



## รายงานการปฏิบัติงานสหกิจศึกษา

การพัฒนาโปรแกรมทดสอบซอฟต์แวร์  
เพื่อวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติ

**Development of Software Testing Programs  
to Analyze the Statistical Process Control in Production Process**

โดย

นางสาววิลาวัลย์ เชื้อนิจ

รายงานนี้เป็นส่วนหนึ่งของรายงานสหกิจศึกษา

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยสยาม

ภาคการศึกษาที่ 2 ปีการศึกษา 2563

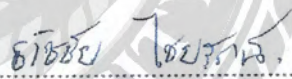
หัวข้อโครงการ      การพัฒนาโปรแกรมทดสอบซอฟต์แวร์เพื่อวิเคราะห์การควบคุมกระบวนการ  
การผลิตเชิงสถิติ  
Development of Software Testing Programs to Analyze the  
Statistical Process Control in Production Process

รายชื่อผู้จัดทำ      นางสาววิลาวัลย์ เชื้อนิจ  
ภาควิชา      วิศวกรรมคอมพิวเตอร์  
อาจารย์ที่ปรึกษา      อาจารย์นลินรัตน์ วิศวภิตติ

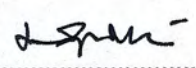
อนุมัติให้รายงานนี้เป็นส่วนหนึ่งของการปฏิบัติงานสหกิจศึกษา ภาควิชาวิศวกรรม  
คอมพิวเตอร์ คณะวิศวกรรมศาสตร์ ภาควิชาการศึกษาที่ 2 ปีการศึกษา 2563

คณะกรรมการสอบโครงการ

  
.....อาจารย์ที่ปรึกษา  
(อาจารย์นลินรัตน์ วิศวภิตติ)

  
.....พนักงานที่ปรึกษา  
(นายรัชชัย ไชยรัตน์)

  
.....กรรมการกลาง  
(ผู้ช่วยศาสตราจารย์พิกิจ สุวดี)

  
.....ผู้ช่วยอธิการบดีและผู้อำนวยการสำนักสหกิจศึกษา  
(ผู้ช่วยศาสตราจารย์ ดร.มารุจ ลิ้มประวัฒน์)

## จดหมายนำส่งรายงาน

วันที่ 4 เดือน พฤษภาคม พ.ศ. 2564

เรื่อง ขอส่งรายงานการปฏิบัติงานสหกิจศึกษา  
เรียน อาจารย์ที่ปรึกษาสหกิจศึกษาภาควิชาวิศวกรรมคอมพิวเตอร์  
อาจารย์ณลินรัตน์ วิศวภิตติ

ตามที่ได้ผู้จัดทำ นางสาววิลาวัลย์ เชื้อนิจ นักศึกษาภาควิชา วิศวกรรมคอมพิวเตอร์ คณะ วิศวกรรมศาสตร์ มหาวิทยาลัยสยาม ได้เข้าปฏิบัติงานในโครงการสหกิจศึกษาระหว่างวันที่ 11 มกราคม พ.ศ 2564 ถึง วันที่ 30 เมษายน 2564 ในตำแหน่ง Test Software Engineer (Intern) ณ บริษัท ลูเมนต้า อินเทอร์เน็ตเนชั่นแนล (ประเทศไทย) จำกัด และได้รับมอบหมายงานจากพนักงานที่ปรึกษาให้ศึกษาและทำรายงานเรื่องการพัฒนาโปรแกรมทดสอบซอฟต์แวร์และการวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติ

บัดนี้การปฏิบัติงานสหกิจศึกษาได้สิ้นสุดแล้วนางสาววิลาวัลย์ เชื้อนิจ จึงขอส่งรายงานดังกล่าวมา จำนวน 1 เล่ม และ CD จำนวน 1 แผ่น เพื่อขอรับคำปรึกษาต่อไป

จึงเรียนมาเพื่อโปรดพิจารณา

ขอแสดงความนับถือ

นางสาววิลาวัลย์ เชื้อนิจ

นักศึกษาสหกิจศึกษา

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยสยาม

## กิตติกรรมประกาศ (Acknowledgement)

การที่ผู้จัดทำได้มาปฏิบัติงานในโครงการสหกิจศึกษา ณ บริษัท ฐเมนต้า อินเตอร์เนชั่นแนล (ประเทศไทย) จำกัด ตั้งแต่วันที่ 11 มกราคม พ.ศ 2564 ถึง วันที่ 30 เมษายน 2564 ในตำแหน่ง Test Software Engineer Intern ส่งผลให้ผู้จัดทำได้รับความรู้และประสบการณ์ทำงานต่าง ๆ ซึ่งจะ เป็นประโยชน์ต่อการประกอบอาชีพในอนาคต โดยในระหว่างปฏิบัติงาน ผู้จัดทำได้รับคำแนะนำ และความช่วยเหลือจากหลายฝ่าย ดังนี้

- |                               |                                     |
|-------------------------------|-------------------------------------|
| 1. คุณวัชรชัย ไชยรัตน์        | Engineering Dev Manager             |
| 2. คุณธนกร ศัจธรรม            | Engineer Staff, Reliability/Test    |
| 3. คุณชนินทร์ โพธิสุข         | Senior Test Engineer                |
| 4. คุณกฤษฎดา มุกดาเจริญชัย    | Engineer Sr, Test Software Engineer |
| 5. Mr.David Yue               | IT Systems Engineer                 |
| 6. อาจารย์ณลินรัตน์ วิสวกิตติ | อาจารย์ที่ปรึกษาสหกิจศึกษา          |

และบุคคลท่านอื่น ๆ ที่ไม่ได้กล่าวนาม

ผู้จัดทำขอขอบพระคุณผู้ที่มีส่วนเกี่ยวข้องทุกท่านที่มีส่วนร่วมในการให้ข้อมูลและเป็นที่ปรึกษาในการทำรายงานฉบับนี้จนเสร็จสมบูรณ์ ตลอดจนให้การดูแลให้ความเข้าใจกับชีวิตของการทำงานจริง จึงขอขอบพระคุณเป็นอย่างสูงไว้ ณ ที่นี้

ผู้จัดทำ

นางสาววิลาวัลย์ เชื้อนิจ

4 พฤษภาคม 2564



**Project Title** : Development of Software Testing Programs to Analyze the Statistical Process Control in Production Process

**Credits** : 6 Units

**By** : Miss Wilawan Chueanit

**Advisor** : Miss Nalinrat Witsawakitti

**Degree** : Bachelor of Engineering

**Major** : Computer Engineering

**Faculty** : Engineering

**Semester / Academic Year** : 2/2020

#### Abstract

The daily product testing of Lumentum International (Thailand) Co., Ltd. consisted of 2 parts testing machine test with the test results saved in Excel file format, and product testing on the testing machine by saved the test results in a text file. Because of the two parts of the testing, neither results of the tests were stored in a compatible database. This made it difficult to use the data to analyze the statistical process control of the manufacturing process in order to control the testing machine and product testing to meet the standard quality.

Therefore, the idea was to development a software testing program to analyze the statistical process control with Visual Basic.Net language. The program will generate an XML file from the test machine test results and the product test on the tester, to be stored as data in a database for use in the analysis of the production process. The test results of the developed program found that the results of daily product testing of Lumentum International (Thailand) Co., Ltd. can be obtained from the company's database, and to showed the results in a graph for use in the analysis of the statistical process control.

**Keywords:** software testing, statistical process control, manufacturing process

Approved by

## สารบัญ

|                 | หน้า |
|-----------------|------|
| จดหมายนำส่ง     | ก    |
| กิตติกรรมประกาศ | ข    |
| บทคัดย่อ        | ค    |
| Abstract        | ง    |

### บทที่ 1 บทนำ

|                                           |   |
|-------------------------------------------|---|
| 1.1 ความเป็นมาและความสำคัญของโครงการ..... | 1 |
| 1.2 วัตถุประสงค์ของโครงการ.....           | 1 |
| 1.3 ขอบเขตของโครงการ.....                 | 2 |
| 1.4 ประโยชน์ที่คาดว่าจะได้รับ.....        | 2 |

### บทที่ 2 การทบทวนเอกสารและวรรณกรรมที่เกี่ยวข้อง

|                                                                        |   |
|------------------------------------------------------------------------|---|
| 2.1 การควบคุมกระบวนการผลิตเชิงสถิติ (Statistical process control)..... | 3 |
| 2.2 แผนภูมิควบคุม (Control Chart).....                                 | 3 |
| 2.3 ภาษาวิซวล เบสิก (Visual Basic).....                                | 6 |
| 2.4 ดอท เน็ต เฟรมเวิร์ก (.NET Framework).....                          | 7 |
| 2.5 เอกซ์เอ็มแอล (XML).....                                            | 7 |
| 2.6 เอกซ์เอ็มแอล มาร์กเกอร์ เวอร์ชัน 1.1 (XML Marker 1.1).....         | 8 |

### บทที่ 3 รายละเอียดและการปฏิบัติงาน

|                                                     |    |
|-----------------------------------------------------|----|
| 3.1 ชื่อและที่ตั้งของสถานประกอบการ.....             | 9  |
| 3.2 ลักษณะการประกอบการ.....                         | 9  |
| 3.3 รูปแบบการจัดองค์กรและการบริหารงานขององค์กร..... | 10 |
| 3.4 ตำแหน่งงานและลักษณะงานที่ได้รับมอบหมาย.....     | 11 |
| 3.5 ชื่อและตำแหน่งงานของพนักงานที่ปรึกษา.....       | 11 |
| 3.6 ระยะเวลาที่ปฏิบัติงาน.....                      | 11 |
| 3.7 แผนการดำเนินงาน.....                            | 11 |
| 3.8 เครื่องมือและอุปกรณ์ที่ใช้.....                 | 14 |

## สารบัญ(ต่อ)

หน้า

### บทที่ 4 ผลการปฏิบัติงานตามโครงการ

|                                                                |    |
|----------------------------------------------------------------|----|
| 4.1 โปรแกรมทดสอบซอฟต์แวร์เพื่อวิเคราะห์การควบคุมกระบวนการ..... | 15 |
| ผลิตเชิงสถิติ                                                  |    |
| 4.2 การทดสอบซอฟต์แวร์ฟังก์ชัน “Main Test”.....                 | 16 |
| 4.3 การทดสอบซอฟต์แวร์ฟังก์ชัน “Power Check”.....               | 17 |
| 4.4 การทดสอบซอฟต์แวร์ฟังก์ชัน “Daily Check”.....               | 19 |
| 4.5 การวิเคราะห์ข้อมูลเพื่อควบคุมกระบวนการเชิงสถิติ .....      | 21 |

### บทที่ 5 สรุปผลและข้อเสนอแนะ

|                                                      |    |
|------------------------------------------------------|----|
| 5.1 สรุปผลโครงการ.....                               | 23 |
| 5.2 สรุปผลการปฏิบัติงานสหกิจศึกษา.....               | 24 |
| บรรณานุกรม                                           |    |
| ภาคผนวก ก ภาพขณะปฏิบัติงานและการนิเทศของอาจารย์..... | 25 |
| ภาคผนวก ข บทความวิชาการ.....                         | 28 |
| ภาคผนวก ค วนิฉ.....                                  | 36 |
| ภาคผนวก ง โค้ดของโปรแกรม.....                        | 38 |
| ประวัติผู้จัดทำ                                      |    |

## สารบัญตาราง

หน้า

|                                                  |    |
|--------------------------------------------------|----|
| ตารางที่ 3.1 ตารางแผนการดำเนินงานของโครงการ..... | 13 |
|--------------------------------------------------|----|





## สารบัญรูปภาพ

หน้า

|                                                                                                                                       |    |
|---------------------------------------------------------------------------------------------------------------------------------------|----|
| รูปที่ 2.1 กราฟแสดงองค์ประกอบของแผนภูมิควบคุม.....                                                                                    | 3  |
| รูปที่ 2.2 กฎข้อที่ 1 มีจุดใดจุดหนึ่งมากกว่า $+3\sigma$ หรือ น้อยกว่า $-3\sigma$ .....                                                | 4  |
| รูปที่ 2.3 กฎข้อที่ 2 มี 9 จุดต่อเนื่องกันที่มากกว่า หรือน้อยกว่า เส้นแกนกลาง.....                                                    | 4  |
| รูปที่ 2.4 กฎข้อที่ 3 มี 6 จุดต่อเนื่องกันที่เพิ่มขึ้นหรือลดลงอย่างต่อเนื่อง.....                                                     | 5  |
| รูปที่ 2.5 กฎข้อที่ 4 มี 14 จุดต่อเนื่องกันที่ขึ้นลงสลับฟันปลา.....                                                                   | 5  |
| รูปที่ 2.6 กฎข้อที่ 5 มี 2 หรือ 3 จุดจาก 3 จุดต่อเนื่องกันที่อยู่ห่างจากค่าเฉลี่ย .....<br>มากกว่า $+2\sigma$ หรือน้อยกว่า $-2\sigma$ | 5  |
| รูปที่ 2.7 กฎข้อที่ 6 มี 4 หรือ 5 จุด ในแถวมีค่ามากกว่า $\pm 1\sigma$ .....<br>จากค่าเฉลี่ยในทิศทางเดียวกัน                           | 5  |
| รูปที่ 2.8 กฎข้อที่ 7 มี 15 จุดต่อเนื่องกันที่อยู่ภายใน $\pm 1\sigma$ จากค่าเฉลี่ย.....                                               | 6  |
| รูปที่ 2.9 กฎข้อที่ 8 มี 8 จุดต่อเนื่องกันที่ไม่อยู่ภายใน $\pm 1\sigma$ จากค่าเฉลี่ย .....                                            | 6  |
| รูปที่ 2.10 โลโก้ของภาววิชาวิศวกรรมที่ออกแบบโดยไม่โครซอฟต์.....                                                                       | 6  |
| รูปที่ 2.11 องค์ประกอบของ คอเทนต์เฟรมเวิร์ก .....                                                                                     | 7  |
| รูปที่ 2.12 หน้าจอแสดงผลของเอกซ์เอ็มแอล มาร์กเกอร์ .....                                                                              | 8  |
| รูปที่ 3.1 แผนที่ตั้งสถานประกอบการ.....                                                                                               | 9  |
| รูปที่ 3.2 การจัดองค์การและการบริหารงานขององค์กร.....                                                                                 | 10 |
| รูปที่ 4.1.1 หน้าจอหลักของโปรแกรมทดสอบซอฟต์แวร์เพื่อวิเคราะห์.....<br>การควบคุมกระบวนการผลิตเชิงสถิติ                                 | 15 |
| รูปที่ 4.2.1 หน้าจอกรอกข้อมูลของฟังก์ชัน “Main Test”.....                                                                             | 16 |
| รูปที่ 4.2.2 หน้าจอทดสอบซอฟต์แวร์ของฟังก์ชัน “Main Test”.....                                                                         | 16 |
| รูปที่ 4.2.3 ไฟล์เอกซ์เอ็มแอลของฟังก์ชัน Main Test.....                                                                               | 17 |
| รูปที่ 4.2.4 หน้าจอผลการทดสอบของฟังก์ชัน “Main Test”.....                                                                             | 17 |
| รูปที่ 4.3.1 หน้าจอกรอกข้อมูลของฟังก์ชัน “Power Check”.....                                                                           | 18 |
| รูปที่ 4.3.2 หน้าจอทดสอบซอฟต์แวร์ของฟังก์ชัน Power Check.....                                                                         | 18 |
| รูปที่ 4.3.3 ไฟล์เอกซ์เอ็มแอลของฟังก์ชัน Power Check.....                                                                             | 19 |
| รูปที่ 4.3.4 แสดงข้อความและผลลัพธ์การทดสอบของฟังก์ชัน Power Check.....                                                                | 19 |
| รูปที่ 4.4.1 หน้าจอ frmInput ของฟังก์ชัน Daily Check.....                                                                             | 20 |
| รูปที่ 4.4.2 หน้าจอหน้าจอตสอบซอฟต์แวร์ของฟังก์ชัน ของฟังก์ชัน Daily Check.....                                                        | 20 |

## สารบัญรูปภาพ (ต่อ)

หน้า

|                                                                           |    |
|---------------------------------------------------------------------------|----|
| รูปที่ 4.4.3 ไฟล์เอกซ์เอ็มแอลของฟังก์ชัน Daily Check.....                 | 21 |
| รูปที่ 4.4.4 แสดงข้อความการทำงานของฟังก์ชัน Daily Check.....              | 21 |
| รูปที่ 4.5.1 ระบบฐานข้อมูลเอพีไออินเทอร์เน็ตเฟส.....                      | 22 |
| รูปที่ 4.5.2 ระบบเอสพีซี.....                                             | 22 |
| รูปที่ ก.1 การเขียนโปรแกรมในการพัฒนาโปรแกรมทดสอบซอฟต์แวร์.....            | 26 |
| รูปที่ ก.2 การนำข้อมูลมาวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติ.....     | 26 |
| รูปที่ ก.3 การนิเทศงานสหกิจศึกษาครั้งที่ 1 วันที่ 17 กุมภาพันธ์ 2564..... | 27 |
| รูปที่ ก.4 การนิเทศงานสหกิจศึกษาครั้งที่ 2 วันที่ 22 เมษายน 2564.....     | 27 |
| รูปที่ ค.1 ไลน์ของโครงการ.....                                            | 37 |



# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาและความสำคัญของปัญหา

บริษัทลูเมนตี้ม อินเตอร์เนชั่นแนล (ประเทศไทย) จำกัด (Lumentum International (Thailand).Co ,Ltd) เป็นบริษัทที่ดำเนินกิจการเกี่ยวกับการผลิตชิ้นส่วนเครื่องเลเซอร์ ชิ้นส่วนอุปกรณ์โทรคมนาคมสำหรับระบบใยแก้วนำแสง กิจการประกอบเครื่องจักร อุปกรณ์เครื่องจักร และ/หรือ อุปกรณ์การสื่อสาร เป็นต้น โดยก่อนที่จะส่งมอบผลิตภัณฑ์ให้กับลูกค้า จะต้องมีการทดสอบผลิตภัณฑ์เพื่อให้ลูกค้าได้รับผลิตภัณฑ์ที่มีคุณภาพตามมาตรฐาน ที่ใช้ทดสอบผลิตภัณฑ์ก่อนที่จะส่งมอบให้กับลูกค้า

โดยในการทดสอบผลิตภัณฑ์จะประกอบด้วย 2 ส่วน ได้แก่

1. การทดสอบเครื่องทดสอบโดยใช้ฟังก์ชันทดสอบที่ชื่อ “Daily Check” เพื่อให้เครื่องทดสอบอยู่ในสถานะที่พร้อมทดสอบผลิตภัณฑ์ ซึ่งบนสายการผลิต (Production line) จะมีเครื่องทดสอบจำนวนมาก ซึ่งในการทดสอบฟังก์ชัน Daily Check ในแต่ละวันจะมีการบันทึกข้อมูลผลการทดสอบในรูปแบบของไฟล์เอกซ์เซล (Excel)
2. การทดสอบผลิตภัณฑ์บนเครื่องทดสอบโดยใช้ฟังก์ชันทดสอบที่ชื่อ “Main Test” และ “Power Check” ในการทดสอบ ซึ่งบันทึกข้อมูลผลการทดสอบในรูปแบบของไฟล์ข้อความ (Text File) บนเครื่องทดสอบ เรียกว่า “Local Tester” และบนเซิร์ฟเวอร์เรียกว่า “Share Folder”

เนื่องจากการทดสอบทั้ง 2 ส่วน ไม่ได้มีการจัดเก็บข้อมูลผลของการทดสอบในลักษณะของฐานข้อมูลที่ใช้งานร่วมกันได้ จึงทำให้ยากต่อการนำข้อมูลไปใช้ในการวิเคราะห์การควบคุมกระบวนการเชิงสถิติ (Statistical Process Control : SPC) ของกระบวนการผลิต เพื่อใช้ควบคุมเครื่องทดสอบและการทดสอบผลิตภัณฑ์ให้มีคุณภาพตามมาตรฐาน

ดังนั้นแผนกวิศวกรทดสอบซอฟต์แวร์ (Test Software Engineer) ของบริษัทและผู้จัดทำจึงมีแนวคิดที่จะพัฒนาโปรแกรมทดสอบซอฟต์แวร์เพื่อวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติ โดยโปรแกรมสามารถสร้างไฟล์เอกซ์เอ็มแอล (XML) จากผลการทดสอบ สำหรับนำไปจัดเก็บเป็นข้อมูลในระบบฐานข้อมูล (Database System) ของบริษัท เพื่อที่จะสามารถนำข้อมูลมาแสดงผลในลักษณะของกราฟสำหรับการวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติ เพื่อนำไปสู่การควบคุมคุณภาพการทดสอบผลิตภัณฑ์อันจะได้มาซึ่งผลิตภัณฑ์ที่มีคุณภาพตามมาตรฐาน

