบเงื่อนไขว่า im เท่ากับ 0 หรือไม่ หากเป็นจริงทำกรอบที่ 9 หากไม่เป็นจริงทำกรอบที่ 10

(9) กรอบที่ 9 กำหนดค่า pathnew เท่ากับ c และกำหนดค่า j เท่ากับ im จากนั้นทำกรอบที่ 11

(10) กรอบที่ 10 กำหนดค่า pathnew เท่ากับ c จากนั้นทำกรอบที่ 17

(11) กรอบที่ 11 ตรวจสอบเงื่อนไขว่า j น้อยกว่า Lbound(TC) หรือไม่ หากเป็นจริงทำกรอบที่ 15 หากไม่เป็นจริงทำกรอบที่ 12

(12) กรอบที่ 12 ตรวจสอบเงื่อนไขว่า TC(im).Referencecut เท่ากับ TC(j).Facecut หรือไม่ หากเป็นจริงทำกรอบที่ 13 หากไม่เป็นจริงทำกรอบที่ 14

(13) กรอบที่ 13 กำหนดค่า pathnew เท่ากับ pathnew &" "& TC(j).Facecut จากนั้นทำกรอบที่ 14

(14) กรอบที่ 14 กำหนดค่า j เท่ากับ j - 1 จากนั้นทำกรอบที่ 11

(15) กรอบที่ 15 กำหนดค่า c เท่ากับ pathnew จากนั้นทำกรอบที่ 16

(16) กรอบที่ 16 ตรวจสอบเงื่อนไขว่า im ต้องไม่เท่าเท่ากับ TC(j).Facecut หรือไม่ หากเป็นจริงทำกรอบที่ 9 หากไม่เป็นจริงทำกรอบที่ 17

(17) กรอบที่ 17 กำหนดค่า b2 เท่ากับ c และส่งค่า b2 ออกจาก Function จากนั้นจบการทำงาน

มหาวิทยาลัยเทคโนโลยีสุรนารี

### 3.2.9 Function PathOne

โดย Function มีหน้าที่เรียงลำดับความสัมพันธ์ใน Path ใหม่ โดยเรียงลำดับจากหลังมาหน้าโดยใช้คำสั่ง Reverse จากนั้นบันทึกค่าลงในตัวแปร b รูปที่ 3.34 Flow Chart แสดงขั้นตอนการทำงานของ Function PathOne

## 3.3 กลุ่มโมดูลแสดงผลลัพธ์ (Output Modules)

กลุ่มโมดูลแสดงผลลัพธ์ (Output Modules) ทำหน้าที่แสดงผลลัพธ์ของโปรแกรมตามแบบขั้นตอนการคำนวณของแผนภูมิฯ โดยรับค่าการคำนวณจากโมดูลย่อยทั้งหลายที่เกี่ยวข้องของโปรแกรมมาแสดงผลลัพธ์ให้ผู้ใช้งานพิจารณาความเหมาะสม

## 3.4 กลุ่มข้อมูลย้อนกลับ (Feedback Modules)

กลุ่มข้อมูลย้อนกลับ (Feedback Modules) เป็นกลุ่มโมดูลเพื่ออำนวยความสะดวกต่อผู้ใช้โปรแกรมในการปรับเปลี่ยนขั้นตอนการตัดโลหะ เช่น เชื่อมโยงกับโมดูลประมวลผล (Processing Modules)

## 3.5 สรุป

ในบทนี้ได้กล่าวถึงรายละเอียดของการทำงานของโมดูลหลักๆของโปรแกรมแผนภูมิความคลาดเคลื่อน ได้แก่โมดูลรับข้อมูล (Input Modules); โมดูลประมวลผล (Processing Modules); โมดูลแสดงผลลัพธ์ (Output Modules); และโมดูลข้อมูลย้อนกลับ (Feedback Modules)

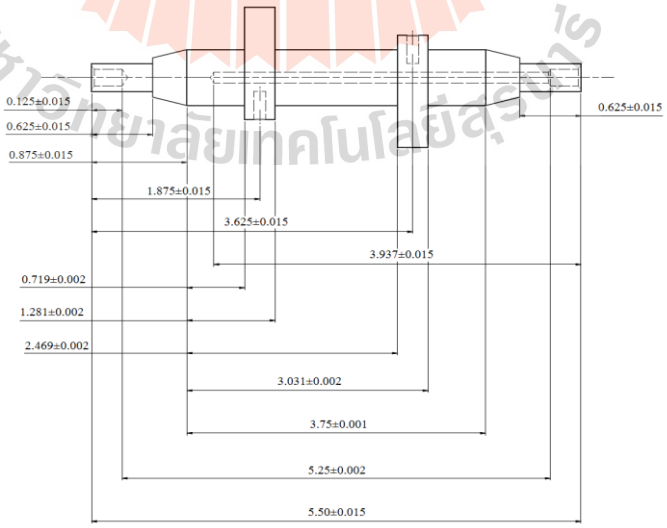
มหาวิทยาลัยเทคโนโลยีสุรนารี

# บทที่ 4
# ตัวอย่างการใช้โปรแกรมแผนภูมิความคลาดเคลื่อน

ในบทนี้เป็นส่วนของการทดสอบการทำงานของโปรแกรมแผนภูมิความคลาดเคลื่อน เปรียบเทียบผลลัพธ์จากการคำนวณตามทฤษฎีด้วยวิธีแผนภูมิต้นไม้ โดยจะแสดงรายละเอียดหลักดังนี้ 1) ส่วนของการรับภาพสเกตช์ชิ้นงาน 2 มิติ; 2) ส่วนของการกำหนดขนาด Blueprint; 3) ส่วนของการกำหนดขั้นตอนในการตัด; 4) ส่วนของการรับข้อมูลและแสดงผลลัพธ์ของแผนภูมิความคลาดเคลื่อน; และ 5) ส่วนปรับปรุงแก้ไขแผนภูมิฯ โดยใช้ตัวอย่างชิ้นงานประกอบการคำนวณดังรูปที่ 4.1

## 4.1 ส่วนของการรับภาพสเกตช์ชิ้นงาน 2 มิติ

ส่วนของการรับภาพสเกตช์ชิ้นงาน 2 มิติ เป็นส่วนแรกที่ผู้ใช้งานโปรแกรมแผนภูมิความคลาดเคลื่อนจะต้องเขียนภาพสเกตช์ของชิ้นงาน ลงในกรอบสี่เหลี่ยมที่กำหนดให้ เพื่อให้โปรแกรมสามารถนำภาพสเกตช์นั้นไปประมวลผลหาจำนวนรอยตัดของชิ้นงาน แสดงดังรูปที่ 4.2 โดยมีขั้นตอนการตัดโลหะในตารางที่ 4.1

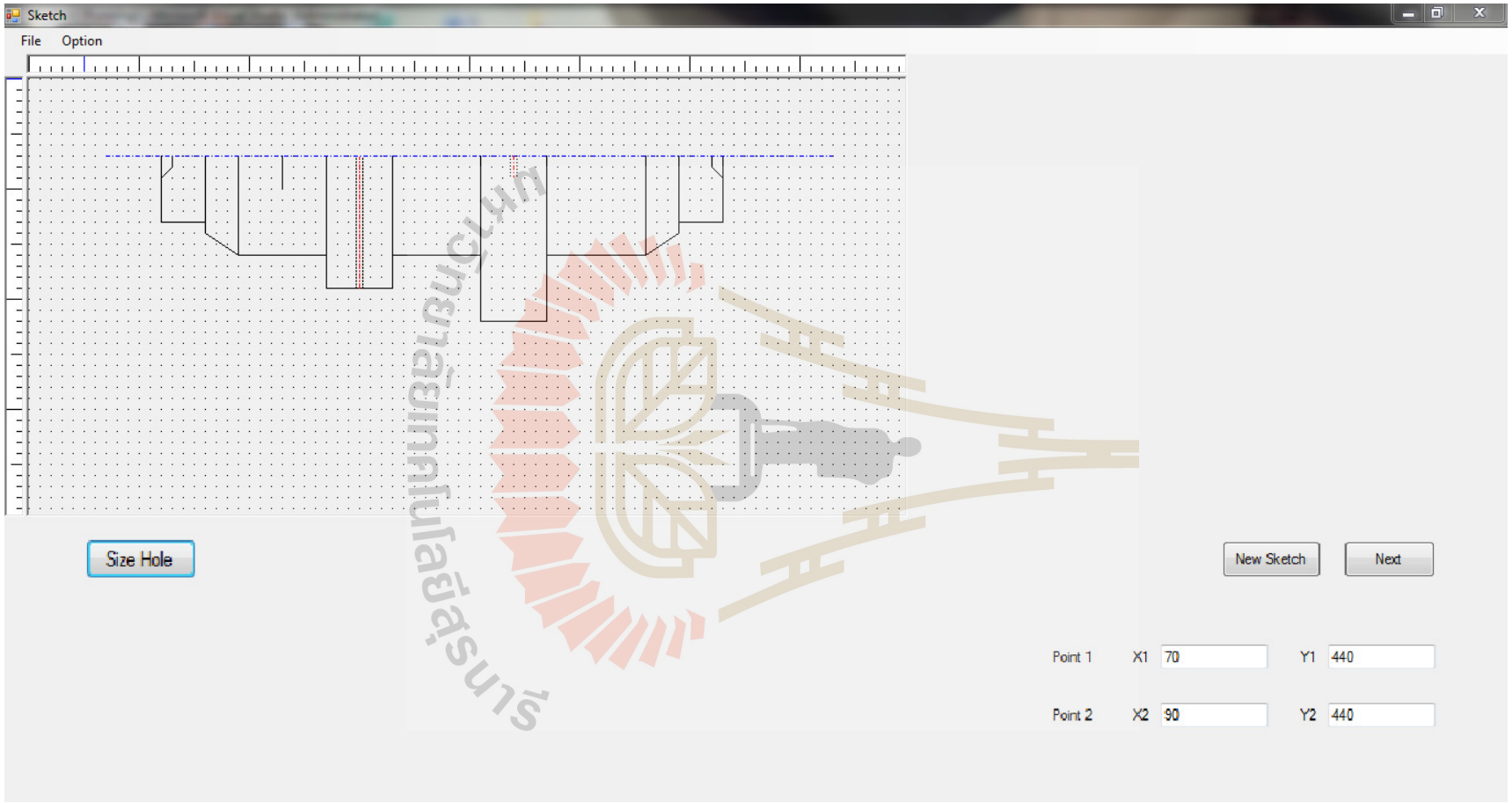



รูปที่ 4.1 แสดงชิ้นงานตัวอย่างประกอบการคำนวณแผนภูมิ

ตารางที่ 4.1 แสดงขั้นตอนการตัดโลหะของชิ้นงาน รูปที่ 4.1

| ลำดับที่ | ชื่อขั้นตอน | ลักษณะการตัดผิว | ผิวอ้างอิง | ผิวตัด | ค่าความคลาดเคลื่อน ±Tol./2 (mm) |
|---|---|---|---|---|---|
| 0 | Operation 00 | กลึงปาดหน้าผิวอ้างอิง | | ตัดผิว A | |
| 1 | Operation 10 | กลึงเซาะร่อง | อ้างอิงผิว A | ตัดผิว F | 0.031 |
| 2 | Operation 20 | กลึงบ่า | อ้างอิงผิว F | ตัดผิว H | 0.031 |
| 3 | Operation 30 | กลึงเซาะร่อง | อ้างอิงผิว F | ตัดผิว I | 0.031 |
| 4 | Operation 40 | กลึงบ่า | อ้างอิงผิว I | ตัดผิว K | 0.031 |
| 5 | Operation 50 | กลึงปาดหน้า | อ้างอิงผิว F | ตัดผิว O | 0.062 |
| 6 | Operation 60 | กลึงบ่า | อ้างอิงผิว F | ตัดผิว K | 0.005 |
| 7 | Operation 70 | กลึงปาดหน้า | อ้างอิงผิว K | ตัดผิว O | 0.005 |
| 8 | Operation 80 | กลึงปาดหน้า | อ้างอิงผิว K | ตัดผิว A | 0.005 |
| 9 | Operation 90 | กลึงเซาะร่อง | อ้างอิงผิว A | ตัดผิว C | 0.015 |
| 10 | Operation 100 | กลึงเรียว | อ้างอิงผิว A | ตัดผิว D | 0.010 |
| 11 | Operation 110 | กลึงเซาะร่อง | อ้างอิงผิว A | ตัดผิว F | 0.003 |
| 12 | Operation 120 | กลึงบ่า | อ้างอิงผิว A | ตัดผิว H | 0.003 |
| 13 | Operation 130 | กลึงเซาะร่อง | อ้างอิงผิว A | ตัดผิว I | 0.003 |
| 14 | Operation 140 | กลึงปาดหน้า | อ้างอิงผิว A | ตัดผิว O | 0.015 |
| 15 | Operation 150 | กลึงบ่า | อ้างอิงผิว A | ตัดผิว K | 0.003 |
| 16 | Operation 160 | กลึงเรียว | อ้างอิงผิว A | ตัดผิว L | 0.010 |
| 17 | Operation 170 | กลึงเซาะร่อง | อ้างอิงผิว O | ตัดผิว M | 0.015 |
| 18 | Operation 180 | เจาะรู | อ้างอิงผิว O | ตัดผิว E | 0.015 |
| 19 | Operation 190 | เจาะรู | อ้างอิงผิว A | ตัดผิว G | 0.015 |
| 20 | Operation 200 | เจาะรู | อ้างอิงผิว A | ตัดผิว J | 0.015 |
| 21 | Operation 210 | เจาะรู | อ้างอิงผิว A | ตัดผิว B | 0.010 |
| 22 | Operation 220 | เจาะรู | อ้างอิงผิว B | ตัดผิว N | 0.002 |
| 23 | Operation 230 | กลึงเรียว | อ้างอิงผิว B | ตัดผิว D | 0.010 |
| 24 | Operation 240 | กลึงบ่า | อ้างอิงผิว D | ตัดผิว L | 0.010 |
| 25 | Operation 250 | กลึงเรียว | อ้างอิงผิว B | ตัดผิว D | 0.005 |
| 26 | Operation 260 | กลึงบ่า | อ้างอิงผิว D | ตัดผิว L | 0.001 |

มหาวิทยาลัยเทคโนโลยีสุรนารี




รูปที่ 4.2 แสดงการสเกตซ์ภาพชิ้นงาน 2 มิติ

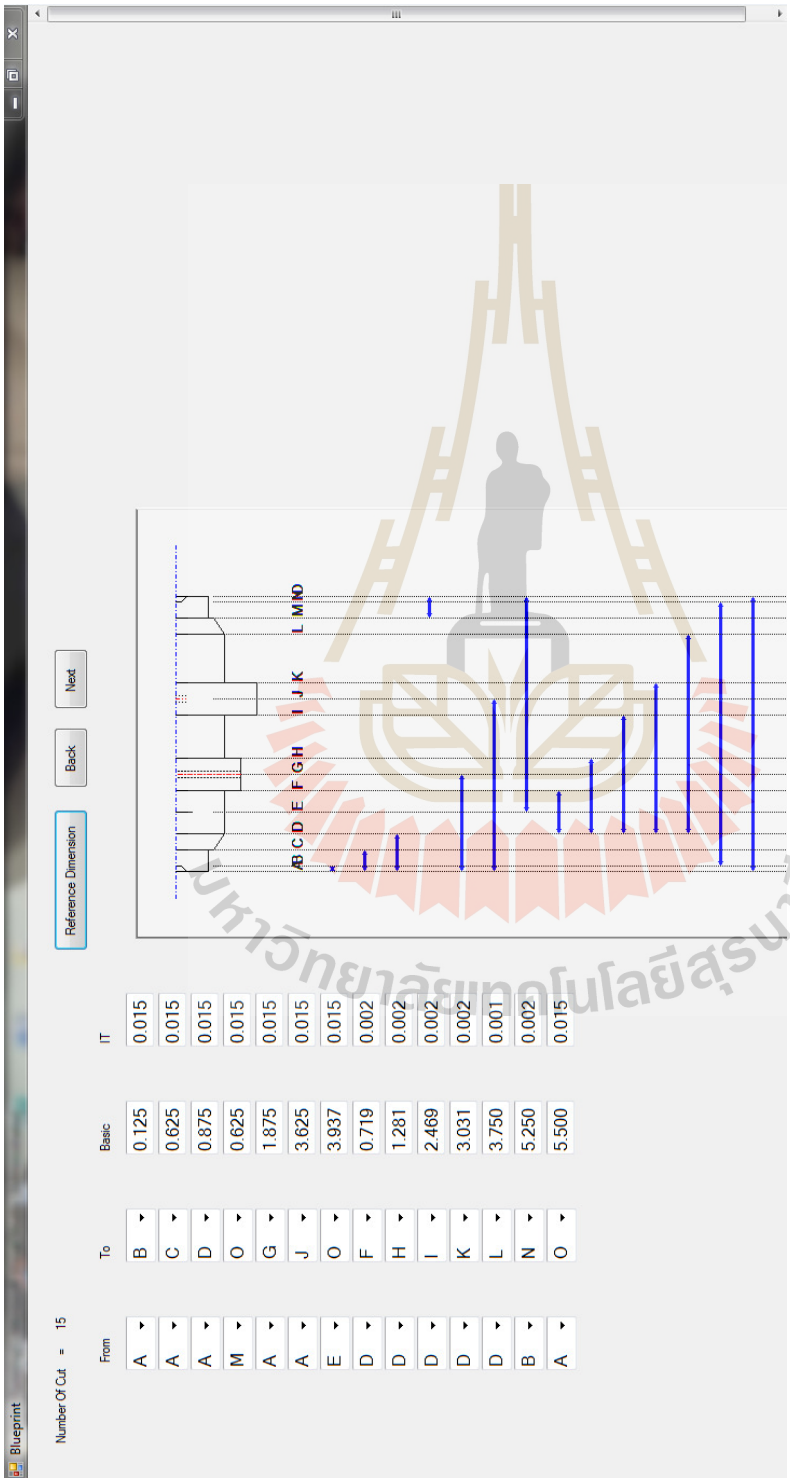
ในรูปที่ 4.3 โปรแกรมประมวลผลจำนวนรอยตัด (Number of cut) เท่ากับ 7 รอยตัด โดยมีชื่อตัวอักษรภาษาอังกฤษพิมพ์ใหญ่ระบุตำแหน่งของรอยตัดดังนี้ A, B, C, D, E, F, และG

รูปที่ 4.3 แสดงผลจำนวนรอยตัดของชิ้นงาน

มหาวิทยาลัยเทคโนโลยีสุรนารี

## 4.2 ส่วนของการกำหนดขนาด Blueprint

ในส่วนนี้โปรแกรมจะแสดง textbox เพื่อรับข้อมูลจากผู้ใช้งาน โดยผู้ใช้งานจะต้องใส่ข้อมูลดังนี้ 1) จุดอ้างอิงของขนาด 2) ขนาดมูลฐาน และ 3) ค่าความคลาดเคลื่อน แสดงดังรูปที่ 4.4

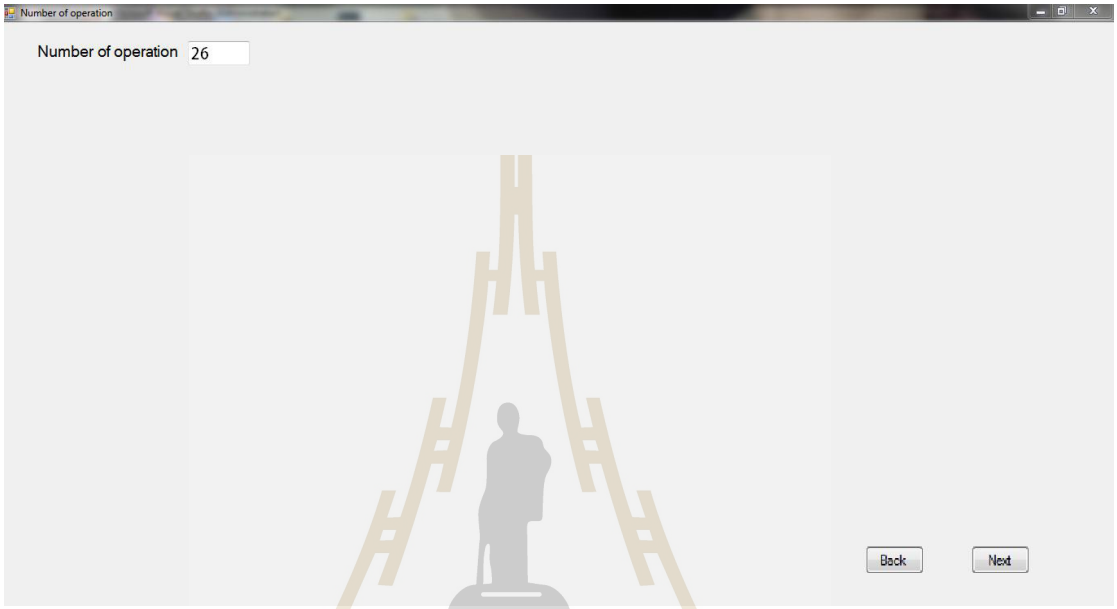



รูปที่ 4.4 แสดงการรับข้อมูลจากแบบ Blueprint (มิติทุกมิติหน่วยเป็นมิลลิเมตร)

มหาวิทยาลัยเทคโนโลยีสุรนารี

## 4.3 ส่วนของการกำหนดขั้นตอนในการตัด

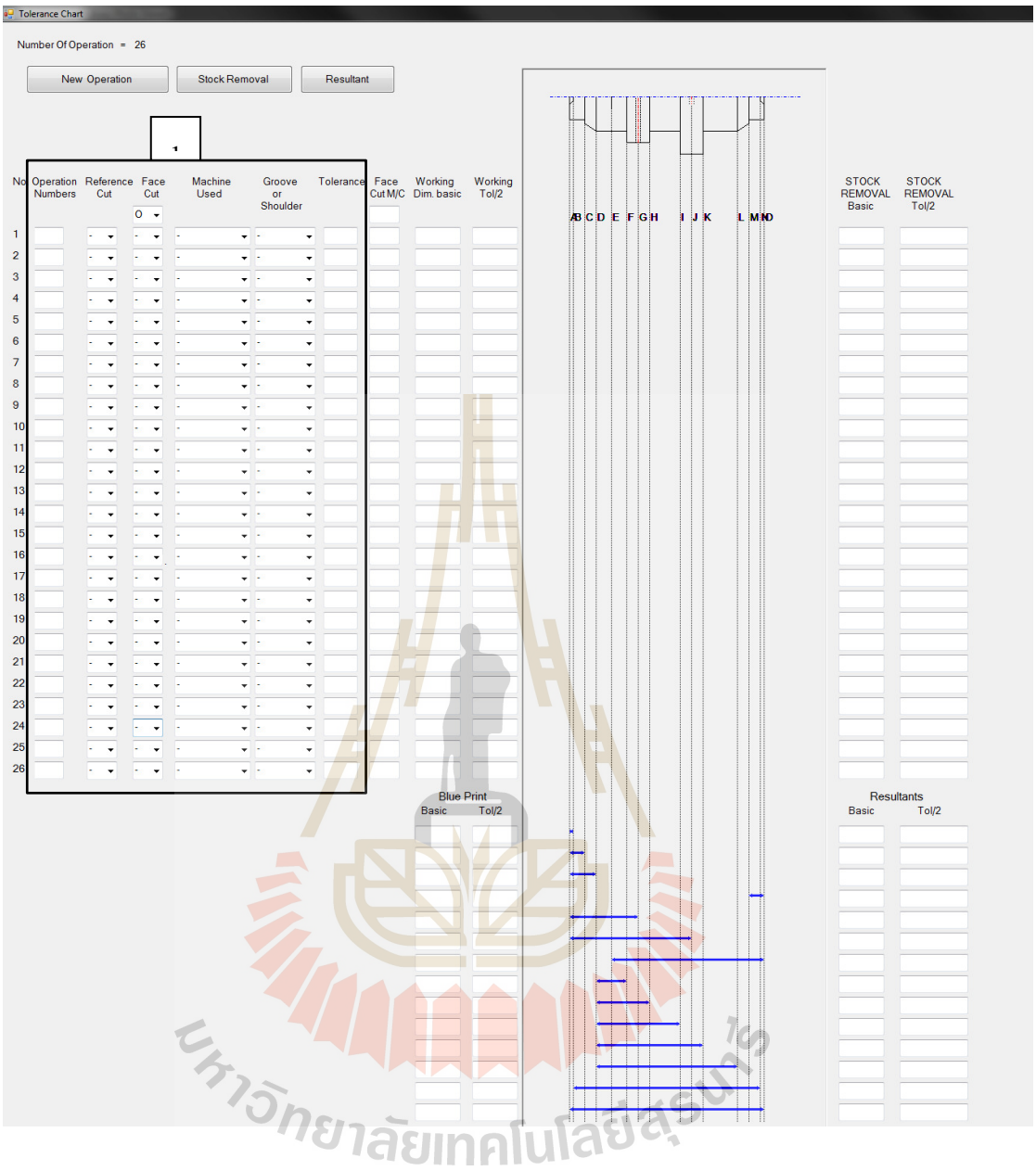
ในส่วนนี้เป็นส่วนที่ผู้ใช้งานโปรแกรมกำหนดจำนวนของขั้นตอนในการตัด ในรูปที่ 4.5 กำหนดขั้นในการตัดโลหะจำนวน 11 ขั้นตอน




รูปที่ 4.5 แสดงการกำหนดจำนวนขั้นตอนการตัดโลหะจำนวน 26 ขั้นตอน

## 4.4 ส่วนของการรับข้อมูลและแสดงผลลัพธ์ของแผนภูมิความคลาดเคลื่อน

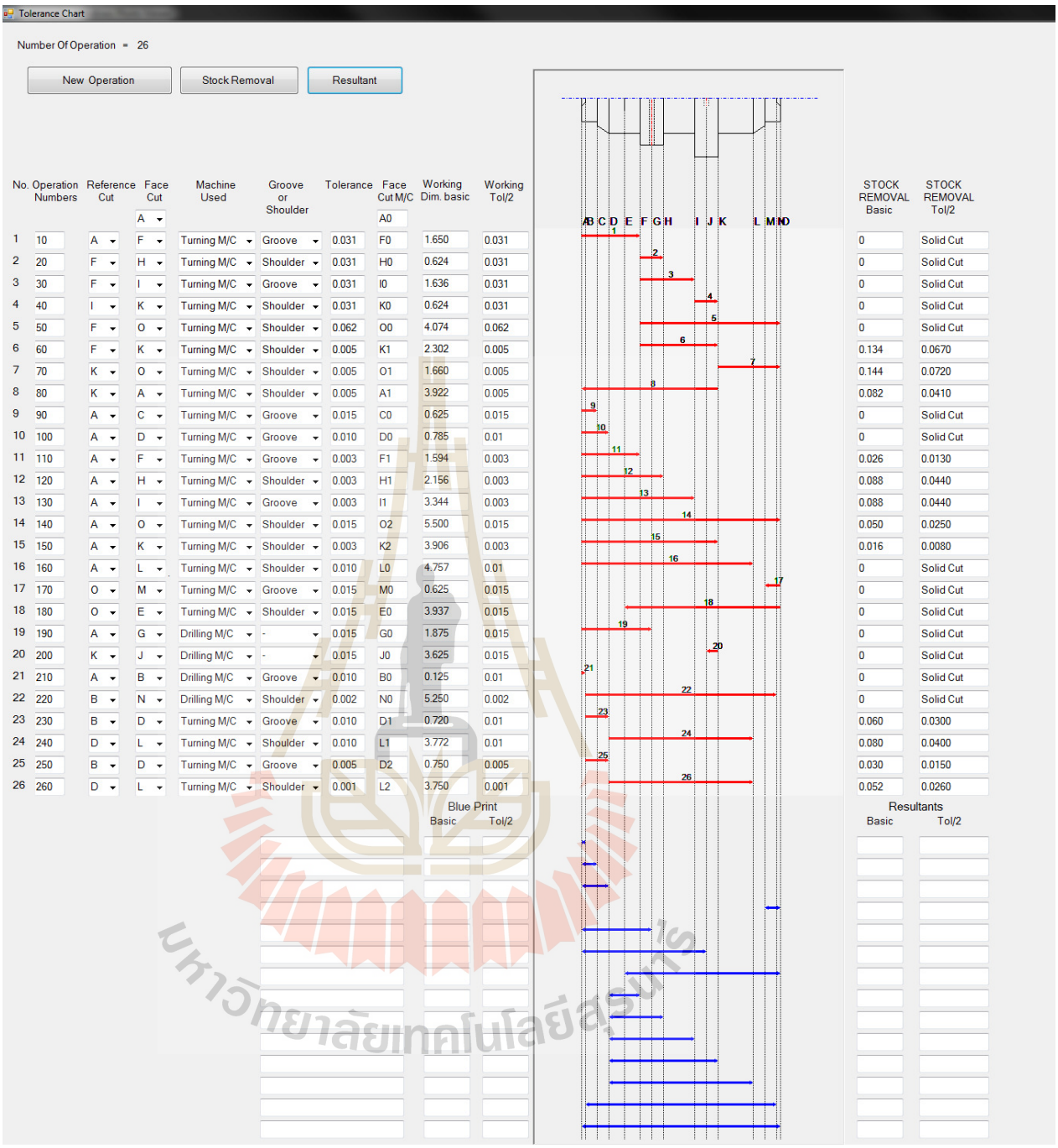
ในส่วนนี้ผู้ใช้งานโปรแกรมแผนภูมิความคลาดคลาดจะต้องใส่ข้อมูลของกระบวนการตัดในแต่ละขั้นตอน เพื่อใช้สำหรับคำนวณหาผลลัพธ์ของแผนภูมิฯ แสดงพื้นที่ใส่ข้อมูลดังรูปที่ 4.6 ในกรอบที่ 1 โดยผู้ใช้งานจะต้องใส่ข้อมูลดังนี้ Operation numbers, จุดอ้างอิงรอยตัด, จุดตัด, เครื่องใช้ที่ใช้ในขั้นตอนการตัด, ลักษณะของรอยตัดเป็นการตัดร่องหรือตัดบ่า, และสุดท้ายความคลาดเคลื่อนของกระบวนการตัดใส่หรือไม่ใส่ก็ได้ หากไม่ใส่โปรแกรมจะประมวลผลมาใส่โดยอัตโนมัติ




รูปที่ 4.6 แสดงส่วนของการรับข้อมูลในการตัดโลหะ

รูปที่ 4.7 ผู้ใช้งานใส่ข้อมูลในการตัดโลหะตามตารางที่ 4.1 และทำการประมวลผลโดยกดปุ่ม Stock Removal โปรแกรมจะแสดงความคลาดเคลื่อนสะสม, ขนาดมูลฐานของความคลาดเคลื่อนสะสม, ขนาดมูลฐานของขั้นตอนการตัด, และความคลาดเคลื่อนของขั้นตอนการตัด

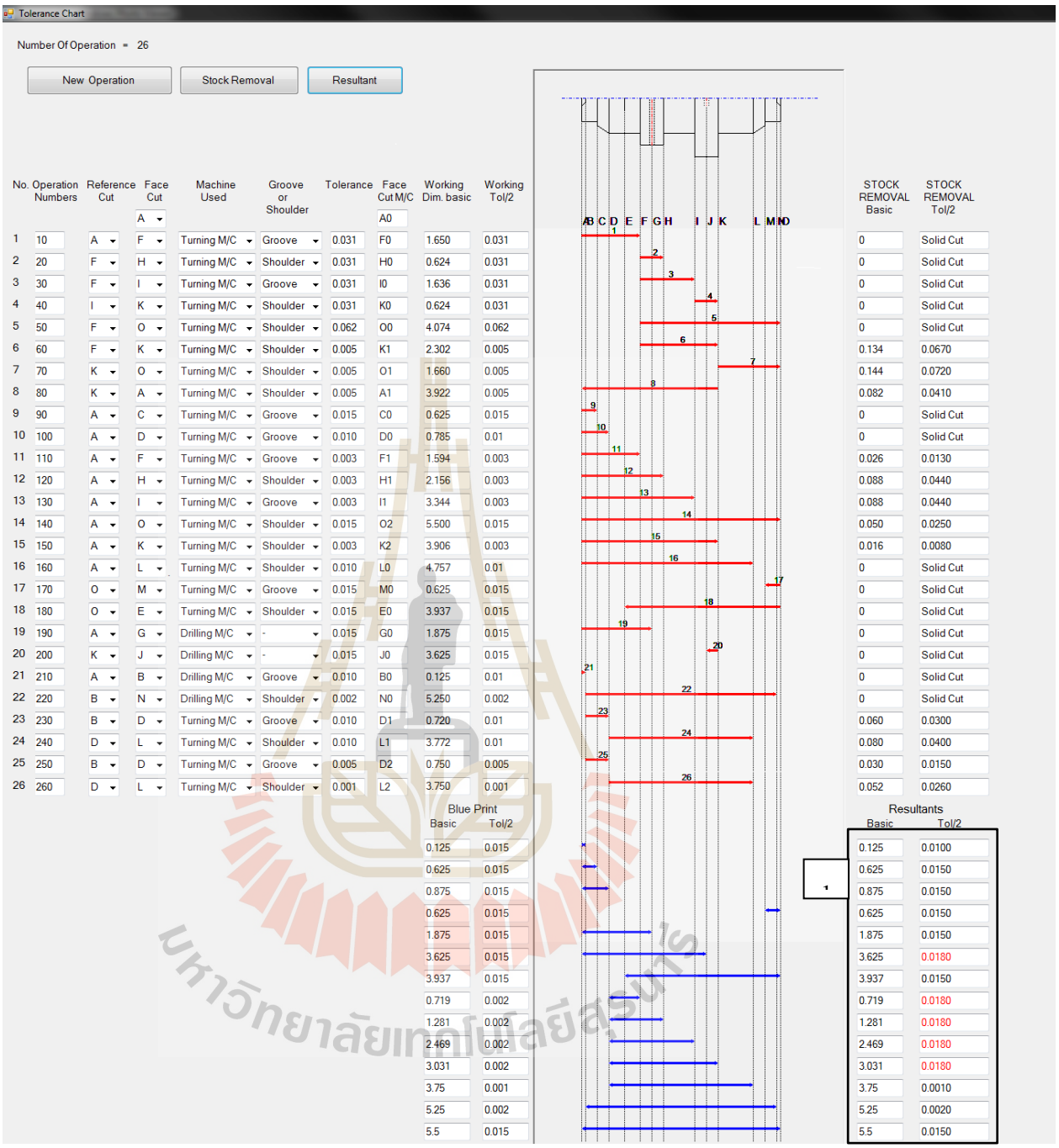
สำหรับโปรแกรมแผนภูมิความคลาดเคลื่อนนี้ได้กำหนดค่า ขนาดมูลฐานของโลหะที่ถูกตัดออกไป (stock removal) เท่ากับค่าความคลาดเคลื่อนคูณสอง เพื่อป้องกันการตัดอากาศ




รูปที่ 4.7 แสดงการรับข้อมูลในการตัดโลหะ และคำนวณความคลาดเคลื่อนสะสม

รูปที่ 4.8 ผู้ใช้งานกดปุ่ม Resultant โปรแกรมจะแสดงผลลัพธ์ของแผนภูมิความเคลื่อนจากรูปที่ 4.8 ในกรอบที่ 1 จะแสดงให้เห็นตัวอย่างว่า หากทำการตัดโลหะตามขั้นตอนของแผนภูมิฯ อาจทำให้ขนาดของชิ้นงานไม่เป็นไปตามแบบกำหนดเนื่องจาก ความคลาดเคลื่อนสะสมของขนาดมูลฐาน จุด A ถึง J เกินขนาดจากแบบกำหนดของชิ้นงาน โดย จุด A ถึง J ได้ความคลาดเคลื่อนสะสมที่ 0.018 ซึ่งโปรแกรมแสดงตัวสีแดงหมายความว่าหากวางแผนขั้นตอนการตัดแบบนี้อาจทำให้ขนาดของชิ้นงานไม่เป็นไปตามแบบกำหนด ผู้ใช้งานจะต้องทำการแก้ไขโดยว่างแผนขั้นตอนการตัดโลหะใหม่ จึงขอแสดงตัวอย่างการปรับปรุงแก้ไขแผนภูมิฯ โดยวางแผนกระบวนการตัดโลหะใหม่ดังตารางที่ 4.2 คือเพิ่มขั้นขั้นในการตัดโลหะจำนวน 4 ขั้นตอนเพื่อปรับปรุงกระบวนการตัดโลหะให้เป็นไปตามแบบที่กำหนดไว้

มหาวิทยาลัยเทคโนโลยีสุรนารี



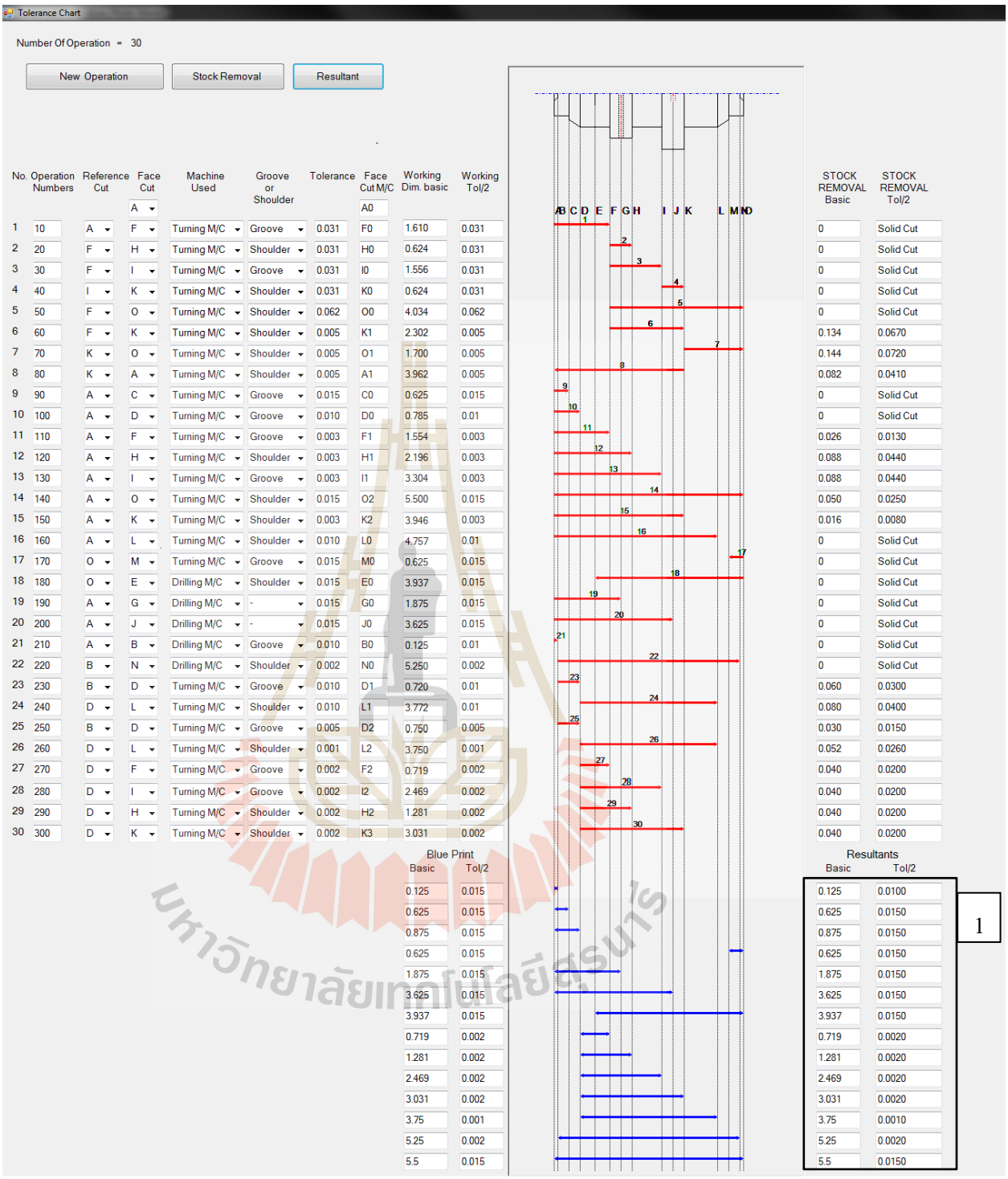

| No. | Operation Numbers | Reference Cut | Face Cut | Machine Used | Groove or Shoulder | Tolerance | Face Cut M/C | Working Dim. basic | Working Tol/2 | STOCK REMOVAL Basic | STOCK REMOVAL Tol/2 | Blue Print Basic | Blue Print Tol/2 | Resultants Basic | Resultants Tol/2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | A | | | | A0 | | | | | | | | |
| 1 | 10 | A | F | Turning M/C | Groove | 0.031 | F0 | 1.650 | 0.031 | 0 | Solid Cut | 0.125 | 0.015 | 0.125 | 0.0100 |
| 2 | 20 | F | H | Turning M/C | Shoulder | 0.031 | H0 | 0.624 | 0.031 | 0 | Solid Cut | 0.625 | 0.015 | 0.625 | 0.0150 |
| 3 | 30 | F | I | Turning M/C | Groove | 0.031 | I0 | 1.636 | 0.031 | 0 | Solid Cut | 0.875 | 0.015 | 0.875 | 0.0150 |
| 4 | 40 | I | K | Turning M/C | Shoulder | 0.031 | K0 | 0.624 | 0.031 | 0 | Solid Cut | 0.625 | 0.015 | 0.625 | 0.0150 |
| 5 | 50 | F | O | Turning M/C | Shoulder | 0.062 | O0 | 4.074 | 0.062 | 0 | Solid Cut | 1.875 | 0.015 | 1.875 | 0.0150 |
| 6 | 60 | F | K | Turning M/C | Shoulder | 0.005 | K1 | 2.302 | 0.005 | 0.134 | 0.0670 | 3.625 | 0.015 | 3.625 | 0.0180 |
| 7 | 70 | K | O | Turning M/C | Shoulder | 0.005 | O1 | 1.660 | 0.005 | 0.144 | 0.0720 | 3.937 | 0.015 | 3.937 | 0.0150 |
| 8 | 80 | K | A | Turning M/C | Shoulder | 0.005 | A1 | 3.922 | 0.005 | 0.082 | 0.0410 | 0.719 | 0.002 | 0.719 | 0.0180 |
| 9 | 90 | A | C | Turning M/C | Groove | 0.015 | C0 | 0.625 | 0.015 | 0 | Solid Cut | 1.281 | 0.002 | 1.281 | 0.0180 |
| 10 | 100 | A | D | Turning M/C | Groove | 0.010 | D0 | 0.785 | 0.01 | 0 | Solid Cut | 2.469 | 0.002 | 2.469 | 0.0180 |
| 11 | 110 | A | F | Turning M/C | Groove | 0.003 | F1 | 1.594 | 0.003 | 0.026 | 0.0130 | 3.031 | 0.002 | 3.031 | 0.0180 |
| 12 | 120 | A | H | Turning M/C | Shoulder | 0.003 | H1 | 2.156 | 0.003 | 0.088 | 0.0440 | 3.75 | 0.001 | 3.75 | 0.0010 |
| 13 | 130 | A | I | Turning M/C | Groove | 0.003 | I1 | 3.344 | 0.003 | 0.088 | 0.0440 | 5.25 | 0.002 | 5.25 | 0.0020 |
| 14 | 140 | A | O | Turning M/C | Shoulder | 0.015 | O2 | 5.500 | 0.015 | 0.050 | 0.0250 | 5.5 | 0.015 | 5.5 | 0.0150 |
| 15 | 150 | A | K | Turning M/C | Shoulder | 0.003 | K2 | 3.906 | 0.003 | 0.016 | 0.0080 | | | | |
| 16 | 160 | A | L | Turning M/C | Shoulder | 0.010 | L0 | 4.757 | 0.01 | 0 | Solid Cut | | | | |
| 17 | 170 | O | M | Turning M/C | Groove | 0.015 | M0 | 0.625 | 0.015 | 0 | Solid Cut | | | | |
| 18 | 180 | O | E | Turning M/C | Shoulder | 0.015 | E0 | 3.937 | 0.015 | 0 | Solid Cut | | | | |
| 19 | 190 | A | G | Drilling M/C | - | 0.015 | G0 | 1.875 | 0.015 | 0 | Solid Cut | | | | |
| 20 | 200 | K | J | Drilling M/C | - | 0.015 | J0 | 3.625 | 0.015 | 0 | Solid Cut | | | | |
| 21 | 210 | A | B | Drilling M/C | Groove | 0.010 | B0 | 0.125 | 0.01 | 0 | Solid Cut | | | | |
| 22 | 220 | B | N | Drilling M/C | Shoulder | 0.002 | N0 | 5.250 | 0.002 | 0 | Solid Cut | | | | |
| 23 | 230 | B | D | Turning M/C | Groove | 0.010 | D1 | 0.720 | 0.01 | 0.060 | 0.0300 | | | | |
| 24 | 240 | D | L | Turning M/C | Shoulder | 0.010 | L1 | 3.772 | 0.01 | 0.080 | 0.0400 | | | | |
| 25 | 250 | B | D | Turning M/C | Groove | 0.005 | D2 | 0.750 | 0.005 | 0.030 | 0.0150 | | | | |
| 26 | 260 | D | L | Turning M/C | Shoulder | 0.001 | L2 | 3.750 | 0.001 | 0.052 | 0.0260 | | | | |

มหาวิทยาลัยเทคโนโลยีสุรนารี

รูปที่ 4.8 แสดงผลลัพธ์ของแผนภูมิความคลาดเคลื่อน

ตารางที่ 4.2 แสดงขั้นตอนการตัดโลหะของชิ้นงาน รูปที่ 1.2 แบบปรับปรุงใหม่

| ลำดับที่ | ชื่อขั้นตอน | ลักษณะการตัดผิว | ผิวอ้างอิง | ผิวตัด | ค่าความคลาดเคลื่อน ±Tol./2 (mm) |
|---|---|---|---|---|---|
| 0 | Operation 00 | กลึงปาดหน้าผิวอ้างอิง | | ตัดผิว A | |
| 1 | Operation 10 | กลึงเซาะร่อง | อ้างอิงผิว A | ตัดผิว F | 0.031 |
| 2 | Operation 20 | กลึงบ่า | อ้างอิงผิว F | ตัดผิว H | 0.031 |
| 3 | Operation 30 | กลึงเซาะร่อง | อ้างอิงผิว F | ตัดผิว I | 0.031 |
| 4 | Operation 40 | กลึงบ่า | อ้างอิงผิว I | ตัดผิว K | 0.031 |
| 5 | Operation 50 | กลึงปาดหน้า | อ้างอิงผิว F | ตัดผิว O | 0.062 |
| 6 | Operation 60 | กลึงบ่า | อ้างอิงผิว F | ตัดผิว K | 0.005 |
| 7 | Operation 70 | กลึงปาดหน้า | อ้างอิงผิว K | ตัดผิว O | 0.005 |
| 8 | Operation 80 | กลึงปาดหน้า | อ้างอิงผิว K | ตัดผิว A | 0.005 |
| 9 | Operation 90 | กลึงเซาะร่อง | อ้างอิงผิว A | ตัดผิว C | 0.015 |
| 10 | Operation 100 | กลึงเรียว | อ้างอิงผิว A | ตัดผิว D | 0.010 |
| 11 | Operation 110 | กลึงเซาะร่อง | อ้างอิงผิว A | ตัดผิว F | 0.003 |
| 12 | Operation 120 | กลึงบ่า | อ้างอิงผิว A | ตัดผิว H | 0.003 |
| 13 | Operation 130 | กลึงเซาะร่อง | อ้างอิงผิว A | ตัดผิว I | 0.003 |
| 14 | Operation 140 | กลึงปาดหน้า | อ้างอิงผิว A | ตัดผิว O | 0.015 |
| 15 | Operation 150 | กลึงบ่า | อ้างอิงผิว A | ตัดผิว K | 0.003 |
| 16 | Operation 160 | กลึงเรียว | อ้างอิงผิว A | ตัดผิว L | 0.010 |
| 17 | Operation 170 | กลึงเซาะร่อง | อ้างอิงผิว O | ตัดผิว M | 0.015 |
| 18 | Operation 180 | เจาะรู | อ้างอิงผิว O | ตัดผิว E | 0.015 |
| 19 | Operation 190 | เจาะรู | อ้างอิงผิว A | ตัดผิว G | 0.015 |
| 20 | Operation 200 | เจาะรู | อ้างอิงผิว A | ตัดผิว J | 0.015 |
| 21 | Operation 210 | เจาะรู | อ้างอิงผิว A | ตัดผิว B | 0.010 |
| 22 | Operation 220 | เจาะรู | อ้างอิงผิว B | ตัดผิว N | 0.002 |
| 23 | Operation 230 | กลึงเรียว | อ้างอิงผิว B | ตัดผิว D | 0.010 |
| 24 | Operation 240 | กลึงบ่า | อ้างอิงผิว D | ตัดผิว L | 0.010 |
| 25 | Operation 250 | กลึงเรียว | อ้างอิงผิว B | ตัดผิว D | 0.005 |
| 26 | Operation 260 | กลึงบ่า | อ้างอิงผิว D | ตัดผิว L | 0.001 |
| 27 | Operation 270 | กลึงเซาะร่อง | อ้างอิงผิว D | ตัดผิว F | 0.002 |
| 28 | Operation 280 | กลึงเซาะร่อง | อ้างอิงผิว D | ตัดผิว I | 0.002 |
| 29 | Operation 290 | กลึงบ่า | อ้างอิงผิว D | ตัดผิว H | 0.002 |
| 30 | Operation 300 | กลึงบ่า | อ้างอิงผิว D | ตัดผิว K | 0.002 |




รูปที่ 4.9 แสดงผลลัพธ์ของแผนภูมิความคลาดเคลื่อน

## 4.5 ส่วนปรับปรุงแก้ไขแผนภูมิความคลาดเคลื่อน

ในรูปที่ 4.9 ผู้ใช้งานปรับปรุงแผนภูมิความคลาดเคลื่อนโดยใส่ข้อมูลขั้นตอนการตัดโลหะใหม่ดังตารางที่ 4.2 จากนั้นทำการประมวลผลลัพธ์ใหม่ จะเห็นได้ว่าผลลัพธ์ในกรอบที่ 1 ไม่แสดงตัวเลขสีแดงจึงทำให้แผนภูมิความคลาดเคลื่อนนี้ สามารถนำไปใช้งานในกระบวนการผลิตชิ้นงานได้ สำหรับรายละเอียดการใช้โปรแกรม ได้อธิบายในภาคผนวก ก.

## 4.6 สรุป

จากการใช้โปรแกรมแผนภูมิความคลาดเคลื่อนสร้างแผนภูมิความคลาดเคลื่อนนั้น โปรแกรมฯให้ผลลัพธ์เป็นที่น่าพอใจไม่ว่าจะเป็นการคำนวณค่าความคลาดเคลื่อนสะสมการปรับเปลี่ยนขั้นตอนในการผลิต หรือ การปรับปรุงเงื่อนไขของการตัดโลหะ ตลอดจนการติดต่อระหว่างโปรแกรมและผู้ใช้ (interaction) และเปรียบเทียบผลการคำนวณระหว่างโปรแกรมและการคำนวณด้วยมือโดยใช้วิธี กราฟต้นไม้ให้ผลลัพธ์ จากการคำนวณเท่ากัน ซึ่งแสดงให้เห็นว่าการสร้างโปรแกรมแผนภูมิความคลาดเคลื่อนนี้มีความถูกต้องตามแนวทางการสร้างแผนภูมิฯ

มหาวิทยาลัยเทคโนโลยีสุรนารี

# บทที่ 5
# สรุปผลงานวิจัย

## 5.1 สรุป

งานวิจัยนี้สามารถบรรลุวัตถุประสงค์ที่ตั้งไว้คือ สามารถสร้างแผนภูมิความคลาดเคลื่อนด้วยโปรแกรม Visual Basic โดยเพิ่มความสามารถของโปรแกรม 3 ส่วนคือ; (1) สามารถเปลี่ยนแปลงลำดับขั้นตอนการตัดโลหะ; (2) สามารถเพิ่มหรือลดขั้นตอนการตัดโลหะโดยวิธีปฏิสัมพันธ์กับผู้ใช้; (3) สามารถตรวจสอบความเหมาะสมของแผนภูมิความคลาดเคลื่อนของทั้งสองส่วนแรก และสามารถกล่าวโดยรวมว่าโปรแกรมแผนภูมิฯให้ผลลัพธ์เป็นที่น่าพอใจไม่ว่าจะเป็นการคำนวณค่าความคลาดเคลื่อนสะสมการปรับเปลี่ยนขั้นตอนในการผลิต หรือการปรับปรุงเงื่อนไขของการตัดโลหะ ตลอดจนการติดต่อระหว่างโปรแกรมและผู้ใช้ (interaction) ซึ่งมีการตอบโต้กับผู้ใช้งานโปรแกรมได้ง่ายและรวดเร็ว สำหรับงานวิจัยทางด้านนี้ในอนาคต ผู้ที่มีความสนใจพัฒนาโปรแกรมแผนภูมิฯ สามารถนำ code ในการเขียนโปรแกรมไปพัฒนา, ปรับปรุงให้โปรแกรมแผนภูมิฯ มีความสามารถยิ่งขึ้นไป

## 5.2 วิจารณ์โปรแกรม

การพัฒนาโปรแกรมแผนภูมิความคลาดเคลื่อนนี้มีลักษณะการเขียนโปรแกรมแบบ modular programing คือ การเขียนโปรแกรมแบบแยกส่วนย่อยหลายๆส่วน ทำให้ง่ายต่อการเรียกใช้งานโปรแกรมเนื่องจากแต่ละส่วนเป็นอิสระต่อกัน ข้อดีของการเขียนโปรแกรมแบบนี้คือสามารถแก้ไขตรวจสอบโปรแกรมได้ง่ายโปรแกรมประมวลผลรวดเร็ว สำหรับการเขียนโปรแกรมแบบชุดคำสั่งเฉพาะทางนั้นโปรแกรมต้องใช้ระยะเวลาในการดำเนินงานค่อยข้างมาก เนื่องจากต้องดำเนินการรันโปรแกรมตามขั้นตอนที่เขียนโปรแกรมกำหนดไว้ จากการทดสอบโปรแกรมแผนภูมิความคลาดเคลื่อนที่พัฒนานี้ สามารถประมวลผลลัพธ์ได้เร็วกว่า โปรแกรมที่พัฒนาโดยโปรแกรมตารางคำนวณในเงื่อนไขกระบวนการตัดโลหะที่เหมือนกัน

## 5.3 ข้อจำกัดของโปรแกรม

ในการพัฒนาโปรแกรมแผนภูมิความคลาดเคลื่อนยังมีข้อจำกัดอยู่บางส่วนคือ โปรแกรมไม่สามารถรับข้อมูลรูปภาพและมิติชิ้นงานจากโปรแกรมเฉพาะทางได้เช่น Autocad หรือ program Drafting อื่นๆ จะต้องใช้การเขียนรูปโดยผู้ใช้งานเอง หากมีการพัฒนาในส่วนนี้เพิ่มเติม ก็สามารถลดเวลาในการเขียนรูปชิ้นงานได้มาก

โปรแกรมแผนภูมิความคลาดเคลื่อนในงานวิจัยนี้ มีขอบเขตสำหรับงานตัดโลหะ (machined part) เท่านั้น; ใช้เฉพาะชิ้นงานตัดโลหะสองมิติที่มีการผลิตแบบไม่ต่อเนื่อง (rotational discrete machine part); ไม่มี Geometric tolerance มาเกี่ยวข้อง; โดยมีขั้นตอนการผลิตด้วยกระบวนการตัดโลหะพื้นฐาน(basic machining processes) โดยเฉพาะอย่างยิ่งการกลึง (turning operation) และการเจียรนัย (grinding operation); และสมมุติความสามารถของเครื่องมือกลอยู่ในสภาวะเฉลี่ยของอายุและการใช้งาน ซึ่งสามารถให้ค่าความคลาดเคลื่อนตามมาตรฐาน ISO 286

## 5.4 ข้อเสนอแนะในการพัฒนาต่อไป

การพัฒนาโปรแกรมถือได้ว่าเป็นไปตามวัตถุประสงค์ แต่ก็ยังมีหลายส่วนที่สามารถนำไปพัฒนาให้ดีขึ้นได้ เช่น การสร้าง cut อัตโนมัติที่ให้ผลลัพธ์ตามแบบกำหนดของชิ้นงานในแนวทางการสร้างแผนภูมิความคลาดเคลื่อน หากพัฒนาในส่วนนี้เพิ่มเติมจะทำให้ผู้ใช้งานลดเวลาในการออกแบบขั้นตอนการผลิตลงได้; การขยายโปรแกรมให้ครอบคลุมถึงงานกัดโลหะ หรือกระบวนการแปรรูปโลหะอื่นๆ (prismatic parts); และรวมไปถึงการประยุกต์ Optimization Techniques ซึ่งก็มีผู้ทำวิจัยมากแล้ว แต่ตราบใดที่ผลงานวิจัยยังไม่สามารถให้คำตอบที่ตรงกันได้ตราบนั้นก็ยังคงมีงานวิจัยทำนองนี้อยู่

มหาวิทยาลัยเทคโนโลยีสุรนารี

# รายการอ้างอิง


Doyle, L.E., (1950). **Tool Engineering: Analysis and Procedure**,Englewood Cliffs, N.J., Prentice-Hall.

Bjorke, O. (1978). **Computer - Aided Tolerancing**, Tapir, Norway.

Ahaluwalia, R. S. and Karolin, A. V. (1984). **CATC A computer - aided tolerance control system**.

Gadzala, J. L. (1959). **Dimensional Control in Precision Manufacturing**: As Applied in Production Machining to Effect Higher Production and Lower Unit Costs, McGraw-Hill.

Wade, O. R. (1966). **Tolerance Control in Design and Manufacturing**, Industrial Press, New York.

Irani, S. A., Mittal, R. O., and Lehtihet, E. A. (1989). **Tolerance chart optimization**. International Journal of Production Research.

Whybrew, K., Britton, G. A., Robinson, D. F., and Sermsuti-anuwat, Y. (1990). **A Graphtheoretic Approach to Tolerance Charting**, International Journal of Advanced Manufacturing Technology.

ชลาลัย วงเวียน. (2559). **การสร้างแผนภูมิความคลาดเคลื่อนด้วยตารางคำนวณ**, วิทยานิพนธ์ระดับปริญญาโท, มหาวิทยาลัยเทคโนโลยีสุรนารี.

Britton, G.A., Whybrew, K. and Sermsuti-Anuwat, Y. (1992). **A Manual Graph Theoretic Method for Teaching Tolerance Charting**,The International Journal of Mechanical Engineering Education.

Sermsuti - Anuwat, Y. (1992). **Computer-aided Process Planning and Fixture Design (CAPPFD)**, Ph.D. Thesis, Mechanical Engineering University of Canterbury, New Zealand.

Ngoi, B. K. A. (1992) **Applying linear programming to tolerance chart balancing**. International Journal of Advanced Manufacturing Technology.

Ji, P.(1993). **A tree approach for Tolerance charting**, International Journal of Production Research.

# ภาคผนวก ก

# คู่มือการใช้งานโปรแกรมแผนภูมิความคลาดเคลื่อน

มหาวิทยาลัยเทคโนโลยีสุรนารี

# คู่มือการใช้โปรแกรมแผนภูมิความคลาดเคลื่อน

## 1.1 การเริ่มต้นใช้งานโปรแกรมแผนภูมิความคลาดเคลื่อน

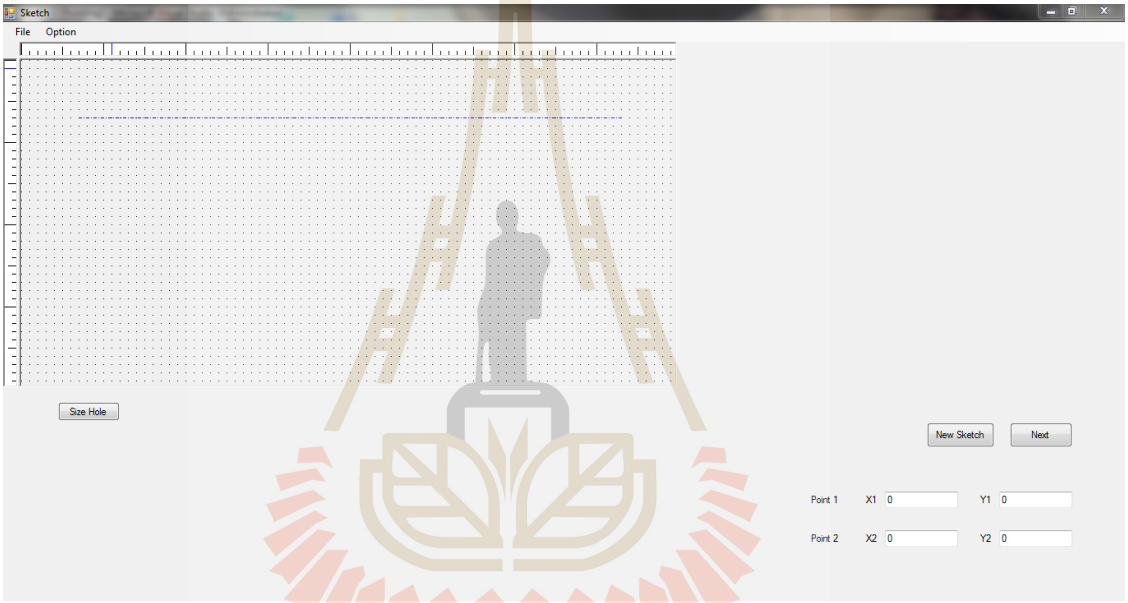
เปิดโปรแกรมแผนภูมความคลาดเคลื่อน โปรแกรมจะแสดงหน้าจอต้อนรับผู้ใช้งานดังในรูป ที่ 1.1

รูปที่ 1.1 แสดงหน้าจอต้อนรับของโปรแกรมแผนภูมิความคลาดเคลื่อน

หลังจากผู้ใช้งานเปิดหน้าจอต้อนรับแล้ว ให้ดำเนินการกดปุ่ม Sketch เพื่อสร้างรูปชิ้นงานตามแบบกำหนด (blueprint)

## 1.2 หน้าจอสเกตซ์รูปชิ้นงาน 2 มิติ

หลังจากดกปุ่ม Sketch ในรูป ที่ 1.1 โปรแกรมจะแสดงหน้าจอ Sketch ดังในรูปที่ 1.2 เพื่อให้ผู้ใช้งาน เขียนเส้นลงไปในกรอบสีเหลี่ยมที่กำหดให้ โปรแกรมจะทำการเขียนเส้น center line ให้ผู้ใช้งาน 1 เส้น ลักษณะการสเกตซ์รูปชิ้นงานควรเป็นแบบ half sketch เพื่อลดเวลาในการสเกตซ์ภพชิ้นงาน วิธีการลากเส้นคือกดปุ่มเมาส์ด้านซ้ายค้างในตำแหน่งจุดเริ่มต้นของเส้น จากนั้นทำการเคลื่อนเมาส์ไปหาตำแหน่งจุดสิ้นสุดของเส้น ทำการปล่อยปุ่มเมาส์ด้านซ้ายโปรแกรมจะทำการสร้างเส้นตรงให้ 1 เส้น ในรูปที่ 1.7 แสดงตัวอย่างการสเกตซ์ชิ้นงาน 2 มิติ แบบ half sketch

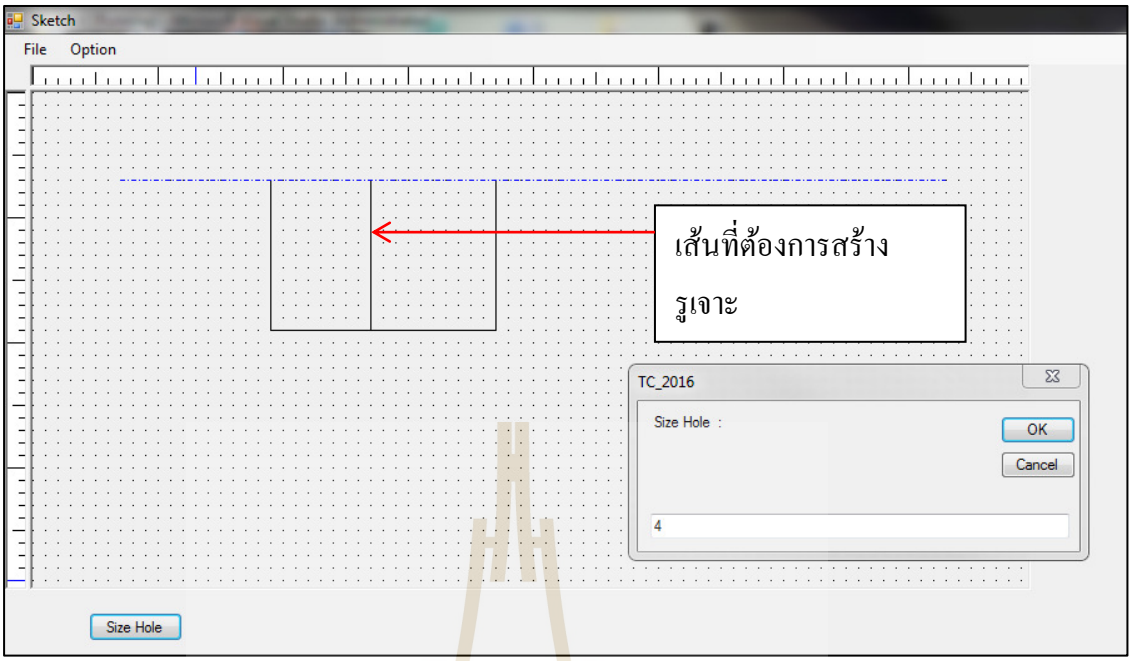



รูปที่ 1.2 แสดงหน้าสำหรับสเกตซ์รูปชิ้นงาน 2 มิติ

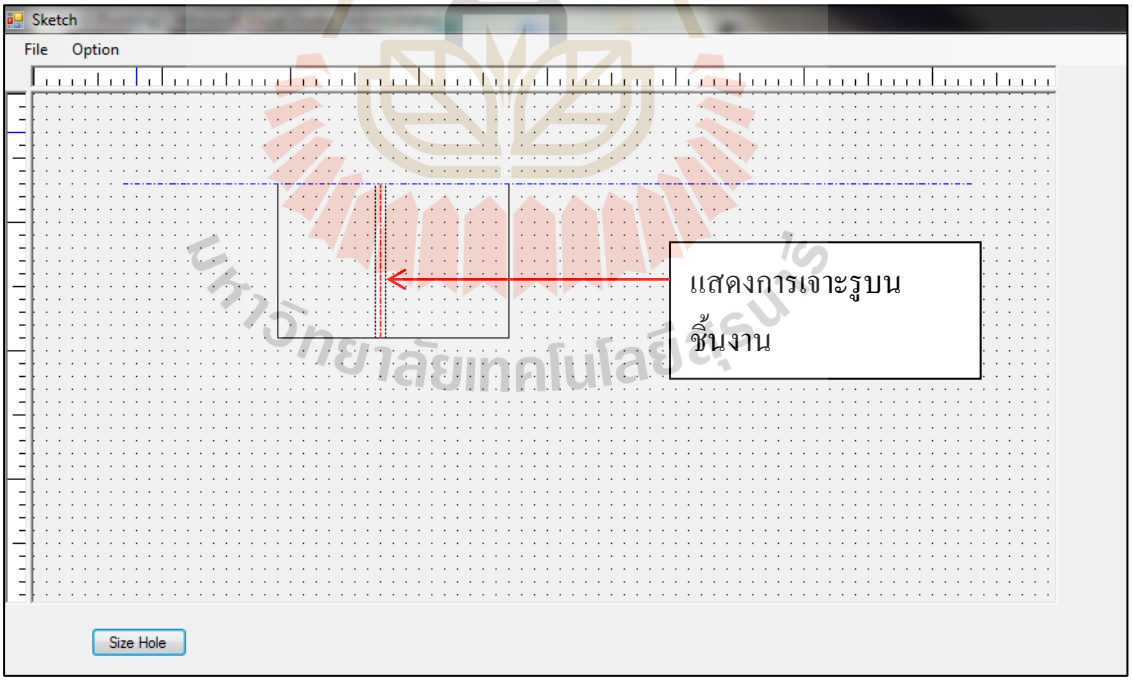
ในส่วนข้องหน้าจอ Sketch นี้ จะมีปุ่มสำหรับประมวลผล 3 ปุ่มด้วยกันดังนี้:-

(1) ปุ่ม Size Hole หากผู้ใช้งานลากเส้นตรงแล้วทำการกดปุ่มนี้ โปรแกรมจะทำการสร้างรูเจาะบนรูปสเกตซ์ โดยให้ผู้ใช้งานกำหนดของรูเจาะตามแบบกำหนด ในรูปที่ 1.3 เส้นการสร้างเส้นรูปเจาะโดยผู้ใช้งานลากสร้างตรงที่ต้องการสร้างรูเจาะ จากนั้นกดปุ่ม Size Hole โปรแกรมจะแสดงกล่องรับขนาดดังรูป ส่วนในรูปที่ 1.4 โปรแกรมแสดงการเจาะรูบนชิ้นงาน

มหาวิทยาลัยเทคโนโลยีสุรนารี




รูปที่ 1.3 แสดงการสร้างรูเจาะบนชิ้นงาน




รูปที่ 1.4 แสดงการรูเจาะบนชิ้นงาน

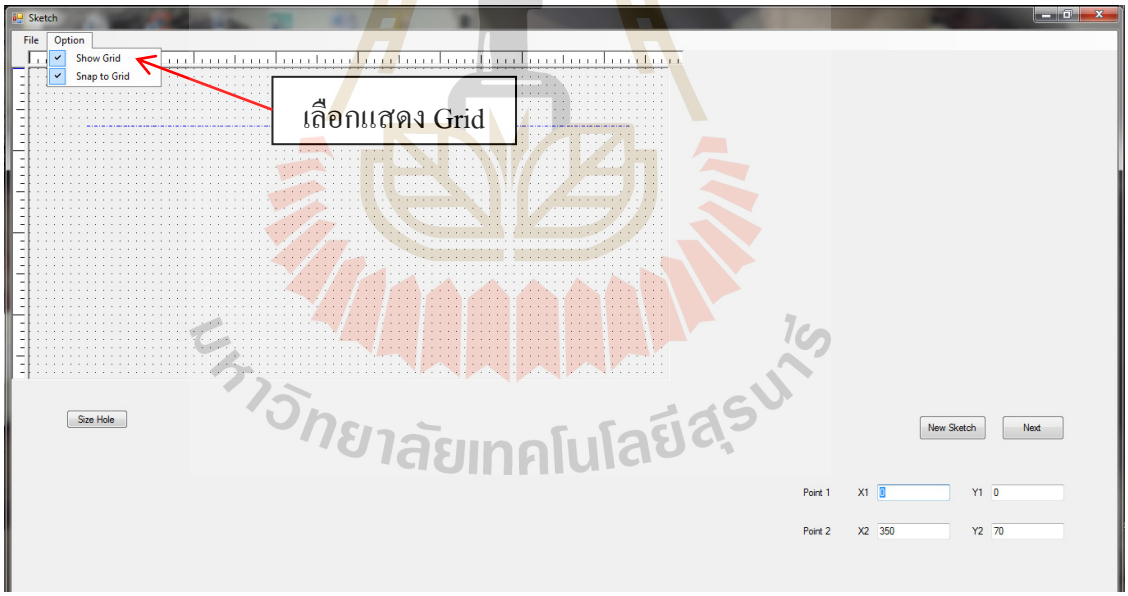
(2) ปุ่ม New Sketch หากผู้ใช้งานต้องการเขียนรูปให้ สามารถทำได้โดยกดปุ่ม New Sketch โปรแกรมจะทำการลบพื้นที่สเกตซ์รูปใหม่ จากนั้นให้ผู้ใช้งานกลับไปยังหน้าจอต้อนรับอีกครั้ง

(3) ปุ่ม Next หากผู้ใช้งานกดปุ่มนี้โปรแกรมดำเนินการต่อโดยประมวลผลจำนวนรอยตัดและแสดงหน้าจอ Number Of Cut ออกมา แสดงดังในรูปที่ 1.8

ในเมนู Option ของหน้าจอ Sketch นี้มีฟังก์ชั่นช่วยในการเขียนเส้น 2 ฟังก์ชั่นดังนี้:-

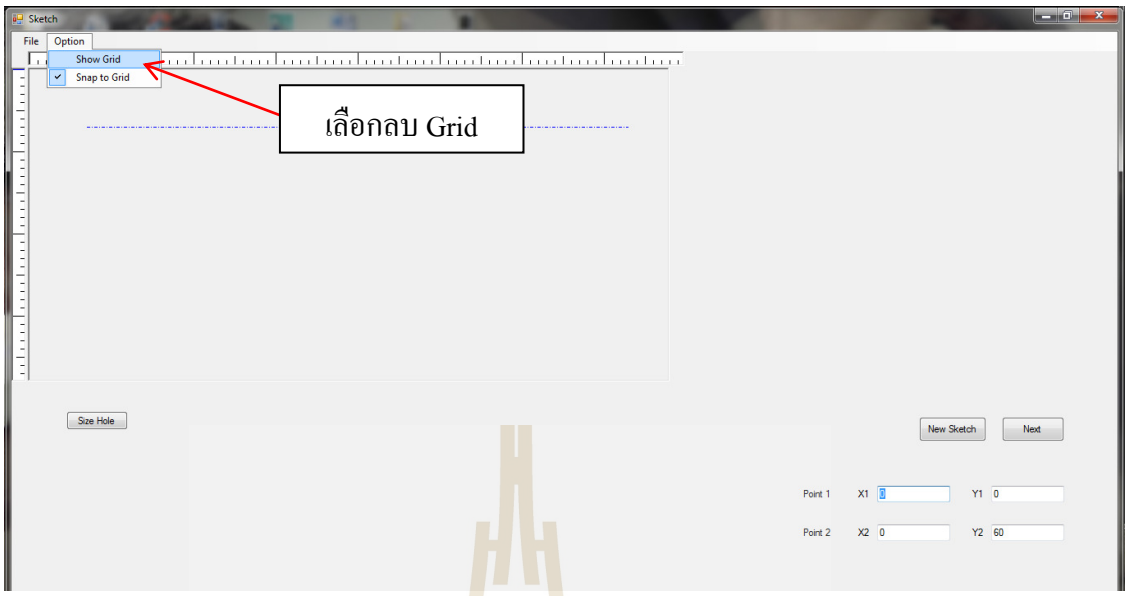
(1) ฟังก์ชั่น Show Grid หากผู้ใช้งานทำเครื่องหมายถูกหน้าข้อความ “Show Grid” โปรแกรมจะแสดง Grid เพื่อช่วยในการเขียนภาพชิ้นงาน ดังแสดงในรูปที่ 1.5 และหากผู้ใช้งานยกเลิกเครื่องหมายถูกหน้าข้อความ “Show Grid” โปรแกรมจะลบ Grid ออก ดังแสดงในรูปที่ 1.6

(2) ฟังก์ชั่น Snap to Grid หากผู้ใช้งานทำเครื่องหมายถูกหน้าข้อความ “Snap to Grid” การเคลื่อนที่ของเมาส์จะเคลื่อนที่แล้วหยุดตรงตำแหน่ง Grid เสมอ และหากผู้ใช้งานยกเลิกเครื่องหมายถูกหน้าข้อความ “Snap to Grid” การเคลื่อนที่ของเมาส์จะเป็นอิสระ