## 1.2 วัตถุประสงค์โครงการ

- 1.2.1 การพัฒนาโปรแกรมทดสอบซอฟต์แวร์เพื่อวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติ
- 1.2.2 เพื่อสร้างไฟล์เอ็กซ์เอ็มแอลจากผลการทดสอบเครื่องทดสอบและผลิตภัณฑ์ของบริษัท
- 1.2.3 เพื่อวิเคราะห์ข้อมูลที่ได้จากผลการทดสอบเครื่องทดสอบและผลิตภัณฑ์ของบริษัท

## 1.3 ขอบเขตของโครงการ

- 1.3.1 โปรแกรมสามารถสร้างไฟล์เอ็กซ์เอ็มแอล (XML) ของผลการทดสอบซอฟต์แวร์จากฟังก์ชัน “Daily Check” ฟังก์ชัน “Main Test” และ “Power Check”
- 1.3.2 ไฟล์เอ็กซ์เอ็มแอลที่สร้างจะถูกจัดเก็บไว้ในระบบฐานข้อมูล (Database System) ของบริษัท
- 1.3.3 โปรแกรมสามารถทำงานร่วมกับโปรแกรมของบริษัทที่มีอยู่เดิมได้
- 1.3.4 โปรแกรมสามารถตรวจสอบการทำงานของเครื่องทดสอบตามพารามิเตอร์ที่สำคัญ (Key parameters) ที่บริษัทกำหนดได้
- 1.3.5 โปรแกรมใช้ในการตรวจสอบการทำงานของเครื่องทดสอบแบบวันต่อวัน

## 1.4 ประโยชน์ที่คาดว่าจะได้รับ

- 1.4.1 ได้โปรแกรมทดสอบซอฟต์แวร์เพื่อวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติสำหรับใช้งานในบริษัท
- 1.4.2 ได้ฐานข้อมูลของผลการทดสอบจากฟังก์ชัน “Daily Check” ฟังก์ชัน “Main Test” และ “Power Check” เพื่อนำไปใช้ในการวิเคราะห์ข้อมูล
- 1.4.3 ได้ควบคุมกระบวนการผลิตเชิงสถิติ เพื่อให้เครื่องทดสอบและการทดสอบผลิตภัณฑ์เป็นไปตามมาตรฐาน

## บทที่ 2

### การทบทวนเอกสารและวรรณกรรมที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงโปรแกรม เทคโนโลยี และฟังก์ชันการทำงานต่างๆ ที่ศึกษาค้นคว้าจากแหล่งต่างๆ เพื่อนำไปใช้ในการพัฒนาโปรแกรมทดสอบซอฟต์แวร์

#### 2.1 การควบคุมกระบวนการผลิตเชิงสถิติ (Statistical Process Control: SPC)

การควบคุมกระบวนการผลิตเชิงสถิติ เป็นวิธีการควบคุมคุณภาพที่ใช้วิธีการทางสถิติในการตรวจสอบและควบคุมกระบวนการผลิต ช่วยให้กระบวนการทำงานในการผลิตมีประสิทธิภาพ การผลิตผลิตภัณฑ์ให้สอดคล้องกับข้อกำหนด (Spec) มากขึ้น ทำให้มีของเสียน้อยลง

**ประโยชน์ของการควบคุมกระบวนการทางผลิตเชิงสถิติ**

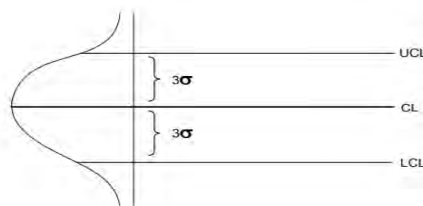
1. สามารถลดการสูญเสียและการผลิตซ้ำของผลิตภัณฑ์
2. ลดต้นทุนของการผลิต
3. ทราบถึงสาเหตุและปัญหาที่เกิดขึ้นจากความแปรผันของกระบวนการการผลิต
4. ป้องกันเพื่อไม่ให้เกิดปัญหา และความสูญเสียของผลิตภัณฑ์
5. สามารถตรวจสอบค่าของพารามิเตอร์กระบวนการผลิต กับข้อกำหนด (Spec) ของการผลิต และข้อกำหนด (Spec) ของผลิตภัณฑ์
6. ช่วยในการควบคุมคุณภาพของผลิตภัณฑ์

#### 2.2 แผนภูมิควบคุม (Control Chart)

แผนภูมิควบคุม คือ แผนภูมิที่ใช้ในการตรวจสอบค่าของตัวพารามิเตอร์ที่ต้องการควบคุมคุณภาพว่าเกิดความแปรผันเกินจากขอบเขตที่กำหนดไว้หรือไม่ หากพบว่าเกินขอบเขตที่กำหนดไว้จะต้องหาสาเหตุของความแปรผัน และดำเนินการแก้ไขก่อนที่จะเกิดความเสียหายแก่ผลิตภัณฑ์

**องค์ประกอบของแผนภูมิควบคุม**

ข้อมูลที่วัดได้จากกระบวนการผลิตจะมีการแจกแจงแบบปกติ (Normal Distribution) ซึ่งมีค่าพารามิเตอร์ที่เกี่ยวข้องอยู่ 2 ค่า คือ ค่าเฉลี่ย ( $\mu$ ) และส่วนเบี่ยงเบนมาตรฐาน ( $\sigma$ ) โดยมีโอกาสหรือความน่าจะเป็นที่ค่าสังเกตที่วัดได้จะอยู่ในช่วง  $\pm 3\sigma$  หรือเท่ากับ 0.9974 สามารถนำมาสร้างกราฟแผนภูมิควบคุม ซึ่งประกอบไปด้วยเส้นสำคัญ 3 เส้น ดังรูปที่ 2.1



รูปที่ 2.1 กราฟแสดงองค์ประกอบของแผนภูมิควบคุม

เส้นแกนกลาง (Center Line : CL) เป็นค่าเฉลี่ยของกระบวนการผลิต คำนวณได้โดยนำค่าจากกลุ่มตัวอย่างทั้งหมดนั้นมาหาค่าเฉลี่ย

ขีดจำกัดควบคุมด้านบน (Upper Control Limit : UCL) เส้นที่มีระยะห่างจากเส้นแกนกลางเท่ากับ  $+3\sigma$

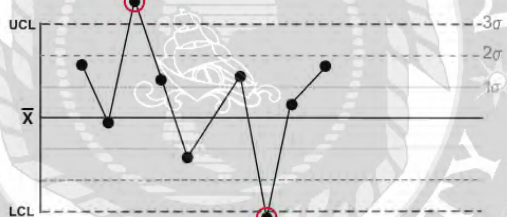
ขีดจำกัดควบคุมด้านล่าง (Lower Control Limit : LCL) เส้นที่มีระยะห่างจากเส้นแกนกลางเท่ากับ  $-3\sigma$

จากรูปที่ 2.1 ขีดจำกัดควบคุมด้านบนและด้านล่าง แสดงถึงความแปรผันที่อยู่ในระดับคุณภาพมาตรฐานที่ผู้ผลิตยอมรับได้ ซึ่งอยู่ในช่วง  $\pm 3\sigma$  เท่านั้น หากค่าสังเกตที่วัดได้กระจายอยู่ภายในขอบเขตดังกล่าว แสดงว่ากระบวนการผลิตอยู่ภายใต้การควบคุม ผลิตภัณฑ์ที่ผลิตจะมีคุณภาพตรงตามมาตรฐานการผลิต หากมีความแปรผันมากเกินไป ทำให้ค่าสังเกตที่วัดได้อยู่นอกเส้นขีดจำกัด แสดงว่ากระบวนการผลิต อยู่นอกเหนือการควบคุม แสดงค่าผลิตภัณฑ์ที่ผลิตไม่มีคุณภาพตามมาตรฐานการผลิต

**ลักษณะและเงื่อนไขของกราฟที่ออกนอกการควบคุม**

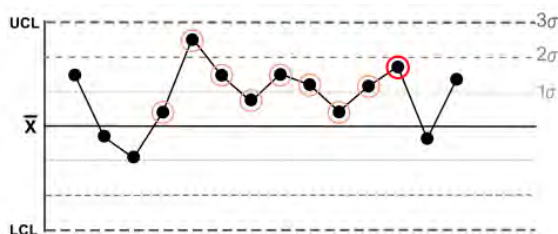
ตามกฎ 8 ข้อ ของเนลสัน (The eight Nelson Rules)

กฎข้อที่ 1 มีจุดใดจุดหนึ่งมากกว่า  $+3\sigma$  หรือ น้อยกว่า  $-3\sigma$



รูปที่ 2.2 กฎข้อที่ 1 มีจุดใดจุดหนึ่งมากกว่า  $+3\sigma$  หรือ น้อยกว่า  $-3\sigma$

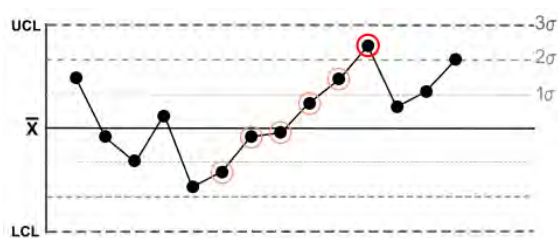
กฎข้อที่ 2 มี 9 จุดต่อเนื่องกันที่มากกว่า หรือน้อยกว่า เส้นแกนกลาง (Center Line)



รูปที่ 2.3 กฎข้อที่ 2 มี 9 จุดต่อเนื่องกันที่มากกว่า หรือน้อยกว่า เส้นแกนกลาง

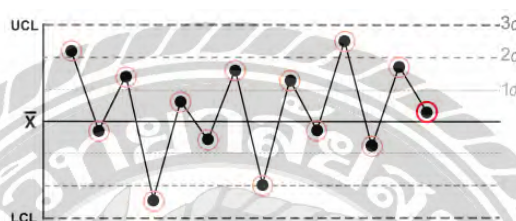


กฎข้อที่ 3 มี 6 จุดต่อเนื่องกันที่เพิ่มขึ้นหรือลดลงอย่างต่อเนื่อง



รูปที่ 2.4 กฎข้อที่ 3 มี 6 จุดต่อเนื่องกันที่เพิ่มขึ้นหรือลดลงอย่างต่อเนื่อง

กฎข้อที่ 4 มี 14 จุดต่อเนื่องกันที่ขึ้นลงสลับฟันปลา



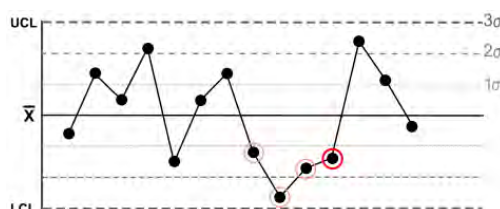
รูปที่ 2.5 กฎข้อที่ 4 มี 14 จุดต่อเนื่องกันที่ขึ้นลงสลับฟันปลา

กฎข้อที่ 5 มี 2 หรือ 3 จุดจาก 3 จุดต่อเนื่องกันที่อยู่ห่างจากค่าเฉลี่ยมากกว่า  $+2\sigma$  หรือน้อยกว่า  $-2\sigma$



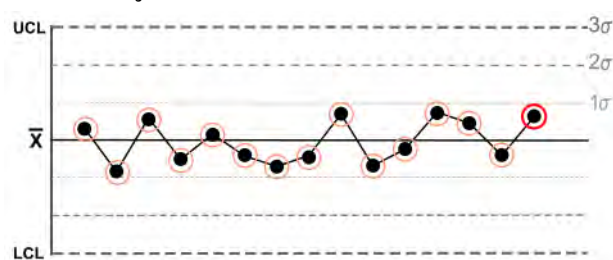
รูปที่ 2.6 กฎข้อที่ 5 มี 2 หรือ 3 จุดจาก 3 จุดต่อเนื่องกันที่อยู่ห่างจากค่าเฉลี่ยมากกว่า  $+2\sigma$  หรือน้อยกว่า  $-2\sigma$

กฎข้อที่ 6 มี 4 หรือ 5 จุด ในแถวมีค่ามากกว่า  $\pm 1\sigma$  จากค่าเฉลี่ยในทิศทางเดียวกัน



รูปที่ 2.7 กฎข้อที่ 6 มี 4 หรือ 5 จุด ในแถวมีค่ามากกว่า  $\pm 1\sigma$  จากค่าเฉลี่ยในทิศทางเดียวกัน

กฎข้อที่ 7 มี 15 จุดต่อเนื่องกันที่อยู่ภายใน  $\pm 1\sigma$  จากค่าเฉลี่ย



รูปที่ 2.8 กฎข้อที่ 7 มี 15 จุดต่อเนื่องกันที่อยู่ภายใน  $\pm 1\sigma$  จากค่าเฉลี่ย

กฎข้อที่ 8 มี 8 จุดต่อเนื่องกันที่ไม่อยู่ภายใน  $\pm 1\sigma$  จากค่าเฉลี่ย



รูปที่ 2.9 กฎข้อที่ 8 มี 8 จุดต่อเนื่องกันที่ไม่อยู่ภายใน  $\pm 1\sigma$  จากค่าเฉลี่ย

### 2.3 ภาษาวิซวลเบสิก (Visual Basic)

วิซวลเบสิก (Visual Basic) เป็นภาษาในการเขียนโปรแกรมแบบการเขียนโปรแกรมที่ขึ้นกับเหตุการณ์ (event-driven programming) โดย ภาษาวิซวลเบสิก นั้นถูกออกแบบมาเพื่อให้ง่ายต่อการเรียนรู้และง่ายต่อการใช้งาน ซึ่งสนับสนุนการพัฒนาโปรแกรมแบบแรพิดแอปพลิเคชัน (rapid application development : RAD) และ กราฟฟิคยูสเซอร์อินเทอร์เฟซ (Graphical User Interface : GUI) การเข้าถึงฐานข้อมูล และอื่นๆ ที่ทำงานภายใต้ ดอทเน็ตเฟรมเวิร์ก (.NET Framework)

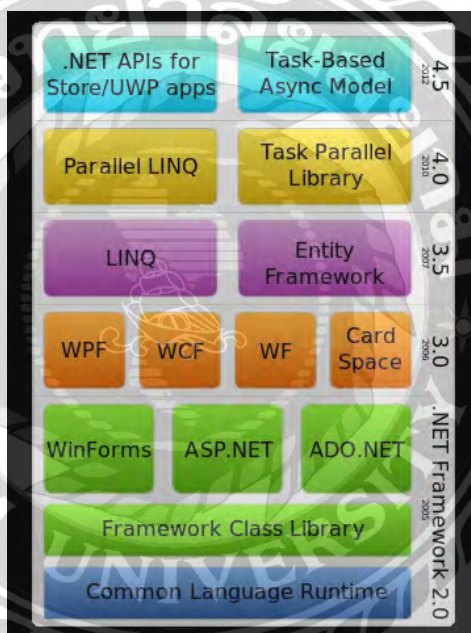
Microsoft®  
VB.net

รูปที่ 2.10 โลโก้ของภาษาวิซวลเบสิกที่ออกแบบโดยไมโครซอฟต์

## 2.4 คอทเน็ตเฟรมเวิร์ก (.NET Framework)

คอทเน็ตเฟรมเวิร์ก (.NET Framework) คือกรอบในการพัฒนาซอฟต์แวร์ (software framework) ที่พัฒนาโดยไมโครซอฟต์ (Microsoft) ที่มีการทำงานหลักบนวินโดวส์ (Windows) ประกอบไปด้วยไลบรารีของคลาสต่างๆ เป็นจำนวนมาก ที่เรียกว่า เฟรมเวิร์กคลาสไลบรารี และมีตัวแปลภาษาสำหรับการเขียนโปรแกรมในภาษาต่างๆ

โปรแกรมที่เขียนโดยคอทเน็ตเฟรมเวิร์ก (.NET Framework) ทำงานภายใต้สภาพแวดล้อมของซอฟต์แวร์ที่เรียกว่า คอมมอนแลงกวิจันไทม์ (Common Language Runtime : CLR) และแอปพลิเคชันเวอร์ชวลแมชีน (Application virtual machine) ที่ทำหน้าที่ให้บริการเซอร์วิสต่าง ๆ เช่น ความปลอดภัย การจัดการหน่วยความจำ การจัดการกับข้อผิดพลาด ทั้งเฟรมเวิร์กคลาสไลบรารี และ คอมมอนแลงกวิจันไทม์ (Common Language Runtime : CLR) เป็นองค์ประกอบของ คอทเน็ตเฟรมเวิร์ก (.NET Framework)



รูปที่ 2.11 องค์ประกอบของ คอทเน็ตเฟรมเวิร์ก (.NET Framework)

## 2.5 เอกซ์เอ็มแอล (XML)

เอกซ์เอ็มแอล มีโครงสร้างที่ประกอบด้วยแท็กเปิด และแท็กปิด เช่นเดียวกับภาษา เอกซ์เอ็มแอล (HTML) แต่ภาษาเอกซ์เอ็มแอล สามารถสร้างแท็กรวมทั้งกำหนดโครงสร้างของข้อมูลได้เอง มีความสะดวกในการจัดการด้านระบบการติดต่อกับผู้ใช้จากโครงสร้างของข้อมูล สามารถนำข้อมูลจากหลายแหล่งมาแสดงผลและประมวลผลร่วมกันได้ และในส่วนของข้อมูลสามารถปรับให้เป็น เอกซ์เอ็มแอล (HTML) ได้ สำหรับประโยชน์ในการใช้งานนั้น จะสามารถนำมาใช้สำหรับการ

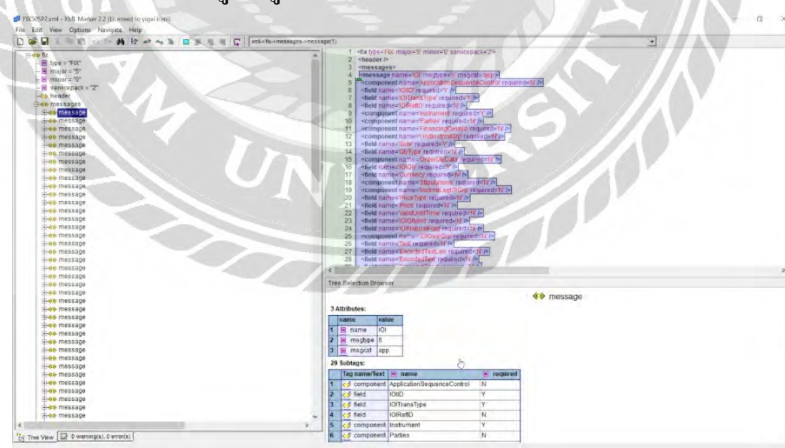
เข้าถึงระบบข้อมูลขนาดใหญ่ใช้กับระบบเครือข่ายในองค์กร หรืออินเทอร์เน็ตเพื่อดูข้อมูลหรือเรียกใช้ข้อมูลที่ทำให้การแสดงผลทางหน้าจอที่รวดเร็ว

### จุดเด่นของ เอกซ์เอ็มแอล (XML)

1. ดูเอกสารได้ง่าย สะดวก และได้ผลดี
2. สนับสนุนการประยุกต์ใช้กับงานต่าง ๆ และสนับสนุน โปรแกรมประยุกต์ต่าง ๆ
3. เขียนง่าย
4. อ่านได้โดยไม่ต้องอาศัยโปรแกรมหรือเครื่องมือช่วยแปล
5. การเขียนเอกซ์เอ็มแอล ทำได้ด้วยการใช้เครื่องมือแก้ไขข้อความ (Text Editor) ทั่วไป และไม่ต้องการเครื่องมือที่ซับซ้อน
6. ใช้เป็นตัวควบคุมข้อมูล ในการขนส่งข้อมูล และสร้างการเชื่อมโยงระหว่างแอปพลิเคชันได้ง่าย
7. สนับสนุนยูนิโคด (UNICODE) ทำให้ใช้ได้กับหลากหลายภาษา และผสมกันได้หลากหลายภาษา

### 2.6 เอกซ์เอ็มแอลมาร์กเกอร์เวอร์ชัน 1.1 (XML Marker 1.1)

เอกซ์เอ็มแอลมาร์กเกอร์ (XML Marker) เป็นเครื่องมือแก้ไข และแสดงข้อมูลเอกซ์เอ็มแอลแบบลำดับชั้นและแบบตาราง โดยจะสร้างการแสดงผลแบบตารางของแท็กที่เลือกโดยการรวบรวมแอตทริบิวต์ (Attribute) ที่ซ้ำกันและชื่อแท็กแล้วจัดเรียงเป็นคอลัมน์ (Column) ผลลัพธ์ที่ได้คือการแสดงผลข้อมูลแบบตาราง ทำให้ดูข้อมูลได้ง่าย



รูปที่ 2.12 หน้าจอแสดงผลของเอกซ์เอ็มแอล มาร์กเกอร์



### บทที่ 3

#### รายละเอียดการปฏิบัติงาน

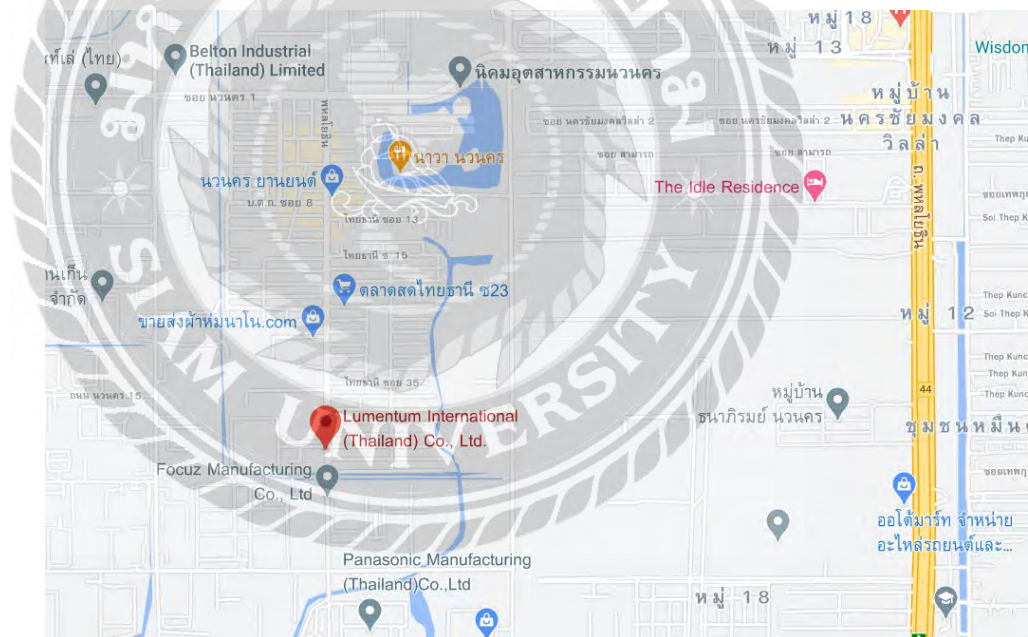
รายละเอียดของงานที่ปฏิบัติ จะกล่าวถึง ชื่อ-ที่ตั้ง ของสถานประกอบการ ลักษณะโดยรวมของสถานประกอบการ รูปแบบการบริหารองค์กร ตำแหน่งงานที่นักศึกษาได้รับมอบหมาย ระยะเวลาที่ปฏิบัติงาน ขั้นตอนวิธีการดำเนินงาน อุปกรณ์เครื่องมือที่ใช้ในการปฏิบัติงานโครงการ สหกิจ

#### 3.1 ชื่อและที่ตั้งของสถานประกอบการ

ชื่อสถานประกอบการ บริษัท ลูเมนตั้ม อินเตอร์เนชั่นแนล (ประเทศไทย) จำกัด  
(Lumentum International (Thailand) Company Limited)

ที่ตั้งสถานประกอบการ 60/129 หมู่ที่ 19 ตำบลคลองหนึ่ง อำเภอคลองหลวง จ.ปทุมธานี  
10120

[www.lumentum.com](http://www.lumentum.com)



รูปที่ 3.1 แผนที่ตั้งสถานประกอบการ

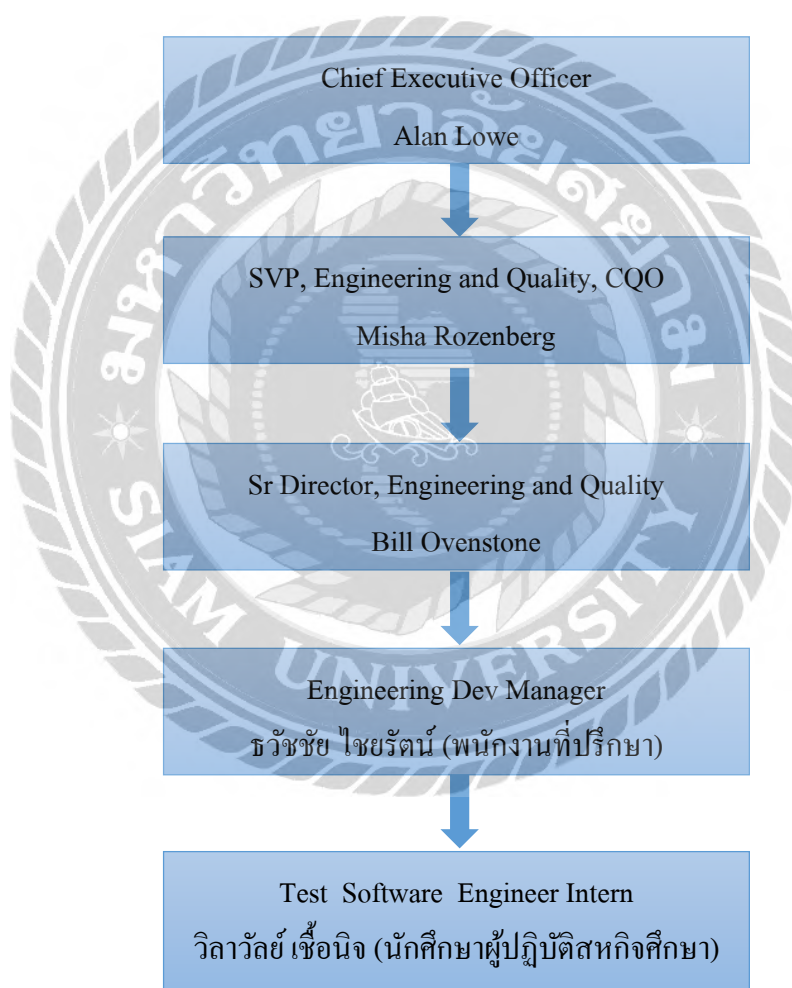
#### 3.2 ลักษณะของสถานประกอบการ

บริษัท ลูเมนตั้ม อินเตอร์เนชั่นแนล (ประเทศไทย) จำกัด (Lumentum International (Thailand) Company Limited) อำเภอคลองหลวง จังหวัดปทุมธานี ได้ดำเนินธุรกิจภายใต้หลักจรรยาบรรณแห่งพันธมิตรธุรกิจผู้มีความรับผิดชอบ (Responsible Business Alliance : RBA) ซึ่ง

ดำเนินการเกี่ยวกับการผลิตชิ้นส่วนเครื่องเลเซอร์ ชิ้นส่วนอุปกรณ์โทรคมนาคมสำหรับระบบใยแก้วนำแสง กิจกรรมประกอบเครื่องจักร อุปกรณ์เครื่องจักร และ/หรือ อุปกรณ์การสื่อสาร เป็นต้น โดยมีสำนักงานใหญ่อยู่ที่รัฐแคลิฟอร์เนีย สหรัฐอเมริกาที่ก่อตั้งมากกว่า 40 ปี และมีบริษัทในประเทศต่าง ๆ ประกอบด้วย ประเทศ แคนาดา จีน ฝรั่งเศส อิตาลี ญี่ปุ่น สโลวีเนีย มาเลเซีย เกาหลีใต้ สวิตเซอร์แลนด์ ไต้หวัน ไทย สหราชอาณาจักร และสหรัฐอเมริกา เป็นต้น อีกทั้งบริษัทยังเป็นผู้นำด้านการขนส่งและการส่งสัญญาณโทรคมนาคม 100G data center laser ships และ 3D sensing diode lasers

### 3.3 รูปแบบการจัดองค์กรและการบริหารงานขององค์กร

การบริหารงานขององค์กรในส่วนงานที่ไปปฏิบัติสหกิจศึกษาสามารถแสดงได้ดังรูปที่ 3.2



รูปที่ 3.2 การจัดองค์กรและการบริหารงานขององค์กร

### 3.4 ตำแหน่งและลักษณะงานที่ได้รับมอบหมาย

ตำแหน่งงานที่ได้รับมอบหมายคือวิศวกรทดสอบซอฟต์แวร์ระดับฝึกหัด (Test Software Engineer Intern) โดยมีหน้าที่ที่ได้รับมอบหมาย ดังนี้

3.4.1 พัฒนาโปรแกรมทดสอบซอฟต์แวร์ เพื่อสร้างไฟล์ผลการทดสอบออกมาในรูปแบบของไฟล์เอกซ์เอ็มแอล (XML)

3.4.2 พัฒนาโปรแกรม เพื่อบันทึกไฟล์ผลการทดสอบในรูปแบบของไฟล์เอกซ์เอ็มแอล (XML) ไปยังระบบฐานข้อมูลของบริษัท (Database)

3.4.3 นำข้อมูลไฟล์ผลการทดสอบจากระบบฐานข้อมูลของบริษัทไปวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติ (Statistical Process Control) เพื่อให้การทดสอบเป็นไปตามมาตรฐาน

### 3.5 ชื่อและตำแหน่งของพนักงานที่ปรึกษา

คุณ ธวัชชัย ไชยรัตน์ ตำแหน่ง ผู้จัดการฝ่ายพัฒนาวิศวกรรม ( Engineering Dev Manager)

### 3.6 ระยะเวลาที่ปฏิบัติงาน

เริ่มปฏิบัติงาน : วันที่ 11 มกราคม พ.ศ. 2564

สิ้นสุดการปฏิบัติงาน : วันที่ 30 เมษายน พ.ศ. 2564

### 3.7 แผนการดำเนินงาน

3.7.1 ศึกษาข้อมูลที่เกี่ยวข้องกับการทำโครงการ

ศึกษาข้อมูลต่างๆ ที่เกี่ยวข้องกับการทำโครงการ เช่น กระบวนการทดสอบผลิตภัณฑ์ของบริษัท และการควบคุมกระบวนการผลิตเชิงสถิติ (Statistical Process Control) โปรแกรมภาษา (Programming Language) ที่จะใช้ในการพัฒนา

3.7.2 รวบรวมความต้องการและวิเคราะห์ระบบ

รวบรวมความต้องการจากพนักงานที่ปรึกษาโดยการสัมภาษณ์ ว่ามีฟังก์ชันการทำงานใดบ้างที่บริษัทต้องการให้มีในโปรแกรมที่จะพัฒนา จากนั้นทำการวิเคราะห์ความต้องการที่รวบรวม ซึ่งพบว่าโปรแกรมจะต้องสามารถสร้างไฟล์ผลการทดสอบผลิตภัณฑ์จาก ฟังก์ชัน “Daily Check”, “Main Test” และ “Power Check” ในรูปแบบไฟล์เอกซ์เอ็มแอล รวมถึงสามารถบันทึกไฟล์เอกซ์เอ็มแอลที่ได้ไว้บนระบบฐานข้อมูล เพื่อนำไปใช้ในวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติ

3.7.3 ออกแบบระบบ

จากการวิเคราะห์โปรแกรม ผู้จัดทำได้ทำการออกแบบโปรแกรมที่จะพัฒนา โดยออกแบบการทำงานของโปรแกรมเป็น 2 ส่วนหลัก ได้แก่



1. ส่วนของการ สร้างไฟล์เอกซ์เอ็มแอลของผลการทดสอบ
2. ส่วนการบันทึกไฟล์เอกซ์เอ็มแอลของผลการทดสอบไปไว้บนระบบฐานข้อมูลของบริษัท

### 3.7.4 พัฒนาระบบ

พัฒนาโปรแกรมออกเป็น 2 ส่วนหลักตามที่ได้ออกแบบไว้ คือส่วนของการสร้างไฟล์เอกซ์เอ็มแอลและส่วนการบันทึกไฟล์เอกซ์เอ็มแอลไปไว้บนระบบฐานข้อมูลของบริษัท โดยพัฒนาโปรแกรมด้วยภาษาวิชวลเบสิกคอตเน็ต (Visual Basic.Net) และใช้ระบบฐานข้อมูลเอพีไออินเตอร์เฟซ (API interface) เป็นระบบจัดการฐานข้อมูล (Database Management System )

### 3.7.5 ทดสอบโปรแกรม

นำโปรแกรมที่พัฒนามาทำการทดสอบการทำงานในส่วนของการทำงานหลักทั้งสองส่วนตามที่ได้ออกแบบไว้ ได้แก่

1. ทดสอบสร้างผลการทดสอบในรูปแบบของไฟล์เอกซ์เอ็มแอลของทั้ง 3 ฟังก์ชัน ได้แก่ ฟังก์ชัน “Daily Check” , “Main Test” และ “Power Check”
2. การบันทึกผลการทดสอบในรูปแบบของไฟล์เอกซ์เอ็มแอลไปไว้บนระบบฐานข้อมูลของบริษัท

### 3.7.6 วิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติ

นำข้อมูลผลการทดสอบความพร้อมของเครื่องทดสอบผลิตภัณฑ์ ที่ได้จากฟังก์ชัน “Daily Check” ซึ่งเก็บอยู่ในรูปแบบของไฟล์ไฟล์เอกซ์เอ็มแอลที่อยู่บนระบบฐานข้อมูล มาพล็อตกราฟ (plot graph) บนระบบเอสพีซี (SPC System) ซึ่งเป็นระบบวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติ เพื่อใช้ข้อมูลกราฟที่ได้ในการวิเคราะห์และทำการควบคุมเครื่องทดสอบให้อยู่ในมาตรฐาน

### 3.7.7 จัดทำเอกสารรายงานผลการดำเนินงานโครงการ

จัดทำเอกสารรายงานผลการดำเนินงานโครงการ เช่น ที่มาและความสำคัญของโครงการ หลักการที่เกี่ยวข้อง และผลทดสอบการทำงานของโปรแกรม ข้อดี ข้อเสีย และข้อเสนอแนะต่างๆ

จากขั้นตอนการทำงานดังกล่าว ผู้จัดทำได้นำหัวข้อการทำงานและกำหนดระยะเวลาในการดำเนินงานของโครงการมากำหนดเป็นตารางแผนการดำเนินงาน ดังตารางที่ 3.1

ตารางที่ 3.1 ตารางแผนการดำเนินงานของโครงการ

| ที่ | หัวข้องาน                                                                                                                  | 2564   |            |        |        |
|-----|----------------------------------------------------------------------------------------------------------------------------|--------|------------|--------|--------|
|     |                                                                                                                            | มกราคม | กุมภาพันธ์ | มีนาคม | เมษายน |
| 1   | ศึกษาเกี่ยวกับโครงการที่สถานประกอบการมอบหมายงานให้                                                                         | ↔      |            |        |        |
| 2   | รวบรวมและวิเคราะห์ความต้องการของโปรแกรมที่จะพัฒนา                                                                          | ↔      |            |        |        |
| 3   | ศึกษาโครงสร้างของไฟล์ เอกซ์เอ็มแอล และ โปรแกรมการสร้างไฟล์เอกซ์เอ็มแอล (XML) ด้วยภาษาวิซวลเบสิก คอทเน็ต (Visual Basic.Net) | ↔      |            |        |        |
| 4   | ออกแบบโปรแกรมที่จะพัฒนาของฟังก์ชัน Daily Check                                                                             | ↔      |            |        |        |
| 5   | พัฒนาโปรแกรมสร้างไฟล์เอกซ์เอ็มแอล (XML) ของฟังก์ชัน Daily Check                                                            | ↔      |            |        |        |
| 6   | ทดสอบโปรแกรมฟังก์ชัน Daily Check บนลายการผลิต                                                                              | ↔      |            |        |        |
| 7   | ออกแบบโปรแกรมที่จะพัฒนาของฟังก์ชัน Main Test และ Power Check                                                               | ↔      |            |        |        |
| 8   | พัฒนาโปรแกรมสร้างไฟล์เอกซ์เอ็มแอล (XML) ของฟังก์ชัน Main Test และ Power Check                                              |        | ↔          |        |        |
| 9   | ทดสอบโปรแกรมฟังก์ชัน Main Test และ Power Check บนลายผลิต                                                                   |        | ↔          |        |        |
| 10  | วิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติ (Statistical Process Control)                                                     |        |            | ↔      | ↔      |
| 11  | ปรับปรุงและแก้ไขข้อผิดพลาดที่เกิดขึ้นก่อนส่งมอบโครงการให้กับบริษัท                                                         |        |            | ↔      | ↔      |
| 12  | จัดทำเอกสารรายงาน                                                                                                          |        |            | ↔      | ↔      |

### 3.8 อุปกรณ์และเครื่องมือที่ใช้

ฮาร์ดแวร์

- คอมพิวเตอร์โน้ตบุ๊ก 1 เครื่อง
- เครื่องทดสอบซอฟต์แวร์ 1 เครื่อง

ซอฟต์แวร์

- วิซวล สตูดิโอ (Visual Studio 2019) 1 ชุด
- เอกซ์เอ็มแอล มาร์กเกอร์ (Xml Marker 1.1) 1 ชุด



## บทที่ 4

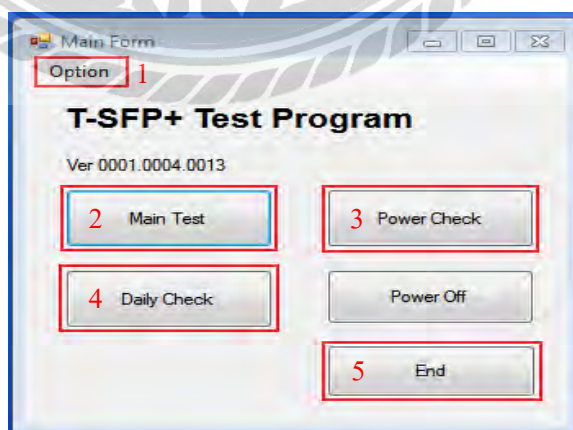
### ผลการปฏิบัติงาน

ผลการปฏิบัติงานตามโครงการพัฒนาโปรแกรมทดสอบซอฟต์แวร์ เพื่อวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติเป็นไปตามความต้องการของบริษัทกำหนดไว้ แต่เนื่องจากข้อมูลของโครงการบางส่วนเป็นสิ่งที่บริษัทต้องคำนึงถึงความปลอดภัยในการนำไปเผยแพร่สู่สาธารณะ ดังนั้นผู้จัดทำจึงนำเสนอผลการทำงานของโปรแกรมทดสอบซอฟต์แวร์ เพื่อวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติมาเพียงบางส่วนเท่าที่บริษัทจะอนุญาตให้สามารถเผยแพร่ได้เท่านั้น

#### 4.1 โปรแกรมทดสอบซอฟต์แวร์เพื่อวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติ

หน้าจอหลักของโปรแกรมทดสอบซอฟต์แวร์ เพื่อวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติชื่อ Main Form โดยเป็นหน้าจอแรกของการเข้าใช้งานโปรแกรม ซึ่งจะประกอบไปด้วยส่วนหลักต่างๆ 5 ส่วนหลัก ดังรูปที่ 4.1.1

1. หมายเลข 1 Option เป็นส่วนที่ใช้ในการเลือกโหมดหรือตั้งค่าการทำงานของโปรแกรม
2. หมายเลข 2 Main Test เป็นปุ่มที่ใช้ในการทดสอบซอฟต์แวร์ฟังก์ชัน “Main Test”
3. หมายเลข 3 Power Check เป็นปุ่มที่ใช้ในการทดสอบซอฟต์แวร์ฟังก์ชัน “Power Check”
4. หมายเลข 4 Daily Check เป็นปุ่มที่ใช้ในการทดสอบซอฟต์แวร์ฟังก์ชัน “Daily Check”
5. หมายเลข 5 End เป็นปุ่มที่ใช้ในการปิดโปรแกรม



รูปที่ 4.1.1 หน้าจอหลักของโปรแกรมทดสอบซอฟต์แวร์  
เพื่อวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติ