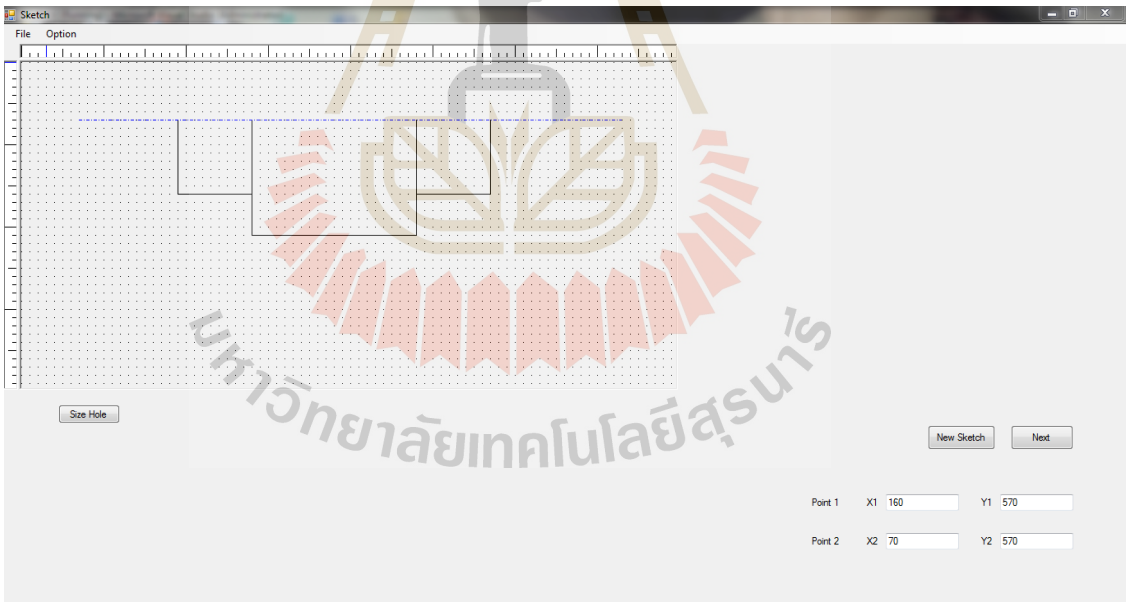



รูปที่ 1.5 แสดง Grid บนพื้นที่สเกตซ์ชิ้นงาน

มหาวิทยาลัยเทคโนโลยีสุรนารี




รูปที่ 1.6 แสดงลบ Grid บนพื้นที่สเกตซ์ชิ้นงาน

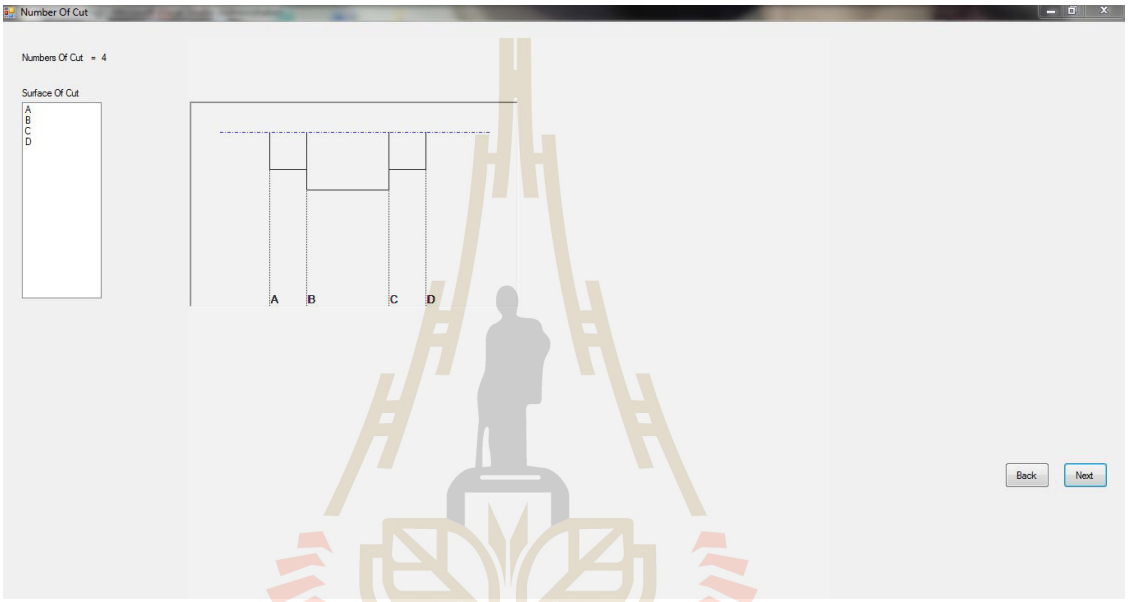



รูปที่ 1.7 แสดงตัวอย่างสเกตซ์ชิ้นงาน 2 มิติ

มหาวิทยาลัยเทคโนโลยีสุรนารี

## 1.3 หน้าจอแสดงผลจำนวนรอยตัด

ในหัวข้อนี้โปรแกรม จะแสดงหน้าจอ Number Of Cut โดยโปรแกรมแสดงข้อมูลดังนี้ จำนวนรอยตัว, ชื่อของรอยตัดโดยเริ่มต้นจาก A, และสุดท้ายแสดงรูปชิ้นงานพร้อมชื่อระบุตำแหน่งของรอยตัด แสดงดังในรูปที่ 1.8 หน้าจอ Number Of Cut นี้ จะมีปุ่มสำหรับประมวลผล 2 ปุ่ม ด้วยกันดังนี้:-




รูปที่ 1.8 แสดงแสดงหน้าจอ Number Of Cut

(1) ปุ่ม Back หากผู้ใช้งานกดปุ่มนี้โปรแกรมจะย้อนกลับไปแสดงหน้าจอ Sketch

(2) ปุ่ม Next หากผู้ใช้งานกดปุ่มนี้โปรแกรมดำเนินการต่อโดยแสดงหน้าจอ Blueprint ออกมา ดังในรูปที่ 1.9

## 1.4 หน้าจอ Blueprint

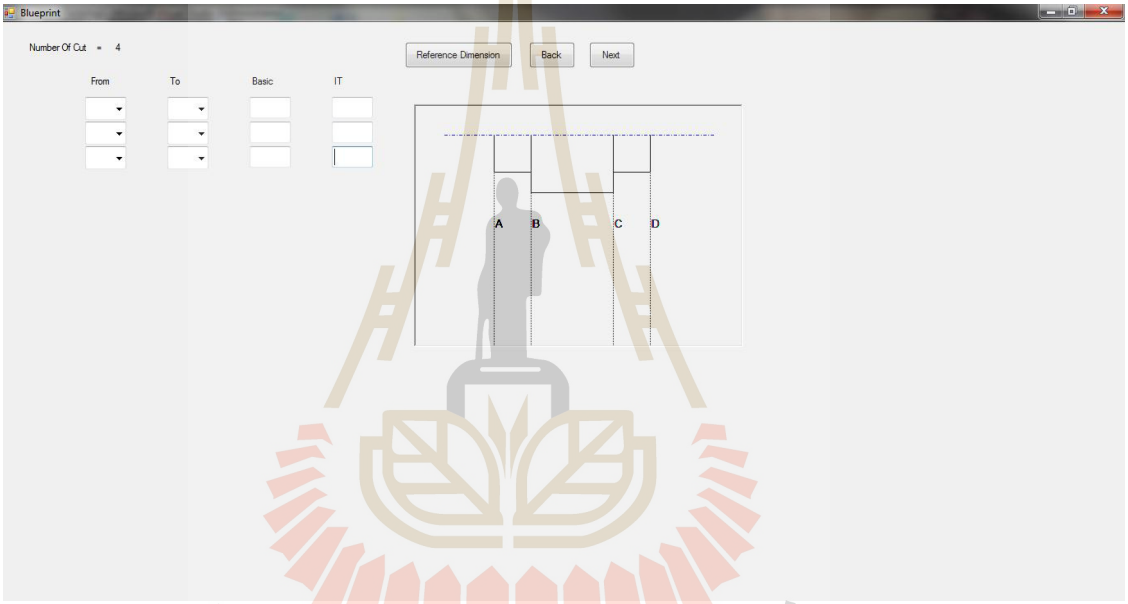
ในหัวข้อนี้โปรแกรม จะแสดงหน้าจอ Blueprint ดังรูปที่ 1.9 โดยโปรแกรมแสดงกล่องรับข้อมูลจากแบบกำหนด โดยผู้ใช้งานจะต้องใส่ข้อมูลดังนี้ 1) ชื่อตำแหน่งของจุดเริ่มต้น และจุดสิ้นสุดของขนาดตามแบบ 2) ผู้ใช้งานใส่ขนาดตามแบบกำหนด และ 3) ผู้ใช้งานใส่ค่าความ

คลาดเคลื่อนตามแบบกำหนด หากผู้ใช้งานใส่ข้อมูลครบถ้วนจะแสดงดังรูปที่ 1.10 ในกรอบที่ 1 หน้าจอ Blueprint นี้จะมีปุ่มใช้งานทั้งหมด 3 ปุ่มดังนี้:-

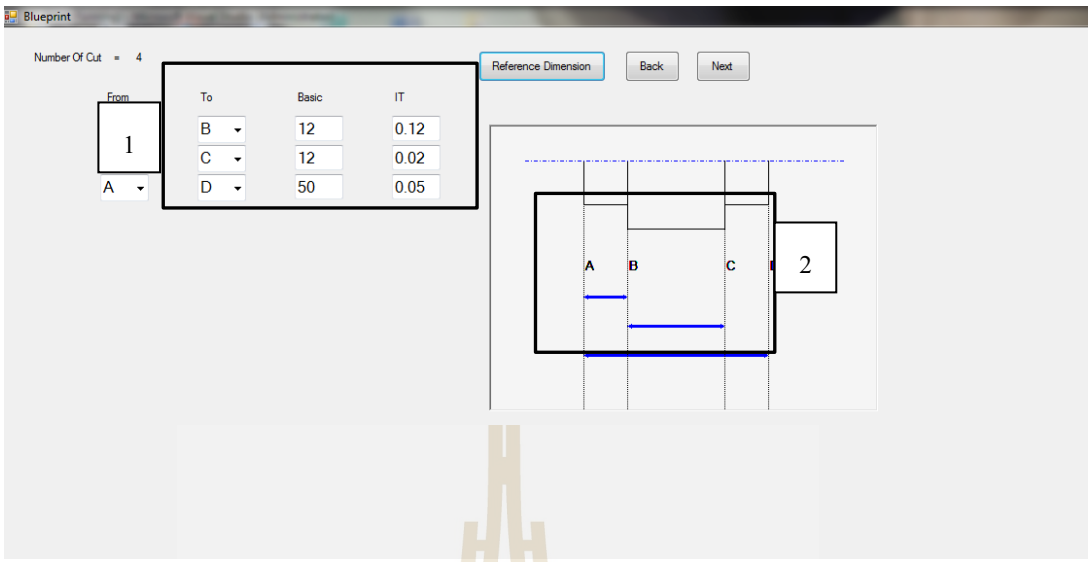
(1) ปุ่ม Reference Dimension หากผู้ใช้งานกดปุ่มนี้โปรแกรมจะแสดงลูกศรแสดงจุดเริ่มต้นและจุดสิ้นสุดของขนาด แสดงดังรูปที่ 1.10 ในกรอบที่ 2

(2) ปุ่ม Back หากผู้ใช้งานกดปุ่มนี้โปรแกรมจะย้อนกลับไปแสดงหน้าจอ Number Of Cut

(3) ปุ่ม Next หากผู้ใช้งานใส่ข้อมูลแล้วเสร็จต้องการดำเนินการต่อให้กดปุ่มนี้ โปรแกรมจะแสดงหน้าจอให้ผู้ใช้งานใส่จำนวนขั้นตอนในการตัดโลหะ (หน้าจอ Number of operation)




รูปที่ 1.9 แสดงหน้าจอรับข้อมูล Blueprint



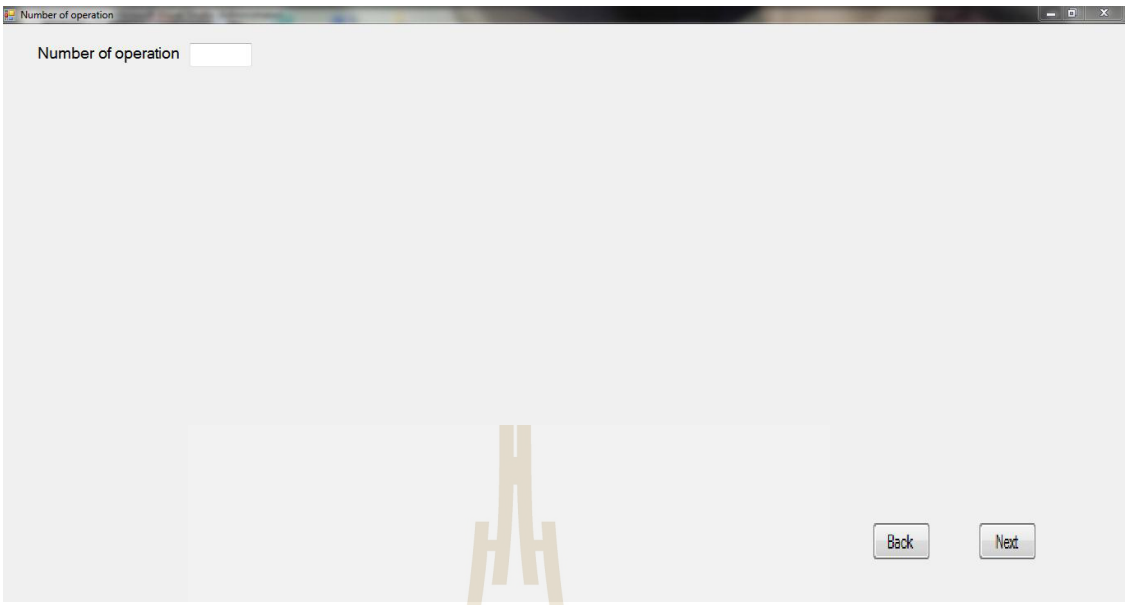

รูปที่ 1.10 แสดงตัวอย่างการรับข้อมูล Blueprint

## 1.5 หน้าจอ Number of operation

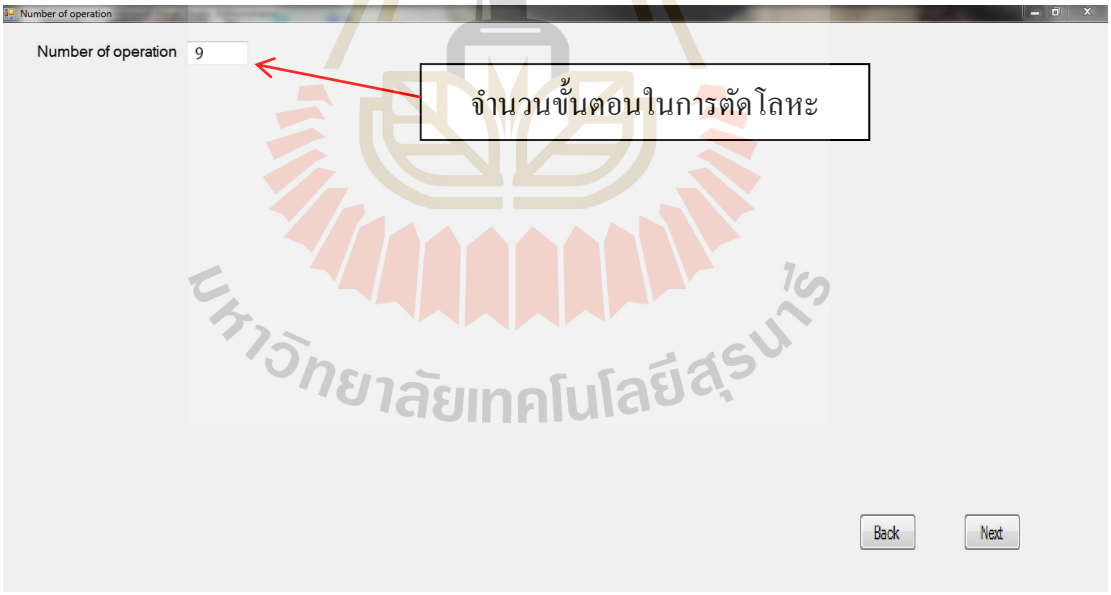
ในหัวข้อนี้โปรแกรม จะแสดงหน้าจอ Number of operation ดังรูปที่ 1.11 โดยโปรแกรมแสดงกล่องรับข้อมูลจำนวนขั้นตอนในการตัดโลหะจากผู้ใช้งาน รูปที่ 1.12 แสดงตัวอย่างการใส่ข้อมูลจำนวนขั้นตอนในการตัดโลหะจำนวน 9 ขั้นตอน ในหน้าจอนี้ จะมีปุ่มใช้งานทั้งหมด 2 ปุ่ม ดังนี้:-

(1) ปุ่ม Back หากผู้ใช้งานกดปุ่มนี้โปรแกรมจะย้อนกลับไปแสดงหน้าจอ Blueprint

(2) ปุ่ม Next หากผู้ใช้งานใส่จำนวนขั้นตอนในการตัดแล้วเสร็จ องการดำเนินการต่อให้กดปุ่ม Next โปรแกรมจะแสดงหน้าจอ Tolerance Chart ซึ่งเป็นหน้าจอใส่ข้อมูลขั้นตอนในการตัดและแสดงผลลัพธ์ของแผนภูมิ




รูปที่ 1.11 แสดงหน้าจอเพื่อรับข้อมูลจำนวนขั้นตอนในการตัดโลหะ




รูปที่ 1.12 แสดงตัวอย่างรับการข้อมูลจำนวนขั้นตอนในการตัดโลหะ

มหาวิทยาลัยเทคโนโลยีสุรนารี

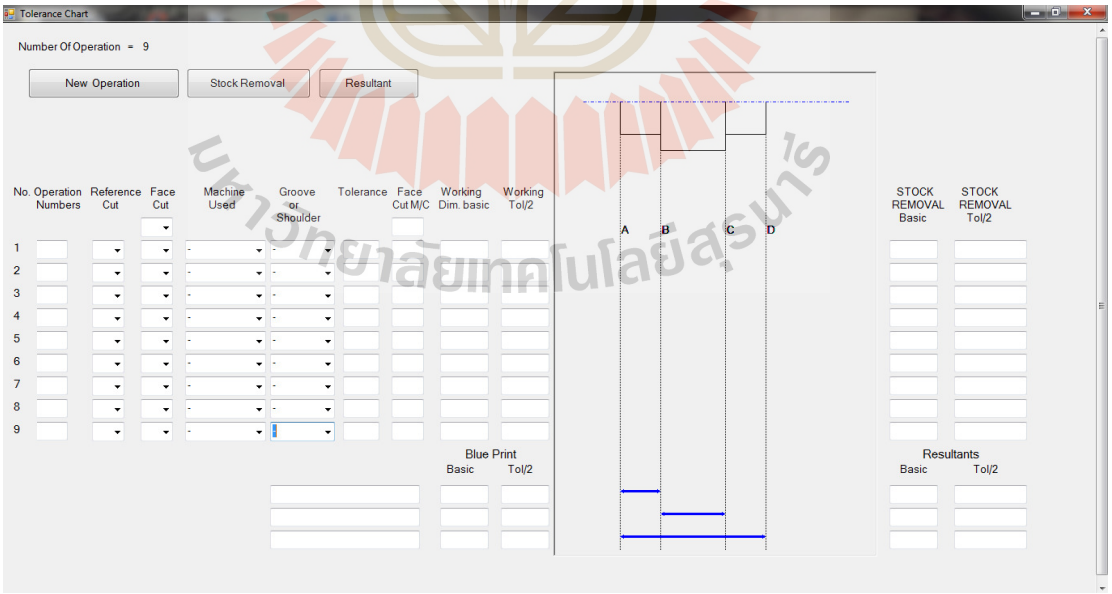
## 1.6 หน้าจอ Tolerance Chart

ในหัวข้อนี้โปรแกรม จะแสดงหน้าจอ Tolerance Chart ดังรูปที่ 1.13 โดยโปรแกรมแสดงกล่องรับข้อมูลให้ผู้ใช้งานใส่ข้อมูลดังนี้ (1) ชื่อขั้นตอนในการตัดโละ (2) ผิวอ้างอิงและผิวตัดในแต่ละขั้นตอนการตัดโลหะ (3) ลักษณะของรอยตัดว่าเป็นบ่าหรือเป็นร่อง และ (4) ค่าความคลาดเคลื่อนในแต่ละขั้นตอนการตัดในข้อนี้หากผู้ใช้งานไม่ใส่ข้อมูลโปรแกรมจะทำการประมวลผลจากสูตรที่ คำนวณ IT รูปที่ 1.14 แสดงตัวอย่างการใส่ข้อมูลขั้นตอนในการตัดโลหะ ในหน้าจอนี้จะมีปุ่มใช้งานทั้งหมด 3 ปุ่มดังนี้:-

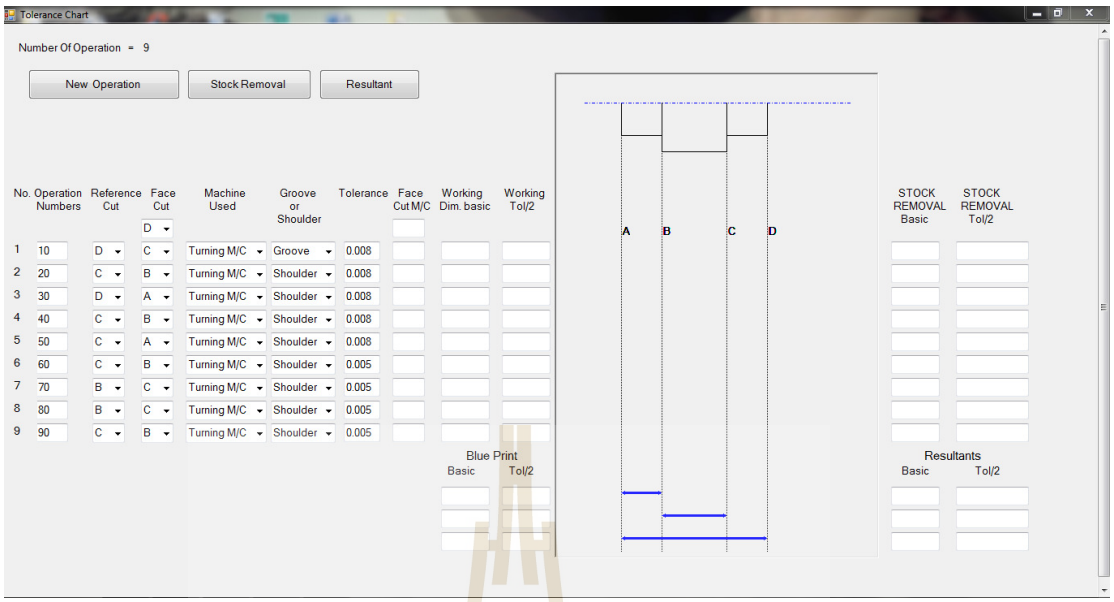
(1) ปุ่ม New Operation หากผู้ใช้งานกดปุ่มนี้โปรแกรมจะย้อนกลับไปแสดงหน้าจอ Number of operation เพื่อให้ผู้ใช้งานกำหนดจำนวนขั้นตอนในการตัดโลหะใหม่ โดยเพิ่มหรือลดจำนวนขั้นตอนในการตัดโลหะพิจารณาโดยผู้ใช้งาน

(2) ปุ่ม Stock Removal หากผู้ใช้งานกดปุ่มนี้โปรแกรมจะแสดงความคลาดเคลื่อนสะสม, ขนาดมูลฐานของความคลาดเคลื่อนสะสม, ขนาดมูลฐานของขั้นตอนการตัด, และความคลาดเคลื่อนของขั้นตอนการตัดในแต่ละขั้นตอน แสดงดังรูปที่ 1.15

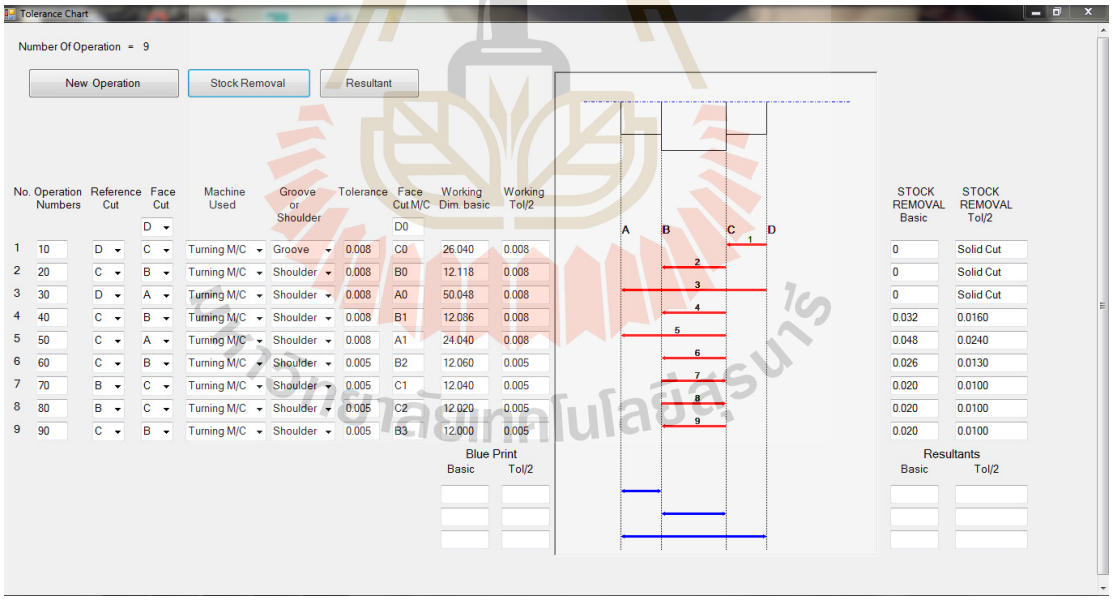
(3) ปุ่ม Resultant หากผู้ใช้งานกดปุ่มนี้ โปรแกรมจะแสดงผลลัพธ์ของแผนภูมิความเคลื่อนดังในรูปที่ 1.16



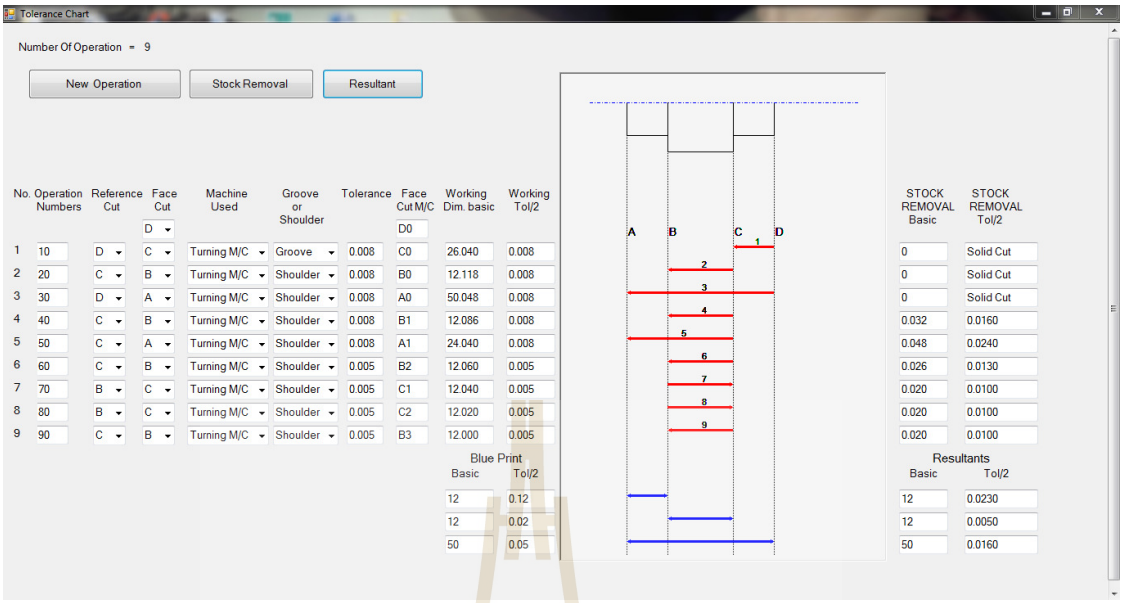

รูปที่ 1.13 แสดงหน้าจอ Tolerance Chart




รูปที่ 1.14 แสดงตัวอย่างการใส่ข้อมูลขั้นตอนในการตัดโลหะ




รูปที่ 1.15 แสดงตัวอย่างหน้าจอหลังกดปุ่ม Stock Removal

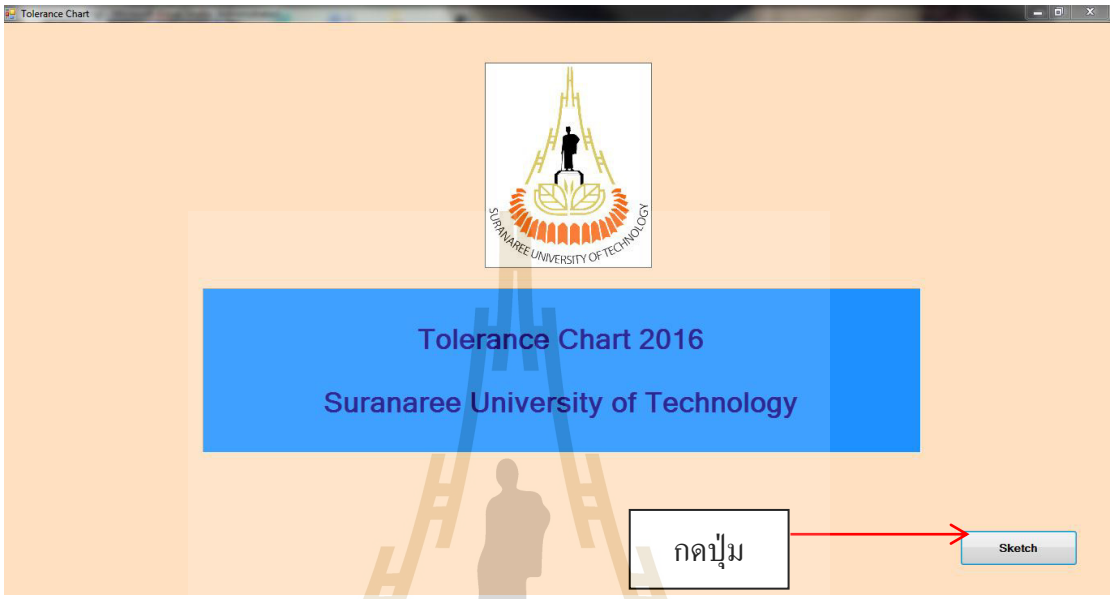



รูปที่ 1.16 แสดงตัวอย่างหน้าจอหลังกดปุ่ม Resultant

มหาวิทยาลัยเทคโนโลยีสุรนารี

## 2.1 ตัวอย่างแสดงการใช้งานโปรแกรมตามลำดับ

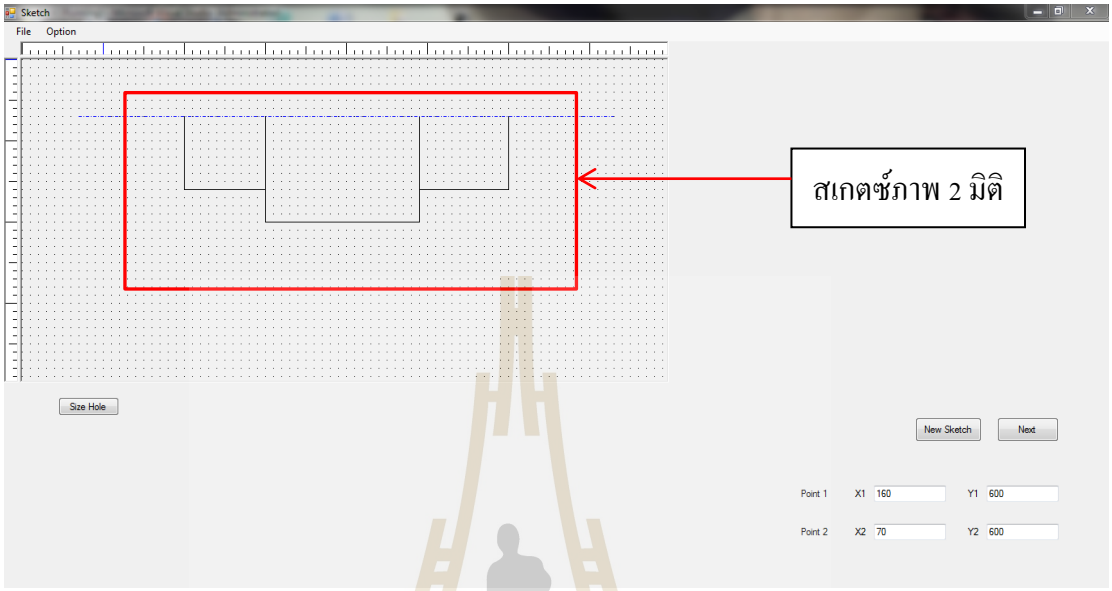
ขั้นตอนที่ 1 หลังจากเปิดโปรแกรมผู้ใช้งานกดปุ่ม Sketch แสดงดังรูปที่ 2.1




รูปที่ 2.1

มหาวิทยาลัยเทคโนโลยีสุรนารี

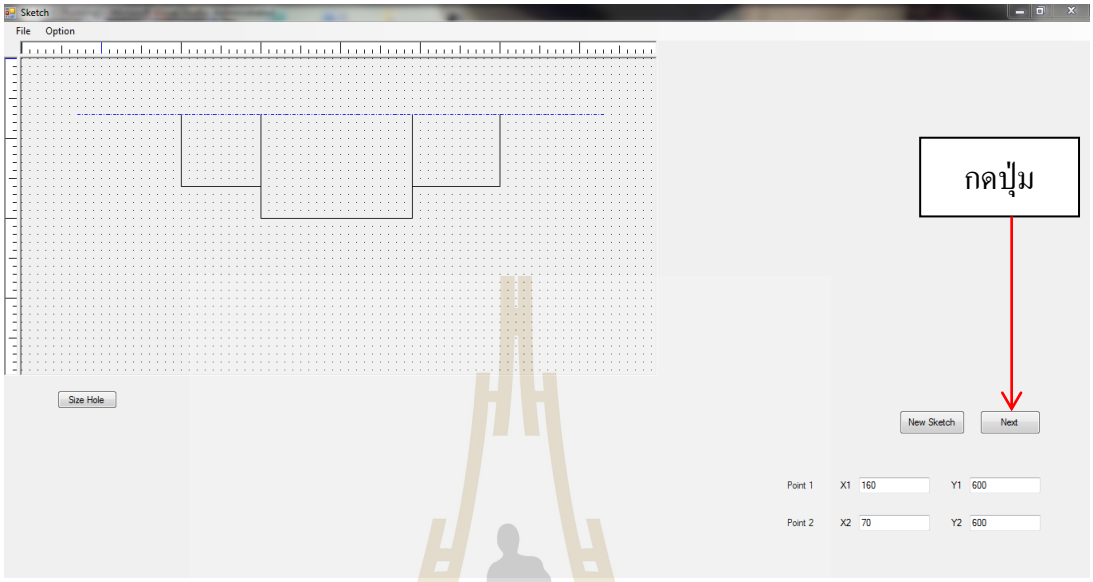
ขั้นตอนที่ 2 โปรแกรมแสดงหน้าจอ Sketch ผู้ใช้งานสเกตซ์ภาพชิ้นงาน 2 มิติ ดังรูป 2.2




รูปที่ 2.2

มหาวิทยาลัยเทคโนโลยีสุรนารี

ขั้นตอนที่ 3 ผู้ใช้งานกดปุ่ม Next แสดงดังรูปที่ 2.3




รูปที่ 2.3

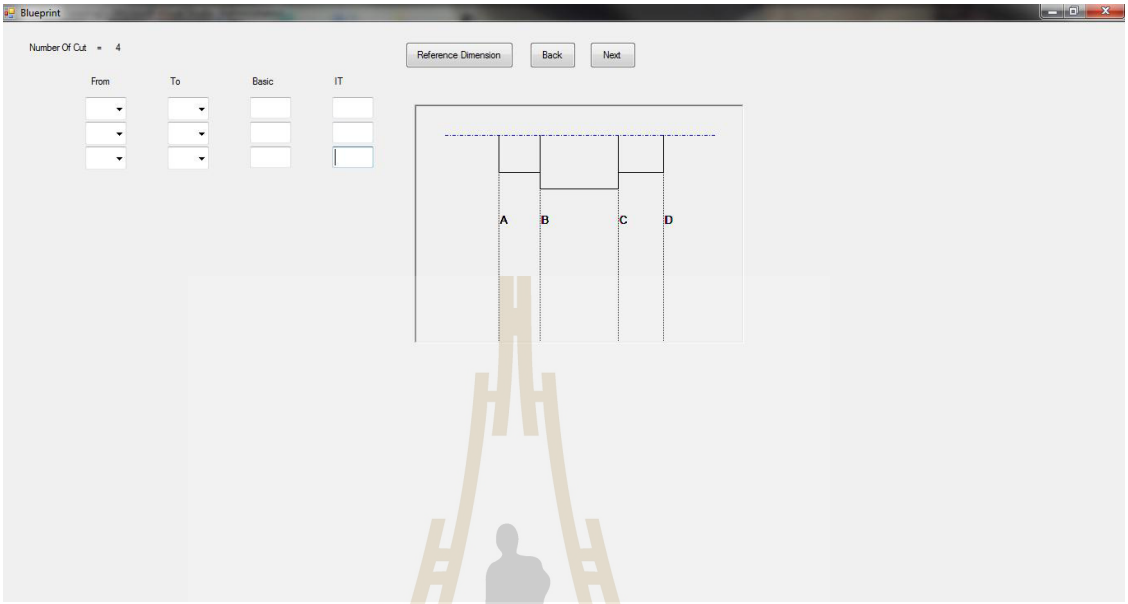
มหาวิทยาลัยเทคโนโลยีสุรนารี

ขั้นตอนที่ 4 โปรแกรมแสดงหน้าจอ Number of cut ผู้ใช้งานกดปุ่ม Next แสดงดังรูปที่ 2.4

รูปที่ 2.4

มหาวิทยาลัยเทคโนโลยีสุรนารี

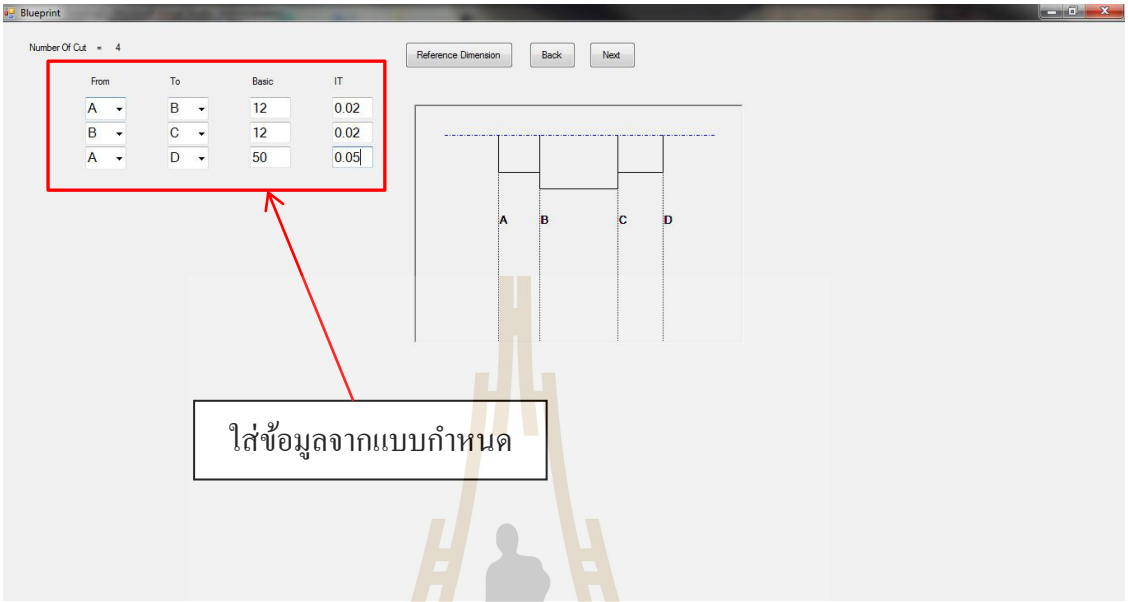
ขั้นตอนที่ 5 โปรแกรมแสดงหน้าจอ Blueprint แสดงดังรูปที่ 2.5




รูปที่ 2.5

มหาวิทยาลัยเทคโนโลยีสุรนารี

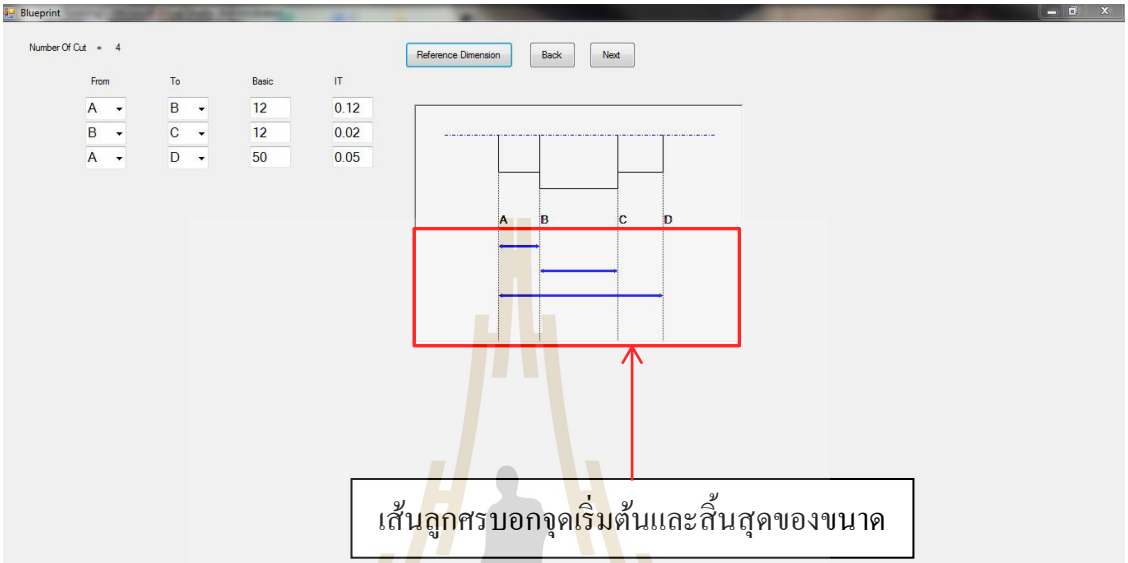
ขั้นตอนที่ 6 ผู้ใช้งานใส่ข้อมูลจากแบบกำหนด ในหน้าจอ Blueprint แสดงดังรูปที่ 2.6




รูปที่ 2.6

มหาวิทยาลัยเทคโนโลยีสุรนารี

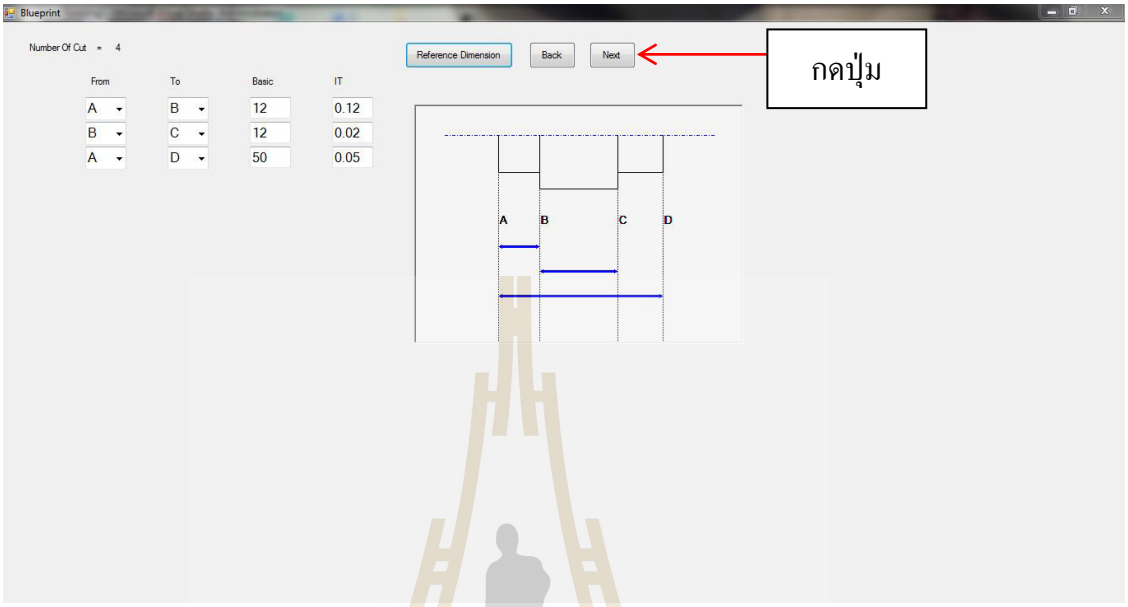
ขั้นตอนที่ 7 ผู้ใช้งานกดปุ่ม Reference Dimension โปรแกรมจะแสดงเส้นลูกศรบอกตำแหน่งจุดเริ่มต้นและจุดสิ้นสุดของขนาด แสดงดังรูปที่ 2.7




รูปที่ 2.7

มหาวิทยาลัยเทคโนโลยีสุรนารี

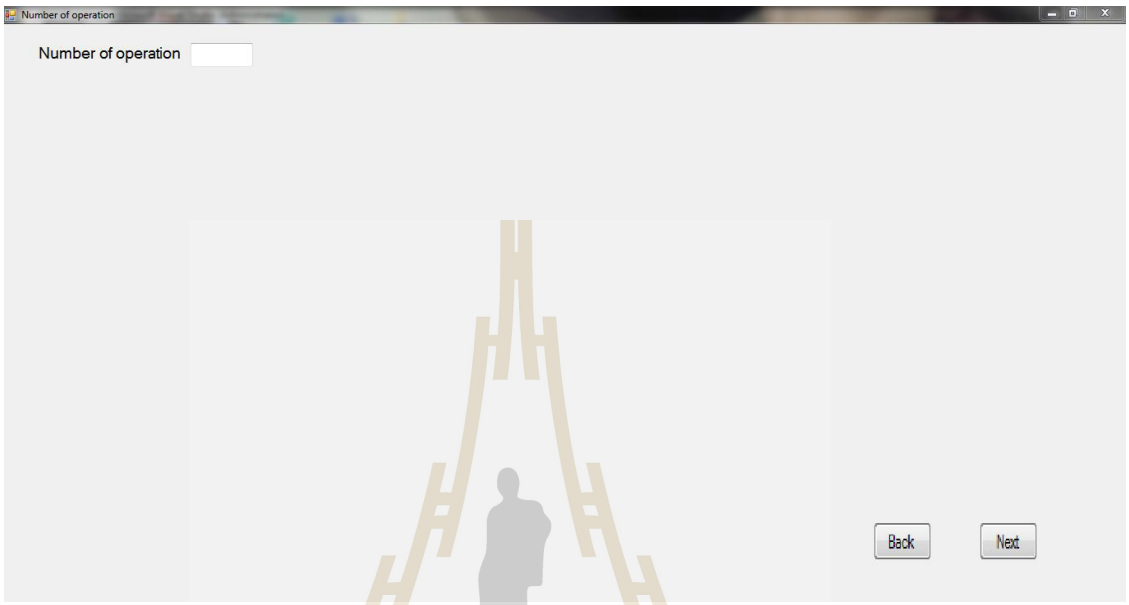
ขั้นตอนที่ 8 หน้าจอ Blueprint ผู้ใช้งานกดปุ่ม Next แสดงดังรูปที่ 2.8




รูปที่ 2.8

มหาวิทยาลัยเทคโนโลยีสุรนารี

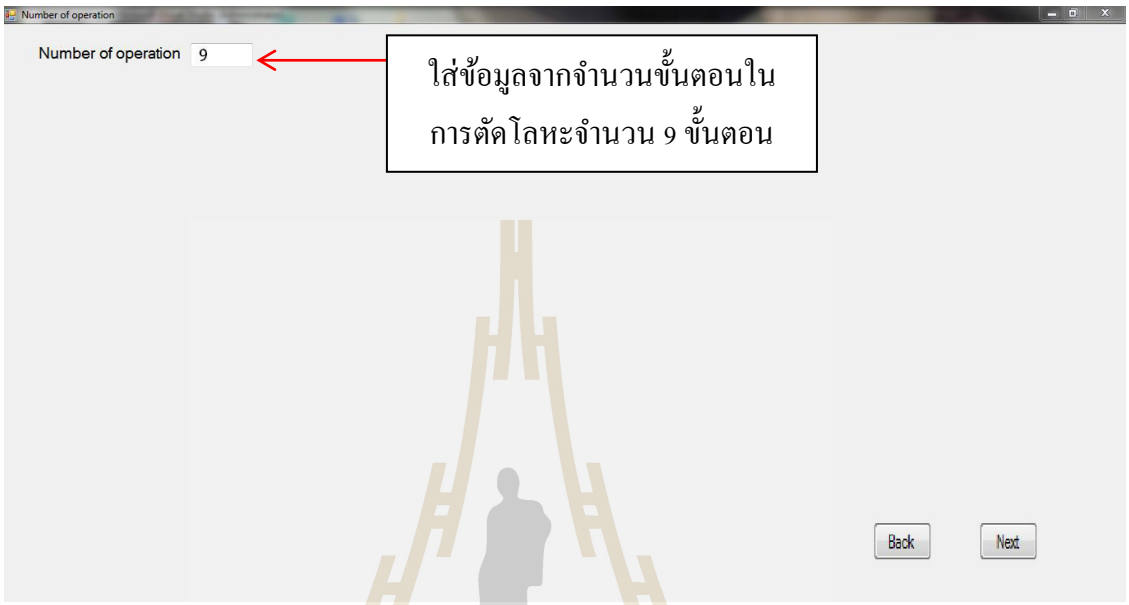
ขั้นตอนที่ 9 โปรแกรมแสดงหน้าจอ Number of operation แสดงดังรูปที่ 2.9




รูปที่ 2.9

มหาวิทยาลัยเทคโนโลยีสุรนารี

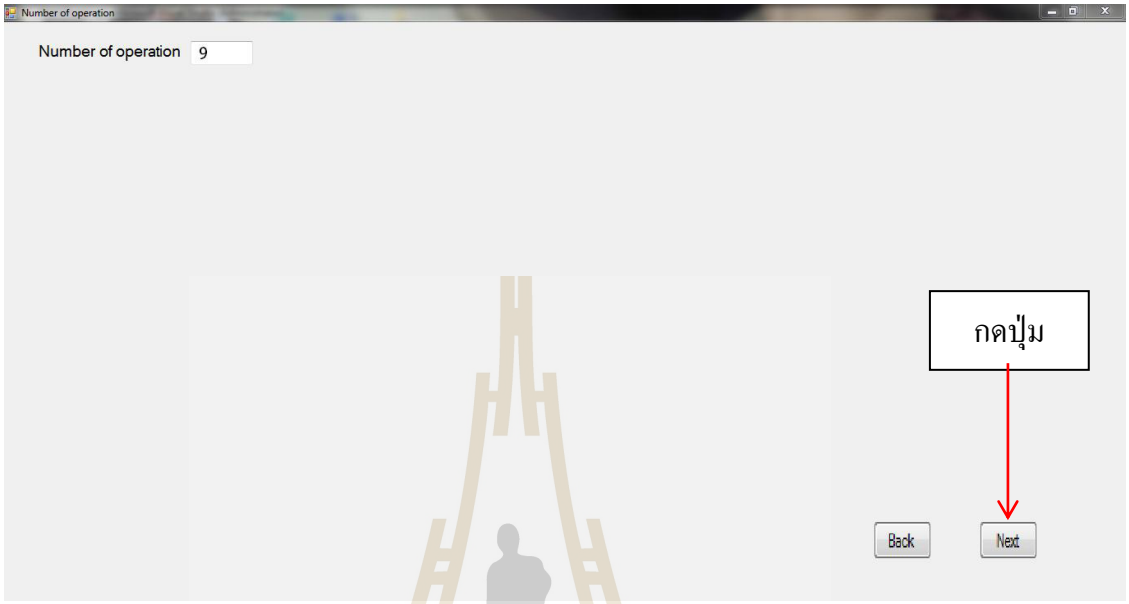
ขั้นตอนที่ 10 ผู้ใช้งานใส่ข้อมูลจำนวนขั้นตอนในการตัดโลหะ ในหน้าจอ Number of operation แสดงดังรูปที่ 2.10




รูปที่ 2.10

มหาวิทยาลัยเทคโนโลยีสุรนารี

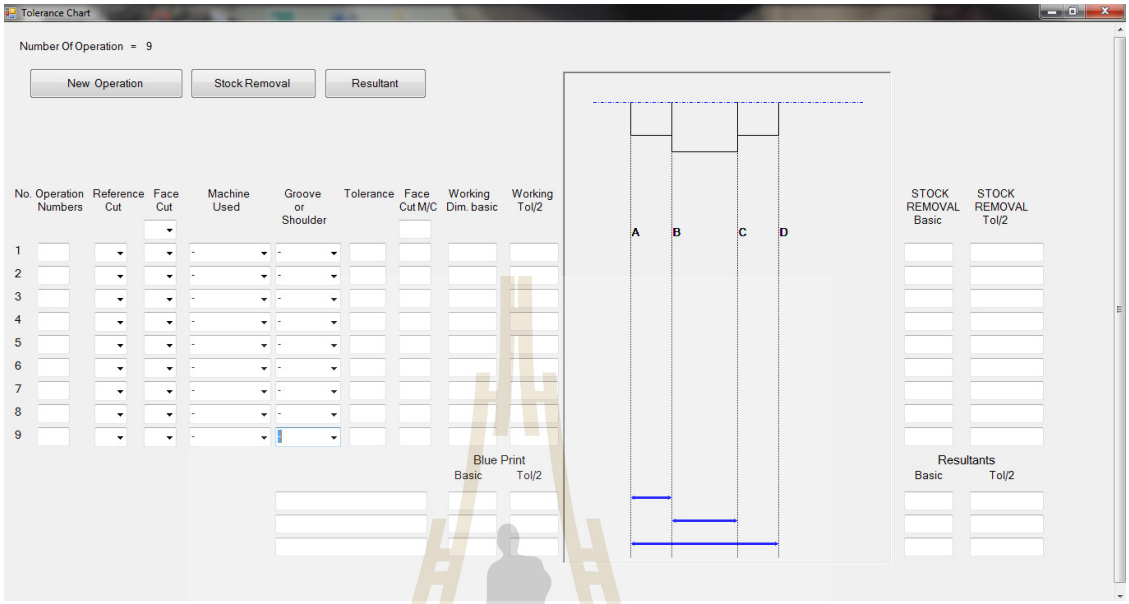
ขั้นตอนที่ 11 หน้าจอ Number of operation ผู้ใช้งานกดปุ่ม Next แสดงดังรูปที่ 2.11




รูปที่ 2.11

มหาวิทยาลัยเทคโนโลยีสุรนารี

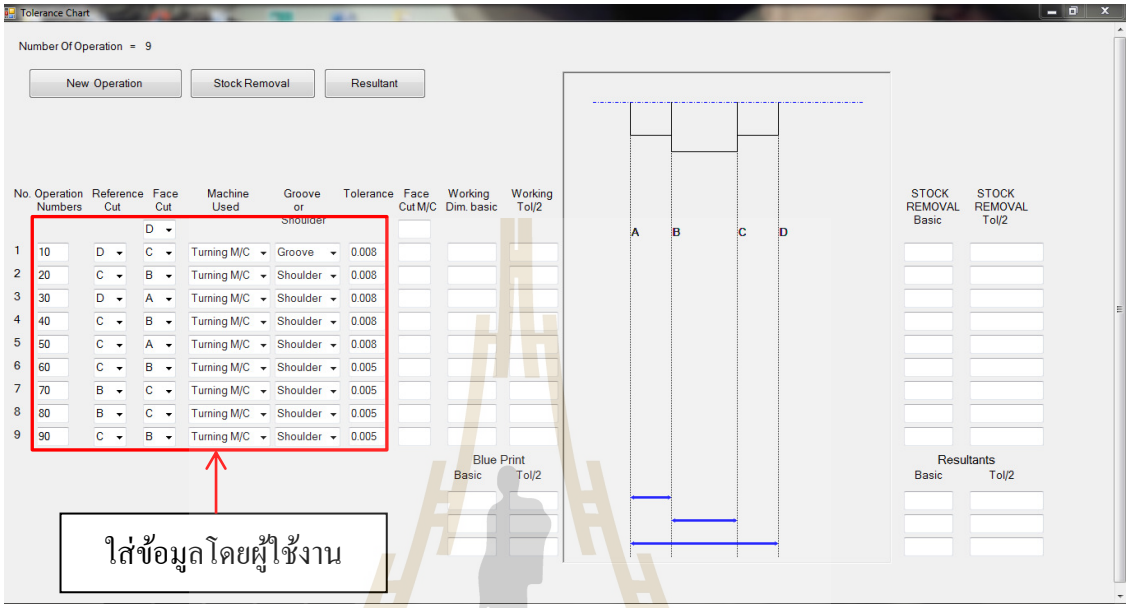
ขั้นตอนที่ 12 โปรแกรมแสดงหน้าจอ Tolerance Chart แสดงดังรูปที่ 2.12




รูปที่ 2.12

มหาวิทยาลัยเทคโนโลยีสุรนารี

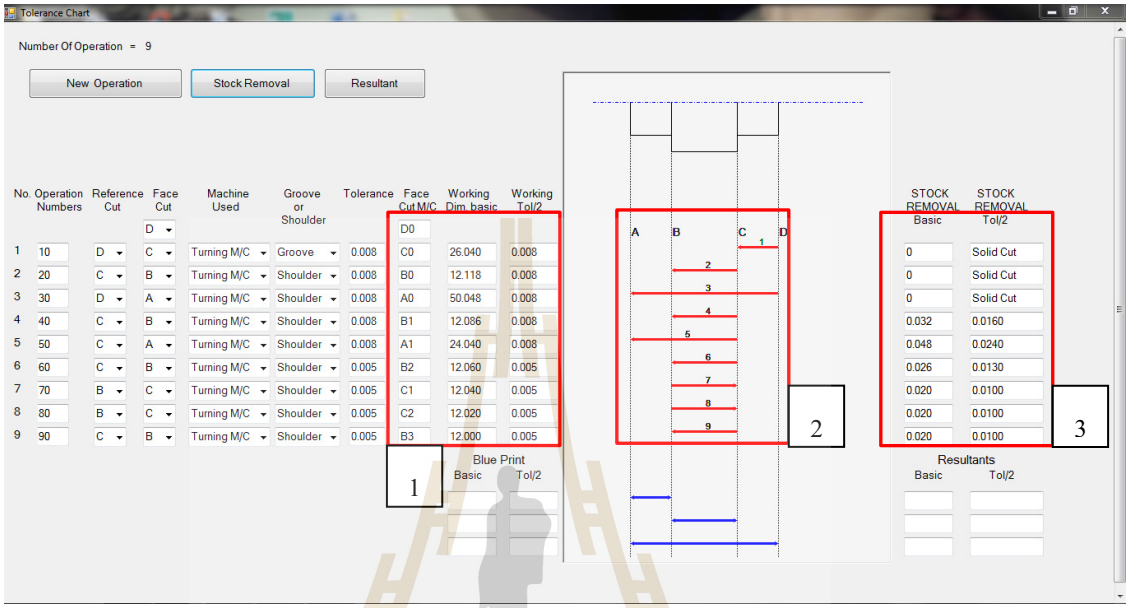
ขั้นตอนที่ 13 ผู้ใช้งานใส่ข้อมูลรายละเอียดของขั้นตอนในการตัดโลหะในแต่ละขั้นตอน ในหน้าจอ Tolerance Chart แสดงดังรูปที่ 2.13




รูปที่ 2.13

มหาวิทยาลัยเทคโนโลยีสุรนารี

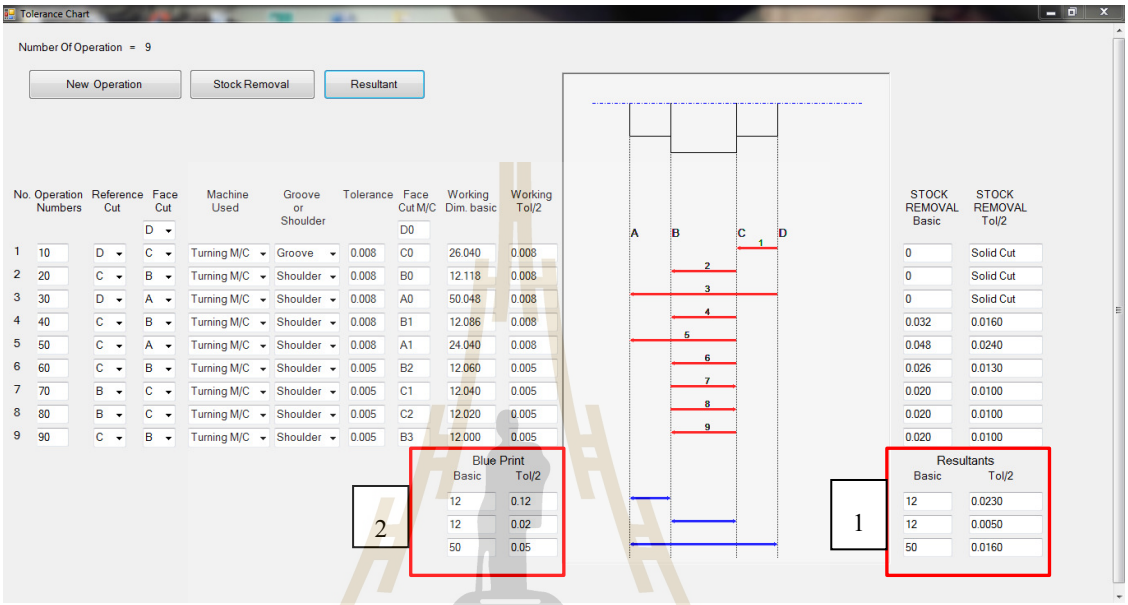
ขั้นตอนที่ 14 หน้าจอ Tolerance Chart ผู้ใช้งานกดปุ่ม Stock Removal โปรแกรมจะแสดงผลในกรอบ 1, 2 และ 3 ของรูปที่ 2.14




รูปที่ 2.14

มหาวิทยาลัยเทคโนโลยีสุรนารี

ขั้นตอนที่ 15 หน้าจอ Tolerance Chart ผู้ใช้งานกดปุ่ม Stock Removal โปรแกรมจะแสดงผลลัพธ์ของแผนภูมิความคลาดเคลื่อนในกรอบที่ 1 ของรูปที่ 2.15 ส่วนกรอบที่ 2 แสดงข้อมูลจากแบบกำหนด




รูปที่ 2.15

**หมายเหตุ** การแก้ไขปรับปรุงข้อมูลการตัดโลหะนั้นสามารถแก้ไขได้ทันที จากนั้นเริ่มทำตามขั้นตอนที่ 14 อีกครั้ง ส่วนในเรื่องการเพิ่มหรือลดขั้นตอนในการตัดโลหะนั้นสามารถ กดปุ่ม New operation จากนั้นเริ่มทำในขั้นตอนที่ 10 โดยผู้ใช้งานไม่ต้องใส่ข้อมูลพื้นฐาน(อ้างอิงข้อมูลเดิม)

# ภาคผนวก ข

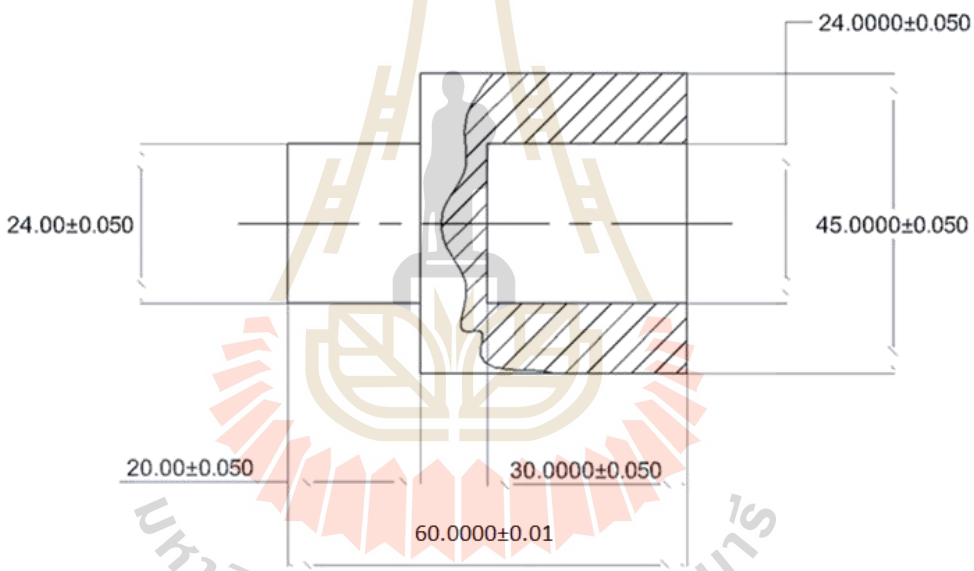
## ความคลาดเคลื่อน (Tolerance of machined)

มหาวิทยาลัยเทคโนโลยีสุรนารี

# ความคลาดเคลื่อน (Tolerance of machined)

## 1. ความคลาดเคลื่อนของขนาด (tolerance)

เนื่องจากเครื่องมือกลทุกชนิดมีความผันแปรตามธรรมชาติในการทำหน้าที่ (natural variation) จึงไม่สามารถผลิตชิ้นงานได้ถูกต้องแน่นอนได้ เช่น เครื่องกลึงไม่สามารถกลึงเพลาให้ได้ขนาด Ø45 mm พอดี ทำได้เพียงขนาดที่ใกล้เคียงเท่านั้น ทำให้ผู้ออกแบบต้องกำหนดค่าความคลาดเคลื่อน (tolerance) ให้กับมิติของชิ้นส่วนที่ออกแบบเสมอ รูปที่ 1.1 แสดงตัวอย่างชิ้นงานระบุความคลาดเคลื่อน




รูปที่ 1.1 แสดงการบอกขนาดของชิ้นงาน

วิธีการระบุค่าความคลาดเคลื่อนแบบระบุค่าความเบี่ยงเบนจากขนาดมูลฐาน (basic size) ใช้กับมิติที่มีความสำคัญต่อหน้าที่ใช้งานของชิ้นส่วน (functional dimensions) มี 3 แบบดังนี้

(1) เบี่ยงเบนด้านเดียว (unilateral tolerancing) คือให้ขนาดความคลาดเคลื่อนเบี่ยงเบนจากขนาดมูลฐานไปทางด้านใดด้านหนึ่งเพียงด้านเดียว เช่น $30_{0}^{+0.024}$ หรือ $24_{-0.010}^{0}$ เป็นต้น วิธีนี้นิยมใช้กับมิติของรูปลักษณ์ที่เป็นชิ้นงานสวม (mating parts)

(2) เบี่ยงเบนสองด้าน (bilateral tolerancing) เช่น 30±0.02 มีขอบเขตด้านสูงและด้านต่ำเบี่ยงเบนจากขนาดมูลฐานเท่าๆกัน (คือเท่ากับครึ่งหนึ่งของความคลาดเคลื่อน) ส่วนมากจะใช้กับมิติความยาว ที่ไม่ใช่มิติของชิ้นงานสวมมิติของชิ้นงานบางอย่างอาจจำเป็นต้องใช้ค่าเบี่ยงเบนด้านบนและด้านล่างไม่เท่ากัน เช่น $30^{+0.030}_{-0.015}$ แบบนี้มักใช้กับงานสวมทางยาว (length fits) ในบางกรณี

(3) บอกเป็นขอบเขตของมิติ (limiting dimensions) บอกค่าขอบเขตของขนาดโดยตรง เช่น $\frac{30.02}{29.98}$ มีขอบเขตด้านสูง = 30.02 mm และขอบเขตด้านต่ำ = 29.98 mm เหมือนกับแบบที่ 2 คือใช้กับมิติใช้งาน (functional dimensions) ของชิ้นส่วนเท่านั้น

## 2. ความสามารถของกระบวนการผลิต

การกำหนดขนาดความคลาดเคลื่อนให้กับมิติของชิ้นส่วนที่ออกแบบขึ้นอยู่กับปัจจัยหลายอย่าง เช่น ชนิดของกระบวนการผลิต, ความชำนาญ และสภาพการใช้งานของเครื่องมือกล เป็นต้น ซึ่งต้องใช้ประสบการณ์ของผู้ออกแบบในการกำหนดค่าความคลาดเคลื่อนที่เหมาะสมได้ ในปัจจุบันได้มีการรวบรวมข้อมูลจากการทดลอง และจากการปฏิบัติงานจริงเกี่ยวกับความสามารถของกระบวนการผลิตพื้นฐาน เพื่อใช้เป็นเกณฑ์ในการกำหนดความคลาดเคลื่อนในการออกแบบ ตารางที่ 1.1 ระบุเกรดของความคลาดเคลื่อนของมิติที่เกิดจากกระบวนการผลิตต่างๆ

สิ่งสำคัญที่ผู้ออกแบบต้องตระหนักอยู่เสมอคือ ค่าความคลาดเคลื่อนยิ่งน้อย ต้นทุนในการผลิตยิ่งสูง จึงมีหลักการกำหนดความคลาดเคลื่อนให้กับมิติโดยกำหนดความคลาดเคลื่อนให้มากที่สุดและต้องไม่เกิดความเสียหายแก่หน้าที่ใช้งาน (function) ของชิ้นงานออกแบบ

มหาวิทยาลัยเทคโนโลยีสุรนารี

**ตารางที่ 1.1** แสดงขนาดความคลาดเคลื่อนของมิติที่เกิดจากกระบวนการผลิตต่างๆ

| กระบวนการผลิต | ความคลาดเคลื่อนโดยประมาณ (mm) | เกรดความคลาดเคลื่อน (IT GRADE) |
|---|---|---|
| การหล่อทราย: ชิ้นงานขนาดเล็ก,ใหญ่ | ±0.30; ±1.60 | IT16 |
| การตีขึ้นรูป (forging): ชิ้นงานขนาดเล็ก,ใหญ่ | ±0.80; ±1.50 | IT15 - IT16 |
| การฉีดโลหะ (die casting) และการทำโมลด์พลาสติก | ±0.40 (ทั่วๆไป) | IT14 |
| การฉีดโลหะผสมสังกะสีแบบเที่ยงตรง (precision) | 0.05-0.20 | IT10 - IT12 |
| การกดอัดโลหะ, การดึงท่อ, และการทำextrution | 0.10-1.40 | IT10 - IT13 |
| การไสโลหะ (planning and shaping) | 0.10-0.30 | IT10 - IT11 |
| การเจาะรู (drilling): ø 6 - 12 mm;<br>ø 12 - 20 mm | 0.10<br>0.18 | IT11 - IT12 |
| การรีม (reaming): ø 6 - 25 mm;<br>มากกว่า 25 mm | 0.02<br>0.035ขึ้นไป | IT7 - IT8 |
| การกัดโลหะ (milling): กัดแบบ gang milling;<br>กัดร่องเล็กๆ | 0.08-0.12<br>0.05-0.08 | IT8 - IT10 |
| การกลึง(turning): ใช้ capstan lathe และturret lathe,<br>และใช้(roller box), งานขนาดไม่เกิน 28 mm<br>การกลึงธรรมดา: ø 25 - 50 mm;<br>มากกว่า 50 mm | 0.05<br>0.10<br>0.12ขึ้นไป | IT8 - IT10 |
| การโบรช(broaching): ต่ำกว่า ø 25 mm;<br>ø 25 - 50 mm | 0.02<br>0.04 | IT7 - IT8 |
| การขัดรู(honing): ไม่เกิน ø50 mm | 0.01-0.016 | IT6 |
| การเจียรนัย(grinding): ต่ำกว่า ø 25 mm<br>ø 25 - 50 mm | 0.007-0.012<br>0.012-0.016 | IT5 - IT6 |
| การขัดผิวระนาบ(lapping): ใช้เครื่องขัด (machine) | 0.002-0.01 | IT4 - IT5 |
| การขัดผิวของมาตรฐาน หรือของเกจอ้างอิงต่างๆ | น้อยกว่า 0.002 | IT01 - IT3 |

การคำนวณค่าความคลาดเคลื่อนของมิติสามารถทำได้ สองวิธีคือ; (1) ใช้สูตรคำนวณดังสมการ1.1; (2) เปิดตารางแสดงค่าตัวเลขของเกรดความคลาดเคลื่อนมาตรฐาน(Standard tolerance grades international Tolerance grades) หรือ IT ดังตาราง 1.2 งานวิจัยนี้ได้ใช้สูตรคำนวณค่า international tolerance (IT) จากสมการที่ 1.1 ในการกำหนดค่าความคลาดเคลื่อนให้กับมิติต่างๆในแต่ละขั้นตอนการตัด ซึ่งสมการดังกล่าวเป็นสมการที่สะดวกต่อการทำงานของโปรแกรมแผนภูมิความคลาดเคลื่อน

$$T=10^{\frac{IT-16}{5}}\left(0.45\times D^{\frac{1}{3}}+0.001\times D\right)$$

สมการที่ 1.1

ตารางที่ 1.2 แสดงค่าตัวเลขของเกรดความคลาดเคลื่อนมาตรฐาน(Standard tolerance grades international Tolerance grades) หรือ IT

| Basic size (mm) | | Standard tolerance grades (IT) | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14[1] | 15[1] | 16[1] | 17[1] | 18[1] |
| over | up to | Tolerances (μm) | | | | | | | | | | | Tolerances (mm) | | | | | | |
| – | 3 | 0.8 | 1.2 | 2 | 3 | 4 | 6 | 10 | 14 | 25 | 40 | 60 | 0.10 | 0.14 | 0.26 | 0.40 | 0.60 | 1.00 | 1.40 |
| 3 | 6 | 1 | 1.5 | 2.5 | 4 | 5 | 8 | 12 | 18 | 30 | 48 | 75 | 0.12 | 0.18 | 0.30 | 0.48 | 0.75 | 1.20 | 1.80 |
| 6 | 10 | 1 | 1.5 | 2.5 | 4 | 6 | 9 | 15 | 22 | 36 | 58 | 90 | 0.15 | 0.22 | 0.36 | 0.58 | 0.90 | 1.50 | 2.20 |
| 10 | 18 | 1.2 | 2 | 3 | 5 | 8 | 11 | 18 | 27 | 43 | 70 | 110 | 0.18 | 0.27 | 0.43 | 0.70 | 1.10 | 1.80 | 2.70 |
| 18 | 30 | 1.5 | 2.5 | 4 | 6 | 9 | 13 | 21 | 33 | 52 | 84 | 130 | 0.21 | 0.33 | 0.52 | 0.84 | 1.30 | 2.10 | 3.30 |
| 30 | 50 | 1.5 | 2.5 | 4 | 7 | 11 | 16 | 25 | 39 | 62 | 100 | 160 | 0.25 | 0.39 | 0.62 | 1.00 | 1.60 | 2.50 | 3.90 |
| 50 | 80 | 2 | 3 | 5 | 8 | 13 | 19 | 30 | 46 | 74 | 120 | 190 | 0.30 | 0.46 | 0.74 | 1.20 | 1.90 | 3.00 | 4.60 |
| 80 | 120 | 2.5 | 4 | 6 | 10 | 15 | 22 | 35 | 54 | 87 | 140 | 220 | 0.35 | 0.54 | 0.87 | 1.40 | 2.20 | 3.50 | 5.40 |
| 120 | 180 | 3.5 | 5 | 8 | 12 | 18 | 25 | 40 | 63 | 100 | 160 | 250 | 0.40 | 0.63 | 1.00 | 1.60 | 2.50 | 4.00 | 6.30 |
| 180 | 250 | 4.5 | 7 | 10 | 14 | 20 | 29 | 46 | 72 | 115 | 185 | 290 | 0.46 | 0.72 | 1.15 | 1.85 | 2.90 | 4.60 | 7.20 |
| 250 | 315 | 6 | 8 | 12 | 16 | 23 | 32 | 52 | 81 | 130 | 210 | 320 | 0.52 | 0.81 | 1.30 | 2.10 | 3.20 | 5.20 | 8.10 |
| 315 | 400 | 7 | 9 | 13 | 18 | 25 | 36 | 57 | 89 | 140 | 230 | 360 | 0.57 | 0.89 | 1.40 | 2.30 | 3.60 | 5.70 | 8.90 |
| 400 | 500 | 8 | 10 | 15 | 20 | 27 | 40 | 63 | 97 | 155 | 250 | 400 | 0.63 | 0.97 | 1.55 | 2.50 | 4.00 | 6.30 | 9.70 |
| 500 | 630 | – | – | – | – | – | 44 | 70 | 110 | 175 | 280 | 440 | 0.70 | 1.10 | 1.75 | 2.80 | 4.40 | 7.00 | 11.00 |