#### 4.2 การทดสอบซอฟต์แวร์ฟังก์ชัน “Main Test”

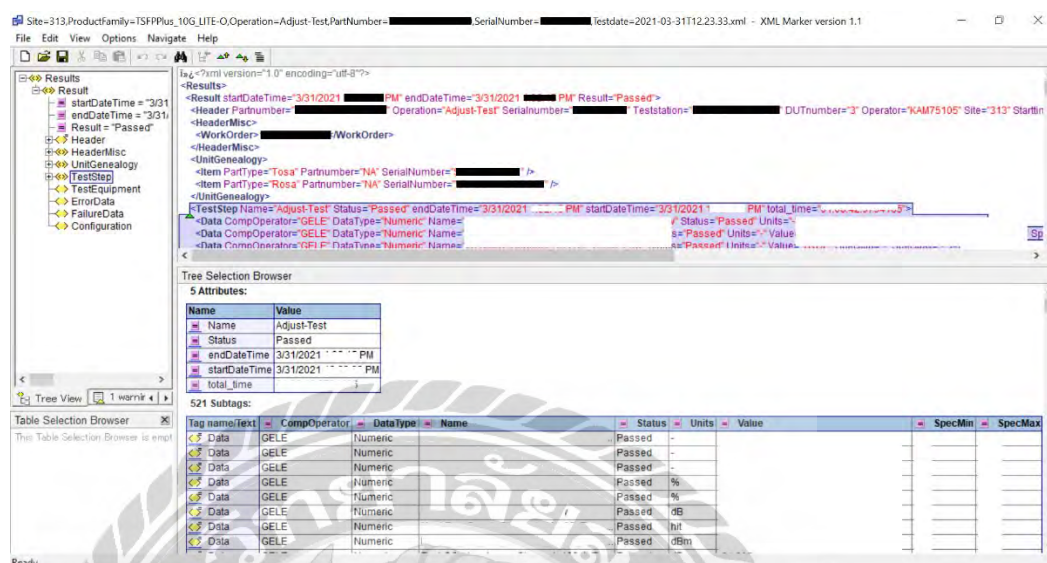
เมื่อผู้ดำเนินการ (Operator) กดปุ่ม Main Test เพื่อจะทำการทดสอบซอฟต์แวร์ฟังก์ชัน “Main Test” โปรแกรมจะเข้าสู่หน้าจอ frmInput ดังรูปที่ 4.2.1 เพื่อให้ผู้ดำเนินการกรอกข้อมูลก่อนที่จะทำการทดสอบซอฟต์แวร์

รูปที่ 4.2.1 หน้าจอกรอกข้อมูลของฟังก์ชัน “Main Test”

หลังจากผู้ดำเนินการกรอกข้อมูลเสร็จแล้ว จะกดปุ่ม Next เพื่อให้โปรแกรมเริ่มดำเนินงาน ในส่วนของการทดสอบซอฟต์แวร์ฟังก์ชัน Main Test จากนั้นโปรแกรมจะเข้าสู่หน้าจอ Measure Progress เพื่อทำการทดสอบซอฟต์แวร์ ดังรูปที่ 4.2.2

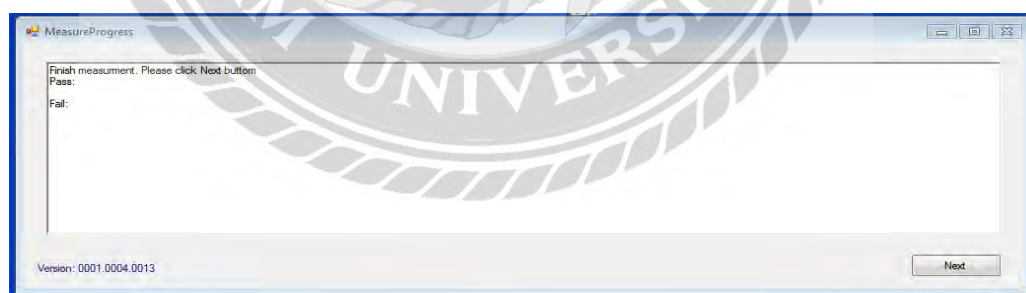
รูปที่ 4.2.2 หน้าจอทดสอบซอฟต์แวร์ของฟังก์ชัน “Main Test”

เมื่อโปรแกรมทำการทดสอบซอฟต์แวร์ฟังก์ชัน “Main Test” เสร็จสิ้น จะสร้างไฟล์ผลการทดสอบในรูปแบบของเอกซ์เอ็มแอล และบันทึกเก็บไว้บนระบบฐานข้อมูล ดังรูปที่ 4.2.3



รูปที่ 4.2.3 ไฟล์เอกซ์เอ็มแอลของฟังก์ชัน Main Test

ซึ่งบนหน้าจอของโปรแกรม จะปรากฏข้อความว่า "Finish measurement. Please click Next bottom" และผลลัพธ์การทดสอบของแต่ละชิ้นงานของผลิตภัณฑ์ว่า ผ่าน (Pass) หรือไม่ผ่าน (Fail) ดังรูปที่ 4.2.4 จากนั้นให้กดปุ่ม Next เพื่อกลับสู่หน้าจอ Main Form



รูปที่ 4.2.4 หน้าจอผลการทดสอบของฟังก์ชัน “Main Test”

#### 4.3 การทดสอบซอฟต์แวร์ฟังก์ชัน “Power Check”

เมื่อผู้ดำเนินการ (Operator) กดปุ่ม Power Check เพื่อจะทำการทดสอบซอฟต์แวร์ฟังก์ชัน Power Check โปรแกรมจะเข้าสู่หน้าจอ frmInput ดังรูปที่ 4.3.1 เพื่อจะให้ผู้ดำเนินการกรอกข้อมูลก่อนที่จะทำการทดสอบซอฟต์แวร์



Version: 0001.0004.0013

MeasureTemp: ☒ Low ☒ Mid ☒ High

InputMode: ☒ FITS ☐ Manual

**Power Check**

Operator:

Information:

| Ch                                    | P/N                  | W/O                  | S/N                  |
|---------------------------------------|----------------------|----------------------|----------------------|
| <input checked="" type="checkbox"/> 1 | <input type="text"/> | <input type="text"/> | <input type="text"/> |
| <input checked="" type="checkbox"/> 2 | <input type="text"/> | <input type="text"/> | <input type="text"/> |
| <input checked="" type="checkbox"/> 3 | <input type="text"/> | <input type="text"/> | <input type="text"/> |
| <input checked="" type="checkbox"/> 4 | <input type="text"/> | <input type="text"/> | <input type="text"/> |

Cancel Next

รูปที่ 4.3.1 หน้าจกรอกข้อมูลของฟังก์ชัน “Power Check”

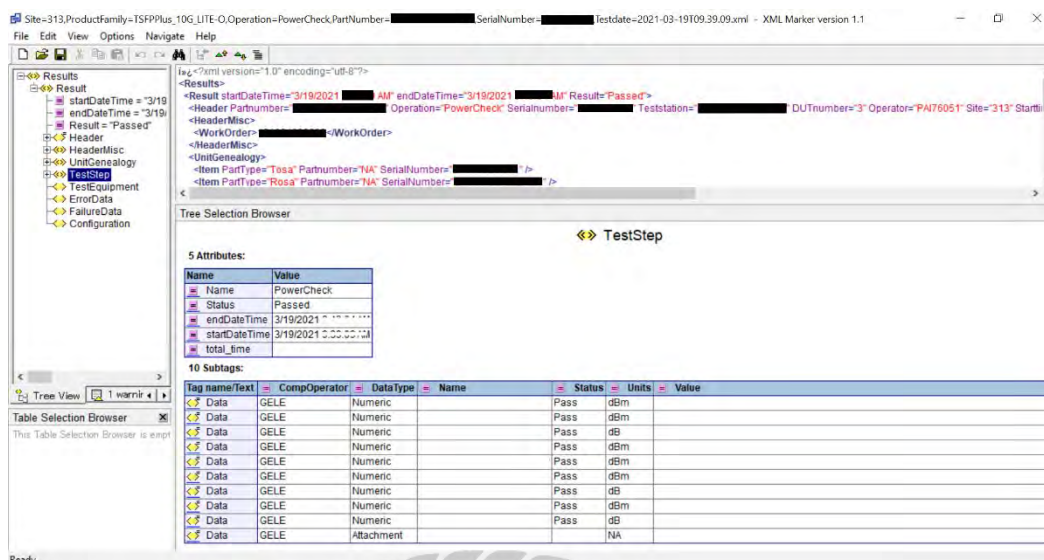
หลังจากผู้ดำเนินการกรอกข้อมูลเสร็จแล้ว จะกดปุ่ม Next เพื่อให้โปรแกรมเริ่มดำเนินงาน ในส่วนของการทดสอบซอฟต์แวร์ฟังก์ชัน “Power Check” จากนั้นโปรแกรมจะเข้าสู่หน้าจอ Measure Progress เพื่อทำการทดสอบซอฟต์แวร์ ดังรูปที่ 4.3.2

Version: 0001.0004.0013

| Ch | ProductName | SerialNumber | WorkOrder | Start                 | Stop                 | Message | Progress |
|----|-------------|--------------|-----------|-----------------------|----------------------|---------|----------|
| 1  |             |              | TEST      | 3/10/2021 11:09:17 AM | 1/1/0001 12:00:00 AM |         | 17%      |

รูปที่ 4.3.2 หน้าจอทดสอบซอฟต์แวร์ของฟังก์ชัน Power Check

เมื่อโปรแกรมทำการทดสอบซอฟต์แวร์ฟังก์ชัน “Power Check” เสร็จสิ้น จะสร้างไฟล์ผลการทดสอบในรูปแบบของเอกซ์เอ็มแอล และบันทึกเก็บไว้บนระบบฐานข้อมูล ดังรูปที่ 4.3.3



รูปที่ 4.3.3 ไฟล์เอกซ์เอ็มแอลของฟังก์ชัน Power Check

ซึ่งบนหน้าจอของโปรแกรม จะปรากฏข้อความว่า "Finish measurement. Please click Next bottom" และผลลัพธ์การทดสอบของแต่ละชิ้นงานของผลิตภัณฑ์ว่า ผ่าน (Pass) หรือไม่ผ่าน (Fail) ดังรูปที่ 4.3.4 จากนั้นให้กดปุ่ม Next เพื่อกลับสู่หน้าจอ Main Form



รูปที่ 4.3.4 แสดงข้อความและผลลัพธ์การทดสอบของฟังก์ชัน Power Check

#### 4.4 การทดสอบซอฟต์แวร์ฟังก์ชัน “Daily Check”

เมื่อผู้ดำเนินการ (Operator) กดปุ่ม Daily Check เพื่อจะทำการทดสอบซอฟต์แวร์ฟังก์ชัน Daily Check โปรแกรมจะเข้าสู่หน้าจอ frmInput ดังรูปที่ 4.4.1 เพื่อให้ผู้ดำเนินการกรอกข้อมูลก่อนที่จะทำการทดสอบซอฟต์แวร์

Version: 0001.0004.0013

MeasureTemp: ☐ Low ☒ Mid ☐ High

InputMode: ☒ FITS ☐ Manual

**Daily Check**

Operator: [Dropdown]

| Ch                                    | P/N        | W/O        | S/N        |
|---------------------------------------|------------|------------|------------|
| <input checked="" type="checkbox"/> 1 | [Redacted] | [Redacted] | [Redacted] |
| <input checked="" type="checkbox"/> 2 | [Redacted] | [Redacted] | [Redacted] |
| <input checked="" type="checkbox"/> 3 | [Redacted] | [Redacted] | [Redacted] |
| <input checked="" type="checkbox"/> 4 | [Redacted] | [Redacted] | [Redacted] |

Buttons: Cancel, Next

รูปที่ 4.4.1 หน้าจอ frmInput ของฟังก์ชัน Daily Check

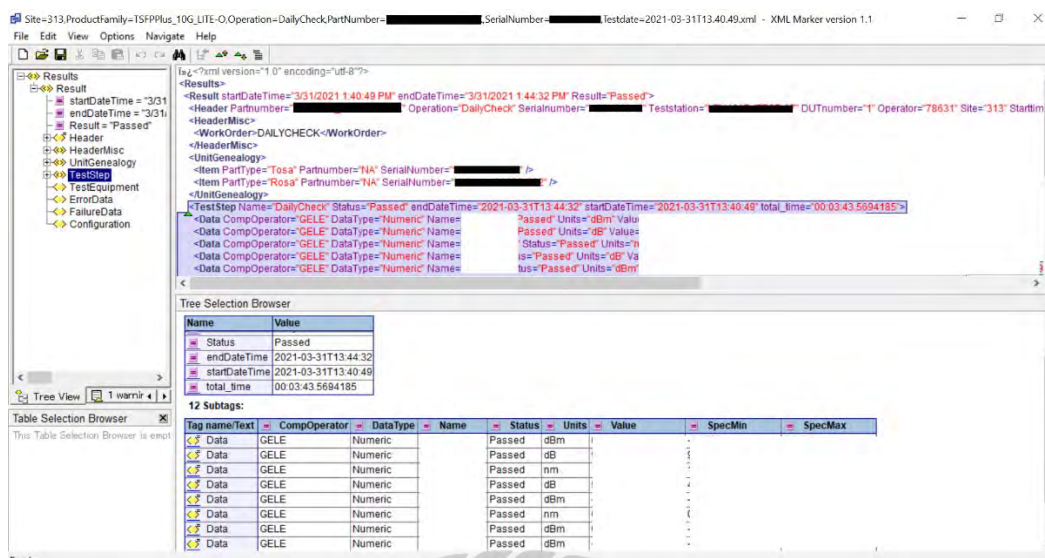
หลังจากผู้ดำเนินการกรอกข้อมูลเสร็จแล้ว จะกดปุ่ม Next เพื่อให้โปรแกรมเริ่มดำเนินงาน ในส่วนของการทดสอบซอฟต์แวร์ฟังก์ชัน Daily Check จากนั้นโปรแกรมจะเข้าสู่หน้าจอ Measure Progress เพื่อทำการทดสอบซอฟต์แวร์ ดังรูปที่ 4.4.2

| Ch | ProductName | SerialNumber | WorkOrder  | Start                 | Stop                 | Message | Progress |
|----|-------------|--------------|------------|-----------------------|----------------------|---------|----------|
| 1  | [Redacted]  | [Redacted]   | [Redacted] | 3/10/2021 10:49:21 AM | 1/1/0001 12:00:00 AM |         | 50%      |
| 2  | [Redacted]  | [Redacted]   | [Redacted] | 3/10/2021 10:49:21 AM | 1/1/0001 12:00:00 AM |         | 67%      |
| 3  | [Redacted]  | [Redacted]   | [Redacted] | 3/10/2021 10:49:21 AM | 1/1/0001 12:00:00 AM |         | 33%      |
| 4  | [Redacted]  | [Redacted]   | [Redacted] | 3/10/2021 10:49:21 AM | 1/1/0001 12:00:00 AM |         | 67%      |

Version: 0001.0004.0025

รูปที่ 4.4.2 หน้าจอหน้าจอทดสอบซอฟต์แวร์ของฟังก์ชัน ของฟังก์ชัน Daily Check

เมื่อโปรแกรมทำการทดสอบซอฟต์แวร์ฟังก์ชัน “Daily Check” เสร็จสิ้น จะสร้างไฟล์ผลการทดสอบในรูปแบบของเอกซ์เอ็มแอล และบันทึกเก็บไว้ในระบบฐานข้อมูล ดังรูปที่ 4.4.3



รูปที่ 4.4.3 ไฟล์เอกซ์เอ็มแอลของฟังก์ชัน Daily Check

ซึ่งบนหน้าจอของโปรแกรม จะปรากฏข้อความว่า "Finish measurement. Please click Next bottom" และผลลัพธ์การทดสอบของแต่ละชิ้นงานของผลิตภัณฑ์ว่า ผ่าน (Pass) หรือไม่ผ่าน (Fail) ดังรูปที่ 4.4.4 จากนั้นให้กดปุ่ม Next เพื่อกลับสู่หน้าจอ Main Form



รูปที่ 4.4.4 แสดงข้อความการทำงานของฟังก์ชัน Daily Check

#### 4.5 การวิเคราะห์ข้อมูลเพื่อควบคุมกระบวนการเชิงสถิติ (Statistical Process Control)

เมื่อโปรแกรมการทดสอบซอฟต์แวร์เสร็จสิ้นทั้ง 3 ฟังก์ชันแล้ว ไฟล์เอกซ์เอ็มแอล(XML) ของผลการทดสอบจะถูกบันทึกไปไว้บนฐานระบบข้อมูลเอพีไออินเตอร์เฟส (API interface) ซึ่งเป็นระบบจัดการฐานข้อมูล (Database Management System) ดังรูปที่ 4.5.1 เพื่อนำข้อมูลไปวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติ (Statistical Process Control)



**LUMENTUM**

Welcome | Witsawat Chuanrit (Test Engineer)

Home / Search / Test Result

Serial number Start time End time Product Family Name Operation name

Part number

Q Search Refresh

Table Filter

| ACTIONS     | TEST HEAD ID | SERIAL NUMBER | PRODUCT FAMILY NAME | PART NUMBER | OPERATION NAME | RESULT | MISCINFO | SITE CODE | SITE NAME |
|-------------|--------------|---------------|---------------------|-------------|----------------|--------|----------|-----------|-----------|
| NAVIGATE TO | 2284583      |               | TSFPPlus_10G_LITE-O |             | Adjust-Test    | Passed |          | 313       | Nava      |
| NAVIGATE TO | 2284584      |               | TSFPPlus_10G_LITE-O |             | Adjust-Test    | Passed |          | 313       | Nava      |
| NAVIGATE TO | 2284582      |               | TSFPPlus_10G_LITE-O |             | Adjust-Test    | Passed |          | 313       | Nava      |
| NAVIGATE TO | 2284581      |               | TSFPPlus_10G_LITE-O |             | Adjust-Test    | Passed |          | 313       | Nava      |
| NAVIGATE TO | 2284545      |               | TSFPPlus_10G_LITE-O |             | DailyCheck     | Passed |          | 313       | Nava      |
| NAVIGATE TO | 2284544      |               | TSFPPlus_10G_LITE-O |             | DailyCheck     | Failed |          | 313       | Nava      |
| NAVIGATE TO | 2284543      |               | TSFPPlus_10G_LITE-O |             | DailyCheck     | Failed |          | 313       | Nava      |
| NAVIGATE TO | 2284542      |               | TSFPPlus_10G_LITE-O |             | DailyCheck     | Passed |          | 313       | Nava      |
| NAVIGATE TO | 2284400      |               | TSFPPlus_10G_LITE-O |             | Adjust-Test    | Passed |          | 313       | Nava      |
| NAVIGATE TO | 2284399      |               | TSFPPlus_10G_LITE-O |             | Adjust-Test    | Passed |          | 313       | Nava      |

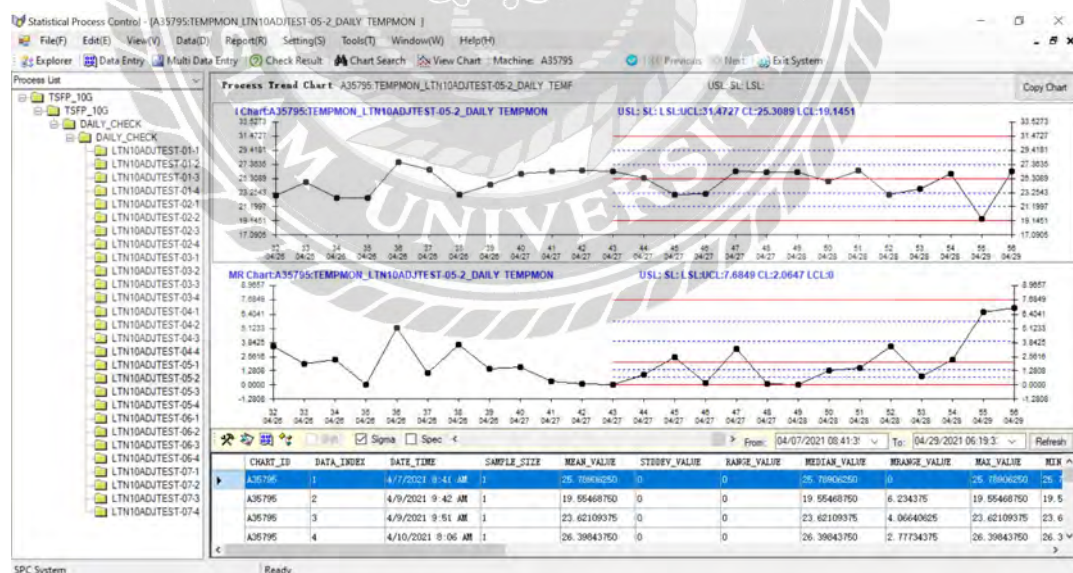
Records Count : 3229 (3.06 Sec)

UI - 1.2.2 | API - 1.7.6.0 | Ingestion DLL - 1.4.8.0 | Data Uploader DLL - 1.0.2.4

Test Report | User Guide | Lumentum © 2021

### รูปที่ 4.5.1 ระบบฐานข้อมูลเอพียออนไลน์

และเมื่อผลการทดสอบฟังก์ชัน “Daily Check” อยู่บนระบบฐานข้อมูลเรียบร้อยแล้ว ระบบเอสพีซี (SPC System) ซึ่งเป็นระบบวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติ จะทำการผลการทดสอบดังกล่าวมาพล็อตกราฟ (plot graph) ดังรูปที่ 4.5.2 เพื่อใช้ในการวิเคราะห์และทำการควบคุมเครื่องทดสอบให้อยู่ในมาตรฐาน