## 3. การวางแผนกระบวนการผลิต (process planning)

การวางแผนกระบวนการผลิตเป็นการกำหนดขั้นตอนในการผลิตและการใช้อุปกรณ์เพื่อให้ได้ผลสำเร็จตามข้อกำหนดในแบบ ส่วนมากจะเกี่ยวข้องกับผลิตภัณฑ์ที่เป็นชิ้นงานตัดโลหะ (machined parts) การวางแผนกระบวนการผลิตนี้อยู่ในความรับผิดชอบของวิศวกร เพื่อให้เกิดแผนกระบวนการผลิต (process plan) ที่เหมาะสม วิศวกรต้องมีความรอบรู้ เกี่ยวกับเครื่องมือกลชนิดต่างๆ ทั้งชนิดที่ใช้งานทั่วไป และใช้งานเฉพาะด้านเกี่ยวกับอุปกรณ์สนับสนุน เช่น อุปกรณ์ยึดชิ้นงาน รู้หลักเศรษฐศาสตร์สำหรับเลือกใช้เครื่องมือกล และต้องมีประสบการณ์ในการทำงาน ด้วยเหตุนี้เอง เพื่อให้การวางแผนกระบวนการผลิตเป็นระบบและนำไปสู่แผนการผลิตที่ดี ประกอบด้วยขั้นตอนการปฏิบัติดังนี้

3.1 วิเคราะห์แบบ (part print analysis) ตรวจสอบแบบทำความเข้าใจและหารายละเอียดที่สำคัญจากแบบชิ้นงาน ที่สำคัญคือ : พื้นที่วิกฤติ (critical areas) ซึ่งมี 2 ชนิดได้แก่

(1) พื้นที่วิกฤติชิ้นงาน (product critical areas) หมายถึงบริเวณที่มีลักษณะพิเศษกว่าบริเวณอื่น เช่น ผิวที่ถูกระบุด้วยค่าความหยาบ, รูปลักษณ์ที่มีค่าความคลาดเคลื่อนเรขาคณิตกำกับ และ รูปลักษณ์ที่มีค่าความคลาดเคลื่อนของขนาด (size tolerance) น้อยเป็นต้น พื้นที่เหล่านี้มีความสำคัญต่อหน้าที่ใช้งานของชิ้นงาน และอาจต้องใช้กรรมวิธีการผลิตพิเศษ หรือต้องใช้ความระมัดระวังในการผลิตมากกว่าส่วนอื่น

(2) พื้นที่วิกฤติกระบวนการผลิต (process critical areas) หมายถึงบริเวณที่มีผลต่อขั้นตอนของการผลิต เช่น บริเวณที่เป็นจุดอ้างอิงของการบอกขนาด (base-line for dimensioning) หรือผิวอ้างอิงของความคลาดเคลื่อนเรขาคณิต เป็นต้น

3.2 ปรึกษาผู้ออกแบบ (consult the product designer) ในขณะที่วิเคราะห์แบบ อาจจำเป็นต้องขอความร่วมมือจากผู้ออกแบบ ให้อธิบายบางจุดของแบบที่ไม่ชัดเจน หรือในบางครั้งผู้วางแผนอาจมองเห็นปัญหาในการผลิตที่สามารถแก้ไขได้โดยการเปลี่ยนแปลงแบบ ตัวอย่างเช่น จุดอ้างอิงของการบอกขนาด อาจไม่เหมาะสมกับการผลิต และทำให้เกิดปัญหาความคลาดเคลื่อนสะสมจนไม่สามารถใช้กระบวนการผลิตที่มีอยู่ได้ ซึ่งปัญหานี้อาจแก้ไขได้โดยการเปลี่ยนแปลงจุดอ้างอิงสำหรับบอกขนาดเสียใหม่ กรณีเช่นนี้การปรึกษาวิศวกรผู้ออกแบบเป็นสิ่งที่ต้องทำเพราะการเปลี่ยนแปลงใดๆในแบบ จะต้องได้รับความเห็นชอบจากผู้ออกแบบเสมอ

สิ่งที่สำคัญคือ การเปลี่ยนแปลงใดๆ จะต้องมีเหตุผลที่เกิดจากการวิเคราะห์อย่างรอบคอบโดยมีเป้าหมายเพื่อให้ผลิตได้ง่ายขึ้นหรือประหยัดค่าใช้จ่าย มิใช่เป็นเหตุผลเพื่อแสดงความ ไม่เห็นด้วยกับแบบของฝ่ายออกแบบเท่านั้น

3.3 ทำรายการของงานที่ต้องทำ (list the operations required) แจกแจงว่ามีงาน (operations) อะไรบ้างที่ต้องทำ และทำที่ส่วนใดของชิ้นงาน ในรายการงานนี้ไม่จำเป็นต้องเรียงลำดับงานที่เกิดขึ้น เป็นเพียงบันทึกงานทั้งหมด และยังไม่มีการพิจารณาปัจจัยอื่นประกอบ อาจทำให้ตารางหรือเป็นรายการของงานธรรมดาก็ได้

3.4 กำหนดกระบวนการผลิตสำหรับงานแต่ละอย่าง (select the process) ผู้วางแผนจะต้องพิจารณาเปรียบเทียบระหว่างกระบวนการผลิตต่างๆที่สามารถทำงานเดียวกันได้ โดยพิจารณาปัจจัยต่างที่เกี่ยวข้อง เช่น ความเที่ยงตรงของเครื่องจักร, คุณภาพของงาน , และอัตราการผลิต เป็นต้น กระบวนการผลิตหรือวิธีการทำงานที่ถูกเลือกควรเป็นชนิดที่เสียค่าใช้จ่ายน้อยที่สุด

3.5 จัดลำดับการทำงาน (sequence the operations) เรียงลำดับการทำงานตั้งแต่เริ่มต้นจนได้ชิ้นงานสำเร็จ ในเรื่องนี้ Eary และ Johnson ได้ให้คำแนะนำการจัดเรียงกลุ่มงานต่างๆ ในกระบวนการตัดโลหะ และขึ้นรูปโลหะ โดยสรุปได้ดังนี้

(1) งานวิกฤติ (critical operations) ได้แก่งานที่ทำให้เกิดพื้นที่วิกฤติ 2 ชนิดที่กล่าวมาแล้ว มีหลักการแยกได้ดังนี้

- ถ้าหากเป็นไปได้ พื้นที่ที่ถูกจัดว่าเป็นพื้นที่วิกฤติเนื่องจากต้องรักษาคุณภาพของผิวควรได้รับการดำเนินการภายหลัง เพื่อหลีกเลี่ยงความเสียหายที่ผิว เช่น รอยขูดขีด หรือรอยกระแทกเนื่องจากเหตุบังเอิญ พื้นที่ส่วนนี้ส่วนใหญ่เป็นพื้นที่วิกฤติชิ้นงาน)

- พื้นที่ซึ่งเป็นพื้นที่วิกฤติเพราะเป็นจุดอ้างอิงของการบอกขนาดจะต้องได้รับการดำเนินการในขั้นตอนต้นๆ ของขั้นตอนการทำงาน (operational sequence ) เพราะพื้นที่เหล่านี้เป็นผิวอ้างอิงในการวัดหรือผิวกำหนดตำแหน่งชิ้นงานสำหรับขั้นตอนการผลิตอื่น จึงถือได้ว่าเป็นพื้นที่วิกฤติกระบวนการผลิต ( process critical areas )

- โดยทั่วไป รูปลักษณ์ที่มีค่าความคลาดเคลื่อนน้อย เป็นพื้นที่วิกฤติที่ควรถูกดำเนินการในตอนต้นของลำดับงาน เพราะว่า สามารถใช้เป็นรูปลักษณ์อ้างอิงสำหรับการตรวจสอบขนาดหรือการตัดโลหะในขั้นตอนต่อๆมา และเนื่องจากมีความคลาดเคลื่อนน้อย, จึงยากต่อการผลิตและมีโอกาสเป็นงานเสียได้มาก เพราะฉะนั้นจึงควรทำตั้งแต่ตอนแรก เพื่อว่าถ้าชิ้นงานเสียในตอนแรก ก็ไม่ต้องทำงานอื่นที่เหลือ จึงเป็นการประหยัดค่าใช้จ่ายในการผลิต

(2) งานรอง (secondary or tie-in operations) เป็นงานที่มีความสำคัญน้อยกว่างานวิกฤติ และไม่จำเป็นต้องใช้ความระมัดระวังในการผลิตมากนัก ค่าความคลาดเคลื่อนที่กำหนดให้กับมิติของรูปลักษณ์ สำหรับงานประเภทนี้ มักเป็นค่าความคลาดเคลื่อนทั่วไป (general tolerance) งานรองอาจถูกทำก่อนหรือหลังงานอื่น ขึ้นอยู่กับเงื่อนไขในการผลิต เช่น การทำ counter bore ซึ่งเป็นงานรอง จะต้องทำหลังการเจาะรู; หรือในการเจาะรู แนวของรูจะต้องตั้งฉากกับผิวชิ้นงาน ดั้งนั้นผิวงานที่รูเปิดออกจะต้องถูกตัดก่อนถึงแม้ว่าเป็นงานรองก็ตาม เป็นต้น

(3) งานเตรียมผิว (qualifying operations) ส่วนใหญ่เกิดขึ้นกับงานตัดโลหะที่ทำกับชิ้นงานหล่อหรืองานตีขึ้นรูป (castings and forgings) เนื่องจากชิ้นงานประเภทนี้มักมีผิวไม่เรียบทำให้ไม่สะดวกต่อการตัดพื้นที่วิกฤติ หรือไม่สามารถทำพื้นที่วิกฤติให้เที่ยงตรงได้ จึงต้องตัดผิวอื่นก่อนเพื่อใช้เป็นผิวกำหนดตำแหน่งชิ้นงานสำหรับทำพื้นที่วิกฤติ

(4) งานเตรียมผิวใหม่ (re- qualifying operations) ในระหว่างการตัดโลหะ บางครั้งงานเตรียมผิวเกิดขึ้นหลายครั้ง เช่น ผิวที่เคยเป็นผิวอ้างอิงในการผลิต (manufacturing datum) อาจเกิดการบิดตัวเนื่องจากการอบชุบ (heat treatment) ดั้งนั้นจึงต้องตัดซ้ำเพื่อให้เกิดผิวอ้างอิงใหม่

3.6 ตรวจสอบลำดับขั้นตอนการทำงานด้วยแผนภูมิความคลาดเคลื่อน แผนภูมิความคลาดเคลื่อน (Tolerance charts) คือแผนภาพที่แสดงการเปลี่ยนแปลงของมิติและความ

คลาดเคลื่อนของขนาดชิ้นงานที่เกิดขึ้นในแต่ละขั้นตอนของการตัดโลหะ แผนภาพเช่นนี้มีประโยชน์สำหรับตรวจสอบว่าลำดับการตัดโลหะ (machining sequence) ที่กำหนดขึ้นนั้นสามารถทำงานได้ตามต้องการหรือไม่ เพราะฉะนั้นการใช้แผนภูมิความคลาดเคลื่อน จึงเป็นเครื่องมือสำหรับการประเมินความเป็นไปได้ของวิธีการตัดโลหะก่อนที่จะเกิดการตัดโลหะจริง และหากพบว่าจำเป็นต้องปรับเปลี่ยนการทำงานใหม่ให้เหมาะสม ก็สามารถใช้แผนภูมิเป็นพื้นฐานในการวิเคราะห์ จึงกล่าวได้ว่าแผนภูมิชนิดนี้เป็นเครื่องมือที่สำคัญอย่างหนึ่งของวิศวกร

3.7 กำหนดวิธีการตรวจสอบ (specify gaging and measuring methods ) ถึงแม้ว่าการออกแบบเครื่องมือการตรวจสอบ เป็นหน้าที่ของฝ่ายออกแบบเครื่องมือ แต่ผู้วางแผนกระบวนการผลิต ( process planners) ก็มีส่วนรับผิดชอบกำหนดวิธีการวัดและตรวจสอบงานในขั้นตอนต่างๆ โดยทำงานประสานกับฝ่ายควบคุมคุณภาพ (Quality control) วิธีการตรวจสอบที่ว่านี้อาจเป็นเพียงการใช้เครื่องมือวัดมาตรฐานธรรมดาหรือการออกแบบเกจ (gage) ขึ้นใช้และรวมไปถึงการใช้เครื่องมืออัตโนมัติ หรือ กึ่งอัตโนมัติ ตรวจสอบด้วย

มหาวิทยาลัยเทคโนโลยีสุรนารี

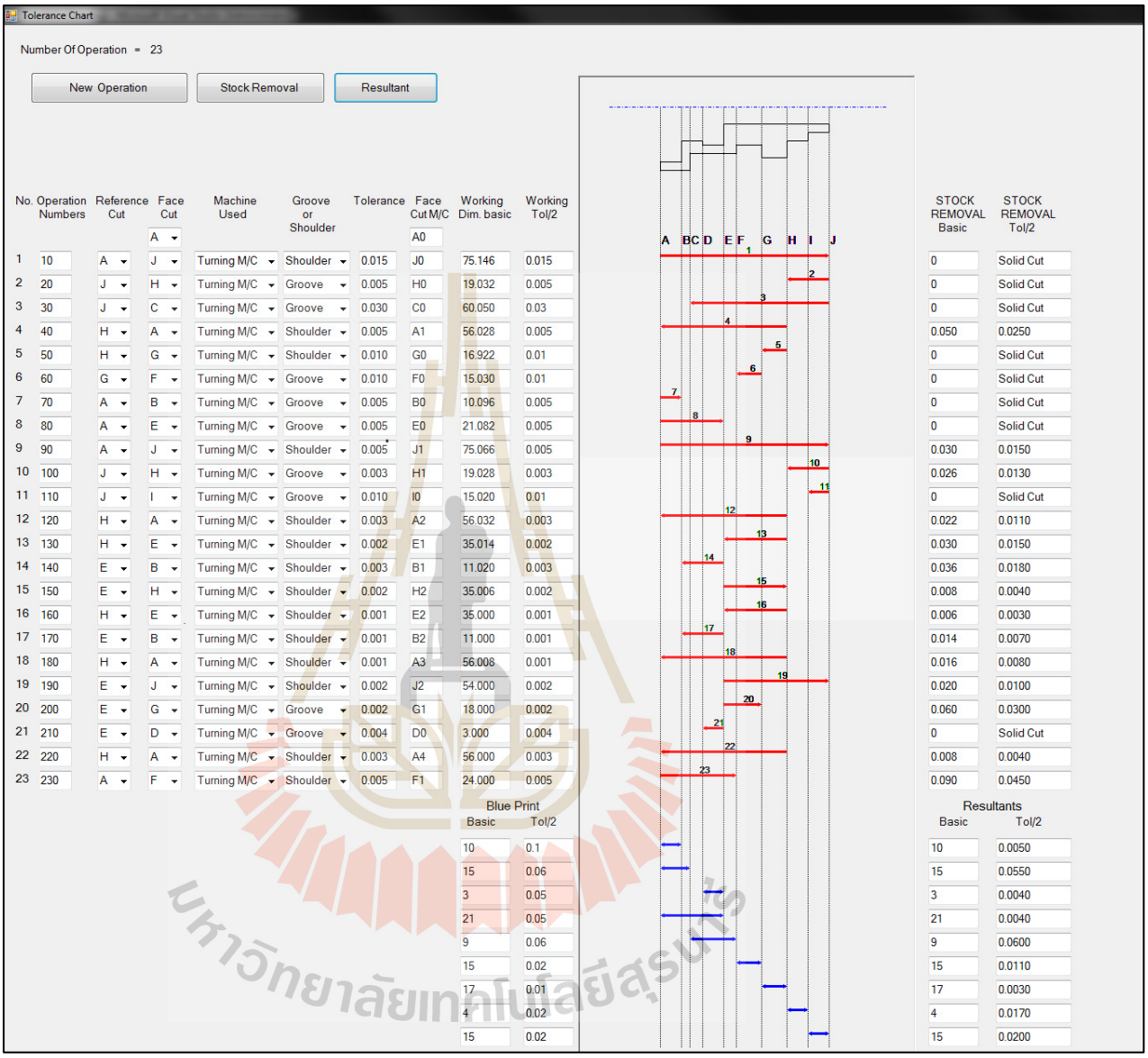
# ภาคผนวก ค

# การทดสอบโปรแกรม

ภาคผนวก ค. แสดงการทดสอบโปรแกรมแผนภูมิความคลาดเคลื่อน จำนวน 5 ตัวอย่าง โดยโปรแกรมจะแสดงรายละเอียดดังนี้ แสดงแบบสเกตซ์ของชิ้นงาน, รายละเอียดเกี่ยวกับความคลาดเคลื่อนของการตัดแต่ละครั้ง, เครื่องมือกลที่ใช้, ค่าความคลาดเคลื่อนและขนาดมูลฐานในแต่ละขั้นตอนการตัด, ผิวที่ถูกตัด-ผิวอ้างอิง ซึ่งเป็นข้อมูลส่วนด้านซ้ายและตอนกลางของแผนภูมิฯ, และผลลัพธ์ของโปรแกรมจะแสดงในส่วนด้านล้างซ้ายของแผนภูมิ แสดงดังในรูปตัวอย่างการทดสอบโปรแกรมที่ 1-5

มหาวิทยาลัยเทคโนโลยีสุรนารี

ตัวอย่างที่ 1



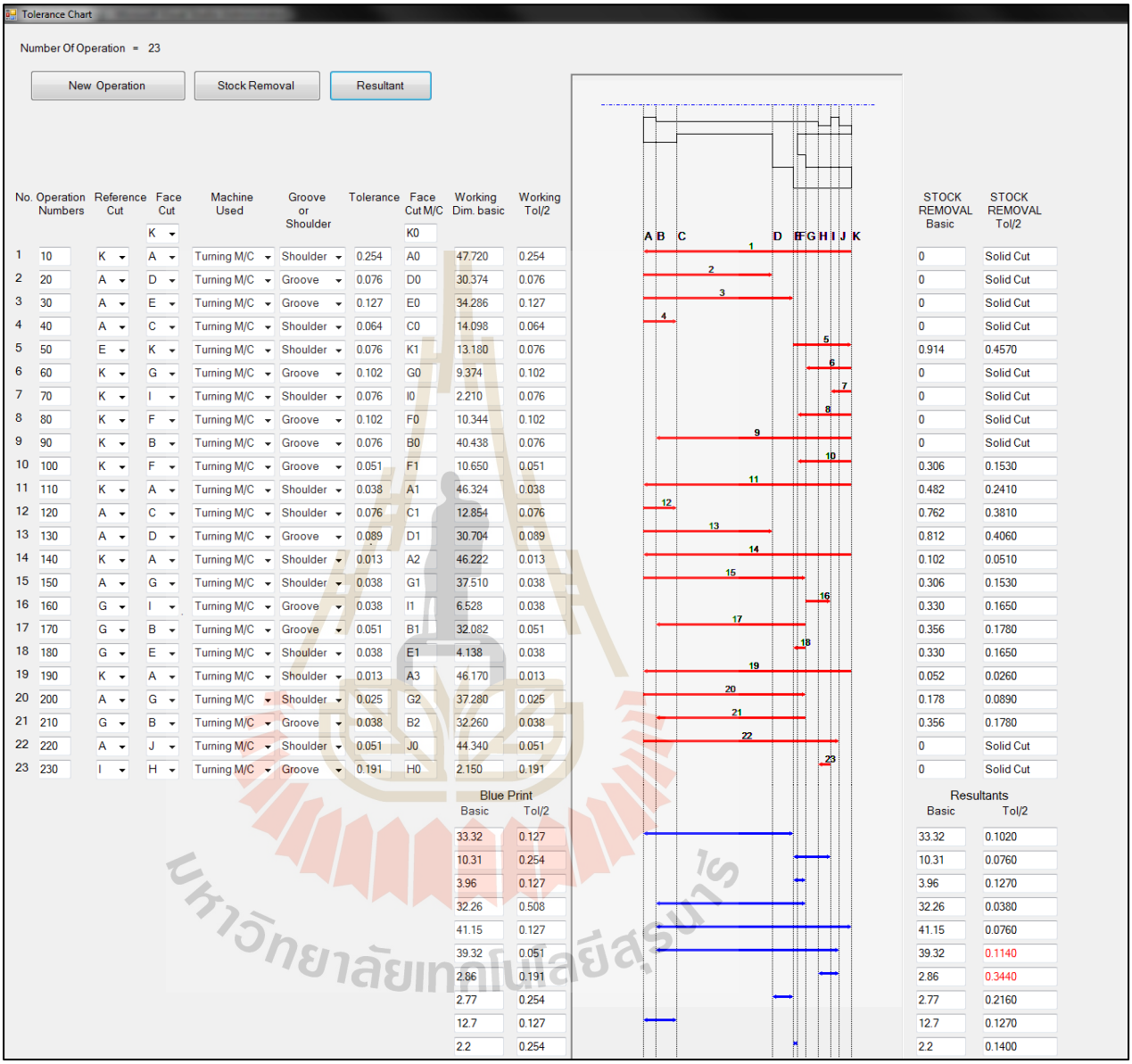

| No. | Operation Numbers | Reference Cut | Face Cut | Machine Used | Groove or Shoulder | Tolerance | Face Cut M/C | Working Dim. basic | Working Tol/2 | STOCK REMOVAL Basic | STOCK REMOVAL Tol/2 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | A | | | | A0 | | | | |
| 1 | 10 | A | J | Turning M/C | Shoulder | 0.015 | J0 | 75.146 | 0.015 | 0 | Solid Cut |
| 2 | 20 | J | H | Turning M/C | Groove | 0.005 | H0 | 19.032 | 0.005 | 0 | Solid Cut |
| 3 | 30 | J | C | Turning M/C | Groove | 0.030 | C0 | 60.050 | 0.03 | 0 | Solid Cut |
| 4 | 40 | H | A | Turning M/C | Shoulder | 0.005 | A1 | 56.028 | 0.005 | 0.050 | 0.0250 |
| 5 | 50 | H | G | Turning M/C | Shoulder | 0.010 | G0 | 16.922 | 0.01 | 0 | Solid Cut |
| 6 | 60 | G | F | Turning M/C | Groove | 0.010 | F0 | 15.030 | 0.01 | 0 | Solid Cut |
| 7 | 70 | A | B | Turning M/C | Groove | 0.005 | B0 | 10.096 | 0.005 | 0 | Solid Cut |
| 8 | 80 | A | E | Turning M/C | Groove | 0.005 | E0 | 21.082 | 0.005 | 0 | Solid Cut |
| 9 | 90 | A | J | Turning M/C | Shoulder | 0.005 | J1 | 75.066 | 0.005 | 0.030 | 0.0150 |
| 10 | 100 | J | H | Turning M/C | Groove | 0.003 | H1 | 19.028 | 0.003 | 0.026 | 0.0130 |
| 11 | 110 | J | I | Turning M/C | Groove | 0.010 | I0 | 15.020 | 0.01 | 0 | Solid Cut |
| 12 | 120 | H | A | Turning M/C | Shoulder | 0.003 | A2 | 56.032 | 0.003 | 0.022 | 0.0110 |
| 13 | 130 | H | E | Turning M/C | Shoulder | 0.002 | E1 | 35.014 | 0.002 | 0.030 | 0.0150 |
| 14 | 140 | E | B | Turning M/C | Shoulder | 0.003 | B1 | 11.020 | 0.003 | 0.036 | 0.0180 |
| 15 | 150 | E | H | Turning M/C | Shoulder | 0.002 | H2 | 35.006 | 0.002 | 0.008 | 0.0040 |
| 16 | 160 | H | E | Turning M/C | Shoulder | 0.001 | E2 | 35.000 | 0.001 | 0.006 | 0.0030 |
| 17 | 170 | E | B | Turning M/C | Shoulder | 0.001 | B2 | 11.000 | 0.001 | 0.014 | 0.0070 |
| 18 | 180 | H | A | Turning M/C | Shoulder | 0.001 | A3 | 56.008 | 0.001 | 0.016 | 0.0080 |
| 19 | 190 | E | J | Turning M/C | Shoulder | 0.002 | J2 | 54.000 | 0.002 | 0.020 | 0.0100 |
| 20 | 200 | E | G | Turning M/C | Groove | 0.002 | G1 | 18.000 | 0.002 | 0.060 | 0.0300 |
| 21 | 210 | E | D | Turning M/C | Groove | 0.004 | D0 | 3.000 | 0.004 | 0 | Solid Cut |
| 22 | 220 | H | A | Turning M/C | Shoulder | 0.003 | A4 | 56.000 | 0.003 | 0.008 | 0.0040 |
| 23 | 230 | A | F | Turning M/C | Shoulder | 0.005 | F1 | 24.000 | 0.005 | 0.090 | 0.0450 |

| Blue Print Basic | Blue Print Tol/2 | Resultants Basic | Resultants Tol/2 |
|---|---|---|---|
| 10 | 0.1 | 10 | 0.0050 |
| 15 | 0.06 | 15 | 0.0550 |
| 3 | 0.05 | 3 | 0.0040 |
| 21 | 0.05 | 21 | 0.0040 |
| 9 | 0.06 | 9 | 0.0600 |
| 15 | 0.02 | 15 | 0.0110 |
| 17 | 0.01 | 17 | 0.0030 |
| 4 | 0.02 | 4 | 0.0170 |
| 15 | 0.02 | 15 | 0.0200 |

รูปตัวอย่างการทดสอบโปรแกรมที่ 1

มหาวิทยาลัยเทคโนโลยีสุรนารี

ตัวอย่างที่ 2

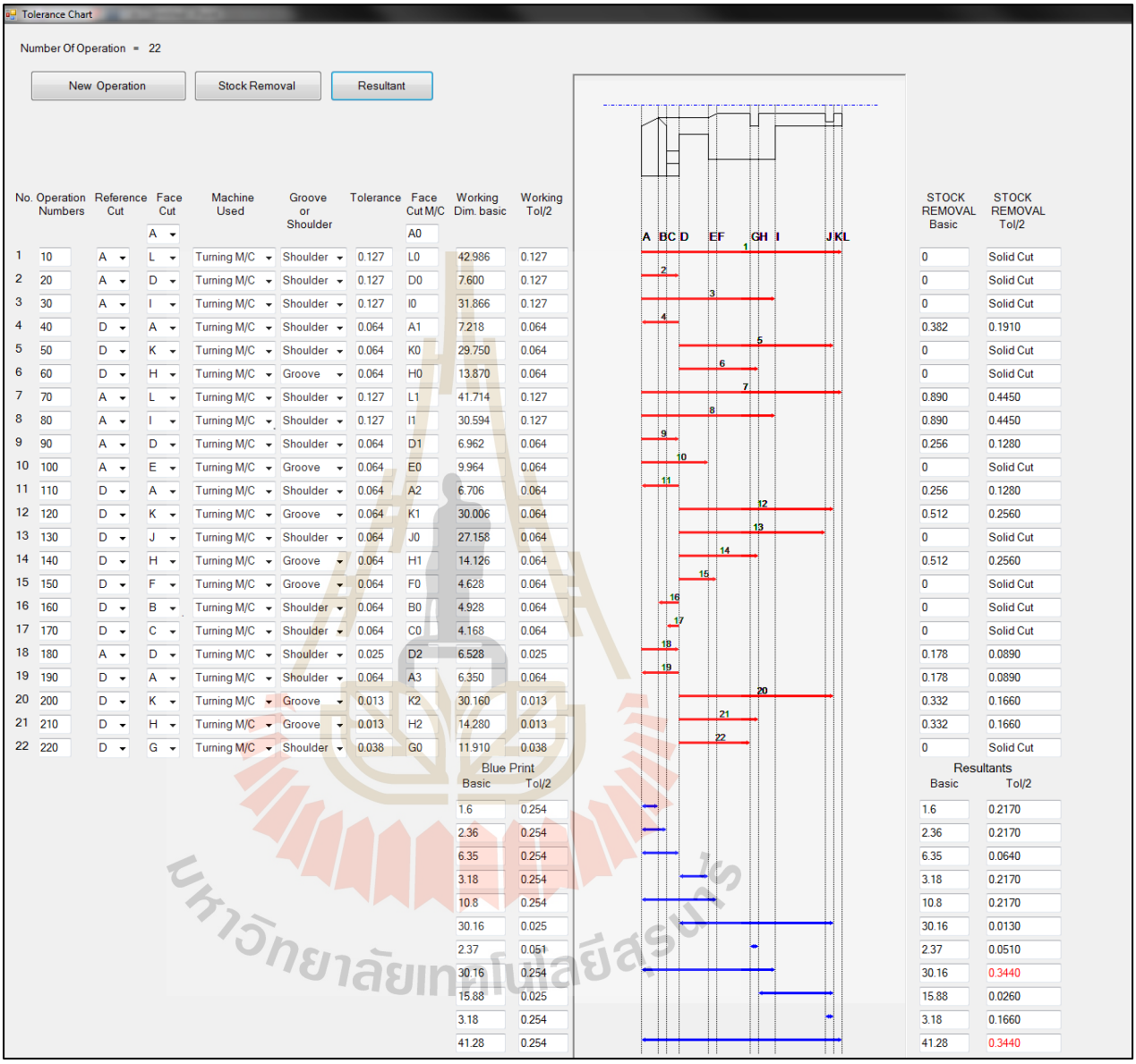



| No. | Operation Numbers | Reference Cut | Face Cut | Machine Used | Groove or Shoulder | Tolerance | Face Cut M/C | Working Dim. basic | Working Tol/2 | STOCK REMOVAL Basic | STOCK REMOVAL Tol/2 | Blue Print Basic | Blue Print Tol/2 | Resultants Basic | Resultants Tol/2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | K | | | | K0 | | | | | | | | |
| 1 | 10 | K | A | Turning M/C | Shoulder | 0.254 | A0 | 47.720 | 0.254 | 0 | Solid Cut | 33.32 | 0.127 | 33.32 | 0.1020 |
| 2 | 20 | A | D | Turning M/C | Groove | 0.076 | D0 | 30.374 | 0.076 | 0 | Solid Cut | 10.31 | 0.254 | 10.31 | 0.0760 |
| 3 | 30 | A | E | Turning M/C | Groove | 0.127 | E0 | 34.286 | 0.127 | 0 | Solid Cut | 3.96 | 0.127 | 3.96 | 0.1270 |
| 4 | 40 | A | C | Turning M/C | Shoulder | 0.064 | C0 | 14.098 | 0.064 | 0 | Solid Cut | 32.26 | 0.508 | 32.26 | 0.0380 |
| 5 | 50 | E | K | Turning M/C | Shoulder | 0.076 | K1 | 13.180 | 0.076 | 0.914 | 0.4570 | 41.15 | 0.127 | 41.15 | 0.0760 |
| 6 | 60 | K | G | Turning M/C | Groove | 0.102 | G0 | 9.374 | 0.102 | 0 | Solid Cut | 39.32 | 0.051 | 39.32 | 0.1140 |
| 7 | 70 | K | I | Turning M/C | Shoulder | 0.076 | I0 | 2.210 | 0.076 | 0 | Solid Cut | 2.86 | 0.191 | 2.86 | 0.3440 |
| 8 | 80 | K | F | Turning M/C | Groove | 0.102 | F0 | 10.344 | 0.102 | 0 | Solid Cut | 2.77 | 0.254 | 2.77 | 0.2160 |
| 9 | 90 | K | B | Turning M/C | Groove | 0.076 | B0 | 40.438 | 0.076 | 0 | Solid Cut | 12.7 | 0.127 | 12.7 | 0.1270 |
| 10 | 100 | K | F | Turning M/C | Groove | 0.051 | F1 | 10.650 | 0.051 | 0.306 | 0.1530 | 2.2 | 0.254 | 2.2 | 0.1400 |
| 11 | 110 | K | A | Turning M/C | Shoulder | 0.038 | A1 | 46.324 | 0.038 | 0.482 | 0.2410 | | | | |
| 12 | 120 | A | C | Turning M/C | Shoulder | 0.076 | C1 | 12.854 | 0.076 | 0.762 | 0.3810 | | | | |
| 13 | 130 | A | D | Turning M/C | Groove | 0.089 | D1 | 30.704 | 0.089 | 0.812 | 0.4060 | | | | |
| 14 | 140 | K | A | Turning M/C | Shoulder | 0.013 | A2 | 46.222 | 0.013 | 0.102 | 0.0510 | | | | |
| 15 | 150 | A | G | Turning M/C | Shoulder | 0.038 | G1 | 37.510 | 0.038 | 0.306 | 0.1530 | | | | |
| 16 | 160 | G | I | Turning M/C | Groove | 0.038 | I1 | 6.528 | 0.038 | 0.330 | 0.1650 | | | | |
| 17 | 170 | G | B | Turning M/C | Groove | 0.051 | B1 | 32.082 | 0.051 | 0.356 | 0.1780 | | | | |
| 18 | 180 | G | E | Turning M/C | Shoulder | 0.038 | E1 | 4.138 | 0.038 | 0.330 | 0.1650 | | | | |
| 19 | 190 | K | A | Turning M/C | Shoulder | 0.013 | A3 | 46.170 | 0.013 | 0.052 | 0.0260 | | | | |
| 20 | 200 | A | G | Turning M/C | Shoulder | 0.025 | G2 | 37.280 | 0.025 | 0.178 | 0.0890 | | | | |
| 21 | 210 | G | B | Turning M/C | Groove | 0.038 | B2 | 32.260 | 0.038 | 0.356 | 0.1780 | | | | |
| 22 | 220 | A | J | Turning M/C | Shoulder | 0.051 | J0 | 44.340 | 0.051 | 0 | Solid Cut | | | | |
| 23 | 230 | I | H | Turning M/C | Groove | 0.191 | H0 | 2.150 | 0.191 | 0 | Solid Cut | | | | |

รูปตัวอย่างการทดสอบโปรแกรมที่ 2

มหาวิทยาลัยเทคโนโลยีสุรนารี

ตัวอย่างที่ 3

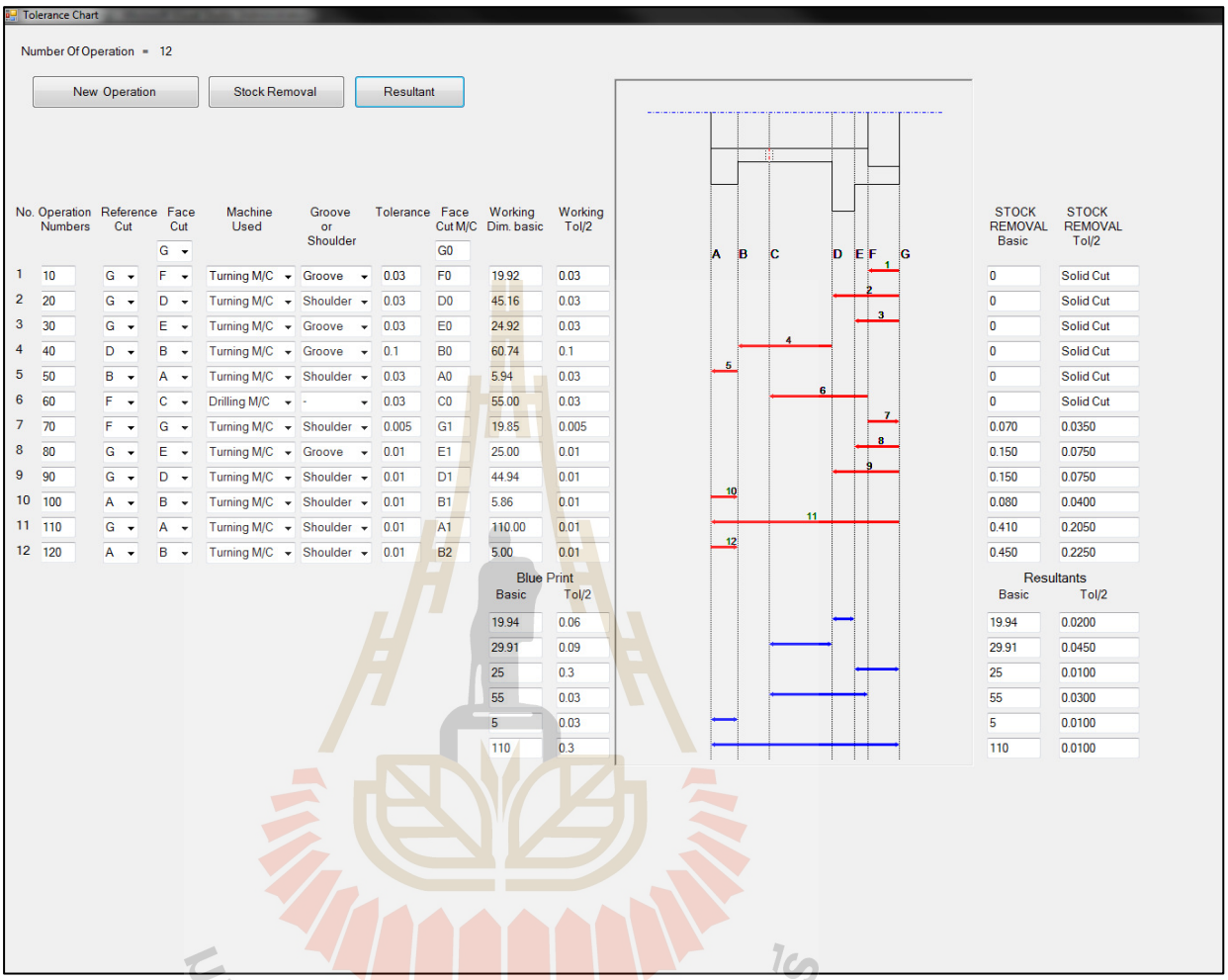



| No. | Operation Numbers | Reference Cut | Face Cut | Machine Used | Groove or Shoulder | Tolerance | Face Cut M/C | Working Dim. basic | Working Tol/2 | STOCK REMOVAL Basic | STOCK REMOVAL Tol/2 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | A | | | | A0 | | | | |
| 1 | 10 | A | L | Turning M/C | Shoulder | 0.127 | L0 | 42.986 | 0.127 | 0 | Solid Cut |
| 2 | 20 | A | D | Turning M/C | Shoulder | 0.127 | D0 | 7.600 | 0.127 | 0 | Solid Cut |
| 3 | 30 | A | I | Turning M/C | Shoulder | 0.127 | I0 | 31.866 | 0.127 | 0 | Solid Cut |
| 4 | 40 | D | A | Turning M/C | Shoulder | 0.064 | A1 | 7.218 | 0.064 | 0.382 | 0.1910 |
| 5 | 50 | D | K | Turning M/C | Shoulder | 0.064 | K0 | 29.750 | 0.064 | 0 | Solid Cut |
| 6 | 60 | D | H | Turning M/C | Groove | 0.064 | H0 | 13.870 | 0.064 | 0 | Solid Cut |
| 7 | 70 | A | L | Turning M/C | Shoulder | 0.127 | L1 | 41.714 | 0.127 | 0.890 | 0.4450 |
| 8 | 80 | A | I | Turning M/C | Shoulder | 0.127 | I1 | 30.594 | 0.127 | 0.890 | 0.4450 |
| 9 | 90 | A | D | Turning M/C | Shoulder | 0.064 | D1 | 6.962 | 0.064 | 0.256 | 0.1280 |
| 10 | 100 | A | E | Turning M/C | Groove | 0.064 | E0 | 9.964 | 0.064 | 0 | Solid Cut |
| 11 | 110 | D | A | Turning M/C | Shoulder | 0.064 | A2 | 6.706 | 0.064 | 0.256 | 0.1280 |
| 12 | 120 | D | K | Turning M/C | Groove | 0.064 | K1 | 30.006 | 0.064 | 0.512 | 0.2560 |
| 13 | 130 | D | J | Turning M/C | Shoulder | 0.064 | J0 | 27.158 | 0.064 | 0 | Solid Cut |
| 14 | 140 | D | H | Turning M/C | Groove | 0.064 | H1 | 14.126 | 0.064 | 0.512 | 0.2560 |
| 15 | 150 | D | F | Turning M/C | Groove | 0.064 | F0 | 4.628 | 0.064 | 0 | Solid Cut |
| 16 | 160 | D | B | Turning M/C | Shoulder | 0.064 | B0 | 4.928 | 0.064 | 0 | Solid Cut |
| 17 | 170 | D | C | Turning M/C | Shoulder | 0.064 | C0 | 4.168 | 0.064 | 0 | Solid Cut |
| 18 | 180 | A | D | Turning M/C | Shoulder | 0.025 | D2 | 6.528 | 0.025 | 0.178 | 0.0890 |
| 19 | 190 | D | A | Turning M/C | Shoulder | 0.064 | A3 | 6.350 | 0.064 | 0.178 | 0.0890 |
| 20 | 200 | D | K | Turning M/C | Groove | 0.013 | K2 | 30.160 | 0.013 | 0.332 | 0.1660 |
| 21 | 210 | D | H | Turning M/C | Groove | 0.013 | H2 | 14.280 | 0.013 | 0.332 | 0.1660 |
| 22 | 220 | D | G | Turning M/C | Shoulder | 0.038 | G0 | 11.910 | 0.038 | 0 | Solid Cut |

| Blue Print Basic | Blue Print Tol/2 | Resultants Basic | Resultants Tol/2 |
|---|---|---|---|
| 1.6 | 0.254 | 1.6 | 0.2170 |
| 2.36 | 0.254 | 2.36 | 0.2170 |
| 6.35 | 0.254 | 6.35 | 0.0640 |
| 3.18 | 0.254 | 3.18 | 0.2170 |
| 10.8 | 0.254 | 10.8 | 0.2170 |
| 30.16 | 0.025 | 30.16 | 0.0130 |
| 2.37 | 0.051 | 2.37 | 0.0510 |
| 30.16 | 0.254 | 30.16 | 0.3440 |
| 15.88 | 0.025 | 15.88 | 0.0260 |
| 3.18 | 0.254 | 3.18 | 0.1660 |
| 41.28 | 0.254 | 41.28 | 0.3440 |

รูปตัวอย่างการทดสอบโปรแกรมที่ 3

ตัวอย่างที่ 4



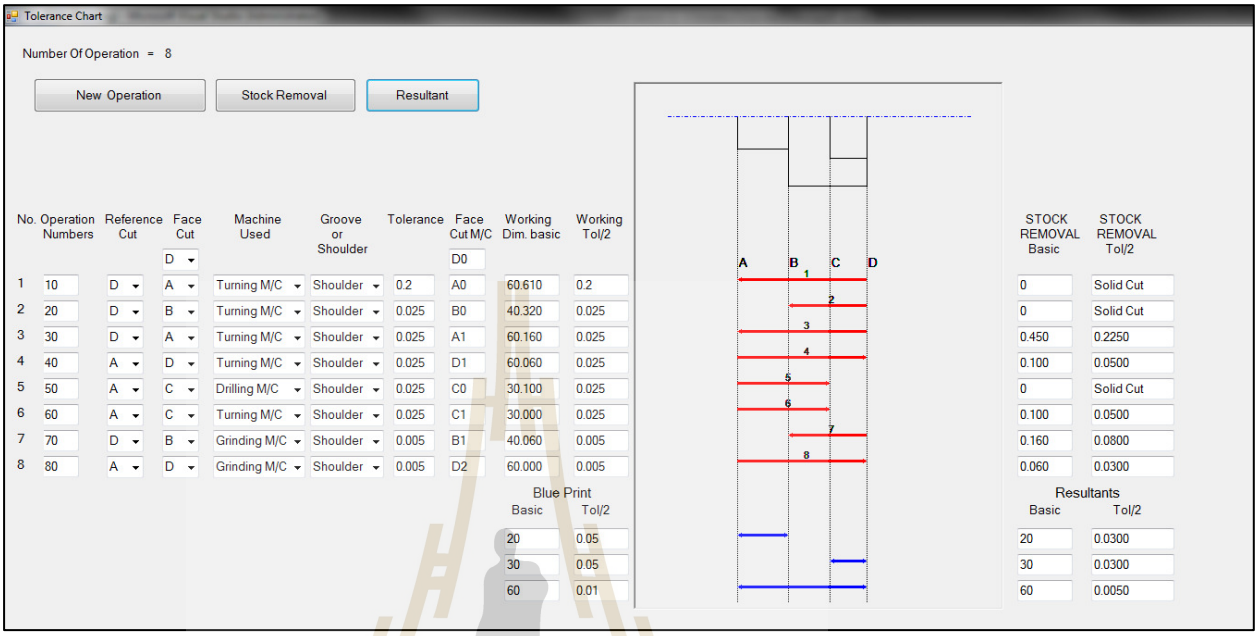

| No. | Operation Numbers | Reference Cut | Face Cut | Machine Used | Groove or Shoulder | Tolerance | Face Cut M/C | Working Dim. basic | Working Tol/2 | STOCK REMOVAL Basic | STOCK REMOVAL Tol/2 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | G | | | | G0 | | | | |
| 1 | 10 | G | F | Turning M/C | Groove | 0.03 | F0 | 19.92 | 0.03 | 0 | Solid Cut |
| 2 | 20 | G | D | Turning M/C | Shoulder | 0.03 | D0 | 45.16 | 0.03 | 0 | Solid Cut |
| 3 | 30 | G | E | Turning M/C | Groove | 0.03 | E0 | 24.92 | 0.03 | 0 | Solid Cut |
| 4 | 40 | D | B | Turning M/C | Groove | 0.1 | B0 | 60.74 | 0.1 | 0 | Solid Cut |
| 5 | 50 | B | A | Turning M/C | Shoulder | 0.03 | A0 | 5.94 | 0.03 | 0 | Solid Cut |
| 6 | 60 | F | C | Drilling M/C | - | 0.03 | C0 | 55.00 | 0.03 | 0 | Solid Cut |
| 7 | 70 | F | G | Turning M/C | Shoulder | 0.005 | G1 | 19.85 | 0.005 | 0.070 | 0.0350 |
| 8 | 80 | G | E | Turning M/C | Groove | 0.01 | E1 | 25.00 | 0.01 | 0.150 | 0.0750 |
| 9 | 90 | G | D | Turning M/C | Shoulder | 0.01 | D1 | 44.94 | 0.01 | 0.150 | 0.0750 |
| 10 | 100 | A | B | Turning M/C | Shoulder | 0.01 | B1 | 5.86 | 0.01 | 0.080 | 0.0400 |
| 11 | 110 | G | A | Turning M/C | Shoulder | 0.01 | A1 | 110.00 | 0.01 | 0.410 | 0.2050 |
| 12 | 120 | A | B | Turning M/C | Shoulder | 0.01 | B2 | 5.00 | 0.01 | 0.450 | 0.2250 |

| Blue Print Basic | Blue Print Tol/2 | Resultants Basic | Resultants Tol/2 |
|---|---|---|---|
| 19.94 | 0.06 | 19.94 | 0.0200 |
| 29.91 | 0.09 | 29.91 | 0.0450 |
| 25 | 0.3 | 25 | 0.0100 |
| 55 | 0.03 | 55 | 0.0300 |
| 5 | 0.03 | 5 | 0.0100 |
| 110 | 0.3 | 110 | 0.0100 |

รูปตัวอย่างการทดสอบโปรแกรมที่ 4

มหาวิทยาลัยเทคโนโลยีสุรนารี

ตัวอย่างที่ 5




| No. | Operation Numbers | Reference Cut | Face Cut | Machine Used | Groove or Shoulder | Tolerance | Face Cut M/C | Working Dim. basic | Working Tol/2 | STOCK REMOVAL Basic | STOCK REMOVAL Tol/2 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | D | | | | D0 | | | | |
| 1 | 10 | D | A | Turning M/C | Shoulder | 0.2 | A0 | 60.610 | 0.2 | 0 | Solid Cut |
| 2 | 20 | D | B | Turning M/C | Shoulder | 0.025 | B0 | 40.320 | 0.025 | 0 | Solid Cut |
| 3 | 30 | D | A | Turning M/C | Shoulder | 0.025 | A1 | 60.160 | 0.025 | 0.450 | 0.2250 |
| 4 | 40 | A | D | Turning M/C | Shoulder | 0.025 | D1 | 60.060 | 0.025 | 0.100 | 0.0500 |
| 5 | 50 | A | C | Drilling M/C | Shoulder | 0.025 | C0 | 30.100 | 0.025 | 0 | Solid Cut |
| 6 | 60 | A | C | Turning M/C | Shoulder | 0.025 | C1 | 30.000 | 0.025 | 0.100 | 0.0500 |
| 7 | 70 | D | B | Grinding M/C | Shoulder | 0.005 | B1 | 40.060 | 0.005 | 0.160 | 0.0800 |
| 8 | 80 | A | D | Grinding M/C | Shoulder | 0.005 | D2 | 60.000 | 0.005 | 0.060 | 0.0300 |

| Blue Print Basic | Blue Print Tol/2 | Resultants Basic | Resultants Tol/2 |
|---|---|---|---|
| 20 | 0.05 | 20 | 0.0300 |
| 30 | 0.05 | 30 | 0.0300 |
| 60 | 0.01 | 60 | 0.0050 |

รูปตัวอย่างการทดสอบโปรแกรมที่ 5

มหาวิทยาลัยเทคโนโลยีสุรนารี

# ภาคผนวก ง

# รายการโปรแกรม (Program Listings)

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
Module Module1
    Public Structure Dimension
        Public FromType As String
        Public ToType As String
        Public Basic As Single
        Public Tolerance As Single
        Public Machine As String
        Public ReferenceCut As String
        Public Facecut As String
    End Structure
    Public Structure SizeHole
        Public XX1 As Integer
        Public YY1 As Integer
        Public XX2 As Integer
        Public YY2 As Integer
        Public SHOLE As Integer
    End Structure
    Public Hole() As SizeHole
    Public BluePrint() As Dimension
    Public TC() As Dimension
    Public NumberOfCut As Integer
    Public Cut() As String
    Public NumberOfOperation As Integer
    Public MC() As String = {"-", "Lapping M/C", "Superfinishing M/C", "Diamond turning M/C", "Grinding
M/C", "Turning M/C", "Milling M/C", "Shaping M/C", "Drilling M/C"}
    Public OperationNumbers() As Integer
    Public Path() As String
    Public PathResult() As String
    Public GenBP(,) As Single
    Public AR1 As New ArrayList
    Public ReversePath() As String
    Public ReversePathResult() As String
    Public Direction() As String = {"-", "Shoulder", "Groove"}
    Public DirectionCut() As Integer
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
    Public NumbersX1() As Integer
    Public NumbersY1() As Integer
    Public NumbersX2() As Integer
    Public NumbersY2() As Integer
    Public searchvalueNews As Integer
    Public positionX() As Integer
    Public positionYmax As Integer
    Public Hcount As Integer = -1
    Public countPh As Integer
End Module
Module Module2
    Public Sub BasicArray(ByVal NumberOfCut As Integer)
        NoSurf = NumberOfCut
        Dim j, n, F, T As Integer
        Dim a, b As Single
        ReDim GenBP(NoSurf, NoSurf)
        For i = 0 To NoSurf - 2
            F = Asc(BluePrint(i).FromType) - 64
            T = Asc(BluePrint(i).ToType) - 64
            If F < T Then
                GenBP(F, T) = BluePrint(i).Basic
                GenBP(T, F) = -BluePrint(i).Basic
            ElseIf F > T Then
                GenBP(F, T) = -BluePrint(i).Basic
                GenBP(T, F) = BluePrint(i).Basic
            End If
        Next i
        For i = 0 To NoSurf - 2
            F = Asc(BluePrint(i).FromType) - 64
            T = Asc(BluePrint(i).ToType) - 64
            For r = 0 To NoSurf
                If T <> r Or F <> r Then
                    If GenBP(T, r) <> 0 Then
                        GenBP(F, r) = GenBP(F, T) + GenBP(T, r)
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
            GenBP(r, F) = -GenBP(F, r)
          End If
        End If
      Next
    Next
    For i = 1 To NoSurf
      For j = 1 To NoSurf

        If i = j Then
          GenBP(i, j) = 0
        ElseIf i <> j Then
          If GenBP(i, j) = 0 Then
            n = 1
            a = 0
            b = 0
            Do Until a <> 0 And b <> 0
              If GenBP(i, n) <> 0 Then
                a = GenBP(i, n)
                If GenBP(n, j) <> 0 Then
                  b = GenBP(n, j)
                End If
              End If
              n = n + 1
            Loop
            If i < j Then
              GenBP(i, j) = a + b
              GenBP(j, i) = -GenBP(i, j)
            ElseIf i > j Then
              GenBP(i, j) = -(a + b)
              GenBP(j, i) = -GenBP(i, j)
            End If
          End If
        End If
      Next
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
    Next
End Sub
Public Function partONE(ByVal keepPart As String) As String()
    Dim b() As String
    AR1.Clear()
    AR1.AddRange(Split(keepPart, " "))
    AR1.Reverse()
    ReDim b(AR1.Count - 1)
    For j As Integer = 0 To AR1.Count - 1
        b(j) = AR1(j)
    Next
    Return b
End Function
Public Function Factorial(ByVal num As Long) As Long
    Dim result As Long = 1
    For i As Integer = 2 To num
        result *= i
    Next i
    Return result
End Function
Public Function CrePath(ByVal b1 As String) As String
    Dim pathnew As String
    Dim k As Integer
    Dim im As Integer
    Dim b2 As String
    Dim c As String
    c = b1
    For mcount = 0 To UBound(TC)
        If c = TC(mcount).Facecut Then
            k = mcount
        End If
    Next
    im = k
    If im = 0 Then
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
            pathnew = c
        Else
            Do Until im = 0
                pathnew = c
                For j As Integer = im To LBound(TC) Step -1
                    If TC(im).ReferenceCut = TC(j).Facecut Then
                        pathnew = pathnew & " " & TC(j).Facecut
                        im = j
                    End If
                Next
                c = pathnew
            Loop
        End If
        b2 = c
        Return b2
    End Function
End Module
Module Module3
    Public NoSurf As Integer
    Public BPform() As String
    Public BPto() As String
    Public BPbasic() As Integer
    Public BPtolerance() As Integer
    Public StockRemovalTolerance() As String
    Public StockRemovalBasic() As String
    Public ResultTolerance() As String
    Public basicnew() As Single
    Public it() As Integer
    Public it2() As Single
End Module
Public Class Form1
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button1.Click
        Me.Hide()
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        Form2.Show()
    End Sub
End Class
Public Class Form2
    Dim searchValue1 As Integer = -1
    Dim searchValue2 As Integer = -1
    Dim searchValue3 As Integer = -1
    Dim searchValue4 As Integer = -1
    Private m_GridX As Integer = 10
    Private m_GridY As Integer = 10
    Private m_PageWid As Integer = 400
    Private m_PageHgt As Integer = 300
    Private m_ShadowThickness As Integer = 20
    Private m_PenOldLine As Pen = Pens.Black
    Private m_PenNewLine As Pen = Pens.Red
    Private m_PenGrid As Pen = Pens.Black
    Private m_PenRulerNormal As Pen = Pens.Blue
    Private m_PenRulerDrawing As Pen = m_PenNewLine
    Private m_ShowGrid As Boolean = False
    Private m_SnapToGrid As Boolean = False
    Private m_Drawing As Boolean = False
    Private m_X1 As Single
    Private m_Y1 As Single
    Private m_X2 As Single
    Private m_Y2 As Single
    Private m_MouseX As Integer = -10
    Private m_MouseY As Integer = -10
    Private m_Points1() As PointF = {}
    Private m_Points2() As PointF = {}
    Private Pk As Integer = 0
    Private Sub Form2_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
MyBase.Load
        searchvalueNews = searchValue1
        TextBox5.Text = searchvalueNews
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        m_ShowGrid = mnuOptionsShowGrid.Checked
        m_SnapToGrid = mnuOptionsSnapToGrid.Checked
        Dim wid As Integer = Me.ClientSize.Width - picLeftRuler.Left - picLeftRuler.Width - 3
        Dim hgt As Integer = Me.ClientSize.Height - picTopRuler.Top - picTopRuler.Height - 3
        picLeftRuler.Size = New Size(picLeftRuler.Width, 400)
        picTopRuler.Size = New Size(800, picTopRuler.Height)
        picCanvas.Size = New Size(800, 400)
        picLeftRuler.Location = New Point(picLeftRuler.Left, picTopRuler.Top + picTopRuler.Height + 2)
        picTopRuler.Location = New Point(picLeftRuler.Left + picLeftRuler.Width + 2, picTopRuler.Top)
        picCanvas.Location = New Point(picTopRuler.Left, picLeftRuler.Top)
        ReDim Preserve Hole(0)
        Hole(0).XX1 = 0
        Hole(0).YY1 = 0
        Hole(0).XX2 = 0
        Hole(0).YY2 = 0
    End Sub
    Private Sub mnuFileExit_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
mnuFileExit.Click
        Me.Close()
    End Sub
    Private Sub mnuOptionsShowGrid_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles mnuOptionsShowGrid.Click
        m_ShowGrid = Not m_ShowGrid
        mnuOptionsShowGrid.Checked = m_ShowGrid
        picCanvas.Invalidate()
    End Sub
    Private Sub mnuOptionsSnapToGrid_Click(ByVal sender As System.Object, ByVal e As System.EventArgs)
Handles mnuOptionsSnapToGrid.Click
        m_SnapToGrid = Not m_SnapToGrid
        mnuOptionsSnapToGrid.Checked = m_SnapToGrid
    End Sub
    Private Sub picCanvas_MouseDown(ByVal sender As Object, ByVal e As
System.Windows.Forms.MouseEventArgs) Handles picCanvas.MouseDown
        If e.Button <> Windows.Forms.MouseButtons.Left Then Exit Sub
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        m_Drawing = True
        m_X1 = e.X
        m_Y1 = e.Y
        SnapToGrid(m_X1, m_Y1)
        m_X2 = m_X1
        m_Y2 = m_Y1
        Dim a As Integer
        a = m_Y1
        searchValue1 = searchValue1 + 1
        ReDim Preserve NumbersY1(searchValue1)
        NumbersY1(searchValue1) = a / 2
        Pk = 1
        ListBox1.Items.Clear()
        For AddNext = 0 To searchValue1
            ListBox1.Items.Add(NumbersY1(AddNext))
        Next AddNext
        Dim b As Integer
        b = m_X1
        searchValue2 = searchValue2 + 1
        ReDim Preserve NumbersX1(searchValue2)
        NumbersX1(searchValue2) = b / 2
        ListBox2.Items.Clear()
        For AddNext = 0 To searchValue2
            ListBox2.Items.Add(NumbersX1(AddNext))
        Next AddNext
        picCanvas.Invalidate()
        ShowMousePosition(m_X2, m_Y2)
        searchvalueNews = searchValue1
        TextBox5.Text = searchvalueNews
    End Sub
    Private Sub picCanvas_MouseMove(ByVal sender As Object, ByVal e As
System.Windows.Forms.MouseEventArgs) Handles picCanvas.MouseMove
        m_X2 = e.X
        m_Y2 = e.Y
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        SnapToGrid(m_X2, m_Y2)
        ShowMousePosition(m_X2, m_Y2)
        If m_Drawing Then picCanvas.Invalidate()
        Dim g As Graphics
        g = picCanvas.CreateGraphics
        Dim redpen As New Drawing.Pen(Color.Red, 1.5)
        redpen.DashStyle = Drawing2D.DashStyle.DashDot
        Dim whitepen As New Drawing.Pen(Color.WhiteSmoke, 1.5)
        whitepen.DashStyle = Drawing2D.DashStyle.Solid
        Dim blackpen As New Drawing.Pen(Color.Black, 1.5)
        blackpen.DashStyle = Drawing2D.DashStyle.Dot
        If Hcount >= 0 Then
            For jcount As Integer = 0 To Hcount
                g.DrawLine(whitepen, Hole(jcount).XX1 * 2, Hole(jcount).YY1 * 2, Hole(jcount).XX2 * 2,
Hole(jcount).YY2 * 2)
            Next
            For jcount As Integer = 0 To Hcount
                g.DrawLine(redpen, Hole(jcount).XX1 * 2, Hole(jcount).YY1 * 2, Hole(jcount).XX2 * 2,
Hole(jcount).YY2 * 2)
            Next
            For jcount As Integer = 0 To Hcount
                g.DrawLine(blackpen, Hole(jcount).XX1 * 2 - Hole(jcount).SHOLE, Hole(jcount).YY1 * 2,
Hole(jcount).XX2 * 2 - Hole(jcount).SHOLE, Hole(jcount).YY2 * 2)
                g.DrawLine(blackpen, Hole(jcount).XX1 * 2 + Hole(jcount).SHOLE, Hole(jcount).YY1 * 2,
Hole(jcount).XX2 * 2 + Hole(jcount).SHOLE, Hole(jcount).YY2 * 2)
            Next
        End If
    End Sub
    Private Sub picCanvas_MouseUp(ByVal sender As Object, ByVal e As
System.Windows.Forms.MouseEventArgs) Handles picCanvas.MouseUp
        If Not m_Drawing Then Exit Sub
        m_Drawing = False
        ShowMousePosition(m_X2, m_Y2)
        Dim c As Integer
```

```
        c = m_Y2
        searchValue3 = searchValue3 + 1
        ReDim Preserve NumbersY2(searchValue3)
        NumbersY2(searchValue3) = c / 2
        ListBox3.Items.Clear()
        For AddNext = 0 To searchValue3
            ListBox3.Items.Add(NumbersY2(AddNext))
        Next AddNext
        Dim d As Integer
        d = m_X2
        searchValue4 = searchValue4 + 1
        ReDim Preserve NumbersX2(searchValue4)
        NumbersX2(searchValue4) = d / 2
        ListBox4.Items.Clear()
        For AddNext = 0 To searchValue4
            ListBox4.Items.Add(NumbersX2(AddNext))
        Next AddNext
        Dim new_index As Integer = m_Points1.Length
        ReDim Preserve m_Points1(new_index)
        ReDim Preserve m_Points2(new_index)
        m_Points1(new_index) = New PointF(m_X1, m_Y1)
        m_Points2(new_index) = New PointF(m_X2, m_Y2)
        picCanvas.Invalidate()
    End Sub
    Private Sub SnapToGrid(ByRef X As Integer, ByRef Y As Integer)
        If Not m_SnapToGrid Then Exit Sub
        Dim ix As Integer = CInt(X / m_GridX)
        Dim iy As Integer = CInt(Y / m_GridY)
        X = ix * m_GridX
        Y = iy * m_GridY
    End Sub
    Private Sub picCanvas_Paint(ByVal sender As Object, ByVal e As System.Windows.Forms.PaintEventArgs)
Handles picCanvas.Paint
        e.Graphics.Clear(picCanvas.BackColor)
```

```
        If m_ShowGrid Then
            For x As Integer = 0 To picCanvas.ClientSize.Width Step m_GridX
                For y As Integer = 0 To picCanvas.ClientSize.Height Step m_GridY
                    e.Graphics.DrawLine(m_PenGrid, x, y, x + 0.5F, y + 0.5F)
                Next y
            Next x
        End If
        For i As Integer = 0 To m_Points1.Length - 1
            e.Graphics.DrawLine(m_PenOldLine, m_Points1(i), m_Points2(i))
        Next i
        If m_Drawing Then
            e.Graphics.DrawLine(m_PenNewLine, m_X1, m_Y1, m_X2, m_Y2)
        End If
        TextBox1.Text = m_Y1
        TextBox2.Text = m_X1
        TextBox3.Text = m_Y2
        TextBox4.Text = m_X2
        Dim p As New Pen(Color.Blue, 1.5)
        Dim pp As New Pen(Color.Blue, 1.5)
        pp.DashStyle = Drawing2D.DashStyle.DashDot
        p.EndCap = Drawing2D.LineCap.ArrowAnchor
        p.StartCap = Drawing2D.LineCap.RoundAnchor
        e.Graphics.DrawLine(pp, 70, 70, 730, 70)
    End Sub
    Private Sub picTopRuler_Paint(ByVal sender As Object, ByVal e As
System.Windows.Forms.PaintEventArgs) Handles picTopRuler.Paint
        e.Graphics.Clear(picTopRuler.BackColor)
        Dim y1 As Integer = picTopRuler.ClientSize.Height
        Dim y2 As Integer = (2 * picTopRuler.ClientSize.Height) \ 3
        Dim y3 As Integer = picTopRuler.ClientSize.Height \ 3
        Dim y4 As Integer = 0
        Dim x As Integer = 0
        For i As Integer = 0 To picTopRuler.ClientSize.Width \ m_GridX
            If i Mod 10 = 0 Then
```

```
            e.Graphics.DrawLine(m_PenGrid, x, y1, x, y4)
        ElseIf i Mod 5 = 0 Then
            e.Graphics.DrawLine(m_PenGrid, x, y1, x, y3)
        Else
            e.Graphics.DrawLine(m_PenGrid, x, y1, x, y2)
        End If
        x += m_GridX
    Next i
    If m_Drawing Then
        e.Graphics.DrawLine(m_PenRulerDrawing, m_MouseX, y1, m_MouseX, 0)
    Else
        e.Graphics.DrawLine(m_PenRulerNormal, m_MouseX, y1, m_MouseX, 0)
    End If
End Sub
Private Sub picLeftRuler_Paint(ByVal sender As Object, ByVal e As
System.Windows.Forms.PaintEventArgs) Handles picLeftRuler.Paint
    e.Graphics.Clear(picLeftRuler.BackColor)
    Dim x1 As Integer = picLeftRuler.ClientSize.Width
    Dim x2 As Integer = (2 * picLeftRuler.ClientSize.Width) \ 3
    Dim x3 As Integer = picLeftRuler.ClientSize.Width \ 3
    Dim x4 As Integer = 0
    Dim y As Integer = 0
    For i As Integer = 0 To picLeftRuler.ClientSize.Height \ m_GridY
        If i Mod 10 = 0 Then
            e.Graphics.DrawLine(m_PenGrid, x1, y, x4, y)
        ElseIf i Mod 5 = 0 Then
            e.Graphics.DrawLine(m_PenGrid, x1, y, x3, y)
        Else
            e.Graphics.DrawLine(m_PenGrid, x1, y, x2, y)
        End If
        y += m_GridY
    Next i
    If m_Drawing Then
        e.Graphics.DrawLine(m_PenRulerDrawing, x1, m_MouseY, x4, m_MouseY)
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        Else
            e.Graphics.DrawLine(m_PenRulerNormal, x1, m_MouseY, x4, m_MouseY)
        End If
    End Sub

    Private Sub ShowMousePosition(ByVal X As Integer, ByVal Y As Integer)
        m_MouseX = X
        m_MouseY = Y
        picTopRuler.Invalidate()
        picLeftRuler.Invalidate()
    End Sub
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button1.Click
        Dim S1() As Integer
        Dim S2() As Integer
        Dim y1() As Integer
        Dim y2() As Integer
        Dim AR1 As New ArrayList
        Dim AR2 As New ArrayList
        Dim ARY1 As New ArrayList
        Dim ARY2 As New ArrayList
        Dim ndx1 As Integer
        Dim ndx2 As Integer
        Dim k As Integer = 0

        If searchvalueNews = -1 Then
            MessageBox.Show("Drawing Line")
        Else
            DumpArray1(NumbersX1, "Contents of NumberY1()")
            For ndx1 = 0 To UBound(NumbersX1)
                ARY1.Add(NumbersY1(ndx1))
                If Not AR1.Contains(NumbersX1(ndx1)) Then
                    AR1.Add(NumbersX1(ndx1))
                End If
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
Next
   AR1.Sort()
ARY1.Sort()
   ReDim S1(AR1.Count - 1)
   For ndx1 = 0 To AR1.Count - 1
      S1(ndx1) = AR1.Item(ndx1)
    Next
ReDim y1(ARY1.Count - 1)
For ndx1 = 0 To ARY1.Count - 1
   y1(ndx1) = ARY1.Item(ndx1)
Next
For ndx2 = 0 To UBound(NumbersX2)
   ARY2.Add(NumbersY2(ndx2))
   If Not AR2.Contains(NumbersX2(ndx2)) Then
      AR2.Add(NumbersX2(ndx2))
   End If
Next
   AR2.Sort()
ARY2.Sort()
ReDim S2(AR2.Count - 1)
   For ndx2 = 0 To AR2.Count - 1
      S2(ndx2) = AR2.Item(ndx2)
   Next
ReDim y2(ARY2.Count - 1)
For ndx2 = 0 To ARY2.Count - 1
   y2(ndx2) = ARY2.Item(ndx2)
Next
   DumpArray2(S2, "Contents of S2()")
If y1(LBound(y1)) > y2(LBound(y2)) Then
   positionYmax = y1(UBound(y1))
Else
   positionYmax = y2(UBound(y2))
End If
Dim a As Integer
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        Dim r As Integer
        For a = 0 To UBound(S1)
            For r = 0 To UBound(S2)
                If S1(a) = S2(r) Then
                    k = k + 1
                    ReDim Preserve positionX(k - 1)
                    positionX(k - 1) = S2(r)
                End If
            Next
        Next
        NumberOfCut = k
        Me.Close()
        Form3.Show()
    End If
End Sub
Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button2.Click
    If Pk = 1 Then
        Dim g As Graphics
        Hcount = Hcount + 1
        g = picCanvas.CreateGraphics
        ReDim Preserve Hole(Hcount)
        Hole(Hcount).XX1 = NumbersX1(UBound(NumbersX1))
        Hole(Hcount).YY1 = NumbersY1(UBound(NumbersY1))
        Hole(Hcount).XX2 = NumbersX2(UBound(NumbersX2))
        Hole(Hcount).YY2 = NumbersY2(UBound(NumbersY2))
        Hole(Hcount).SHOLE = InputBox("Size Hole :  ")
        ListBox5.Items.Clear()
        For AddNext = 0 To Hcount
            ListBox5.Items.Add(Hole(AddNext).XX1)
        Next AddNext
    End If
End Sub
```

```
    Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button3.Click
        Me.Close()
        Form1.Show()
    End Sub
End Class
Public Class Form3
    Public arr() As String
    Dim index As Integer
    Dim hg As Integer
    Private Sub Form3_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
MyBase.Load
        PictureBox1.Size = New Size(400, 250)
        Label2.Text = NumberOfCut
        Dim k As Integer = NumberOfCut - 1
        Cut = (SelectNames(k))
        ListBox1.Items.Clear()
        For AddNext = 0 To UBound(Cut)
            ListBox1.Items.Add(Cut(AddNext))
        Next AddNext
        For AddNext = 0 To UBound(positionX)
            ListBox2.Items.Add(positionX(AddNext))
        Next AddNext
    End Sub
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button1.Click
        Me.Close()
        Form4.Show()
    End Sub
    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button2.Click
        Me.Close()
        Form2.Show()
    End Sub
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
    Private Sub Button5_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button5.Click
        TextBox2.Text = positionYmax
    End Sub
    Private Sub me_MouseMove(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Me.MouseMove
        Dim g As Graphics
        g = PictureBox1.CreateGraphics
        Dim blackpen As New Drawing.Pen(Color.Black)
        Dim blackpen1 As New Drawing.Pen(Color.Black)
        blackpen1.DashStyle = Drawing2D.DashStyle.Dot
        For ndx2 = 0 To UBound(NumbersX1)
            g.DrawLine(blackpen, NumbersX1(ndx2), NumbersY1(ndx2), NumbersX2(ndx2), NumbersY2(ndx2))
        Next
        For ndx4 = 0 To UBound(positionX)
            g.DrawLine(blackpen1, positionX(ndx4), 35, positionX(ndx4), 400)
        Next
        Dim pp As New Pen(Color.Blue, 1.5)
        Dim drawFont As New Font("Arial", 10)
        Dim drawBrush As New SolidBrush(Color.Black)
        pp.DashStyle = Drawing2D.DashStyle.DashDot
        g.DrawLine(pp, 35, 35, 365, 35)
        For ndx4 = 0 To UBound(Cut)
            g.DrawString(Cut(ndx4), drawFont, drawBrush, positionX(ndx4), 230)
        Next
        Dim redpen As New Drawing.Pen(Color.Red, 1.5)
        redpen.DashStyle = Drawing2D.DashStyle.DashDot
        Dim whitepen As New Drawing.Pen(Color.WhiteSmoke, 1.5)
        whitepen.DashStyle = Drawing2D.DashStyle.Solid
        Dim blackpen2 As New Drawing.Pen(Color.Black, 1.5)
        blackpen2.DashStyle = Drawing2D.DashStyle.Dot
        If Hcount >= 0 Then
            For jcount As Integer = 0 To UBound(Hole)
                g.DrawLine(whitepen, Hole(jcount).XX1, Hole(jcount).YY1, Hole(jcount).XX2, Hole(jcount).YY2)
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
            Next
            For jcount As Integer = 0 To UBound(Hole)
                g.DrawLine(redpen, Hole(jcount).XX1, Hole(jcount).YY1, Hole(jcount).XX2, Hole(jcount).YY2)
            Next
            For jcount As Integer = 0 To UBound(Hole)
                g.DrawLine(blackpen2, Hole(jcount).XX1 - Hole(jcount).SHOLE, Hole(jcount).YY1,
Hole(jcount).XX2 - Hole(jcount).SHOLE, Hole(jcount).YY2)
                g.DrawLine(blackpen2, Hole(jcount).XX1 + Hole(jcount).SHOLE, Hole(jcount).YY1,
Hole(jcount).XX2 + Hole(jcount).SHOLE, Hole(jcount).YY2)
            Next
        End If
    End Sub
End Class
Public Class Form4
    Dim k As Integer = NumberOfCut - 2
    Dim f As Integer = NumberOfCut - 1
    Dim comB1() As ComboBox
    Dim comB2() As ComboBox
    Dim txtF1() As TextBox
    Dim txtF2() As TextBox
    Dim ss As Integer
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button1.Click
        Me.Close()
        Form3.Show()
    End Sub
    Private Sub Form4_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
MyBase.Load
        Label2.Text = NumberOfCut
        Dim Nbox1 As Integer
        Dim Nbox2 As Integer
        Dim Nbox3 As Integer
        Dim Nbox4 As Integer
        Dim i As Integer
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
Nbox1 = k
ReDim comB1(Nbox1)
For i = 0 To Nbox1
    comB1(i) = New ComboBox()
    comB1(i).Height = 22
    comB1(i).Width = 50
    comB1(i).Top = 91 + i * 30
    comB1(i).Left = 100
    comB1(i).Font = New Font(Font.Size, 12)
    Me.Controls.Add(comB1(i))
Next
For addnext = 0 To Nbox1
    For i = 0 To UBound(Cut)
        comB1(addnext).Items.Add(Cut(i))
    Next
Next
Nbox2 = k
ReDim comB2(Nbox2)
For i = 0 To Nbox2
    comB2(i) = New ComboBox()
    comB2(i).Height = 22
    comB2(i).Width = 50
    comB2(i).Top = 91 + i * 30
    comB2(i).Left = 200
    comB2(i).Font = New Font(Font.Size, 12)
    Me.Controls.Add(comB2(i))
Next
For addnext = 0 To Nbox2
    For i = 0 To UBound(Cut)
        comB2(addnext).Items.Add(Cut(i))
    Next
Next
Nbox3 = k
ReDim txtF1(Nbox3)
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        For i = 0 To Nbox3
            txtF1(i) = New TextBox()
            txtF1(i).Height = 22
            txtF1(i).Width = 50
            txtF1(i).Top = 91 + i * 30
            txtF1(i).Left = 300
            txtF1(i).Font = New Font(Font.Size, 12)
            Me.Controls.Add(txtF1(i))
        Next
        Nbox4 = k
        ReDim txtF2(Nbox4)
        For i = 0 To Nbox4
            txtF2(i) = New TextBox()
            txtF2(i).Height = 22
            txtF2(i).Width = 50
            txtF2(i).Top = 91 + i * 30
            txtF2(i).Left = 400
            txtF2(i).Font = New Font(Font.Size, 12)
            Me.Controls.Add(txtF2(i))
        Next
        PictureBox1.Size = New Size(400, positionYmax + 100 + (30 * f))
        PictureBox1.Location = New Point(500, 100)
    End Sub
    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button2.Click
        ReDim BluePrint(k)
        On Error GoTo Err_Handler
            For jcount As Integer = 0 To UBound(BluePrint)
                BluePrint(jcount).FromType = comB1(jcount).SelectedItem
                BluePrint(jcount).ToType = comB2(jcount).SelectedItem
                BluePrint(jcount).Basic = txtF1(jcount).Text
                BluePrint(jcount).Tolerance = txtF2(jcount).Text
            Next
            Me.Close()
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        Form5.Show()
    Exit Sub
Err_Handler:
    MsgBox("Input Data", vbOKOnly, "Error")
  End Sub
  Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button4.Click
    ReDim BluePrint(k)
    For jcount As Integer = 0 To UBound(BluePrint)
      BluePrint(jcount).FromType = comB1(jcount).SelectedItem
      BluePrint(jcount).ToType = comB2(jcount).SelectedItem
      BluePrint(jcount).Basic = txtF1(jcount).Text
      BluePrint(jcount).Tolerance = txtF2(jcount).Text
    Next
    Dim g As Graphics
    Dim Px1 As Integer
    Dim Px2 As Integer
    g = PictureBox1.CreateGraphics
    g.Clear(Color.WhiteSmoke)
    Dim p As New Pen(Color.Blue, 2.5)
    p.EndCap = Drawing2D.LineCap.ArrowAnchor
    p.StartCap = Drawing2D.LineCap.ArrowAnchor
    For gcount As Integer = 0 To UBound(BluePrint)
      Px1 = Asc(BluePrint(gcount).FromType) - 65
      Px2 = Asc(BluePrint(gcount).ToType) - 65
      g.DrawLine(p, positionX(Px1), positionYmax + 70 + (30 * gcount), positionX(Px2), positionYmax + 70
+ (30 * gcount))
    Next
  End Sub
  Private Sub me_MouseMove(ByVal sender As Object, ByVal e As
System.Windows.Forms.MouseEventArgs) Handles Me.MouseMove
    Dim g As Graphics
    g = PictureBox1.CreateGraphics
    Dim blackpen As New Drawing.Pen(Color.Black)
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        Dim blackpen1 As New Drawing.Pen(Color.Black)
        blackpen1.DashStyle = Drawing2D.DashStyle.Dot
        For ndx2 = 0 To UBound(NumbersX1)
            g.DrawLine(blackpen, NumbersX1(ndx2), NumbersY1(ndx2), NumbersX2(ndx2), NumbersY2(ndx2))
        Next
        For ndx4 = 0 To UBound(positionX)
            g.DrawLine(blackpen1, positionX(ndx4), 70, positionX(ndx4), positionYmax + 100 + (30 * f))
        Next
        Dim pp As New Pen(Color.Blue, 1.5)
        Dim drawFont As New Font("Arial", 10)
        Dim drawBrush As New SolidBrush(Color.Black)
        pp.DashStyle = Drawing2D.DashStyle.DashDot
        g.DrawLine(pp, 35, 35, 365, 35)
        For ndx4 = 0 To UBound(Cut)
            g.DrawString(Cut(ndx4), drawFont, drawBrush, positionX(ndx4), positionYmax + 30)
        Next
        Dim redpen As New Drawing.Pen(Color.Red, 1.5)
        redpen.DashStyle = Drawing2D.DashStyle.DashDot
        Dim whitepen As New Drawing.Pen(Color.WhiteSmoke, 1.5)
        whitepen.DashStyle = Drawing2D.DashStyle.Solid
        Dim blackpen2 As New Drawing.Pen(Color.Black, 1.5)
        blackpen2.DashStyle = Drawing2D.DashStyle.Dot
        If Hcount >= 0 Then
            For jcount As Integer = 0 To UBound(Hole)
                g.DrawLine(whitepen, Hole(jcount).XX1, Hole(jcount).YY1, Hole(jcount).XX2, Hole(jcount).YY2)
            Next
            For jcount As Integer = 0 To UBound(Hole)
                g.DrawLine(redpen, Hole(jcount).XX1, Hole(jcount).YY1, Hole(jcount).XX2, Hole(jcount).YY2)
            Next
            For jcount As Integer = 0 To UBound(Hole)
                g.DrawLine(blackpen2, Hole(jcount).XX1 - Hole(jcount).SHOLE, Hole(jcount).YY1,
Hole(jcount).XX2 - Hole(jcount).SHOLE, Hole(jcount).YY2)
                g.DrawLine(blackpen2, Hole(jcount).XX1 + Hole(jcount).SHOLE, Hole(jcount).YY1,
Hole(jcount).XX2 + Hole(jcount).SHOLE, Hole(jcount).YY2)
```