### รูปที่ 4.5.2 ระบบเอสพีซี

## บทที่ 5

### สรุปรายงานและข้อเสนอแนะ

ผลจากการปฏิบัติงานสหกิจศึกษาในบริษัทลูเมนต้า อินเทอร์เน็ต (ประเทศไทย) จำกัด ในตำแหน่งวิศวกรทดสอบซอฟต์แวร์ระดับฝึกหัด (Software Test Engineer Intern) ซึ่งได้รับมอบหมายให้พัฒนาโปรแกรมทดสอบซอฟต์แวร์เพื่อวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติ สามารถสรุปรายงานและข้อเสนอแนะของผลการปฏิบัติงานได้เป็น 2 ส่วนหลักๆ ดังนี้

#### 5.1 สรุปผลโครงการ

จากการนำโปรแกรมที่พัฒนาไปทดสอบการใช้งานในบริษัท พบว่ามีผลสรุป ข้อจำกัด และข้อเสนอแนะ ดังนี้

##### 5.1.1 ผลสรุปการทำโครงการ

โปรแกรมทดสอบซอฟต์แวร์ที่พัฒนาขึ้น สามารถนำไปใช้งานร่วมกับระบบงานเดิมของบริษัทได้ตรงตามความต้องการของบริษัท โดยโปรแกรมสามารถสร้างไฟล์เอกซ์เอ็มแอลจากผลการทดสอบซอฟต์แวร์ฟังก์ชัน “Main Check” ฟังก์ชัน “Power Check” และฟังก์ชัน “Daily Check” ในกระบวนการทดสอบผลิตภัณฑ์ และนำไปจัดเก็บบนระบบฐานข้อมูลซึ่งใช้โปรแกรมเอพีไอ อินเทอร์เน็ต (API Interface) เป็นระบบจัดการฐานข้อมูล เพื่อให้โปรแกรมเอสพีซี (SPC) ซึ่งเป็นระบบการควบคุมกระบวนการผลิตเชิงสถิติ (Statistical Process Control) นำไฟล์เอกซ์เอ็มแอลจากการทดสอบซอฟต์แวร์ฟังก์ชัน “Daily Check” ที่จัดเก็บบนระบบฐานข้อมูลมาทำการวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติ เพื่อควบคุมเครื่องทดสอบให้อยู่ในมาตรฐานตามที่กำหนดและได้ผลิตภัณฑ์ที่มีคุณภาพตามมาตรฐาน

##### 5.1.2 ข้อจำกัดของโครงการ

โปรแกรมทดสอบซอฟต์แวร์ที่พัฒนาสามารถทำงานได้บนเครื่องทดสอบผลิตภัณฑ์ที่ใช้งานด้วยบัญชีผู้ใช้ (Account) ที่ตรงกับเครื่องทดสอบเท่านั้น ซึ่งถ้าเครื่องทดสอบเข้าใช้งานด้วยบัญชีผู้ใช้ (Account) ที่ไม่ใช่ของเครื่องทดสอบ จะทำให้โปรแกรมไม่สามารถเข้าถึงข้อมูลบนเน็ตเวิร์คแชร์ไดรฟ์ (Network Share Drive) ซึ่งเป็นสาเหตุให้ไม่สามารถบันทึกไฟล์เอกซ์เอ็มแอลที่โปรแกรมทดสอบซอฟต์แวร์สร้างขึ้นไปยังระบบฐานข้อมูลได้

##### 5.1.3 ข้อเสนอแนะ

บริษัทควรกำหนดนโยบายให้การใช้งานเครื่องทดสอบผลิตภัณฑ์ ต้องใช้บัญชีผู้ใช้ที่ตรงกับของเครื่องทดสอบเท่านั้น



## 5.2 สรุปผลการปฏิบัติงานสหกิจศึกษา

จากการไปปฏิบัติงานสหกิจศึกษา พบว่ามีข้อดี ข้อจำกัด และข้อเสนอแนะ ดังนี้

### 5.2.1 ข้อดีของการปฏิบัติงานสหกิจศึกษา

1. ได้ศึกษาและฝึกทักษะการเขียนโปรแกรมเชิงวัตถุ (Object-Oriented Programming: OOP)
2. ได้ศึกษาและฝึกทักษะการเขียนโปรแกรมด้วยภาษาวิซวลเบสิกเน็ต (Visual Basic.Net)
3. ได้ฝึกทักษะในการสื่อสารภาษาอังกฤษ เช่น การติดต่อกับงาน การส่งอีเมลล์
4. ได้ฝึกทักษะการทำงานร่วมกับคนในทีม
5. ได้เข้าใจการระบบการทำงานในบริษัท และสามารถนำไปประยุกต์ใช้ในการประกอบอาชีพในอนาคต
6. ได้ฝึกการมีความรับผิดชอบต่องานที่ได้รับมอบหมาย
7. ได้รับความรู้ และประสบการณ์ใหม่ๆ จากบุคลากรที่มีความรู้ความสามารถในองค์กร
8. ได้ฝึกการปรับตัวให้เข้ากับบุคคลและสถานประกอบการที่ปฏิบัติงาน

### 5.2.2 ข้อจำกัดของการปฏิบัติงานสหกิจศึกษา

1. การสื่อสารภาษาอังกฤษกับพนักงานชาวต่างชาติของบริษัท
2. การเขียนโปรแกรมเชิงวัตถุ (Object-Oriented Programming: OOP)
3. ภาษาโปรแกรมที่ใช้ในการพัฒนาโปรแกรม

### 5.2.3 ข้อเสนอแนะ

1. ฝึกทักษะการสื่อสารภาษาอังกฤษทั้งการฟัง พูด อ่านเขียน
2. ศึกษาและฝึกทักษะแนวคิดการเขียน โปรแกรมเชิงวัตถุเพิ่มเติม จากแหล่งข้อมูลต่างๆ เช่น บทเรียนออนไลน์ หรือ YouTube
3. ฝึกทักษะการเรียนรู้ด้วยตนเองจากแหล่งข้อมูลต่างๆ เพื่อพร้อมรับการเรียนรู้ภาษาโปรแกรม เนื่องจากแต่ละองค์กรใช้ภาษาโปรแกรมที่แตกต่างกัน

## บรรณานุกรม

งานบริหารทรัพยากรสุขภาพ. (ม.ป.ป.). การอ่านแผนภูมิควบคุม(Control Chart). เข้าถึงได้จาก

[https://www.si.mahidol.ac.th/th/division/um/admin/download\\_files/52\\_48\\_1mg4rg6.pdf](https://www.si.mahidol.ac.th/th/division/um/admin/download_files/52_48_1mg4rg6.pdf)

พชัย ขาวเหลือง.(2550). การควบคุมคุณภาพเชิงสถิติ. เข้าถึงได้จาก

<http://www.math.cmru.ac.th/web56/option/blog/blogdata/200417002643.pdf>

Marcuscode. (2559). แนะนำภาษา Visual Basic. เข้าถึงได้จาก

<http://marcuscode.com/lang/visual-basic/introduction>

Mindphp. (2560). XML คืออะไร. เข้าถึงได้จาก

<https://www.mindphp.com/คู่มือ/73-คืออะไร/2129-xml-คืออะไร.html>

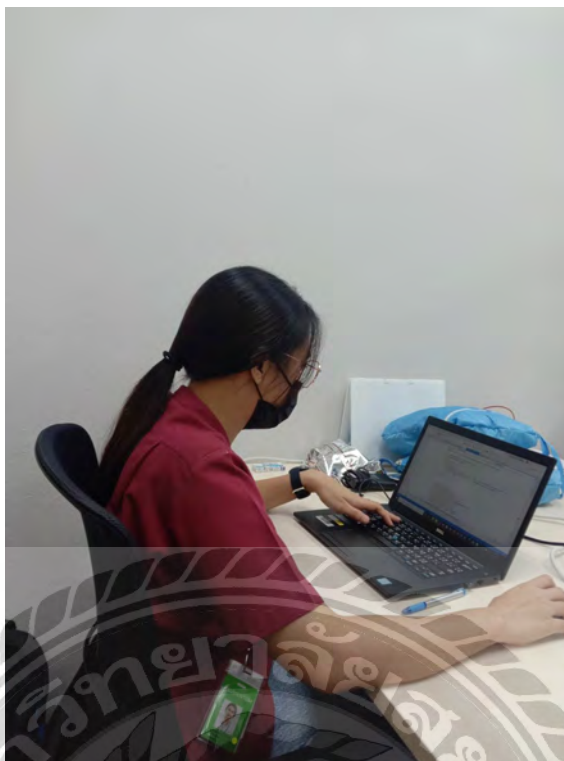
Spc Thailand. (2562). Statistical process control (SPC)คือ?. เข้าถึงได้จาก

<https://www.spc-thailand.com/post/statistical-process-control-spc>

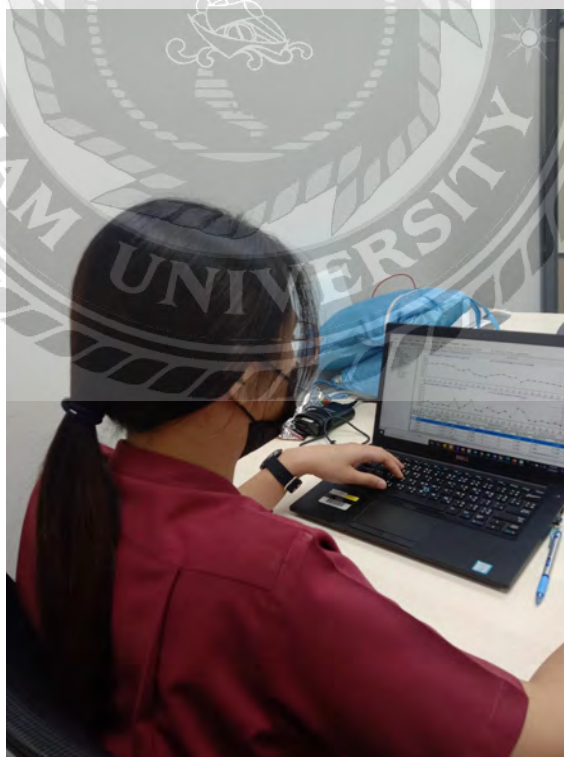




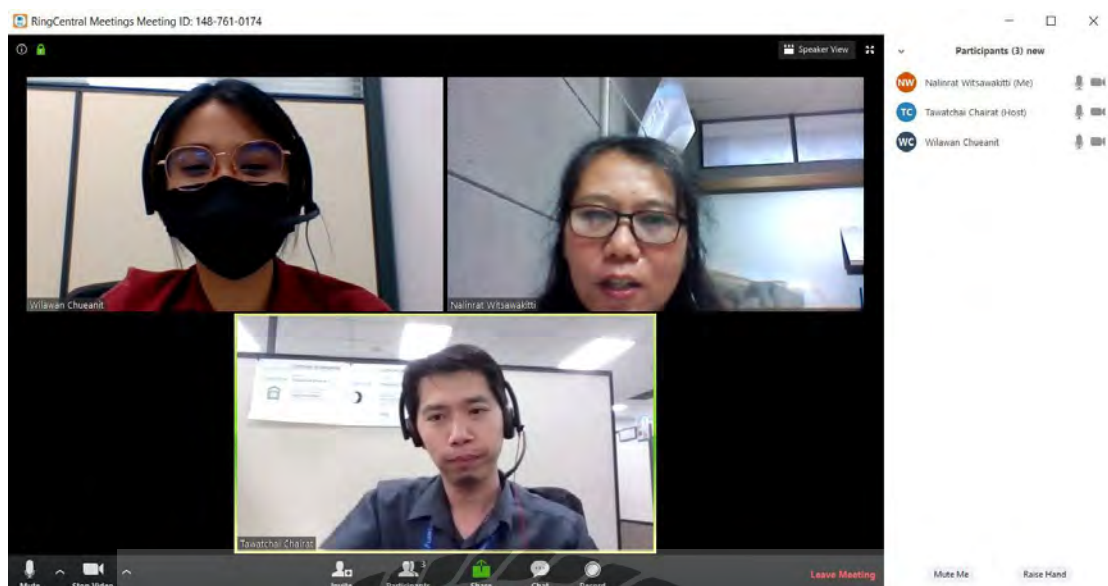
ภาคผนวก ก  
ภาพขณะปฏิบัติงานและการนิเทศของอาจารย์



รูปที่ ก.1 การเขียนโปรแกรมในการพัฒนาโปรแกรมทดสอบซอฟต์แวร์



รูปที่ ก.2 การนำข้อมูลมาวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติ



รูปที่ ก.3 การนิเทศงานสหกิจศึกษาครั้งที่ 1 วันที่ 17 กุมภาพันธ์ 2564  
(ช่วงการระบาดของโรคติดเชื้อไวรัสโคโรนา 2019 ระลอกใหม่เดือนมกราคม 2564)



รูปที่ ก.4 การนิเทศงานสหกิจศึกษาครั้งที่ 2 วันที่ 22 เมษายน 2564  
(ช่วงการระบาดของโรคติดเชื้อไวรัสโคโรนา 2019 ระลอกใหม่เดือนเมษายน 2564)



ภาคผนวก ข  
บทความวิชาการ



# การพัฒนาโปรแกรมทดสอบซอฟต์แวร์เพื่อวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติ

## Development of Software Testing Programs to Analyze the Statistical Process Control in Production Process

วิลาวลัย เชื้อนิจ<sup>1</sup>

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยสยาม

Email : wilawan.chu@siam.edu<sup>1</sup>

### บทคัดย่อ

การทดสอบผลิตภัณฑ์ในแต่ละวันของบริษัท ลูเมนตัม อินเตอร์เนชั่นแนล (ประเทศไทย) จำกัด ประกอบด้วย 2 ส่วน ได้แก่ การทดสอบเครื่องทดสอบ โดยบันทึกผลการทดสอบในรูปแบบของไฟล์เอกซ์เซล และการทดสอบผลิตภัณฑ์บนเครื่องทดสอบโดยบันทึกผลการทดสอบในรูปแบบของไฟล์ข้อความ แต่เนื่องจากการทดสอบทั้ง 2 ส่วนไม่ได้มีการจัดเก็บข้อมูลผลของการทดสอบในลักษณะของฐานข้อมูลที่ใช้งานร่วมกันได้ จึงทำให้ยากต่อการนำข้อมูลไปใช้วิเคราะห์การควบคุมกระบวนการเชิงสถิติของกระบวนการผลิตเพื่อใช้ควบคุมเครื่องทดสอบและการทดสอบผลิตภัณฑ์ให้มีคุณภาพตามมาตรฐาน

ดังนั้นจึงมีแนวคิดที่จะพัฒนาโปรแกรมทดสอบซอฟต์แวร์เพื่อวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติด้วยภาษาวิซวลเบสิกคอตเน็ต โดยโปรแกรมจะสร้างไฟล์เอกซ์เอ็มแอลจากผลการทดสอบเครื่องทดสอบและการทดสอบผลิตภัณฑ์บนเครื่องทดสอบ เพื่อนำไปจัดเก็บเป็นข้อมูลในระบบฐานข้อมูลสำหรับการใช้ในการวิเคราะห์กระบวนการผลิต ซึ่งจากผลการทดสอบโปรแกรมที่พัฒนาพบว่า สามารถนำผลการทดสอบผลิตภัณฑ์ในแต่ละวันของบริษัท ลูเมนตัม อินเตอร์เนชั่นแนล (ประเทศไทย) จำกัด จากระบบฐานข้อมูลของบริษัท มาแสดงผลในลักษณะของกราฟสำหรับการใช้ในการวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติได้

### Abstract

The daily product testing of Lumentum International (Thailand) Co., Ltd. consisted of 2 parts testing machine test with the test results saved in Excel file format, and product testing on the testing machine by saved the test results in a text file. Because of the two parts of the testing, neither results of the tests were stored in a compatible database. This made it difficult to use the data to analyze the statistical process control of the manufacturing process in order to control the testing machine and product testing to meet the standard quality.

Therefore, the idea was to development a software testing program to analyze the statistical process control with Visual Basic.Net language. The program will generate an XML file from the test machine test results and the product test on the tester, to be stored as data in a database for use in the analysis of the production process. The test results of the developed program found that the results of daily product testing of Lumentum International (Thailand) Co., Ltd. can be obtained from the company's database, and to showed the results in a graph for use in the analysis of the statistical process control.

### วัตถุประสงค์

1. การพัฒนาโปรแกรมทดสอบซอฟต์แวร์เพื่อวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติ
2. เพื่อสร้างไฟล์เอกซ์เอ็มแอลจากผลการทดสอบเครื่องทดสอบและผลิตภัณฑ์ของบริษัท
3. เพื่อวิเคราะห์ข้อมูลที่ได้จากผลการทดสอบเครื่องทดสอบและผลิตภัณฑ์ของบริษัท

### ขอบเขต

1. โปรแกรมสามารถสร้างไฟล์เอกซ์เอ็มแอล (XML) ของผลการทดสอบซอฟต์แวร์จากฟังก์ชัน “Daily Check” ฟังก์ชัน “Main Test” และ “Power Check”
2. ไฟล์เอกซ์เอ็มแอลที่สร้างจะถูกจัดเก็บไว้บนระบบฐานข้อมูล (Database System) ของบริษัท
3. โปรแกรมสามารถทำงานร่วมกับโปรแกรมของบริษัทที่มีอยู่เดิมได้
4. โปรแกรมสามารถตรวจสอบการทำงานของเครื่องทดสอบตามพารามิเตอร์ที่สำคัญ (Key parameters) ที่บริษัทกำหนดได้
5. โปรแกรมใช้ในการตรวจสอบการทำงานของเครื่องทดสอบแบบวันต่อวัน

### ประโยชน์ที่ได้รับ

1. ได้โปรแกรมทดสอบซอฟต์แวร์เพื่อวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติ สำหรับใช้งานในบริษัท
2. ได้ฐานข้อมูลของผลการทดสอบจากฟังก์ชัน “Daily Check” ฟังก์ชัน “Main Test” และ “Power Check” เพื่อนำไปใช้ในการวิเคราะห์ข้อมูล
3. ได้ควบคุมกระบวนการผลิตเชิงสถิติ เพื่อให้เครื่องทดสอบและการทดสอบผลิตภัณฑ์เป็นไปตามมาตรฐาน

### ขั้นตอนและวิธีการดำเนินงาน

1. ศึกษาข้อมูลที่เกี่ยวข้องกับการทำโครงการ  
ศึกษาข้อมูลต่างๆ ที่เกี่ยวข้องกับการทำโครงการ เช่น กระบวนการทดสอบผลิตภัณฑ์ของบริษัท และการควบคุมกระบวนการผลิตเชิงสถิติ (Statistical Process Control) โปรแกรมภาษา (Programming Language) ที่จะใช้ในการพัฒนา
2. รวบรวมความต้องการและวิเคราะห์ระบบ

รวบรวมความต้องการจากพนักงานที่ปรึกษาโดยการสัมภาษณ์ ว่ามีฟังก์ชันการทำงานใดบ้างที่บริษัทต้องการให้มีในโปรแกรมที่จะพัฒนา จากนั้นทำการวิเคราะห์ความต้องการที่รวบรวม ซึ่งพบว่าโปรแกรมจะต้องสามารถสร้างไฟล์ผลการทดสอบผลิตภัณฑ์จาก ฟังก์ชัน “Daily Check”, “Main Test” และ “Power Check” ในรูปแบบไฟล์เอกซ์เอ็มแอล รวมถึงสามารถบันทึกไฟล์เอกซ์เอ็มแอลที่ได้ไว้บนระบบฐานข้อมูล เพื่อนำไปใช้ในการวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติ

#### 3. ออกแบบระบบ

จากการวิเคราะห์โปรแกรม ผู้จัดทำได้ทำการออกแบบโปรแกรมที่จะพัฒนา โดยออกแบบการทำงานของโปรแกรมเป็น 2 ส่วนหลัก ได้แก่

1. ส่วนของการ สร้างไฟล์เอกซ์เอ็มแอลของผลการทดสอบ
2. ส่วนการบันทึกไฟล์เอกซ์เอ็มแอลของผลการทดสอบไปไว้บนระบบฐานข้อมูลของบริษัท
4. พัฒนาระบบ

พัฒนาโปรแกรมออกเป็น 2 ส่วนหลักตามที่ได้ออกแบบไว้ คือส่วนของการสร้างไฟล์เอกซ์เอ็มแอลและส่วนการบันทึกไฟล์เอกซ์เอ็มแอลไปไว้บนระบบฐานข้อมูลของบริษัท โดยพัฒนาโปรแกรมด้วยภาษาวิซวลเบสิกคอตเน็ต (Visual Basic.Net) และใช้ระบบฐานข้อมูลเอพีไออินเตอร์เฟส (API interface) เป็นระบบจัดการฐานข้อมูล (Database Management System)

## 5. ทดสอบโปรแกรม

นำโปรแกรมที่พัฒนามาทำการทดสอบการทำงานในส่วนของการทำงานหลักทั้งสองส่วนตามที่ออกแบบไว้ได้แก่