```
        Next
      End If
    End Sub
End Class
Public Class Form5

    Private Sub Form5_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
MyBase.Load
      ListBox1.Items.Add(".....From")
      For kcount As Integer = 0 To UBound(BluePrint)
        ListBox1.Items.Add(BluePrint(kcount).FromType)
      Next
      ListBox1.Items.Add(".....TO1")
      For kcount As Integer = 0 To UBound(BluePrint)
        ListBox1.Items.Add(BluePrint(kcount).ToType)
      Next
      ListBox1.Items.Add(".....Basic1")
      For kcount As Integer = 0 To UBound(BluePrint)
        ListBox1.Items.Add(BluePrint(kcount).Basic)
      Next
      ListBox1.Items.Add(".....TOL1")
      For kcount As Integer = 0 To UBound(BluePrint)
        ListBox1.Items.Add(BluePrint(kcount).Tolerance)
      Next
      BasicArray(NumberOfCut)
    End Sub
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button1.Click
      On Error GoTo Err_Handler
      NumberOfOperation = TextBox1.Text
      Me.Hide()
      Form6.Show()
      Exit Sub
Err_Handler:
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        MsgBox("Input Data", vbOKOnly, "Error")
    End Sub
    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button2.Click
        Me.Hide()
        Form4.Show()
    End Sub
    Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button4.Click
        Dim y, z As Integer
        y = TextBox2.Text
        z = TextBox3.Text
        TextBox4.Text = GenBP(y, z)
    End Sub
     Private Sub TextBox1_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles TextBox1.KeyPress
        Select Case Asc(e.KeyChar)
            Case 48 To 57
                e.Handled = False
            Case 8, 13, 46
                e.Handled = False
            Case Else
                e.Handled = True
                MessageBox.Show("สามารถกดได้แค่ตัวเลข")
        End Select
    End Sub
    Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button3.Click
        TextBox5.Text = TC(2).Machine
    End Sub
End Class
Public Class Form6
    Dim k As Integer = NumberOfOperation
    Dim comB1() As ComboBox
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
    Dim comB2() As ComboBox
    Dim comB3() As ComboBox
    Dim comB4() As ComboBox
    Dim txtF1() As TextBox
    Dim txtF2() As TextBox
    Dim txtF3() As TextBox
    Dim txtF4() As TextBox
    Dim txtF5() As TextBox
    Dim txtF6() As TextBox
    Dim txtF7() As TextBox
    Dim txtF8() As TextBox
    Dim txtF9() As TextBox
    Dim txtF10() As TextBox
    Dim txtF11() As TextBox
    Dim txtF12() As TextBox
    Dim txtF13() As TextBox
    Dim labE1() As Label
    Dim list1 As ListBox
    Dim list2 As ListBox
    Dim list3 As ListBox
    Dim list4 As ListBox
    Dim ss As Integer
    Dim count1() As Integer
    Dim count2() As Integer
    Dim basic2() As Integer
    Dim FromS As String
    Dim ToS As String
    Dim finalP As String
    Private Sub Form6_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
MyBase.Load
        ReDim TC(NumberOfOperation)
        Label21.Text = NumberOfOperation
        Dim i As Integer
        ReDim comB1(UBound(TC))
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
For i = 1 To UBound(TC)
    comB1(i) = New ComboBox()
    comB1(i).Height = 24
    comB1(i).Width = 100
    comB1(i).Top = 240 + i * 28
    comB1(i).Left = 225
    comB1(i).Font = New Font(Font.Size, 10)
    Me.Controls.Add(comB1(i))
Next
For addnext = 1 To UBound(TC)
    For i = 0 To UBound(MC)
        comB1(addnext).Items.Add(MC(i))
    Next
Next
ReDim labE1(UBound(TC))
For i = 1 To UBound(TC)
    labE1(i) = New Label()
    labE1(i).Height = 24
    labE1(i).Width = 32
    labE1(i).Top = 240 + i * 28
    labE1(i).Left = 10
    labE1(i).Font = New Font(Font.Size, 10)
    Me.Controls.Add(labE1(i))
Next
For addnext = 1 To UBound(TC)
    labE1(addnext).Text = (addnext)
Next
ReDim txtF1(UBound(TC))
For i = 1 To UBound(TC)
    txtF1(i) = New TextBox()
    txtF1(i).Height = 24
    txtF1(i).Width = 40
    txtF1(i).Top = 240 + i * 28
    txtF1(i).Left = 40
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
    txtF1(i).Font = New Font(Font.Size, 10)
    Me.Controls.Add(txtF1(i))
Next
  ReDim txtF2(UBound(TC))
For i = 1 To UBound(TC)
    txtF2(i) = New TextBox()
    txtF2(i).Height = 24
    txtF2(i).Width = 45
    txtF2(i).Top = 240 + i * 28
    txtF2(i).Left = 420
    txtF2(i).Font = New Font(Font.Size, 10)
    Me.Controls.Add(txtF2(i))
Next
ReDim txtF3(UBound(TC))
For i = 0 To UBound(TC)
    txtF3(i) = New TextBox()
    txtF3(i).Height = 24
    txtF3(i).Width = 40
    txtF3(i).Top = 240 + i * 28
    txtF3(i).Left = 480
    txtF3(i).Font = New Font(Font.Size, 10)
    Me.Controls.Add(txtF3(i))
Next
ReDim comB2(UBound(TC))
For i = 1 To UBound(TC)
    comB2(i) = New ComboBox()
    comB2(i).Height = 24
    comB2(i).Width = 40
    comB2(i).Top = 240 + i * 28
    comB2(i).Left = 110
    comB2(i).Font = New Font(Font.Size, 10)
    Me.Controls.Add(comB2(i))
Next
For addnext = 1 To UBound(TC)
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
    comB2(addnext).Items.Add("-")
    For i = 0 To UBound(Cut)
        comB2(addnext).Items.Add(Cut(i))
    Next

Next
ReDim comB3(UBound(TC))
For i = 0 To UBound(TC)
    comB3(i) = New ComboBox()
    comB3(i).Height = 24
    comB3(i).Width = 40
    comB3(i).Top = 240 + i * 28
    comB3(i).Left = 170
    comB3(i).Font = New Font(Font.Size, 10)
    Me.Controls.Add(comB3(i))
Next
comB3(0).Items.Add(Cut(LBound(Cut)))
comB3(0).Items.Add(Cut(UBound(Cut)))
For addnext = 1 To UBound(TC)
    comB3(addnext).Items.Add("-")
    For i = 0 To UBound(Cut)
        comB3(addnext).Items.Add(Cut(i))
    Next
Next
ReDim txtF5(UBound(TC))
For i = 1 To UBound(TC)
    txtF5(i) = New TextBox()
    txtF5(i).Height = 24
    txtF5(i).Width = 60
    txtF5(i).Top = 240 + i * 28
    txtF5(i).Left = 540
    txtF5(i).Font = New Font(Font.Size, 10)
    Me.Controls.Add(txtF5(i))
Next
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
ReDim txtF6(UBound(TC))
For i = 1 To UBound(TC)
    txtF6(i) = New TextBox()
    txtF6(i).Height = 24
    txtF6(i).Width = 60
    txtF6(i).Top = 240 + i * 28
    txtF6(i).Left = 615
    txtF6(i).Font = New Font(Font.Size, 10)
    Me.Controls.Add(txtF6(i))
Next
ReDim txtF7(UBound(TC))
For i = 1 To UBound(TC)
    txtF7(i) = New TextBox()
    txtF7(i).Height = 24
    txtF7(i).Width = 60
    txtF7(i).Top = 240 + i * 28
    txtF7(i).Left = 1095
    txtF7(i).Font = New Font(Font.Size, 10)
    Me.Controls.Add(txtF7(i))
Next
ReDim txtF8(UBound(TC))
For i = 1 To UBound(TC)
    txtF8(i) = New TextBox()
    txtF8(i).Height = 24
    txtF8(i).Width = 90
    txtF8(i).Top = 240 + i * 28
    txtF8(i).Left = 1175
    txtF8(i).Font = New Font(Font.Size, 10)
    Me.Controls.Add(txtF8(i))
Next
ReDim comB4(UBound(TC))
For i = 1 To UBound(TC)
    comB4(i) = New ComboBox()
    comB4(i).Height = 24
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        comB4(i).Width = 80
        comB4(i).Top = 240 + i * 28
        comB4(i).Left = 330
        comB4(i).Font = New Font(Font.Size, 10)
        Me.Controls.Add(comB4(i))
    Next
    For addnext = 1 To UBound(TC)
        For i = 0 To UBound(Direction)
            comB4(addnext).Items.Add(Direction(i))
        Next
    Next
        ReDim txtF10(UBound(BluePrint) + 1)
        For i = 1 To (UBound(BluePrint)) + 1
            txtF10(i) = New TextBox()
            txtF10(i).Height = 24
            txtF10(i).Width = 60
            txtF10(i).Top = (290 + (UBound(TC) * 28) + (i * 28))
            txtF10(i).Left = 540
            txtF10(i).Font = New Font(Font.Size, 10)
            Me.Controls.Add(txtF10(i))
        Next
        ReDim txtF11(UBound(BluePrint) + 1)
        For i = 1 To (UBound(BluePrint)) + 1
            txtF11(i) = New TextBox()
            txtF11(i).Height = 24
            txtF11(i).Width = 60
            txtF11(i).Top = (290 + (UBound(TC) * 28) + (i * 28))
            txtF11(i).Left = 615
            txtF11(i).Font = New Font(Font.Size, 10)
            Me.Controls.Add(txtF11(i))
        Next
        ReDim txtF12(UBound(BluePrint) + 1)
        For i = 1 To (UBound(BluePrint)) + 1
            txtF12(i) = New TextBox()
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        txtF12(i).Height = 24
        txtF12(i).Width = 60
        txtF12(i).Top = (290 + (UBound(TC) * 28) + (i * 28))
        txtF12(i).Left = 1095
        txtF12(i).Font = New Font(Font.Size, 10)
        Me.Controls.Add(txtF12(i))
    Next
    ReDim txtF13(UBound(BluePrint) + 1)
    For i = 1 To (UBound(BluePrint)) + 1
        txtF13(i) = New TextBox()
        txtF13(i).Height = 24
        txtF13(i).Width = 90
        txtF13(i).Top = (290 + (UBound(TC) * 28) + (i * 28))
        txtF13(i).Left = 1175
        txtF13(i).Font = New Font(Font.Size, 10)
        Me.Controls.Add(txtF13(i))
    Next
ReDim txtF9(UBound(BluePrint) + 1)
For i = 1 To (UBound(BluePrint)) + 1
    txtF9(i) = New TextBox()
    txtF9(i).Height = 24
    txtF9(i).Width = 185
    txtF9(i).Top = (290 + (UBound(TC) * 28) + (i * 28))
    txtF9(i).Left = 330
    txtF9(i).Font = New Font(Font.Size, 10)
    Me.Controls.Add(txtF9(i))
Next
PictureBox1.Size = New Size(400, (290 + (UBound(TC) * 28) + (UBound(BluePrint) * 28)))
PictureBox1.Location = New Point(680, 60)
Label17.Location = New Point(620, (290 + (UBound(TC) * 28)))
Label18.Location = New Point(1105, (290 + (UBound(TC) * 28)))
Label7.Location = New Point(545, (290 + (UBound(TC) * 28)))
Label2.Location = New Point(568, (270 + (UBound(TC) * 28)))
Label16.Location = New Point(1133, (270 + (UBound(TC) * 28)))
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        Label20.Location = New Point(1195, (290 + (UBound(TC) * 28)))
        countPh = 0
    End Sub
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button1.Click
        Me.Close()
        Form5.Show()
    End Sub
    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button2.Click
        Dim g As Graphics
        g = PictureBox1.CreateGraphics
        g.Clear(Color.WhiteSmoke)
        TC(0).Machine = MC(5)
        For jcount As Integer = 1 To UBound(TC)
            TC(jcount).Machine = comB1(jcount).SelectedItem
        Next
        ReDim count1(UBound(TC))
        For mcount As Integer = 0 To UBound(TC)
            Dim a As Integer = -1
            For kcount As Integer = 0 To mcount
                If TC(mcount).ToType = TC(kcount).ToType Then
                    a = a + 1
                    count1(mcount) = a
                End If
            Next
        Next
        For xcount As Integer = 0 To UBound(TC)
            ListBox2.Items.Add(count1(xcount))
        Next
        For kcount As Integer = 0 To UBound(txtF3)
            If TC(kcount).ToType = "-" Then
                TC(kcount).Facecut = "-"
                txtF3(kcount).Text = "-"
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        Else
            txtF3(kcount).Text = (TC(kcount).ToType & count1(kcount))
            TC(kcount).Facecut = (TC(kcount).ToType & count1(kcount))
            ListBox2.Items.Add(TC(kcount).Facecut)
        End If
    Next
    ReDim count2(UBound(TC))
    For mcount As Integer = 0 To UBound(TC)
        Dim h As Integer = -1
        For kcount As Integer = 0 To mcount
            If TC(mcount).FromType = TC(kcount).ToType Then
                h = h + 1
                count2(mcount) = h
            End If
        Next
    Next

    For kcount As Integer = 0 To UBound(TC)
        If TC(kcount).FromType = "-" Then
            TC(kcount).ReferenceCut = "-"
        Else
            TC(kcount).ReferenceCut = (TC(kcount).FromType & count2(kcount))
            ListBox2.Items.Add(TC(kcount).ReferenceCut)
        End If
    Next
    ReDim basicnew(UBound(TC))
    For jcount As Integer = 1 To UBound(TC)
        If TC(jcount).FromType = "-" Or TC(jcount).ToType = "-" Then
            basicnew(jcount) = 0
        Else
            basicnew(jcount) = Math.Abs(GenBP(Asc(TC(jcount).FromType) - 64, Asc(TC(jcount).ToType) -
64))
        End If
    Next
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        ReDim it(UBound(TC))
        For jcount As Integer = 1 To UBound(TC)
          If txtF2(jcount).Text = "" Then
            If TC(jcount).Machine = "-" Then
              TC(jcount).Tolerance = 0
              txtF6(jcount).Text = TC(jcount).Tolerance.ToString("##0")
            ElseIf TC(jcount).Machine = "Lapping M/C" Then 'IT2
              it(jcount) = 2
              TC(jcount).Tolerance = ((0.45 * (basicnew(jcount) ^ (1 / 3))) + (0.001 * basicnew(jcount))) * (10 ^
((it(jcount) - 16) / 5)) / 2
              txtF6(jcount).Text = TC(jcount).Tolerance.ToString("##0.0000")
            ElseIf TC(jcount).Machine = "Superfinishing M/C" Then 'IT4
              it(jcount) = 4
              TC(jcount).Tolerance = ((0.45 * (basicnew(jcount) ^ (1 / 3))) + (0.001 * basicnew(jcount))) * (10 ^
((it(jcount) - 16) / 5)) / 2
              txtF6(jcount).Text = TC(jcount).Tolerance.ToString("##0.0000")
            ElseIf TC(jcount).Machine = "Diamond turning M/C" Then 'IT4
              it(jcount) = 4
              TC(jcount).Tolerance = ((0.45 * (basicnew(jcount) ^ (1 / 3))) + (0.001 * basicnew(jcount))) * (10 ^
((it(jcount) - 16) / 5)) / 2
              txtF6(jcount).Text = TC(jcount).Tolerance.ToString("##0.0000")
            ElseIf TC(jcount).Machine = "Grinding M/C" Then 'IT5
              it(jcount) = 5
              TC(jcount).Tolerance = ((0.45 * (basicnew(jcount) ^ (1 / 3))) + (0.001 * basicnew(jcount))) * (10 ^
((it(jcount) - 16) / 5)) / 2
              txtF6(jcount).Text = TC(jcount).Tolerance.ToString("##0.0000")
            ElseIf TC(jcount).Machine = "Turning M/C" Then 'IT9
              it(jcount) = 9
              TC(jcount).Tolerance = ((0.45 * (basicnew(jcount) ^ (1 / 3))) + (0.001 * basicnew(jcount))) * (10 ^
((it(jcount) - 16) / 5)) / 2
              txtF6(jcount).Text = TC(jcount).Tolerance.ToString("##0.0000")
            ElseIf TC(jcount).Machine = "Milling M/C" Then 'IT9
              it(jcount) = 9
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
            TC(jcount).Tolerance = ((0.45 * (basicnew(jcount) ^ (1 / 3))) + (0.001 * basicnew(jcount))) * (10 ^
((it(jcount) - 16) / 5)) / 2
            txtF6(jcount).Text = TC(jcount).Tolerance.ToString("##0.0000")
        ElseIf TC(jcount).Machine = "Shaping M/C" Then 'IT10
            it(jcount) = 10
            TC(jcount).Tolerance = ((0.45 * (basicnew(jcount) ^ (1 / 3))) + (0.001 * basicnew(jcount))) * (10 ^
((it(jcount) - 16) / 5)) / 2
            txtF6(jcount).Text = TC(jcount).Tolerance.ToString("##0.0000")
        ElseIf TC(jcount).Machine = "Drilling M/C" Then 'IT11
            it(jcount) = 11
            TC(jcount).Tolerance = ((0.45 * (basicnew(jcount) ^ (1 / 3))) + (0.001 * basicnew(jcount))) * (10 ^
((it(jcount) - 16) / 5)) / 2
            txtF6(jcount).Text = TC(jcount).Tolerance.ToString("##0.0000")
        End If
      Else
        TC(jcount).Tolerance = txtF2(jcount).Text
        txtF6(jcount).Text = TC(jcount).Tolerance
      End If
    Next
    ReDim Path(UBound(TC))
    ReDim ReversePath(UBound(TC))
    ReDim StockRemovalTolerance(UBound(TC))
    ReDim StockRemovalBasic(UBound(TC))
    Dim pathnew(1) As String
    Dim c(1) As String
    Dim k As Integer
    Dim f As String
    Dim num As Integer
    Dim b1 As String
    For gcount As Integer = 1 To UBound(TC)
      If count1(gcount) = 0 Then
        StockRemovalTolerance(gcount) = "Solid Cut"
        StockRemovalBasic(gcount) = 0
        Path(gcount) = "Solid Cut"
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
Else
    k = gcount
    c(0) = TC(k).Facecut
    num = TC(k).Facecut.Substring(1, 1) - 1
    c(1) = TC(k).Facecut.Substring(0, 1) & num
    For i = 0 To 1
        b1 = c(i)
        pathnew(i) = (CrePath(b1))
    Next
    Dim a As String
    Dim d() As String
    For i As Integer = 0 To 1
        a = pathnew(i)
        d = (partONE(a))
        f = String.Join(" ", d)
        ListBox2.Items.Add(c)
        ReversePath(i) = f
    Next
    FromS = ReversePath(0)
    ToS = ReversePath(1)
    Dim strArrayF() As String
    Dim strArrayT() As String
    Dim strArrayR() As String
    strArrayF = Split(FromS, " ")
    strArrayT = Split(ToS, " ")
    Dim ii As Integer
    Dim kk As Integer
    ii = UBound(strArrayF)
    kk = 1
    Do Until kk = 0
        For j = UBound(strArrayT) To LBound(strArrayT) Step -1
            If strArrayF(ii) = strArrayT(j) Then
                finalP = ""
                For kcount As Integer = ii + 1 To UBound(strArrayF)
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
                finalP = finalP & " " & strArrayF(kcount)
            Next
            For jcount As Integer = ii + 1 To UBound(strArrayT)
                finalP = finalP & " " & strArrayT(jcount)
            Next
            kk = 0
          End If
        Next
        ii = ii - 1
      Loop
      finalP = finalP.TrimStart()
      Path(gcount) = finalP
      Dim memoryTol As Single
      memoryTol = 0
      strArrayR = Split(finalP, " ")
      For i = LBound(strArrayR) To UBound(strArrayR)
        For j = 0 To UBound(TC)
          If strArrayR(i) = TC(j).Facecut Then
            memoryTol = memoryTol + TC(j).Tolerance
          End If
        Next
      Next
      StockRemovalTolerance(gcount) = memoryTol.ToString("##0.0000")
      StockRemovalBasic(gcount) = (StockRemovalTolerance(gcount) * 2).ToString("##0.000")
    End If
  Next



  For kcount As Integer = 1 To UBound(Path)
    ListBox3.Items.Add(Path(kcount))
  Next
  For jcount As Integer = 1 To UBound(TC)
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
    txtF7(jcount).Text = StockRemovalBasic(jcount)
Next
For kcount As Integer = 1 To UBound(TC)
    txtF8(kcount).Text = StockRemovalTolerance(kcount)
Next
ReDim DirectionCut(UBound(TC))
For kcount As Integer = 1 To UBound(TC)
    If comB4(kcount).SelectedItem = "-" Then
        DirectionCut(kcount) = 1
    ElseIf comB4(kcount).SelectedItem = "Shoulder" Then
        DirectionCut(kcount) = 1
    ElseIf comB4(kcount).SelectedItem = "Groove" Then
        DirectionCut(kcount) = -1
    End If
Next
countPh = 1
Dim ci(1) As String
Dim Bak() As Single
ReDim Bak(UBound(TC))
Dim bas() As Single
ReDim bas(UBound(TC))
For i As Integer = UBound(TC) To 1 Step -1
    ci(0) = TC(i).FromType
    ci(1) = TC(i).ToType
    For j As Integer = 0 To 1
        For mcount = (i + 1) To UBound(TC)
            If ci(j) = TC(mcount).ToType Then
                Bak(i) = Bak(i) + ((StockRemovalBasic(mcount)) * (DirectionCut(mcount)))
            End If
        Next
    Next
Next

For kcount As Integer = 1 To UBound(TC)
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        ListBox4.Items.Add(Bak(kcount))
    Next
    For kcount As Integer = 1 To UBound(TC)
        bas(kcount) = basicnew(kcount) + Bak(kcount)
        txtF5(kcount).Text = bas(kcount).ToString("##0.000")
    Next
End Sub
Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Button4.Click
    Dim K As Integer
    Dim num1() As Integer
    Dim num2() As Integer
    Dim FromS As String
    Dim ToS As String
    Dim finalP1 As String
    Dim kk As Integer
    Dim xx As Integer
    K = NumberOfCut - 2
    ReDim num1(K)
    ReDim num2(K)
    For i As Integer = 0 To K
        kk = UBound(TC)
        Do Until kk = -1
            If BluePrint(i).FromType = TC(kk).ToType Then
                num1(i) = kk
                BluePrint(i).Facecut = TC(num1(i)).Facecut
                kk = -1
            Else
                kk = kk - 1
            End If
        Loop

        xx = UBound(TC)
        Do Until xx = -1
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        If BluePrint(i).ToType = TC(xx).ToType Then
            num2(i) = xx
            BluePrint(i).ReferenceCut = TC(num2(i)).Facecut
            xx = -1
        Else
            xx = xx - 1
        End If
    Loop
Next
Dim N As Integer
N = NumberOfCut - 2
ReDim PathResult(N)
ReDim ReversePathResult(N)
ReDim ResultTolerance(N)
Dim pathnew(1) As String
Dim c(1) As String
Dim k1 As Integer
Dim f As String
Dim b1 As String
For gcount As Integer = 0 To N
    k1 = gcount
    c(0) = BluePrint(k1).Facecut
    c(1) = BluePrint(k1).ReferenceCut
    For i = 0 To 1
        b1 = c(i)
        pathnew(i) = (CrePath(b1))
    Next
    Dim a As String
    Dim d() As String
    For i As Integer = 0 To 1
        a = pathnew(i)
        d = (partONE(a))
        f = String.Join(" ", d)
        ReversePathResult(i) = f
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
Next
FromS = ReversePathResult(0)
ToS = ReversePathResult(1)
Dim strArrayF() As String
Dim strArrayT() As String
Dim strArrayR() As String
strArrayF = Split(FromS, " ")
strArrayT = Split(ToS, " ")
Dim ii As Integer
Dim kk1 As Integer
ii = UBound(strArrayF)
kk1 = 1
Do Until kk1 = 0
   For j = UBound(strArrayT) To LBound(strArrayT) Step -1
      If strArrayF(ii) = strArrayT(j) Then
         finalP = ""
         For kcount As Integer = ii + 1 To UBound(strArrayF)
            finalP = finalP & " " & strArrayF(kcount)
         Next
         For jcount As Integer = ii + 1 To UBound(strArrayT)
            finalP = finalP & " " & strArrayT(jcount)
         Next
         kk1 = 0
      End If
   Next
   ii = ii - 1
Loop
finalP1 = finalP.TrimStart()
PathResult(gcount) = finalP1
Dim memoryTol As Single
memoryTol = 0
strArrayR = Split(finalP1, " ")
For i = LBound(strArrayR) To UBound(strArrayR)
   For j = 0 To UBound(TC)
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
            If strArrayR(i) = TC(j).Facecut Then
                memoryTol = memoryTol + TC(j).Tolerance
            End If
        Next
    Next
    ResultTolerance(gcount) = memoryTol.ToString("##0.0000")
Next
For kcount As Integer = 0 To UBound(BluePrint)
    If ResultTolerance(kcount) - 0.001 < BluePrint(kcount).Tolerance Then
        txtF13(kcount + 1).ForeColor = Color.Black
    Else
        txtF13(kcount + 1).ForeColor = Color.Red
    End If
    txtF10(kcount + 1).Text = BluePrint(kcount).Basic
    txtF11(kcount + 1).Text = BluePrint(kcount).Tolerance
    txtF12(kcount + 1).Text = BluePrint(kcount).Basic
    txtF13(kcount + 1).Text = ResultTolerance(kcount)
Next
For ncount As Integer = 1 To N + 1
    txtF9(ncount).Text = PathResult(ncount - 1)
Next
End Sub


Private Sub me_mousemove(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Me.MouseMove
    Dim g As Graphics
    g = PictureBox1.CreateGraphics
    Dim blackpen As New Drawing.Pen(Color.Black)
    Dim blackpen1 As New Drawing.Pen(Color.Black)
    blackpen1.DashStyle = Drawing2D.DashStyle.Dot
    For ndx2 = 0 To UBound(NumbersX1)
        g.DrawLine(blackpen, NumbersX1(ndx2), NumbersY1(ndx2), NumbersX2(ndx2), NumbersY2(ndx2))
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        Next
        For ndx4 = 0 To UBound(positionX)
            g.DrawLine(blackpen1, positionX(ndx4), 35, positionX(ndx4), (280 + (UBound(TC) * 28)) +
(UBound(BluePrint) * 28))
        Next
        Dim pp As New Pen(Color.Blue, 1.5)
        Dim drawFont As New Font("Arial", 10)
        Dim drawBrush As New SolidBrush(Color.Black)
        pp.DashStyle = Drawing2D.DashStyle.DashDot
        g.DrawLine(pp, 35, 35, 365, 35)
        For ndx4 = 0 To UBound(Cut)
            g.DrawString(Cut(ndx4), drawFont, drawBrush, positionX(ndx4), 185)
        Next
        Dim redpen As New Drawing.Pen(Color.Red, 1.5)
        redpen.DashStyle = Drawing2D.DashStyle.DashDot
        Dim whitepen As New Drawing.Pen(Color.WhiteSmoke, 1.5)
        whitepen.DashStyle = Drawing2D.DashStyle.Solid
        Dim blackpen2 As New Drawing.Pen(Color.Black, 1.5)
        blackpen2.DashStyle = Drawing2D.DashStyle.Dot
        If Hcount >= 0 Then
            For jcount As Integer = 0 To UBound(Hole)
                g.DrawLine(whitepen, Hole(jcount).XX1, Hole(jcount).YY1, Hole(jcount).XX2, Hole(jcount).YY2)
            Next
            For jcount As Integer = 0 To UBound(Hole)

                g.DrawLine(redpen, Hole(jcount).XX1, Hole(jcount).YY1, Hole(jcount).XX2, Hole(jcount).YY2)
            Next
            For jcount As Integer = 0 To UBound(Hole)
                g.DrawLine(blackpen2, Hole(jcount).XX1 - Hole(jcount).SHOLE, Hole(jcount).YY1,
Hole(jcount).XX2 - Hole(jcount).SHOLE, Hole(jcount).YY2)
                g.DrawLine(blackpen2, Hole(jcount).XX1 + Hole(jcount).SHOLE, Hole(jcount).YY1,
Hole(jcount).XX2 + Hole(jcount).SHOLE, Hole(jcount).YY2)
            Next
        End If
```

มหาวิทยาลัยเทคโนโลยีสุรนารี

```
        Dim Px1 As Integer
        Dim Px2 As Integer
        Dim p As New Pen(Color.Blue, 2.5)
        p.EndCap = Drawing2D.LineCap.ArrowAnchor
        p.StartCap = Drawing2D.LineCap.ArrowAnchor
        For gcount As Integer = 0 To UBound(BluePrint)
            Px1 = Asc(BluePrint(gcount).FromType) - 65
            Px2 = Asc(BluePrint(gcount).ToType) - 65
            g.DrawLine(p, positionX(Px1), 263 + (UBound(TC) * 28) + (28 * gcount), positionX(Px2), 263 +
(UBound(TC) * 28) + (28 * gcount))
        Next
        Dim Pxh1 As Integer
        Dim Pxh2 As Integer
        Dim ph As New Pen(Color.Red, 2.5)
        ph.EndCap = Drawing2D.LineCap.ArrowAnchor
        ph.StartCap = Drawing2D.LineCap.NoAnchor
        Dim drawFont1 As New Font("Arial", 8)
        If countPh = 1 Then
            For gcount As Integer = 1 To UBound(TC)
                Pxh1 = Asc(TC(gcount).FromType) - 65
                Pxh2 = Asc(TC(gcount).ToType) - 65
                g.DrawLine(ph, positionX(Pxh1), 183 + (28 * gcount), positionX(Pxh2), 183 + (28 * gcount))
                If positionX(Pxh1) > positionX(Pxh2) Then
                    g.DrawString(gcount, drawFont1, drawBrush, positionX(Pxh1) - (((positionX(Pxh1) -
positionX(Pxh2)) / 2)), 170 + (28 * gcount))
                Else
                    g.DrawString(gcount, drawFont1, drawBrush, positionX(Pxh2) - (((positionX(Pxh2) -
positionX(Pxh1)) / 2)), 170 + (28 * gcount))
                End If
            Next
        End If
    End Sub
End Class
```

# ภาคผนวก จ

## บทความวิชาการที่ได้รับการตีพิมพ์เผยแพร่

มหาวิทยาลัยเทคโนโลยีสุรนารี

## รายชื่อบทความวิชาการที่ได้รับการตีพิมพ์เผยแพร่

คงเดช คำศรี และ ยงยุทธ เสริมสุธีอนุวัฒน์ (2559). **การใช้คอมพิวเตอร์ช่วยสร้างแผนภูมิความคลาดเคลื่อนสำหรับวางแผนการจัดลำดับการตัดโลหะ**. การประชุมวิชาการระดับชาติ ครั้งที่ 13, 8-9 ธันวาคม 2559, มหาวิทยาลัยเกษตรศาสตร์ วิทยาเขตกำแพงแสน, ประเทศไทย

มหาวิทยาลัยเทคโนโลยีสุรนารี

# การใช้คอมพิวเตอร์ช่วยสร้างแผนภูมิความคลาดเคลื่อนสำหรับวางแผนการจัดลำดับการตัดโลหะ

## Computer-aided tolerance charting for planning machining sequence

**คงเดช คำศรี**[1], ยงยุทธ เสริมสุธีอนุวัฒน์[1]

Khongdet Khumsri[1], Yongyooth Sermsuti-anuwat[1]

### บทคัดย่อ

ในอุตสาหกรรมการผลิตชิ้นงานด้วยกรรมวิธีตัดโลหะ (machining) การวางแผนกระบวนการผลิตมีส่วนสำคัญอย่างมากต่อความถูกต้องของผลผลิตตามข้อกำหนด แผนภูมิความคลาดเคลื่อน (tolerance chart) เป็นเครื่องมือที่ใช้ตรวจสอบความเหมาะสมของขั้นตอนการตัดโลหะกับข้อกำหนดในแบบของชิ้นงานก่อนทำการผลิตจริง บทความนี้กล่าวถึงการพัฒนาโปรแกรมช่วยสร้างแผนภูมิความคลาดเคลื่อนด้วย Visual Basic โดยมีจุดมุ่งหมายเพื่อให้ได้โปรแกรมที่สามารถปรับเปลี่ยนขั้นตอนการตัดโลหะอย่างอัตโนมัติ การคำนวณในโปรแกรมอาศัย Rooted-Tree Graph เป็นหลัก แม้ว่าการพัฒนายังไม่เสร็จสิ้นอย่างสมบูรณ์ แต่โปรแกรมก็สามารถแสดงผลลัพธ์ของแผนภูมิความคลาดเคลื่อนได้อย่างถูกต้อง และสามารถใช้ประเมินขั้นตอนของการตัดโลหะเพื่อสนองความต้องในแบบวิศวกรรมได้

คำสำคัญ: แผนภูมิความคลาดเคลื่อน, การวิเคราะห์ความคลาดเคลื่อน, ความคลาดเคลื่อนสะสม, แผนภาพต้นไม้

### Abstract

A tolerance chart is a graphical representation of a machining sequence for producing a machined part. The technique has used for more than half a century in aircraft and car manufacturing industries for process planning prior to machining an intricate part. However, due to its error proneness and time consumption in charting, many attempts have been made on computerized tolerance charting. This paper describes another attempt basing the charting calculations on the rooted-tree graph concept. Although, at present, the program is under the developing stage, it is able to produce all the tolerance stacks and verify the practical feasibility of a machining sequence.

Keywords: tolerance charts, tolerance analysis, tolerance stacks, rooted-tree graph

E-mail: khongdet_87@hotmail.com

---

[1] สาขาวิชาวิศวกรรมอุตสาหการ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี นครราชสีมา 30000

[1] Industrial Engineering, Faculty of Engineering, Suranaree University of Technology, NakhonRatchasima 3000

มหาวิทยาลัยเทคโนโลยีสุรนารี

## 1.บทนำ

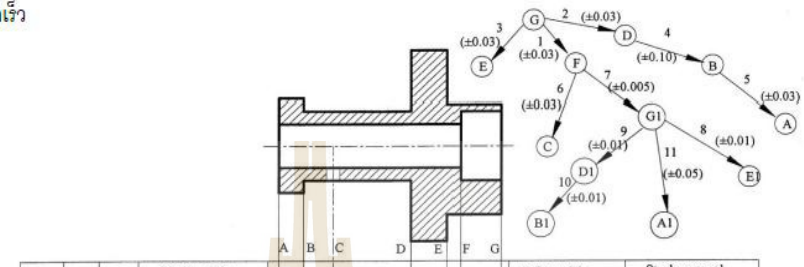
ในอุตสาหกรรมการผลิตชิ้นงานด้วยกรรมวิธีตัดโลหะ (machining) การวางแผนกระบวนการผลิตมีส่วนสำคัญต่อผลผลิตให้เป็นไปตามข้อกำหนด แผนภูมิความคลาดเคลื่อน (Tolerance Chart) เป็นเครื่องมือสำคัญอย่างหนึ่งสำหรับตรวจสอบความเหมาะสมของขั้นตอนการตัดโลหะอย่างมีประสิทธิผล โดยเฉพาะอย่างยิ่งในกรณีของการตัดโลหะที่ซับซ้อน และมีการเปลี่ยนแปลงการจับชิ้นงานหรือเปลี่ยนวิธีการตัดโลหะหลายๆครั้ง ซึ่งเป็นสาเหตุให้เกิดคลาดคลาดเคลื่อนสะสม แผนภูมิความคลาดเคลื่อนสามารถชี้ให้เห็นถึงปัญหาความคลาดเคลื่อนสะสมแม้ว่าการตัดโลหะจริงจะยังไม่เกิดขึ้นก็ตาม

แผนภูมิความคลาดเคลื่อน (tolerance charts) เป็นแผนภาพที่แสดงขั้นตอนการตัดโลหะอย่างเป็นระบบ โดยแสดงให้เห็นถึงการเปลี่ยนแปลงของขนาด (dimension), ส่วนของโลหะที่ถูกตัดออกไปในแต่ละขั้นตอนการตัด และความคลาดเคลื่อนของมิติที่เป็นผลมาจากการตัดโลหะ ตั้งแต่การตัดโลหะครั้งแรกจนถึงครั้งสุดท้าย เทคนิคการวิเคราะห์ความคลาดเคลื่อนชนิดนี้ เป็นที่รู้จักกันมาไม่น้อยกว่า 60 ปี ในหนังสือของ Doyle (1950) [1] แสดงการใช้แผนภูมิแผนภูมิความคลาดเคลื่อนกับการตัดเฟืองฟันตรง (spur gear) อย่างละเอียด

การวิเคราะห์ความคลาดเคลื่อนด้วยคอมพิวเตอร์เริ่มเป็นงานวิจัยแพร่หลาย เมื่อ Bjork (1978) [2] รายงานการสร้างโปรแกรมคำนวณความคลาดเคลื่อนสะสม (tolerance stacks) ด้วยคอมพิวเตอร์ขนาดใหญ่ (mainframe computer) Bjork ใช้ Tolerance Chain สร้างความสัมพันธ์ระหว่างความคลาดเคลื่อนของมิติ แต่ในช่วงเวลานั้นคอมพิวเตอร์ยังมีขีดความสามารถจำกัด ต่อมาในปี 1984 บทความของ Ahaluwalia และ Karolin [3] รายงานการพัฒนาโปรแกรมคอมพิวเตอร์สำหรับสร้างแผนภูมิความคลาดเคลื่อน ซึ่งถือได้ว่าเป็นจุดเริ่มต้นของการใช้คอมพิวเตอร์วิเคราะห์ความคลาดเคลื่อนด้วยหลักการของแผนภูมิความคลาดเคลื่อน

เนื่องจากการคำนวณในแผนภูมิความคลาดเคลื่อนเกิดความผิดพลาดได้ง่าย Gadzala (1959) [4] จึงเสนอวิธี Method of Traces ประกอบการคำนวณหาความคลาดเคลื่อนสะสม; ส่วน O.R. Wade (1967) [5] อาศัย Balance dimensions เป็นหลักในการคำนวณความคลาดเคลื่อน อย่างไรก็ตาม, ถึงแม่ว่าวิธีทั้งสองจะสามารถลดความผิดพลาดจากการคำนวณได้ แต่ยังคงเสียเวลามากในการตรวจสอบ ต่อมาในปี 1990 Whybrew และคณะ ได้พัฒนาวิธีการใช้กราฟต้นไม้ (Rooted-Tree Graph) ช่วยสร้างแผนภูมิความคลาดเคลื่อน ซึ่งเป็นวิธีที่นอกจากจะง่ายต่อการทำความเข้าในและไม่ยุ่งยากซับซ้อนแล้ว ยังมีลักษณะเป็นระบบที่ใช้ตรวจสอบความผิดพลาดได้ง่าย [6] และล่าสุด Chalalai (2016) ได้ใช้เทคนิคกราฟต้นไม้ (rooted-tree graph) เป็นพื้นฐานในการสร้างแผนภูมิแผนภูมิความคลาดเคลื่อนบนโปรแกรมตารางคำนวณ (spreadsheet) [7] ซึ่งโปรแกรมดังกล่าว สามารถแสดงผลลัพธ์เป็นแผนภูมิความคลาดเคลื่อน และแก้ไขเปลี่ยนแปลงความคาดเคลื่อนที่เกี่ยวข้องได้ นอกจากนี้ยังใช้ graphics functions ของโปรแกรมตารางคำนวณในการรับค่า Input และแสดงผล Output เป็นแผนภูมิความคลาดเคลื่อน; แต่เนื่องจากเซลล์ในตารางคำนวณนั้นถูกจำกัดด้วยฟังก์ชั่น

ทำให้ความสามารถในการเปลี่ยนแปลงขั้นตอนในการตัดโลหะ เช่นการเพิ่มหรือลดงานตัดโลหะเป็นไปได้ อย่างยากลำบาก บทความนี้จึงรายงานการพัฒนาโปรแกรมสร้างแผนภูมิความคาดเคลื่อนด้วยให้สามารถมีปฏิสัมพันธ์ กับผู้ใช้ได้มากขึ้นเพื่อปรับเปลี่ยนลำดับของการตัดโลหะและ/หรือ เงื่อนไขของการตัดโลหะได้อย่างรวดเร็ว




| Op. no. | Face cut | M/C used | Working Dim. Basic | Working Dim. ±Tol./2 | Balance Dim. Basic | Balance Dim. ±Tol./2 | Stock removal Basic | Stock removal ±Tol./2 |
|---|---|---|---|---|---|---|---|---|
| 10 | F | | 20.15 | 0.03 | | | Solid cut | |
| 20 | D | | 45.54 | 0.03 | | | Solid cut | |
| 30 | E | | 25 | 0.03 | | | Solid cut | |
| 40 | B | | 59.30 | 0.10 | | | Solid cut | |
| 50 | A | | 5.90 | 0.03 | | | Solid cut | |
| 60 | C | | 55 | 0.03 | | | Solid cut | |
| 70 | G1 | | 19.85 | 0.005 | | | 0.30 | 0.035 |
| 80 | E1 | | 25 | 0.01 | | | 0.30 | 0.075 |
| 90 | D1 | | 44.94 | 0.01 | | | 0.30 | 0.075 |
| 100 | B1 | | 60.06 | 0.01 | | | 0.40 | 0.215 |
| 110 | A1 | | 110 | 0.05 | | | 0.50 | 0.245 |
| | | | Blue print Dim. | | Resultants | | | |
| | | | 19.94 | 0.06 | 19.94 | 0.02 | | |
| | | | 29.91 | 0.09 | 29.91 | 0.045 | | |
| | | | 25 | 0.30 | 25 | 0.01 | | |
| | | | 55 | 0.03 | 55 | 0.03 | | |
| | | | 5 | 0.30 | 5 | 0.03 | | |
| | | | 110 | 0.30 | 110 | 0.05 | | |

รูปที่ 1 แสดงแผนภูมิความคลาดเคลื่อน

**2. องค์ประกอบของแผนภูมิความคลาดเคลื่อน (Components of a tolerance chart)**

ในรูปที่ 1 แสดงส่วนต่างๆของแผนภูมความคลาดเคลื่อน ซึ่งประกอบด้วยส่วนสำคัญ ดังนี้:

1. ภาพสเกตช์ ของชิ้นงานซึ่งอยู่ตอนบนของแผนภูมิฯ ในกรณีที่เป็นชิ้นงานสมมาตรรอบแกนในแนวนอนอาจจะใช้เป็นภาพครึ่งส่วน (half sketch) ก็ได้

2. เส้นดิ่งจากภาพสเก็ตช์ แสดงผิวชิ้นงาน มีรหัสชื่อผิวเป็นตัวอักษร เช่น A, B, C, D, E, F และ G เป็นต้น

3. ขั้นตอนหรือลำดับของการตัดโลหะ แสดงอยู่ที่บริเวณตอนกลางของแผนภูมิ ใช้ลูกศรเป็นสัญลักษณ์แทนการตัด (machining cut) แต่ละครั้ง: หัวลูกศรกำหนดผิวที่ถูกตัด (face cut); หางลูกศรเป็นวงกลม แสดงผิวอ้างอิงในการตัด; และหมายเลขที่ลูกศรแต่ละเส้น คือเลขลำดับการตัด (cut no.)

4. รายละเอียดเกี่ยวกับการตัดโลหะแต่ละครั้ง (cut) อยู่ใน 4 สดมภ์ซ้ายสุดของแผนภูมฯ ได้แก่: หมายเลขงาน (operation no), รหัสชื่อผิวที่เกิดจากการตัด (face cut), เครื่องมือกลที่ใช้ (M/C used – แต่ ในรูปที่ 1 ไม่ได้แสดงรายละเอียดนี้), และ ขนาดมูลฐานและความคลาดเคลื่อนของการตัดแต่ละครั้ง (working dimensions)

5. สภมภ์ขวามือสุด 2 สดมภ์ แสดงขนาดมูลฐาน (basic size) และความคลาดเคลื่อน (±Tol./2) โลหะที่ถูกตัดออกไป (stock removal) ในแต่ละครั้ง

ถัดจาก 2 สดมภ์นี้เข้ามาด้านใน เป็นสดมภ์ของ Balance dimensions ที่อาจจำเป็นต้องใช้ หรือไม่ก็ได้ ขึ้นอยู่กับวิธีการคำนวณความคลาดเคลื่อนสะสม

6. สดมภ์ Blueprint Dim.ซึ่งอยู่ด้านล่างซ้ายของแผนภูมิ แสดงข้อกำหนดของมิติความยาวตามแบบวิศวกรรมของชิ้นงาน โดยทั่วไปจะใช้แผนภูมิฯกับมิติความยาวเท่านั้น

7. ผลของการคำนวณจากแผนภูมิจะอยู่ใน สดมภ์ Resultants อยู่ทางด้านล่างขวามือ ซึ่งประกอบด้วย สดมภ์ย่อยของ ขนาดมูลฐาน (basic size) และค่าความคลาดเคลื่อน (±Tol./2)

จะสังเกต เห็นได้ว่า ตรงกลางระหว่าง สดมภ์ Blueprint และ Resultants ใช้สัญลักษณ์ dot-to-dot แทนผิวที่เป็นผลลัพธ์จากขั้นตอนของการตัดโลหะทั้งหมด

โดยทั่วไป การสร้างแผนภูมิความคลาดเคลื่อนจะเริ่มต้นจาก: (1) สเก็ตช์ภาพชิ้นงาน; (2) ลากเส้นดิ่ง (A, B, C, ...); (3) ใส่ข้อมูลของการตัดแต่ละครั้งในสดมภ์ด้านซ้ายมือของแผนภูมิทั้งหมด ยกเว้นขนาดมูลฐาน (Basic Working Dim.); (4) คำนวณค่าความคลาดเคลื่อนของ stock ที่ถูกตัดออกไป; (5) กำหนดขนาดมูลฐานของ stock removal ที่เหมาะสม; (6) คำนวณความคลาดเคลื่อนสะสมใน สดมภ์ Resultants; และ (7) ย้อนกลับไปคำนวณหาขนาดมูลฐานของการตัดแต่ละครั้ง (Basic Working Dim.)

**3. แผนภาพต้นไม้กับแผนภูมความคลาดเคลื่อน**

ในรูปที่ 1 แสดงแผนภาพต้นไม้ที่ตอนบนของแผนภูมิฯเพื่อใช้ประกอบการคำนวณด้วยมือ ซึ่งรายละเอียดของวิธีการสร้างแผนภาพชนิดนี้ปรากฏอยู่ในบทความของ Whybrew และคณะ (1990) แล้ว [6] ในที่นี้จะสรุปวิธีการดังกล่าวและวิธีใช้งานเพียงสั้นๆดังนี้:

เริ่มต้นด้วยการเขียนโหนด (node) แรกของผิวที่ถูกตัดครั้งแรก; จากนั้นลากลูกศร อ้างอิงจากผิวแรกนี้ ไปยังผิวที่ถูกตัด; ถ้าผิวที่ถูกตัดใหม่ เป็นผิวอ้างอิงของการตัดต่อๆไป, การลากเส้นลูกศร จะดำเนินต่อไป จนครบรอยตัดทั้งหมด ในการเขียนกราฟชนิดนี้ มีหลักที่สำคัญอย่างหนึ่งคือ ผิวที่เกิดจากการตัดซ้ำ จะถูกระบุด้วยตัวเลขห้อยท้าย เช่น G1 เกิดจากการตัดซ้ำครั้งที่ 1 เป็นต้น

เราสามารถนำแผนภาพต้นไม้ (rooted-tree graph) ไปใช้เป็นพื้นฐานในการคำนวณหลายอย่างในแผนภูมิฯ [8] ตัวอย่างเช่น การคำนวณค่าความคลาดเคลื่อนสะสมของ Stock removal ที่เกิดจากการตัดครั้งที่ 7, คือผลรวมของค่าความคลาดเคลื่อนในเส้นของกราฟต้นไม้ ระหว่างโหนด G และ G1 = ±(0.03 + 0.005) = ±0.035 (ดูรูปที่ 2 ประกอบ); หรือ ค่าคลาดเคลื่อนสะสมของ Resultants ผิว D กับ E จะมีค่าเท่ากับ ผลรวมของความคลาดเคลื่อนในเส้นกราฟระหว่าง โหนด D1 และ E1 = ±(0.01 + 0.01) = ± 0.02 เป็นต้น.

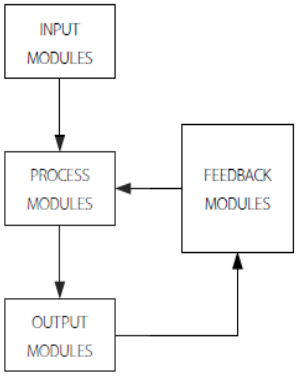
**4. โครงสร้างของโปรแกรม**

โปรแกรมแผนภูมิความคลาดเคลื่อน (Tolerance Charting Program) เขียนด้วยชุดคำสั่งของโปรแกรม

Microsoft Visual Studio ซึ่งเป็นชุดพัฒนาโปรแกรม ที่สะดวกต่อการใช้สร้างโปรแกรมสำเร็จรูปบนระบบปฏิบัติการวินโดวส์

โปรแกรมแผนภูมิความคลาดเคลื่อนที่ได้รับการพัฒนา มีส่วนประกอบหลัก 4 ส่วนคือ:

(1) **กลุ่มโมดูลรับข้อมูล** (Input Modules): ทำหน้าที่เกี่ยวกับการรับข้อมูล สำหรับการสร้างแผนภูมิฯ และเก็บไว้ในตัวแปรต่างๆเพื่อประมวลผลในแต่ละขั้นตอน ต่อไปในกลุ่มโมดูลนี้แบ่งเป็นกลุ่มโมดูลย่อย 2 กลุ่มคือ: กลุ่มแรกรับค่าต่างๆจากภาพสเกตช์ชิ้นงาน ได้แก่: รหัสตัวอักษรกำกับผิวงานในแนวดิ่ง, ขนาดของมิติตามแบบของชิ้นงาน (ขนาดมูลฐานและความคลาดเคลื่อน), และ ผิวที่กำหนดขนาดของแต่ละมิติ; ส่วนกลุ่มโมดูลย่อยที่ 2 ทำหน้าที่รับค่าข้อมูลที่เกี่ยวข้องกับการสร้างแผนภูมิความคลาดเคลื่อนเช่น จำนวนขั้นตอนในการตัด เครื่องมือกลที่ใช้ในการผลิตของแต่ละขั้นตอน และผิวอ้างอิงและผิวที่ถูกตัดในแต่ละครั้งของการตัด




รูปที่ 3 โครงสร้างโปรแกรมแผนภูมิความคลาดเคลื่อน

(2) **กลุ่มโมดูลประมวลผล** (Processing Modules) มีหน้าที่ประมวลผลตามขั้นตอนของการสร้างแผนภูมิฯ และเป็นกลุ่มโมดูลหลักของโปรแกรม ซึ่งประกอบไปด้วยโมดูลย่อยๆที่ทำหน้าที่ต่างๆกัน นับตั้งแต่การเรียกตัวแปรมาใช้, การกำหนดรหัสผิวชิ้นงานใหม่จากข้อมูล input, สร้างเส้นทางของ rooted tree สำหรับความคลาดเคลื่อนของ stock removal และของ resultant dimensions, จนไปถึงการหาขนาดมูลฐานของ working dimensions. นอกจากนี้ยังประกอบด้วย utility modules อื่นๆ เช่น กำหนดค่าความคลาดเคลื่อนสำหรับการตัดแต่ละครั้ง เป็น IT Grade Number (ในกรณีที่ไม่มีข้อมูลส่วนนี้จากผู้ใช้ หรือผู้ใช้ต้องการให้โปรแกรมกำหนดให้) และโมดูลแสดงความสัมพันธ์ของรอยตัด (cut) เพื่อให้ผู้ใช้ปรับความคลาดเคลื่อน เป็นต้น

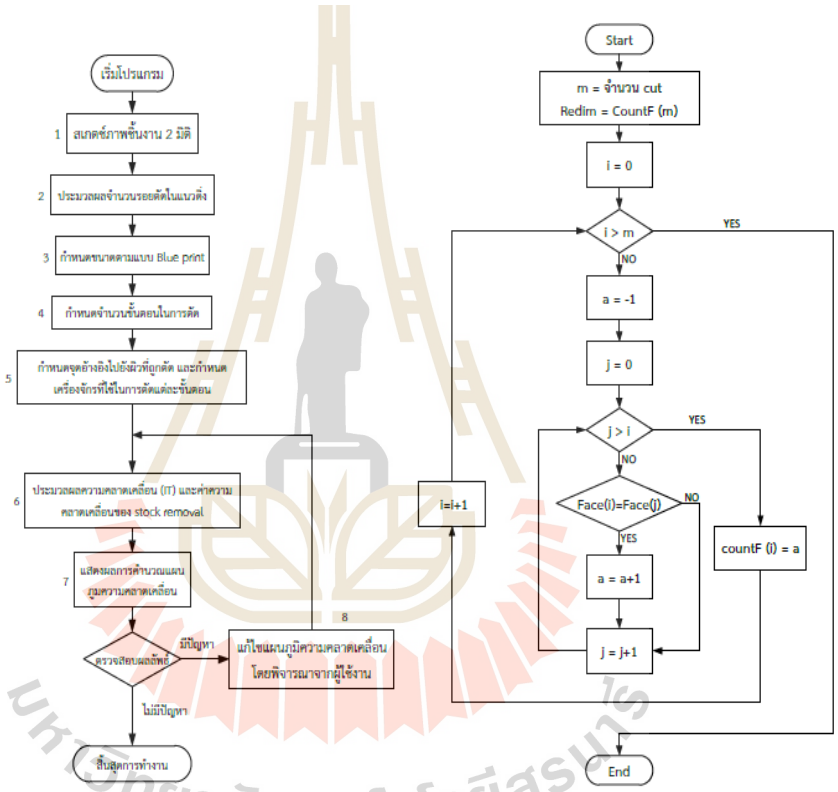
(3) **กลุ่มโมดูลแสดงผลลัพธ์** (Output Modules) ทำหน้าที่แสดงผลลัพธ์ของโปรแกรมตามแบบขั้นตอนการคำนวณของแผนภูมิฯ โดยรับค่าการคำนวณจากโมดูลย่อยทั้งหลายที่เกี่ยวข้องของโปรแกรมมาแสดงผลลัพธ์ให้ผู้ใช้งานพิจารณาความเหมาะสม

(4) **กลุ่มข้อมูลย้อนกลับ** (Feedback Modules) เป็นกลุ่มโมดูลเพื่ออำนวยความสะดวกต่อผู้ใช้โปรแกรมในการปรับเปลี่ยนขั้นตอนการตัดโลหะ เช่น เชื่อมโยงกับโมดูลประมวลผล (Processing Modules) แสดงความสัมพันธ์ของความคลาดเคลื่อน, หรือ เมื่อปรับเปลี่ยน ขั้นตอนของการตัดโลหะ โปรแกรมจะดำเนินการเปลี่ยนแปลงโดยผ่านการทำหน้าที่ของโมดูลในกลุ่มนี้ โดยไม่ต้องเริ่มใส่ข้อมูลเบื้องต้นใหม่

**5. การทำงานของโปรแกรม**

Flow chart ในรูปที่ 4 แสดงลำดับการทำงานโดยทั่วไปของโปรแกรมแผนภูมิความคลาดเคลื่อน เริ่มต้น

ด้วยการรับข้อมูลซึ่งอ้างอิงถึงภาพสเก็ตช์ของชิ้นงาน และรายละเอียดจากแบบวิศวกรรม (กรอบหมายเลข 1 ถึง 5); จากนั้นนำข้อมูลมาประมวล หาค่าความคลาดเคลื่อนสะสมต่างๆ และ ขนาดมูลฐานของ Basic working dim. (กรอบหมายเลข 6); ผลจากการประมวลในขั้นตอนประมวลผลจะถูกส่งไปแสดงออกเป็นแผนภูมิฯโดย Outputs Modules (กรอบหมายเลข 7); และ ถ้าเกิดปัญหาความคลาดเคลื่อนสะสม (tolerance-stack problem) ผู้ใช้โปรแกรมสามารถแก้ไขข้อมูลของกระบวนการตัดโลหะได้ (กรอบหมายเลข 8) จากนั้นโปรแกรมจะประมวลผลใหม่ (ทำซ้ำ กรอบหมายเลข 6 และ 7).




รูปที่ 4 Flowchart แสดงขั้นตอนการทำงาน

รูปที่ 5 แสดงโมดูลประมวลผลผิวตัด

**6. Flow chart ของโมดูลกำหนดผิวตัดชิ้นงานตามลำดับ**

โมดุลนี้อยู่ในกลุ่มของ Processing Modules ทำหน้าที่กำหนดรหัสชื่อผิวชิ้นงานตามลำดับขั้นในการตัดโลหะ ซึ่งมีความสำคัญต่อการสร้างเส้นทาง (path) ของ rooted tree ที่ประกอบด้วย ความคลาดเคลื่อนสะสมระหว่างผิวที่ถูกตัด เพราะว่าในความเป็นจริงแล้ว ไม่มีการสร้าง rooted tree เพื่อหา path แต่ถ้าชื่อผิวของชิ้นถูกกำหนดให้เป็นไปตามลำดับตัดโลหะ, ขั้นตอนการตัดโลหะนั้นก็คือ rooted tree.

โมดูล input จะรับค่าผิวอ้างอิงและผิวที่ถูกตัดจากผู้ใช้โปรแกรมเป็นรหัสอักษรตัวเดียว ซึ่งจะถูกเปลี่ยนให้เป็นโหนดของ rooted tree (รหัสตัวอักษรตามด้วยตัวเลข) ด้วยเงื่อนไขดังนี้: ผิวที่ไม่ถูกตัดซ้ำ ตามด้วยเลข 0; ผิวที่ถูกตัดซ้ำครั้งที่หนึ่ง ตามด้วยเลข 1; ถูกตัดซ้ำครั้งที่สอง ตามด้วยเลข 2, เป็นต้น ในตารางที่ 1 แสดงชื่อผิวบนชิ้นงานก่อน และหลัง การเปลี่ยนแปลง

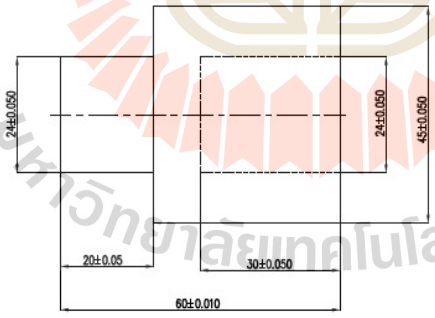
ตารางที่ 1 แสดงผลลัพธ์ของผิวตัดชิ้นงาน

| | Input | | Output | |
|---|---|---|---|---|
| operation | reference cut | face cut | reference cut | face cut |
| 10 | G | F | G0 | F0 |
| 20 | G | D | G0 | D0 |
| 30 | D | E | D0 | E0 |
| 40 | D | B | D0 | B0 |
| 50 | B | A | B0 | A0 |
| 60 | F | C | F0 | C0 |
| 70 | F | G | F0 | G1 |
| 80 | G | D | G1 | D1 |
| 90 | D | B | D1 | B1 |
| 100 | G | A | G1 | A1 |

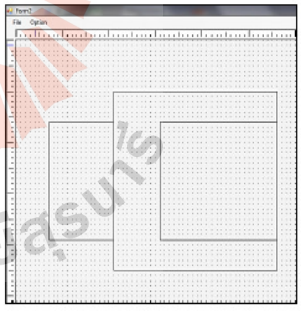
สำหรับการทำงานของโมดูลนี้ สามารถแสดงได้เป็นflow chart ในรูป 5 โดยเริ่มจากการนำจำนวนครั้งของการตัดโลหะทั้งหมดเก็บไว้ในตัวแปร m กำหนดค่า a เท่ากับ -1 จากนั้นทำการวนลูปที่ 1 เท่ากับค่า m เก็บจำนวนลูปในตัวแปร i และวนลูปที่ 2 เท่ากับค่า i เก็บจำนวนลูปในตัวแปร j เงื่อนไขหากตัวแปร facecut(i) เท่ากับตัวแปร facecut(j) นำค่าตัวแปร a บวก 1 จากนั้นนำค่า a(i) ไปเก็บไว้ในตัวแปร CountF(i) ซึ่งอาเรย์ 1 มิตินี้ คือหมายเลขระบุจำนวนครั้งของการตัดซ้ำของผิวแต่ละผิวบนชิ้นงาน

**7. ตัวอย่างการใช้โปรแกรม**

รูปที่ 6 แสดงแบบของชิ้นงานจากเอกสารอ้างอิง [9] ซึ่งเป็นตัวอย่างสำหรับการคำนวณในแผนภูมิความคลาดเคลื่อนด้วยโปรแกรมที่พัฒนาขึ้น การรับข้อมูลอาศัยหน้าจอ 3 หน้า: (1) รูปชิ้นงาน (รูปที่ 7 ); (2) มิติของชิ้นงานในแบบวิศวกรรม (รูปที่ 8);และ (3) รายละเอียดเกี่ยวกับความคลาดเคลื่อนของการตัดแต่ละครั้ง, เครื่องมือกลที่ใช้, และผิวที่ถูกตัด-ผิวอ้างอิง (รูปที่ 9) ซึ่งเป็นข้อมูลส่วนด้านซ้ายและตอนกลางของแผนภูมิฯ



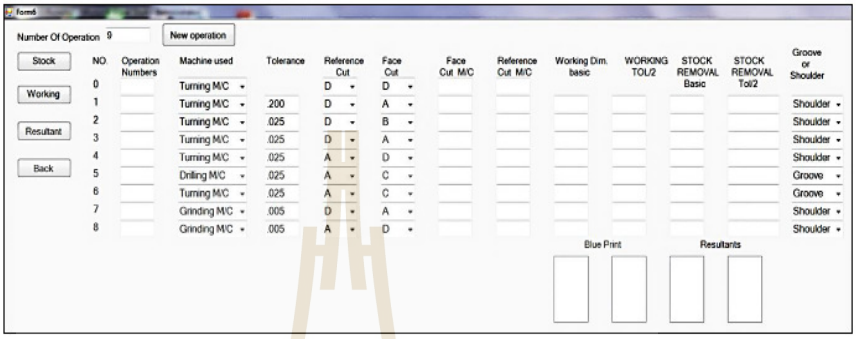

รูปที่ 6 แบบของชิ้นงาน



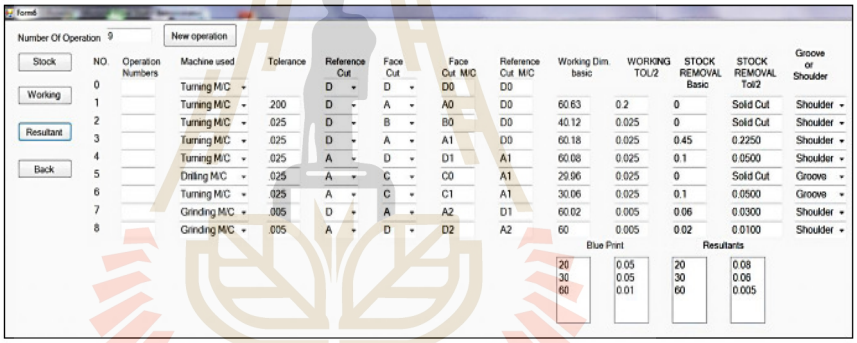
รูปที่ 7 ภาพสเก็ตช์ของชิ้นงาน

รูปที่ 8 การรับข้อมูลขนาดของชิ้นงาน

ผลลัพธ์จากโปรแกรม ในส่วนของการคำนวณค่าความคลาดเคลื่อนสะสมที่มิติสุดท้ายของการตัดโลหะ (finished dimensions) แสดงในรูปที่ 10 ซึ่งก็คือทุกส่วนของแผนภูมิความคลาดเคลื่อน ยกเว้นภาพสเก็ตช์ของชิ้นงานเท่านั้น ที่ยังไม่ได้บรรจุในแผนภูมิ และเป็นงานที่จะทำต่อไป




รูปที่ 9 รับข้อมูลการตัดโลหะ




รูปที่ 10 แสดงผลลัพธ์ของโปรแกรม

## 8. สรุป

ถึงแม้ว่าการพัฒนาโปรแกรมแผนภูมิความคลาดเคลื่อนนี้ยังไม่เสร็จสิ้นอย่างสมบูรณ์ แต่ก็สามารถกล่าวโดยรวมว่าโปรแกรมฯให้ผลลัพธ์เป็นที่น่าพอใจไม่ว่าจะเป็นการคำนวณค่าความคลาดเคลื่อนสะสมการปรับเปลี่ยนขั้นตอนในการผลิต หรือ การปรับปรุงเงื่อนไขของการตัดโลหะ ตลอดจนการติดต่อระหว่างโปรแกรมและผู้ใช้ (interaction) อย่างไรก็ตาม ยังคงมีงานที่ต้องปรับปรุงโปรแกรมในรายละเอียดอยู่อีกหลายอย่าง เพื่อให้บรรลุเป้าหมายของการสร้างโปรแกรมนี้

ในอเมริกาและยุโรป มีการใช้แผนภูมิความคลาดเคลื่อนในอุตสาหกรรมการตัดโลหะของชิ้นส่วนเครื่องบินและชิ้นส่วนรถยนต์ มาตั้งแต่ก่อนปี 1955 ด้วย แต่เทคนิคชนิดนี้ กลับไม่มีผู้กล่าวถึงในประเทศไทยก่อนหน้านี้เลย ไม่ว่าในสถาบันการศึกษาหรือในภาคอุตสาหกรรม

สาขาวิชาวิศวกรรมอุตสาหการ ที่มหาวิทยาลัยเทคโนโลยีสุรนารี เริ่มต้นงานวิจัยด้านนี้ เมื่อ 2 – 3 ปีที่ผ่านมาในรูปของงานวิจัยของนักศึกษาระดับปริญญาโท แม้ว่าจะล่าช้ากว่า สิงคโปร์ หรือ มาเลเซียอยู่บ้าง แต่ก็ยังไม่สายเกินไป และหวังว่า เมื่องานวิจัยนี้ เสร็จสมบูรณ์ จะเป็นจุดเริ่มต้นสำหรับผู้ต้องการบุกเบิกงานวิจัยทางด้าน

"ความคลาดเคลื่อน"และ"การวิเคราะห์ความคลาดเคลื่อน" เพราะในอุตสาหกรรมการผลิตและประกอบชิ้นงานโลหะ เราไม่อาจหลีกเลี่ยงสิ่งเหล่านี้ได้เลย และหวังต่อไปอีกว่า ในอนาคตอันใกล้นี้จะมีผู้ศึกษาวิจัย พัฒนาโปรแกรมแผนภูมิความคลาดเคลื่อนให้ครอบคลุมความความเคลื่อนได้หลากหลายชนิดขึ้นและรวมไปถึงการประยุกต์ Optimization Techniques ซึ่งก็มีผู้ทำวิจัยมากแล้ว แต่ตราบใดที่ผลงานวิจัยยังไม่สามารถให้คำตอบที่ตรงกันได้ ตราบนั้นก็ยังคงมีงานวิจัยทำนองนี้อยู่

## เอกสารอ้างอิง


[1] Doyle, L.E., 1950. Tool Engineering: Analysis and Procedure,Englewood Cliffs, N.J., Prentice-Hall.

[2] Bjorke, O. 1978. Computer-Aided Tolerancing, Tapir, Norway.

[3] Ahaluwalia, R. S. and Karolin, A. V. 1984. CATC A computer- aided tolerance control system.

[4] Gadzala, J. L. 1959. Dimensional Control in Precision Manufacturing: As Applied in Production Machining to Effect Higher Production and Lower Unit Costs, McGraw-Hill.

[5] Wade, O. R. 1967. Tolerance Control in Design and Manufacturing, Industrial Press, New York.

[6] Whybrew, K., Britton, G. A., Robinson, D. F., and Sermsuti-anuwat, Y. 1990. A Graphtheoretic Approach to Tolerance Charting, International Journal of Advanced Manufacturing Technology.

[7] ชลาลัย วงเวียน. 2559. การสร้างแผนภูมิความคลาดเคลื่อนด้วยตารางคำนวณ, วิทยานิพนธ์ระดับปริญญาโท, มหาวิทยาลัยเทคโนโลยีสุรนารี.

[8] G.A. Britton, K. Whybrew, and Y. Sermsuti-Anuwat, 1992. A Manual Graph Theoretic Method for Teaching Tolerance Charting,The International Journal of Mechanical Engineering Education.

[9] Sermsuti - Anuwat, Y. 1992. Computer-aided Process Planning and Fixture Design (CAPPFD), Ph.D.

Thesis, Mechanical Engineering University of Canterbury, New Zealand.


มหาวิทยาลัยเทคโนโลยีสุรนารี

# ประวัติผู้เขียน

นายคงเดช คำศรี เกิดเมื่อวันที่ 25 กรกฎาคม พ.ศ. 2530 ที่จังหวัดกรุงเทพมหานคร ปัจจุบันภูมิลำเนาอยู่ที่ 631 หมู่ 3 ถนนราชสีมา-โชคชัย ตำบลหนองบัวศาลา อำเภอเมือง จังหวัดนครราชสีมา จบการศึกษาระดับประถมศึกษาและระดับมัธยมต้น จากโรงเรียนมารีย์วิทยา (นครราชสีมา) จากนั้น เริ่มศึกษาต่อในระดับประกาศนียบัตรวิชาชีพ (ปวช.) ในสาขาวิชาเครื่องมือกล วิทยาลัยเทคนิคนครราชสีมา และสำเร็จการศึกษาในปี พ.ศ.2548 ต่อมาได้เข้าศึกษาต่อในระดับประกาศนียบัตรวิชาชีพชั้นสูง (ปวส.) ในสาขาวิชาช่างกลโรงงาน และสำเร็จการศึกษาในปี พ.ศ. 2550 ต่อมาได้เข้าศึกษาต่อในระดับปริญญาตรี สาขาวิชาวิศวกรรมอุตสาหการ สำนักวิชาวิศวกรรมศาสตร์และสถาปัตยกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีราชมงคลอีสาน จังหวัดนครราชสีมา และสำเร็จการศึกษาในปี พ.ศ. 2553 หลังสำเร็จการศึกษาได้มีโอกาสเข้าทำงานที่บริษัท อิชิตัน กรุ๊ป จำกัด (มหาชน) ที่จังหวัดพระนครศรีอยุธยา เป็นเวลา 2 ปี ต่อมาได้เข้าศึกษาต่อในระดับปริญญาโท สาขาวิชาวิศวกรรมอุตสาหการ สำนักวิชาวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีสุรนารี ในปี พ.ศ. 2557 ขณะที่ศึกษาอยู่ได้มีโอกาสเป็นผู้ช่วยสอนและวิจัยในสาขาวิชาวิศวกรรมอุตสาหการ มหาวิทยาลัยเทคโนโลยีสุรนารี ซึ่งทำให้ผู้วิจัยได้นำประสบการณ์ และความรู้ที่ได้จากการเป็นผู้ช่วยสอนและวิจัยมาประยุกต์ใช้กับงานวิจัยได้เป็นอย่างดี

ผลงานวิจัย : ได้เสนอบทความเข้าร่วมในการประชุมวิชาการระดับชาติครั้งที่ 13 ประจำปี พ.ศ.2559 เรื่องการใช้คอมพิวเตอร์ช่วยสร้างแผนภูมิความคลาดเคลื่อนสำหรับวางแผนการจัดลำดับการตัดโลหะ มหาวิทยาลัยเกษตรศาสตร์ วิทยาเขตกำแพงแสน

มหาวิทยาลัยเทคโนโลยีสุรนารี