1. ทดสอบสร้างผลการทดสอบในรูปแบบของไฟล์เอกซ์เอ็มแอลของทั้ง 3 ฟังก์ชัน ได้แก่ ฟังก์ชัน “Daily Check”, “Main Test” และ “Power Check”

2. การบันทึกผลการทดสอบในรูปแบบของไฟล์เอกซ์เอ็มแอลไปไว้บนระบบฐานข้อมูลของบริษัท

## 6. วิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติ

นำข้อมูลผลการทดสอบความพร้อมของเครื่องทดสอบผลิตภัณฑ์ ที่ได้จากฟังก์ชัน “Daily Check” ซึ่งเก็บอยู่ในรูปแบบของไฟล์ไฟล์เอกซ์เอ็มแอลที่อยู่บนระบบฐานข้อมูล มาพล็อตกราฟ (plot graph) บนระบบเอสพีซี (SPC System) ซึ่งเป็นระบบวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติ เพื่อใช้ข้อมูลกราฟที่ได้ในการวิเคราะห์และทำการควบคุมเครื่องทดสอบให้อยู่ในมาตรฐาน

## 7. จัดทำเอกสารรายงานผลการดำเนินงานโครงการ

จัดทำเอกสารรายงานผลการดำเนินงานโครงการ เช่น ที่มาและความสำคัญของโครงการ หลักการที่เกี่ยวข้อง และผลทดสอบการทำงานของโปรแกรม ข้อดี ข้อเสีย และข้อเสนอแนะต่างๆ

## อุปกรณ์และเครื่องมือที่ใช้

### ฮาร์ดแวร์

- คอมพิวเตอร์โน้ตบุ๊ก 1 เครื่อง
- เครื่องทดสอบซอฟต์แวร์ 1 เครื่อง

### ซอฟต์แวร์

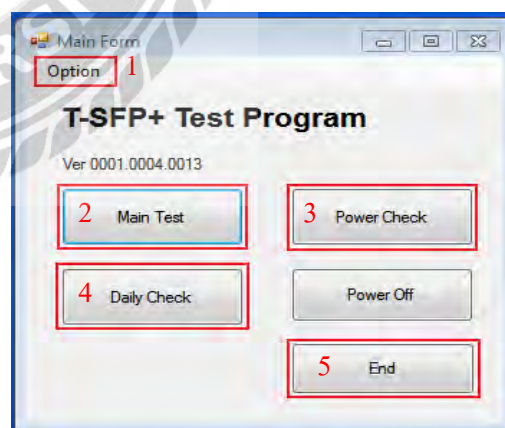
- วิชาล สตูดิโอ (Visual Studio 2019) 1 ชุด
- เอกซ์เอ็มแอล มาร์กเกอร์ (Xml Marker) 1 ชุด

## การทำงานของโปรแกรม

### 1. โปรแกรมทดสอบซอฟต์แวร์เพื่อวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติ

หน้าจอหลักของโปรแกรมทดสอบซอฟต์แวร์ เพื่อวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติชื่อ Main Form โดยเป็นหน้าจอแรกของการเข้าใช้งานโปรแกรม ซึ่งจะประกอบไปด้วยส่วนหลักต่างๆ 5 ส่วนหลัก ดังรูปที่ 1.1

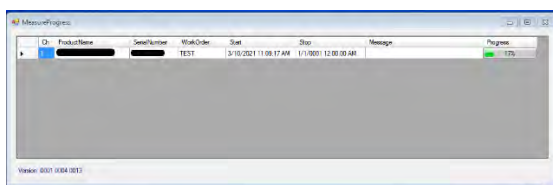
1. หมายเลข 1 Option เป็นส่วนที่ใช้ในการเลือกโหมดหรือตั้งค่าการทำงานของโปรแกรม
2. หมายเลข 2 Main Test เป็นปุ่มที่ใช้ในการทดสอบซอฟต์แวร์ฟังก์ชัน “Main Test”
3. หมายเลข 3 Power Check เป็นปุ่มที่ใช้ในการทดสอบซอฟต์แวร์ฟังก์ชัน “Power Check”
4. หมายเลข 4 Daily Check เป็นปุ่มที่ใช้ในการทดสอบซอฟต์แวร์ฟังก์ชัน “Daily Check”
5. หมายเลข 5 End เป็นปุ่มที่ใช้ในการปิดโปรแกรม



รูปที่ 1.1 หน้าจอหลักของโปรแกรมทดสอบซอฟต์แวร์เพื่อวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติ



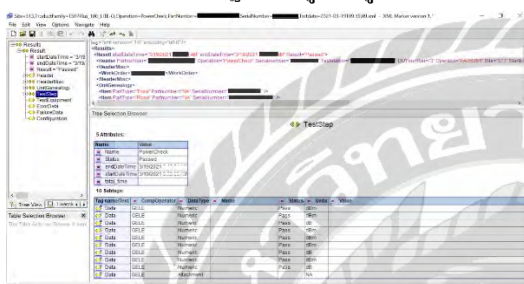




รูปที่ 3.2 หน้าจอทดสอบซอฟต์แวร์ของฟังก์ชัน

Power Check

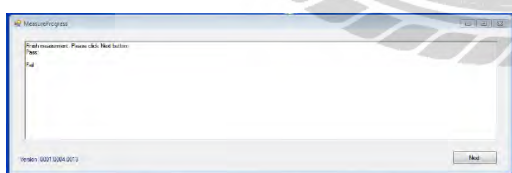
เมื่อโปรแกรมทำการทดสอบซอฟต์แวร์ฟังก์ชัน “Power Check” เสร็จสิ้น จะสร้างไฟล์ผลการทดสอบในรูปแบบของเอกซ์เอ็มแอล และบันทึกเก็บไว้บนระบบฐานข้อมูล ดังรูปที่ 3.3



รูปที่ 3.3 ไฟล์เอกซ์เอ็มแอลของฟังก์ชัน Power

Check

ซึ่งบนหน้าจอของโปรแกรม จะปรากฏข้อความว่า "Finish measurement. Please click Next bottom" และผลลัพธ์การทดสอบของแต่ละชิ้นงานของผลิตภัณฑ์ว่า ผ่าน (Pass) หรือไม่ผ่าน (Fail) ดังรูปที่ 4.3.4 จากนั้นให้กดปุ่ม Next เพื่อกลับสู่หน้าจอ Main Form



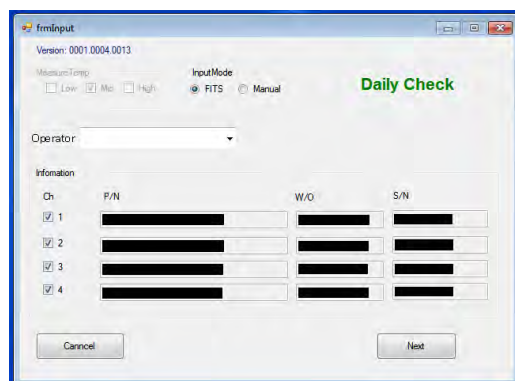
รูปที่ 3.4 แสดงข้อความและผลลัพธ์การทดสอบ

ของฟังก์ชัน Power Check

#### 4 การทดสอบซอฟต์แวร์ฟังก์ชัน “Daily Check”

เมื่อผู้ดำเนินการ (Operator) กดปุ่ม Daily Check เพื่อจะทำการทดสอบซอฟต์แวร์ฟังก์ชัน Daily Check โปรแกรมจะเข้าสู่หน้าจอ firmInput ดัง

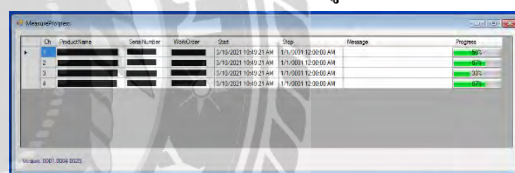
รูปที่ 4.1 เพื่อให้ผู้ดำเนินการกรอกข้อมูล ก่อนที่จะทำการทดสอบซอฟต์แวร์



รูปที่ 4.1 หน้าจอ firmInput ของฟังก์ชัน Daily

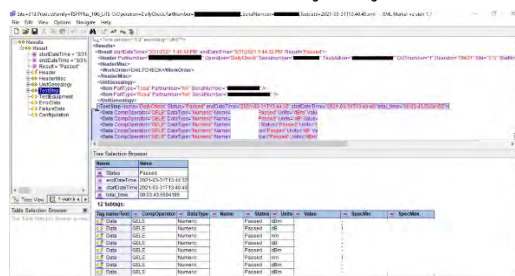
Check

หลังจากผู้ดำเนินการกรอกข้อมูลเสร็จแล้วจะกดปุ่ม Next เพื่อให้โปรแกรมเริ่มดำเนินงาน ในส่วนของการทดสอบซอฟต์แวร์ฟังก์ชัน Daily Check จากนั้นโปรแกรมจะเข้าสู่หน้าจอ Measure Progress เพื่อทำการทดสอบซอฟต์แวร์ ดังรูปที่ 4.4.2



รูปที่ 4.2 หน้าจอหน้าจอตสอบซอฟต์แวร์ของฟังก์ชัน ของฟังก์ชัน Daily Check

เมื่อโปรแกรมทำการทดสอบซอฟต์แวร์ฟังก์ชัน “Daily Check” เสร็จสิ้น จะสร้างไฟล์ผลการทดสอบในรูปแบบของเอกซ์เอ็มแอล และบันทึกเก็บไว้บนระบบฐานข้อมูล ดังรูปที่ 4.3



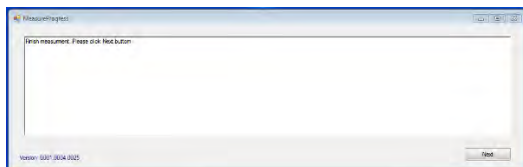
รูปที่ 4.3 ไฟล์เอกซ์เอ็มแอลของฟังก์ชัน Daily

Check

ซึ่งบนหน้าจอของโปรแกรม จะปรากฏ

ข้อความว่า "Finish measurement. Please click Next bottom" และผลลัพธ์การทดสอบของแต่ละชิ้นงานของผลิตภัณฑ์ว่า ผ่าน (Pass) หรือไม่ผ่าน (Fail) ดังรูปที่ 4.4 จากนั้นให้กดปุ่ม Next เพื่อกลับสู่หน้าจอ

Main Form

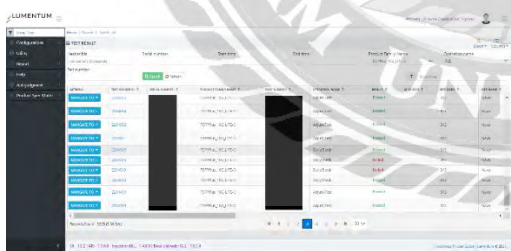


รูปที่ 4.4 แสดงข้อความการทำงานของฟังก์ชัน

Daily Check

## 5 การวิเคราะห์ข้อมูลเพื่อควบคุมกระบวนการเชิงสถิติ (Statistical Process Control)

เมื่อโปรแกรมการทดสอบซอฟต์แวร์เสร็จสิ้นทั้ง 3 ฟังก์ชันแล้ว ไฟล์เอกซ์เอ็มแอล(XML) ของผลการทดสอบจะถูกบันทึกไปไว้บนฐานระบบข้อมูลเอพีไออินเตอร์เฟซ (API interface) ซึ่งเป็นระบบจัดการฐานข้อมูล (Database Management System) ดังรูปที่ 5.1 เพื่อจะนำข้อมูลไปวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติ (Statistical Process Control)



รูปที่ 5.1 ระบบฐานข้อมูลเอพีไออินเตอร์เฟซ

และเมื่อผลการทดสอบฟังก์ชัน "Daily Check" อยู่บนระบบฐานข้อมูลเรียบร้อยแล้ว ระบบเอสพีซี (SPC System) ซึ่งเป็นระบบวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติ จะทำการผลการทดสอบดังกล่าวมาพล็อตกราฟ (plot graph) ดังรูปที่ 5.2 เพื่อใช้ในการวิเคราะห์และทำการควบคุมเครื่องทดสอบให้อยู่ในมาตรฐาน



รูปที่ 5.2 ระบบเอสพีซี

## สรุปผล

โปรแกรมทดสอบซอฟต์แวร์ที่พัฒนาขึ้นสามารถนำไปใช้งานร่วมกับระบบงานเดิมของบริษัทได้ตรงตามความต้องการของบริษัท โดยโปรแกรมสามารถสร้างไฟล์เอกซ์เอ็มแอลจากผลการทดสอบซอฟต์แวร์ฟังก์ชัน "Main Check" ฟังก์ชัน "Power Check" และฟังก์ชัน "Daily Check" ในกระบวนการทดสอบผลิตภัณฑ์ และนำไปจัดเก็บบนระบบฐานข้อมูลซึ่งใช้โปรแกรมเอพีไออินเตอร์เฟซ (API Interface) เป็นระบบจัดการฐานข้อมูล เพื่อให้โปรแกรมเอสพีซี (SPC) ซึ่งเป็นระบบการควบคุมกระบวนการผลิตเชิงสถิติ (Statistical Process Control) นำไฟล์เอกซ์เอ็มแอลจากการทดสอบซอฟต์แวร์ฟังก์ชัน ฟังก์ชัน "Daily Check" ที่จัดเก็บบนระบบฐานข้อมูลมาทำการวิเคราะห์การควบคุมกระบวนการผลิตเชิงสถิติ เพื่อควบคุมเครื่องทดสอบให้อยู่ในมาตรฐานตามที่กำหนดและได้ผลิตภัณฑ์ที่มีคุณภาพตามมาตรฐาน

## กิตติกรรมประกาศ

การที่ผู้จัดทำได้มาปฏิบัติงานในโครงการสหกิจศึกษา ณ บริษัท ลูเมนต้า อินเตอร์เนชั่นแนล (ประเทศไทย) จำกัด ตั้งแต่ วันที่ 11 มกราคม พ.ศ. 2563 ถึงวันที่ 30 เมษายน พ.ศ. 2563 ส่งผลให้ผู้จัดทำได้รับความรู้ทั้งประสบการณ์ทำงานต่าง ๆ ที่เป็นประโยชน์ต่อการเรียนและการปฏิบัติงานในอนาคตเกี่ยวกับการปฏิบัติงานในตำแหน่ง Test Software Engineer (Intern) ณ บริษัท ลูเมนต้า อินเตอร์เนชั่นแนล (ประเทศไทย) จำกัด สามารถนำความรู้ที่ได้ไปใช้ในการประกอบอาชีพในอนาคตโดยได้รับความร่วมมือจาก บริษัท ลูเมนต้า อินเตอร์เนชั่นแนล (ประเทศไทย) จำกัด ได้สอน ได้เรียนรู้งาน และปัญหาที่พบในการทำงานในแผนกต่างๆ จึงขอขอบคุณมา ณ ที่นี้ และสนับสนุนจากหลายฝ่าย ดังนี้

1. คุณธวัชชัย ไชยรัตน์  
Engineering Dev Manager
2. คุณธนกร สัจธรรม  
Engineer Staff, Reliability/Test
3. คุณชนินทร์ โพธิสุข  
Senior Test Engineer
4. คุณกฤษฎา มุกดาเจริญชัย  
Engineer Sr, Test Software Engineer
5. Mr.David Yue  
IT Systems Enginee
6. อาจารย์ณลินรัตน์ วิสวกิตติ  
อาจารย์ที่ปรึกษาสหกิจศึกษา

และบุคคลท่านอื่น ๆ ที่ไม่ได้กล่าวนามทุกท่านที่ได้ให้คำแนะนำช่วยเหลือในการจัดทำรายงาน

ผู้จัดทำขอขอบพระคุณผู้ที่มีส่วนเกี่ยวข้องทุกท่านที่มีส่วนร่วมในการให้ข้อมูลและเป็นที่ปรึกษาในการทำรายงานฉบับนี้จนเสร็จสมบูรณ์ ตลอดจนให้การดูแลให้ความเข้าใจกับชีวิตของการทำงานจริงซึ่งขอขอบพระคุณเป็นอย่างสูงไว้ ณ ที่นี้ด้วย

## เอกสารอ้างอิง

- งานบริหารทรัพยากรสุขภาพ. (ม.ป.ป.). การอ่านแผนภูมิควบคุม(Control Chart). เข้าถึงได้จาก [https://www.si.mahidol.ac.th/th/division/um/admin/download\\_files/52\\_48\\_1mg4rg6.pdf](https://www.si.mahidol.ac.th/th/division/um/admin/download_files/52_48_1mg4rg6.pdf)
- พชัย ขาวเหลือง.(2550). การควบคุมคุณภาพเชิงสถิติ. เข้าถึงได้จาก <http://www.math.cmru.ac.th/web56/option/blog/blogdata/200417002643.pdf>
- Marcuscode. (2559). แนะนำภาษา Visual Basic. เข้าถึงได้จาก <http://marcuscode.com/lang/visual-basic/introduction>
- Mindphp. (2560). XML คืออะไร. เข้าถึงได้จาก <https://www.mindphp.com/คู่มือ/73-คืออะไร/2129-xml-คืออะไร.html>
- SpC Thailand. (2562). Statistical process control (SPC)คืออะไร?. เข้าถึงได้จาก <https://www.spc-thailand.com/post/statistical-process-control-spc>



ภาคผนวก ค

ไวนิล



# การพัฒนาโปรแกรมทดสอบซอฟต์แวร์เพื่อวิเคราะห์กระบวนการผลิตเชิงสถิติ

วิภาวดี เชาวนิต  
Wiawan Chuanit

ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยสยาม กรุงเทพมหานคร 10160

Department of Computer Engineering, Faculty of Engineering, Siam University, Bangkok 10160

ပဏ္ဍိတ

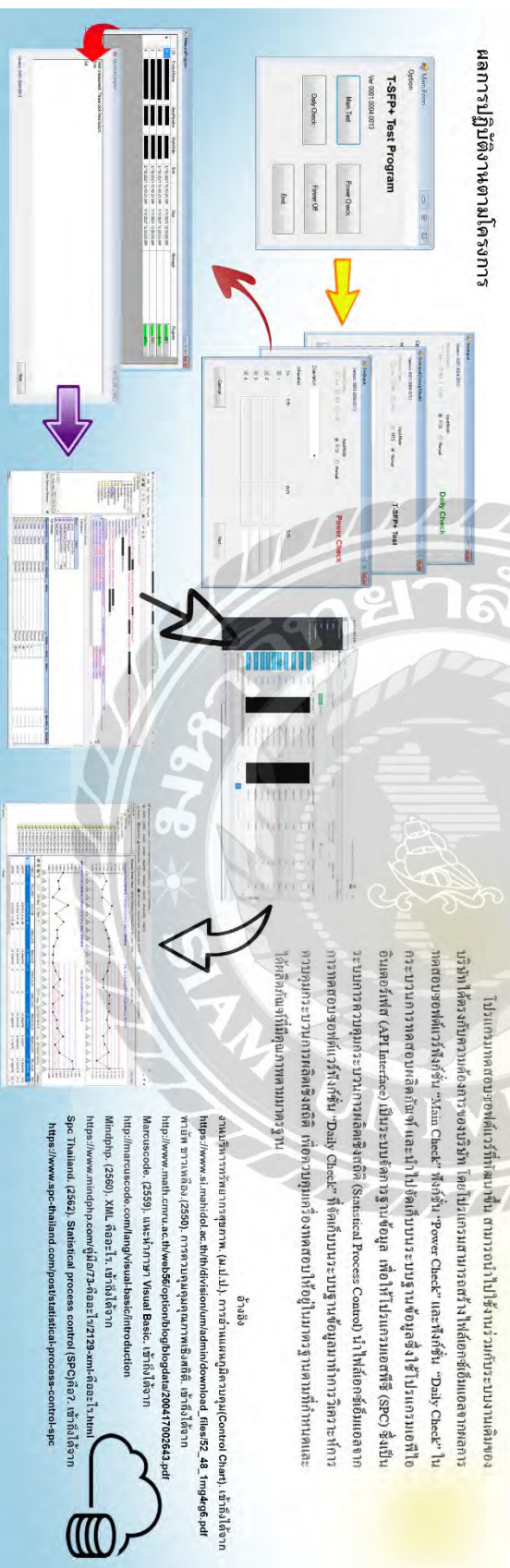
## ความสำคัญของโครงการ

การที่นำมาประกอบของพืชพันธุ์พืชอาหารจากภูมิปัญญาภาคผลิตจึงสอดคล้องกับ  
การวิจัยภาคเกษตรเขต โดยที่กรมจะสร้างให้เกิดประโยชน์ของผลจากการทดลอง  
การวิจัยและการทดสอบผลิตภัณฑ์กับเครือข่ายเกษตรกร เพื่อนำไปใช้กับผู้ใช้ภูมิปัญญา  
ความรู้ในการจัดการทรัพยากรตามกระบวนการผลิตเชิงผลิต

วัดสุทัศนเทพวรารามราชวรมหาวิหาร

1. การพัฒนาโปรแกรมทดสอบเพื่อวัดทักษะการคำนวณการคูณเชิงลิต
2. เมื่อสร้างไฟล์ข้อสอบเองจากผลการทดสอบและเฉลยข้อสอบมีค่า
3. เพื่อวัดทักษะข้อสอบที่แจ้งการทดสอบก่อนทดสอบและเฉลยข้อสอบมีค่า

## ผลการปฏิบัติงานตามโครงการ



รูปที่ ค.1 ไวนิลของโครงการ



ภาคผนวก ง  
โค้ดของโปรแกรม



1. โค้ดในส่วนของฟังก์ชัน DailyCheck ที่ให้สร้างไฟล์เอกซ์เอ็มแอล (XML) และบันทึกไว้ที่ฐานข้อมูล

```
Private Sub WriteXMLExecute(ByVal Data As DailyCheckDataClass, ByVal WorkerName As String, Partnumber As String, Serialnumber As String, SoftwareVersion As String, WorkOrder As String, TOSASerialnumber As String, ROSASerialnumber As String, TestStartTime As String, Site_TDS As String, TDSServerLocation As String)
```

```
    Dim DateFormat As String = TestStartTime
```

```
    Dim dDate As DateTime = DateTime.Parse(DateFormat, CultureInfo.InvariantCulture)
```

```
    Dim strDayFirst As String = Format(dDate, "yyyy-MM-ddTHH.mm.ss")
```

```
    DateFormat = strDayFirst
```

```
    Dim FileName As String
```

```
    Dim testerName As String
```

```
    FileName = "Site=" + Site_TDS + ",ProductFamily=TSFPPlus_10G_LITE-O,
```

```
        Operation=DailyCheck,PartNumber=" + Partnumber + ",SerialNumber=" +
```

```
        Serialnumber + ",Testdate=" + DateFormat + ".xml"
```

```
    Dim TDSOutput As Results
```

```
    TDSOutput = New Results()
```

```
    TDSOutput.ComputerName = System.Net.Dns.GetHostName
```

```
    TDSOutput.StartTime = CType(TestStartTime, Date)
```

```
    TDSOutput.EndTime = Date.Now()
```

```
    ' Create XmlWriterSettings.
```

```
    Dim settings As XmlWriterSettings = New XmlWriterSettings()
```

```
    settings.Indent = True
```

```
    Try
```

```
        If Not Directory.Exists("C:\TsfpLogsDailyCheck") Then
```

```
            Directory.CreateDirectory("C:\TsfpLogsDailyCheck")
```

```
        End If
```

' Create XmlWriter.

Using writer As XmlWriter = XmlWriter.Create("C:\TsfpLogsDailyCheck\" + FileName,  
settings)

writer.WriteStartDocument()

writer.WriteStartElement("Results") ' Root.

writer.WriteStartElement("Result") 'start Result element.

writer.WriteAttributeString("startDateTime", TDSOutput.StartTime.ToString)

writer.WriteAttributeString("endDateTime", TDSOutput.EndTime.ToString)

Dim strResult As String

If Data.Judge = Units.JudgeState.Pass Then

    strResult = "Passed"

Else

    strResult = "Failed"

End If

writer.WriteAttributeString("Result", strResult)

writer.WriteStartElement("Header") 'start Header element.

writer.WriteAttributeString("Partnumber", Partnumber)

writer.WriteAttributeString("Operation", "DailyCheck")

writer.WriteAttributeString("Serialnumber", Serialnumber)

writer.WriteAttributeString("Teststation", TDSOutput.ComputerName)

writer.WriteAttributeString("DUTnumber", Data.ChannelNumber.ToString)

writer.WriteAttributeString("Operator", WorkerName)

writer.WriteAttributeString("Site", Site\_TDS)

writer.WriteAttributeString("Starttime", TDSOutput.StartTime.ToString("s"))

writer.WriteAttributeString("Cycletime\_m", Convert.ToString(TDSOutput.EndTime –  
TDSOutput.StartTime), "HH.mm")

writer.WriteAttributeString("Processstep", "DailyCheck")

writer.WriteAttributeString("SoftwareApplicationVersion", SoftwareVersion)

writer.WriteAttributeString("Miscinfo", String.Format("CH0{0}",

Data.ChannelNumber))

```
writer.WriteString("Workordernumber", WorkOrder)
writer.WriteEndElement() 'end Header element.}
```

```
writer.WriteStartElement("HeaderMisc") 'start HeaderMisc element.
writer.WriteString("WorkOrder", WorkOrder)
writer.WriteEndElement() 'end HeaderMisc element.
```

```
writer.WriteStartElement("UnitGenealogy") 'start HeaderMisc element.
writer.WriteStartElement("Item")
writer.WriteString("PartType", "Tosa")
writer.WriteString("Partnumber", "NA")
writer.WriteString("SerialNumber", TOSASerialnumber)
writer.WriteEndElement()
writer.WriteStartElement("Item")
writer.WriteString("PartType", "Rosa")
writer.WriteString("Partnumber", "NA")
writer.WriteString("SerialNumber", ROSASerialnumber)
writer.WriteEndElement()
writer.WriteEndElement() 'end HeaderMisc element.
```

```
writer.WriteStartElement("TestStep") 'start TestStep element.
writer.WriteString("Name", "DailyCheck")
writer.WriteString("Status", strResult)
writer.WriteString("endDateTime", TDSOutput.EndTime.ToString("s"))
writer.WriteString("startDateTime", TDSOutput.StartTime.ToString("s"))
writer.WriteString("total_time", Convert.ToString(TDSOutput.EndTime –
TDSOutput.StartTime))
```

```
Dim CompOperator As String = "GELE"
```

```
' Measure Data
```

```
Dim ValuePf As Double =
```

```
CType(Data.MeasureData.Pf(Units.OpticalPower.Units.dBm).ToString, Double)
```

```
Dim SpecMinPf As Double =
```

```
CType(Convert.ToString(Data.ReferenceData.Pf(Units.OpticalPower.Units.dBm) -  
Data.Spec.Pf(Units.OpticalPower.Units.dBm)), Double)
```

```
Dim SpecMaxPf As Double =
```

```
CType(Convert.ToString(Data.ReferenceData.Pf(Units.OpticalPower.Units.dBm) +  
Data.Spec.Pf(Units.OpticalPower.Units.dBm)), Double)
```

```
Dim StatusPf As String
```

```
If ValuePf >= SpecMinPf AndAlso ValuePf <= SpecMaxPf Then
```

```
    StatusPf = "Passed"
```

```
Else
```

```
    StatusPf = "Failed"
```

```
End If
```

```
writer.WriteStartElement("Data") 'Pf
```

```
writer.WriteAttributeString("CompOperator", CompOperator)
```

```
writer.WriteAttributeString("DataType", "Numeric")
```

```
writer.WriteAttributeString("Name", "Pf")
```

```
writer.WriteAttributeString("Status", StatusPf)
```

```
writer.WriteAttributeString("Units", "dBm")
```

```
writer.WriteAttributeString("Value", ValuePf.ToString)
```

```
writer.WriteAttributeString("SpecMin", SpecMinPf.ToString)
```

```
writer.WriteAttributeString("SpecMax", SpecMaxPf.ToString)
```

```
writer.WriteEndElement()
```



```

Dim ValueEr As Double =
CType(Data.MeasureData.ER(Units.Decibel.Units.dB).ToString, Double)

Dim SpecMinEr As Double =
CType(Convert.ToString(Data.ReferenceData.ER(Units.Decibel.Units.dB) -
Data.Spec.ER(Units.Decibel.Units.dB)), Double)

Dim SpecMaxEr As Double =
CType(Convert.ToString(Data.ReferenceData.ER(Units.Decibel.Units.dB) +
Data.Spec.ER(Units.Decibel.Units.dB)), Double)

Dim StatusEr As String
If ValueEr >= SpecMinEr AndAlso ValueEr <= SpecMaxEr Then
    StatusEr = "Passed"
Else
    StatusEr = "Failed"
End If
writer.WriteStartElement("Data") 'Er
writer.WriteAttributeString("CompOperator", CompOperator)
writer.WriteAttributeString("DataType", "Numeric")
writer.WriteAttributeString("Name", "Er")
writer.WriteAttributeString("Status", StatusEr)
writer.WriteAttributeString("Units", "dB")
writer.WriteAttributeString("Value", ValueEr.ToString)
writer.WriteAttributeString("SpecMin", SpecMinEr.ToString)
writer.WriteAttributeString("SpecMax", SpecMaxEr.ToString)
writer.WriteEndElement()

Dim ValueWavelength As Double =
CType(Data.MeasureData.Wavelength(Units.Wavelength.Units.nm).ToString, Double)

Dim SpecMinWavelength As Double =
CType(Convert.ToString(Data.ReferenceData.Wavelength(Units.Wavelength.Units.nm) -
Data.Spec.Wavelength(Units.Wavelength.Units.nm)), Double)

```

```

Dim SpecMaxWavelength As Double =
CType(Convert.ToString(Data.ReferenceData.Wavelength(Units.Wavelength.Units.nm) +
Data.Spec.Wavelength(Units.Wavelength.Units.nm)), Double)

Dim StatusWavelength As String

If ValueWavelength >= SpecMinWavelength AndAlso ValueWavelength <=
SpecMaxWavelength Then

    StatusWavelength = "Passed"

Else

    StatusWavelength = "Failed"

End If

writer.WriteStartElement("Data") 'Wavelength
writer.WriteAttributeString("CompOperator", CompOperator)
writer.WriteAttributeString("DataType", "Numeric")
writer.WriteAttributeString("Name", "Wavelength")
writer.WriteAttributeString("Status", StatusWavelength)
writer.WriteAttributeString("Units", "nm")
writer.WriteAttributeString("Value", ValueWavelength.ToString)
writer.WriteAttributeString("SpecMin", SpecMinWavelength.ToString)
writer.WriteAttributeString("SpecMax", SpecMaxWavelength.ToString)
writer.WriteEndElement()

Dim ValueSMSR As Double =
CType(Data.MeasureData.SMSR(Units.Decibel.Units.dB).ToString, Double)

Dim SpecMinSMSR As Double =
CType(Convert.ToString(Data.ReferenceData.SMSR(Units.Decibel.Units.dB) -
Data.Spec.SMSR(Units.Decibel.Units.dB)), Double)

Dim SpecMaxSMSR As Double =
CType(Convert.ToString(Data.ReferenceData.SMSR(Units.Decibel.Units.dB) +
Data.Spec.SMSR(Units.Decibel.Units.dB)), Double)

Dim StatusSMSR As String

If ValueSMSR >= SpecMinSMSR AndAlso ValueSMSR <= SpecMaxSMSR Then

```

```

        StatusSMSR = "Passed"
    Else
        StatusSMSR = "Failed"
    End If

    writer.WriteStartElement("Data") 'SMSR
    writer.WriteAttributeString("CompOperator", CompOperator)
    writer.WriteAttributeString("DataType", "Numeric")
    writer.WriteAttributeString("Name", "SMSR")
    writer.WriteAttributeString("Status", StatusSMSR)
    writer.WriteAttributeString("Units", "dB")
    writer.WriteAttributeString("Value", ValueSMSR.ToString)
    writer.WriteAttributeString("SpecMin", SpecMinSMSR.ToString)
    writer.WriteAttributeString("SpecMax", SpecMaxSMSR.ToString)
    writer.WriteEndElement()

    Dim ValueTxPmin As Double =
CType(Data.MeasureData.TxPmin(Units.OpticalPower.Units.dBm).ToString, Double)

    Dim SpecMinTxPmin As Double =
CType(Convert.ToString(Data.ReferenceData.TxPmin(Units.OpticalPower.Units.dBm) -
Data.Spec.TxPmin(Units.OpticalPower.Units.dBm)), Double)

    Dim SpecMaxTxPmin As Double =
CType(Convert.ToString(Data.ReferenceData.TxPmin(Units.OpticalPower.Units.dBm) +
Data.Spec.TxPmin(Units.OpticalPower.Units.dBm)), Double)

    Dim StatusTxPmin As String

    If ValueTxPmin >= SpecMinTxPmin AndAlso ValueTxPmin <= SpecMaxTxPmin
Then
        StatusTxPmin = "Passed"
    Else
        StatusTxPmin = "Failed"
    End If

    writer.WriteStartElement("Data") 'TxPmin

```

```

writer.WriteString("CompOperator", CompOperator)
writer.WriteString("DataType", "Numeric")
writer.WriteString("Name", "TxPmin")
writer.WriteString("Status", StatusTxPmin)
writer.WriteString("Units", "dBm")
writer.WriteString("Value", ValueTxPmin.ToString)
writer.WriteString("SpecMin", SpecMinTxPmin.ToString)
writer.WriteString("SpecMax", SpecMaxTxPmin.ToString)
writer.WriteEndElement()

```

```

Dim ValueSW As Double =
CType(Data.MeasureData.SW(Units.Wavelength.Units.nm).ToString, Double)
Dim SpecMinSW As Double =
CType(Convert.ToString(Data.ReferenceData.SW(Units.Wavelength.Units.nm) -
Data.Spec.SW(Units.Wavelength.Units.nm)), Double)
Dim SpecMaxSW As Double =
CType(Convert.ToString(Data.ReferenceData.SW(Units.Wavelength.Units.nm) +
Data.Spec.SW(Units.Wavelength.Units.nm)), Double)
Dim StatusSW As String
If ValueSW >= SpecMinSW AndAlso ValueSW <= SpecMaxSW Then
    StatusSW = "Passed"
Else
    StatusSW = "Failed"
End If
writer.WriteStartElement("Data") 'SW
writer.WriteString("CompOperator", CompOperator)
writer.WriteString("DataType", "Numeric")
writer.WriteString("Name", "SW")
writer.WriteString("Status", StatusSW)
writer.WriteString("Units", "nm")
writer.WriteString("Value", ValueSW.ToString)

```

```

writer.WriteString("SpecMin", SpecMinSW.ToString)
writer.WriteString("SpecMax", SpecMaxSW.ToString)
writer.WriteEndElement()

Dim ValueRxPmin2 As Double =
CType(Data.MeasureData.RxPmin2(Units.OpticalPower.Units.dBm).ToString, Double)

Dim SpecMinRxPmin2 As Double =
CType(Convert.ToString(Data.ReferenceData.RxPmin2(Units.OpticalPower.Units.dBm) -
Data.Spec.RxPmin2(Units.OpticalPower.Units.dBm)), Double)

Dim SpecMaxRxPmin2 As Double =
CType(Convert.ToString(Data.ReferenceData.RxPmin2(Units.OpticalPower.Units.dBm) +
Data.Spec.RxPmin2(Units.OpticalPower.Units.dBm)), Double)

Dim StatusRxPmin2 As String
If ValueRxPmin2 >= SpecMinRxPmin2 AndAlso ValueRxPmin2 <=
SpecMaxRxPmin2 Then
    StatusRxPmin2 = "Passed"
Else
    StatusRxPmin2 = "Failed"
End If

writer.WriteStartElement("Data") 'RxPmin2
writer.WriteString("CompOperator", CompOperator)
writer.WriteString("DataType", "Numeric")
writer.WriteString("Name", "RxPmin2")

writer.WriteString("Status", StatusRxPmin2)

writer.WriteString("Units", "dBm")

writer.WriteString("Value", ValueRxPmin2.ToString)
writer.WriteString("SpecMin", SpecMinRxPmin2.ToString)
writer.WriteString("SpecMax", SpecMaxRxPmin2.ToString)
writer.WriteEndElement()

```



```

Dim ValueRxPowMon As Double =
CType(Data.MeasureData.RxPowMon(Units.OpticalPower.Units.dBm).ToString, Double)

Dim SpecMinRxPowMon As Double =
CType(Convert.ToString(Data.ReferenceData.RxPowMon(Units.OpticalPower.Units.dBm) -
Data.Spec.RxPowMon(Units.OpticalPower.Units.dBm)), Double)

Dim SpecMaxRxPowMon As Double =
CType(Convert.ToString(Data.ReferenceData.RxPowMon(Units.OpticalPower.Units.dBm) +
Data.Spec.RxPowMon(Units.OpticalPower.Units.dBm)), Double)

Dim StatusRxPowMon As String
If ValueRxPowMon >= SpecMinRxPowMon AndAlso ValueRxPowMon <=
SpecMaxRxPowMon Then
    StatusRxPowMon = "Passed"
Else
    StatusRxPowMon = "Failed"
End If
writer.WriteStartElement("Data") 'RxPowMon
writer.WriteString("CompOperator", CompOperator)
writer.WriteString("DataType", "Numeric")
writer.WriteString("Name", "RxPowMon")
writer.WriteString("Status", StatusRxPowMon)
writer.WriteString("Units", "dBm")
writer.WriteString("Value", ValueRxPowMon.ToString)
writer.WriteString("SpecMin", SpecMinRxPowMon.ToString)
writer.WriteString("SpecMax", SpecMaxRxPowMon.ToString)
writer.WriteEndElement()

Dim ValueTempMon As Double =
CType(Data.MeasureData.TempMon(Temperature.Units.C).ToString, Double)

Dim SpecMinTempMon As Double =
CType(Convert.ToString(Data.ReferenceData.TempMon(Temperature.Units.C) -
Data.Spec.TempMon(Temperature.Units.C)), Double)

```

```

    Dim SpecMaxTempMon As Double =
CType(Convert.ToString(Data.ReferenceData.TempMon(Temperature.Units.C) +
Data.Spec.TempMon(Temperature.Units.C)), Double)

    Dim StatusTempMon As String

    If ValueTempMon >= SpecMinTempMon AndAlso ValueTempMon <=
SpecMaxTempMon Then

        StatusTempMon = "Passed"

    Else

        StatusTempMon = "Failed"

    End If

    writer.WriteStartElement("Data") 'TempMon
    writer.WriteAttributeString("CompOperator", CompOperator)
    writer.WriteAttributeString("DataType", "Numeric")
    writer.WriteAttributeString("Name", "TempMon")
    writer.WriteAttributeString("Status", StatusTempMon)
    writer.WriteAttributeString("Units", "C")
    writer.WriteAttributeString("Value", ValueTempMon.ToString)
    writer.WriteAttributeString("SpecMin", SpecMinTempMon.ToString)
    writer.WriteAttributeString("SpecMax", SpecMaxTempMon.ToString)
    writer.WriteEndElement()

    Dim ValueVoltMon1 As Double =
CType(Data.MeasureData.VoltMon1(Voltage.Units.V).ToString, Double)

    Dim SpecMinVoltMon1 As Double =
CType(Convert.ToString(Data.ReferenceData.VoltMon1(Voltage.Units.V) -
Data.Spec.VoltMon1(Voltage.Units.V)), Double)

    Dim SpecMaxVoltMon1 As Double =
CType(Convert.ToString(Data.ReferenceData.VoltMon1(Voltage.Units.V) +
Data.Spec.VoltMon1(Voltage.Units.V)), Double)

    Dim StatusVoltMon1 As String

    If ValueVoltMon1 >= SpecMinVoltMon1 AndAlso ValueVoltMon1 <=
SpecMaxVoltMon1 Then

```

```

        StatusVoltMon1 = "Passed"
    Else
        StatusVoltMon1 = "Failed"
    End If

    writer.WriteStartElement("Data") 'VoltMon1
    writer.WriteAttributeString("CompOperator", CompOperator)
    writer.WriteAttributeString("DataType", "Numeric")
    writer.WriteAttributeString("Name", "VoltMon1")
    writer.WriteAttributeString("Status", StatusVoltMon1)
    writer.WriteAttributeString("Units", "V")
    writer.WriteAttributeString("Value", ValueVoltMon1.ToString)
    writer.WriteAttributeString("SpecMin", SpecMinVoltMon1.ToString)
    writer.WriteAttributeString("SpecMax", SpecMaxVoltMon1.ToString)
    writer.WriteEndElement()

    Dim ValueVoltMon2 As Double =
CType(Data.MeasureData.VoltMon2(Voltage.Units.V).ToString, Double)
    Dim SpecMinVoltMon2 As Double =
CType(Convert.ToString(Data.ReferenceData.VoltMon2(Voltage.Units.V) -
Data.Spec.VoltMon2(Voltage.Units.V)), Double)
    Dim SpecMaxVoltMon2 As Double =
CType(Convert.ToString(Data.ReferenceData.VoltMon2(Voltage.Units.V) +
Data.Spec.VoltMon2(Voltage.Units.V)), Double)

    Dim StatusVoltMon2 As String
    If ValueVoltMon2 >= SpecMinVoltMon2 AndAlso ValueVoltMon2 <=
SpecMaxVoltMon2 Then
        StatusVoltMon2 = "Passed"
    Else
        StatusVoltMon2 = "Failed"
    End If

    writer.WriteStartElement("Data") 'VoltMon2

```

```

writer.WriteString("CompOperator", CompOperator)
writer.WriteString("DataType", "Numeric")
writer.WriteString("Name", "VoltMon2")
writer.WriteString("Status", StatusVoltMon2)
writer.WriteString("Units", "V")
writer.WriteString("Value", ValueVoltMon2.ToString)
writer.WriteString("SpecMin", SpecMinVoltMon2.ToString)
writer.WriteString("SpecMax", SpecMaxVoltMon2.ToString)
writer.WriteEndElement()

```

```

Dim ValueRefRxPowMon As Double =
CType(Data.MeasureData.RefRxPowMon(Units.OpticalPower.Units.dBm).ToString, Double)

Dim SpecMinRefRxPowMon As Double =
CType(Convert.ToString(Data.ReferenceData.RefRxPowMon(Units.OpticalPower.Units.dBm) -
Data.Spec.RefRxPowMon(Units.OpticalPower.Units.dBm)), Double)

Dim SpecMaxRefRxPowMon As Double =
CType(Convert.ToString(Data.ReferenceData.RefRxPowMon(Units.OpticalPower.Units.dBm) +
Data.Spec.RefRxPowMon(Units.OpticalPower.Units.dBm)), Double)

Dim StatusRefRxPowMon As String

If ValueRefRxPowMon >= SpecMinRefRxPowMon AndAlso ValueRefRxPowMon <=
SpecMaxRefRxPowMon Then
    StatusRefRxPowMon = "Passed"
Else
    StatusRefRxPowMon = "Failed"
End If

writer.WriteStartElement("Data") 'RefRxPowMon
writer.WriteString("CompOperator", CompOperator)
writer.WriteString("DataType", "Numeric")
writer.WriteString("Name", "RefRxPowMon")
writer.WriteString("Status", StatusRefRxPowMon)
writer.WriteString("Units", "dBm")
writer.WriteString("Value", ValueRefRxPowMon.ToString)

```

```

writer.WriteString("SpecMin", SpecMinRefRxPowMon.ToString)
writer.WriteString("SpecMax", SpecMaxRefRxPowMon.ToString)
writer.WriteEndElement()
writer.WriteEndElement() 'end TestStep element.

```

```

writer.WriteStartElement("TestEquipment") 'start TestEquipment element.
writer.WriteEndElement() 'end TestEquipment element.

```

```

writer.WriteStartElement("ErrorData") 'start ErrorData element.
writer.WriteEndElement() 'end ErrorData element.

```

```

writer.WriteStartElement("FailureData") 'start FailureData element.
writer.WriteEndElement() 'end FailureData element.

```

```

writer.WriteStartElement("Configuration") 'start Configuration element.
writer.WriteEndElement() 'end Configuration element.

```

```

writer.WriteEndElement() 'end Result element.
writer.WriteEndElement() 'end root
writer.WriteEndDocument()

```

```
End Using
```

```
If Site_TDS = "313" Then
```

```

    My.Computer.FileSystem.CopyFile("C:\TsfpLogsDailyCheck\" + FileName,
    TDSServerLocation + FileName)

```

```
End If
```

```
Catch ex As Exception
```

```
Using writer As New IO.StreamWriter("C:\XMLDailyCheckLogs.txt", True)
```

```
    writer.WriteLine(DateTime.Now.ToString("yyyy-MM-dd HH:mm:ss :") + ex.ToString)
```

```
End Using
```

```
End Try
```

```
End Sub
```

2. โค้ดในส่วนของฟังก์ชัน Main Test และ Power Check ที่ให้สร้างไฟล์เอกซ์เอ็มแอล (XML) และบันทึกไว้ที่ฐานข้อมูล

ส่วนที่ 1

```
Public Sub WriteXMLData(TestStartTime As String)

    Dim DutCount As Integer = TestEquipment.DutMaxCount
    Dim TOSASerialNumberArray(DutCount - 1) As String
    Dim ROSASerialNumberArray(DutCount - 1) As String
    Dim OperationName As String = ""

    For ChNo As Integer = 1 To TestEquipment.DutMaxCount
        If TestEquipment.DutChannel(ChNo).IsMeasure Then
            Try
                Dim Sp As New StoredProcedure("Thailand")
                Dim Data As ProductPartsDataDetail =
                Sp.GetProductsData(TestEquipment.DutChannel(ChNo).Dut.ProductData.SerialNumber)
                If Data.SerialNumber <> "" Then
                    TOSASerialNumberArray(ChNo - 1) = Data.PartsSerialNumberTOSA
                    ROSASerialNumberArray(ChNo - 1) = Data.PartsSerialNumberROSA
                Else
                    TOSASerialNumberArray(ChNo - 1) = "NA"
                    ROSASerialNumberArray(ChNo - 1) = "NA"
                End If
                If IsPwrChkMode Then
                    OperationName = "PowerCheck"
                Else
                    OperationName = "Adjust-Test"
                End If
                Catch ex As Exception
                    Using writer As New IO.StreamWriter("C:\XMLLogTest.txt", True)
                        writer.WriteLine(DateTime.Now.ToString("yyyy-MM-dd HH:mm:ss :") +
                        ex.ToString)
                    End Using
                End Try
            End Try
        End If
    End For
End Sub
```



```

SaveXMLData(ChNo, TOSASerialNumberArray(ChNo - 1), ROSASerialNumberArray(ChNo -
1), OperationName, TestStartTime)

    End If

Next

End Sub

```

## ส่วนที่ 2

```

Public Sub SaveXMLData(ByVal ChNo As Integer, ByVal TOSASerialNumber As String,
ByVal ROSASerialNumber As String, ByVal Operation As String, TestStartTime As String)

```

```

    Try

        Dim PartNumber As String =
TestEquipment.DutChannel(ChNo).Dut.ProductData.ProductName

        Dim SerialNumber As String =
TestEquipment.DutChannel(ChNo).Dut.ProductData.SerialNumber

        Dim DUTNumber As String =
TestEquipment.DutChannel(ChNo).Dut.ProductData.ChannelNumber.ToString

        Dim OperatorName As String =
TestEquipment.DutChannel(ChNo).Dut.ProductData.OperatorName

        Dim ProgramVersion As String =
TestEquipment.DutChannel(ChNo).Dut.ProductData.ProgramVersion

        Dim WorkOrder As String =
TestEquipment.DutChannel(ChNo).Dut.ProductData.WorkOrder


        Dim DutObj As MsaTSfpPlus.DutTSfpPlusBase =
CType(TestEquipment.DutChannel(ChNo).Dut, MsaTSfpPlus.DutTSfpPlusBase)

        Dut = DutObj


'Result

        Dim Result As String = ""

        Dim ChResult As Units.JudgeState = Dut.TestDataTable.Judge

```

```
Dim strNote As String = ""
```

```
Dim SubstrNote As String = ""
```

```
Dim failure_japan As String
```

```
If Dut.ProductData.AdjustErrMsg.ToString = "" AndAlso
```

```
Dut.ProductData.TestErrMsg.ToString = "" Then
```

```
    Dut.ProductData.JudgeResult = ChResult
```

```
    Result = "Passed"
```

```
Else
```

```
    Dut.ProductData.JudgeResult = Units.JudgeState.Fail
```

```
    Result = "Failed"
```

```
If Dut.ProductData.AdjustErrMsg.ToString = "" Then
```

```
    strNote = Dut.ProductData.TestErrMsg.ToString
```

```
    SubstrNote = Left(strNote, 50)
```

```
Else
```

```
    strNote = Dut.ProductData.AdjustErrMsg.ToString
```

```
    SubstrNote = Left(strNote, 50)
```

```
End If
```

```
End If
```

```
Dim LocalSaveDirName As String = String.Format("{0}\{1}\{2}\{3}\{4}",
```

```
EquipmentInfo.LocalSaveDir,
```

```
    DutObj.ProductData.ProductName.Substring(0, 3),
```

```
DutObj.ProductData.ProductName,
```

```
    DutObj.ProductData.SerialNumber.Substring(0, 4),
```

```
DutObj.ProductData.SerialNumber)
```

```
Dim Waveform_list As New List(Of String)
```

```
'Copy Attachment
```

```

'Disable from FBN site first due to server is not ready yet.
If CopyImageEnabling = "TRUE" Then
    Try
        If IsPwrChkMode Then
            Dim TestDataDirName As String = "PowerCheck"
            Dim _PowerCheckfileName As String =
String.Format("{0}_{1}_{2:yyyyMMddHHmmss}.txt", DutObj.ProductData.SerialNumber,
TestDataDirName, DutObj.ProductData.StartDate)
            CopyFile(String.Format("{0}\{1}\{2}", LocalSaveDirName, TestDataDirName,
_PowerCheckfileName), ServerAttachment)
        Else
            Dim TestDataDirName As String = "WaveBer"
            Dim _WaveBerfileName As String =
String.Format("{0}_{1}_{2:yyyyMMddHHmmss}.txt", DutObj.ProductData.SerialNumber,
TestDataDirName, DutObj.ProductData.StartDate)
            CopyFile(String.Format("{0}\{1}\{2}", LocalSaveDirName, TestDataDirName,
_WaveBerfileName), ServerAttachment)

            Dim BerPlotDataDirName As String = "BerPlot"
            Dim _BerplotfileName As String =
String.Format("{0}_{1}_{2:yyyyMMddHHmmss}.txt", DutObj.ProductData.SerialNumber,
BerPlotDataDirName, DutObj.ProductData.StartDate)
            CopyFile(String.Format("{0}\{1}\{2}", LocalSaveDirName,
BerPlotDataDirName, _BerplotfileName), ServerAttachment)

'Loop existing file at Waveform path
Dim WaveformDirName As String = "Waveform"
Dim DirWaveForm As New DirectoryInfo(String.Format("{0}\{1}",
LocalSaveDirName, WaveformDirName)) ' Make a reference to a directory.
Dim fileArray As FileInfo() = DirWaveForm.GetFiles() ' Get a reference to each
file in that directory.
Dim _fileWaveform As FileInfo ' Display the names of the files.

```

```

For Each _fileWaveform In fileArray
    If _fileWaveform.Name.Contains(String.Format("{0:yyyyMMddHHmmss}",
DutObj.ProductData.StartDate)) Then
        Waveform_list.Add(_fileWaveform.Name)
        CopyFile(String.Format("{0}\{1}\{2}", LocalSaveDirName,
WaveformDirName, _fileWaveform.Name), ServerImage)
    End If
Next _fileWaveform

'Rom
Dim RomDataDirName As String = "Rom"
Dim _CRomfileName As String =
String.Format("{0}_{1}_{2:yyyyMMddHHmmss}.txt", DutObj.ProductData.SerialNumber,
"CustomerRom", DutObj.ProductData.StartDate)
CopyFile(String.Format("{0}\{1}\{2}", LocalSaveDirName, RomDataDirName,
_CRomfileName), ServerAttachment)

Dim _VRomfileName As String =
String.Format("{0}_{1}_{2:yyyyMMddHHmmss}.txt", DutObj.ProductData.SerialNumber,
"VendorRom", DutObj.ProductData.StartDate)
CopyFile(String.Format("{0}\{1}\{2}", LocalSaveDirName, RomDataDirName,
_VRomfileName), ServerAttachment)

'Adjust
Dim AdjustLogDirName As String = "Adjust"
Dim _AdjustLogfileName As String =
String.Format("{0}_{1}_{2:yyyyMMddHHmmss}.txt", DutObj.ProductData.SerialNumber,
"AdjustLog", DutObj.ProductData.StartDate)
CopyFile(String.Format("{0}\{1}\{2}", LocalSaveDirName, AdjustLogDirName,
_AdjustLogfileName), ServerAttachment)

If DutObj.ProductData.IsAdjust Then

```

```

        Dim AdjustDataDirName As String = "Adjust"

        Dim _SaveAdjustDatafileName As String =
String.Format("{0}_{1}_{2:yyyyMMddHHmmss}.txt", DutObj.ProductData.SerialNumber,
AdjustDataDirName, DutObj.ProductData.StartDate)

        CopyFile(String.Format("{0}\{1}\{2}", LocalSaveDirName,
AdjustLogDirName, _SaveAdjustDatafileName), ServerAttachment)

        Dim ChannelDataDirName As String = "Channel"

        Dim _SaveChannelDatafileName As String =
String.Format("{0}_{1}_{2:yyyyMMddHHmmss}.txt", DutObj.ProductData.SerialNumber,
ChannelDataDirName, DutObj.ProductData.StartDate)

        CopyFile(String.Format("{0}\{1}\{2}", LocalSaveDirName,
AdjustLogDirName, _SaveChannelDatafileName), ServerAttachment)

    End If
End If
Catch ex As Exception
    Using Attachment As New IO.StreamWriter("C:\XMLLogTest.txt", True)
        Attachment.WriteLine(DateTime.Now.ToString("yyyy-MM-dd HH:mm:ss :") +
ex.ToString)
    End Using
End Try
End If

'Create File Name Format
Dim DateFormat As String = TestStartTime
Dim dDate As DateTime = DateTime.Parse(DateFormat, CultureInfo.InvariantCulture)
Dim strDayFirst As String = Format(dDate, "yyyy-MM-ddTHH.mm.ss")
DateFormat = strDayFirst
Dim FileName As String

```

```

FileName = "Site=" + SITE_TDS + ",ProductFamily=TSFPPlus_10G_LITE-
O,Operation=" + Operation + ",PartNumber=" + PartNumber + ",SerialNumber=" +
SerialNumber + ",Testdate=" + DateFormat + ".xml"

```

```
Dim TDSOutput As Results
```

```
TDSOutput = New Results()
```

```
TDSOutput.ComputerName = System.Net.Dns.GetHostName
```

```
TDSOutput.StartTime = CType(TestStartTime, Date)
```

```
TDSOutput.EndTime = Date.Now()
```

```
' Create XmlWriterSettings.
```

```
Dim settings As XmlWriterSettings = New XmlWriterSettings()
```

```
settings.Indent = True
```

```
If Not Directory.Exists("C:\TsfpLogsTest") Then
```

```
    Directory.CreateDirectory("C:\TsfpLogsTest")
```

```
End If
```

```
' Create XmlWriter.
```

```
Using writer As XmlWriter = XmlWriter.Create("C:\TsfpLogsTest\" + FileName,
settings)
```

```
    ' Begin writing.
```

```
    writer.WriteStartDocument()
```

```
    writer.WriteStartElement("Results") ' Root.
```

```
    writer.WriteStartElement("Result") 'start Result element.
```

```
    writer.WriteAttributeString("startDateTime", TDSOutput.StartTime.ToString)
```

```
    writer.WriteAttributeString("endDateTime", TDSOutput.EndTime.ToString)
```

```
    writer.WriteAttributeString("Result", Result)
```

```
    writer.WriteStartElement("Header") 'start Header element.
```

```
    writer.WriteAttributeString("Partnumber", PartNumber)
```

```
    writer.WriteAttributeString("Operation", Operation)
```



```

writer.WriteString("Serialnumber", SerialNumber)
writer.WriteString("Teststation", TDSOutput.ComputerName)
writer.WriteString("DUTnumber", DUTNumber)
writer.WriteString("Operator", OperatorName)
writer.WriteString("Site", SITE_TDS)
writer.WriteString("Starttime", TDSOutput.StartTime.ToString("s"))
writer.WriteString("Cycletime_m", Convert.ToString(TDSOutput.EndTime -
TDSOutput.StartTime))

writer.WriteString("Processstep", Operation)
writer.WriteString("SoftwareAplicationVersion", ProgramVersion)
writer.WriteString("FailureMode", SubstrNote)
writer.WriteString("Miscinfo", String.Format("CH0{0}", DUTNumber))
writer.WriteString("Workordernumber", WorkOrder)
writer.WriteEndElement() 'end Header element.

writer.WriteStartElement("HeaderMisc") 'start HeaderMisc element.
writer.WriteElementString("WorkOrder", WorkOrder)
writer.WriteEndElement() 'end HeaderMisc element.

writer.WriteStartElement("UnitGenealogy") 'start HeaderMisc element.
writer.WriteStartElement("Item")
writer.WriteString("PartType", "Tosa")
writer.WriteString("Partnumber", "NA")

writer.WriteString("SerialNumber", TOSASerialNumber)
writer.WriteEndElement()

writer.WriteStartElement("Item")
writer.WriteString("PartType", "Rosa")
writer.WriteString("Partnumber", "NA")

writer.WriteString("SerialNumber", ROSASerialNumber)
writer.WriteEndElement()

writer.WriteEndElement() 'end HeaderMisc element.

```

```

writer.WriteStartElement("TestStep") 'start TestStep element.
writer.WriteAttributeString("Name", Operation)
writer.WriteAttributeString("Status", Result)
writer.WriteAttributeString("endDateTime", TDSOutput.EndTime.ToString)
writer.WriteAttributeString("startDateTime", TDSOutput.StartTime.ToString)
writer.WriteAttributeString("total_time", Convert.ToString(TDSOutput.EndTime -
TDSOutput.StartTime))

```

```

Dim Name As String = ""
Dim Status As String = ""
Dim Unit As String = ""
Dim Value As String = ""
Dim MinSpec As String = ""
Dim MaxSpec As String = ""
Dim Temp As String = ""
Dim Condition As String = ""
Dim Status_Parameter As String = ""
Dim GELE As String = "GELE"
Dim LOG As String = "LOG"

Dim TDSAttrName As New List(Of String)
Dim Duplicate_index As Integer = 0
Dim TDS_Flag As Boolean = True
Dim TDSAttrNameOrg As String

```

'Data

Try

```
Dut.TestDataTable.Update()
```

```
Dim Judge As Units.JudgeState = Dut.TestDataTable.Judge
```

```

If Dut.ProductData.AdjustErrMsg.ToString = "" AndAlso
Dut.ProductData.TestErrMsg.ToString = "" Then
    Dut.ProductData.JudgeResult = Judge
Else
    Dut.ProductData.JudgeResult = Units.JudgeState.Fail
End If

```

```

Dut.ProductDataTable.Update()

```

```

If Dut.ProductData.AdjustErrMsg.ToString = "" Then
    Dut.ProductData.TestErrMsg.Append(Dut.TestDataTable.GetTestErrorComment)
End If

```

```

With Dut.TestDataTable.DataTable.Rows

```

```

    ForRowIndex As Integer = 0 To .Count - 1

```

```

        TDS_Flag = True

```

```

        TDSAttrNameOrg = ""

```

```

        Dim Row As System.Data.DataRow =

```

```

Dut.TestDataTable.DataTable.Rows.Item(ForRowIndex)

```

```

        Name = Row.Item("DesignSymbol").ToString

```

```

        Status = Row.Item("Judge").ToString

```

```

        If Status = "Pass" Then

```

```

            Status_Parameter = "Passed"

```

```

        Else

```

```

            Status_Parameter = "Failed"

```

```

        End If

```

```

        Unit = Row.Item("Unit").ToString

```

```

        Value = Row.Item("Result").ToString

```

```

MinSpec = Row.Item("MinSpec").ToString
MaxSpec = Row.Item("MaxSpec").ToString
Temp = Row.Item("Temperature").ToString
Condition = Row.Item("Condition2").ToString

'Check name
TDSAttrNameOrg = String.Format("{0}_Channel_{1}_Temp_{2}", Name,
Condition, Temp)

```

```

If Not (TDSAttrName.Contains(TDSAttrNameOrg)) Then
    TDSAttrName.Add(TDSAttrNameOrg)
Else
    Duplicate_index = 0
    While (TDS_Flag)
        Duplicate_index += 1
        If Not (TDSAttrName.Contains(String.Format("{0}_{1}",
TDSAttrNameOrg, Duplicate_index))) Then
            TDSAttrName.Add(String.Format("{0}_{1}", TDSAttrNameOrg,
Duplicate_index))
            TDS_Flag = False
        End If
    End While
End If

```

```

If MinSpec = "" Or MinSpec = "-" Then
    writer.WriteStartElement("Data")
    writer.WriteAttributeString("CompOperator", LOG)
    writer.WriteAttributeString("DataType", "String")
    writer.WriteAttributeString("Name", TDSAttrName(RowIndex))
    writer.WriteAttributeString("Status", "Done")
    writer.WriteAttributeString("Units", Unit)
    writer.WriteAttributeString("Value", Value)

```

```

        writer.WriteString("SpecMin", MinSpec)
        writer.WriteString("SpecMax", MaxSpec)
        writer.WriteEndElement()
    Else
        writer.WriteStartElement("Data")
        writer.WriteString("CompOperator", GELE)
        writer.WriteString("DataType", "Numeric")
        writer.WriteString("Name", TDSAttrName(RowIndex))
        writer.WriteString("Status", Status_Parameter)
        writer.WriteString("Units", Unit)
        writer.WriteString("Value", Value)
        writer.WriteString("SpecMin", MinSpec)
        writer.WriteString("SpecMax", MaxSpec)
        writer.WriteEndElement()
    End If
Next
End With

Catch ex As Exception
    Using TestStep As New IO.StreamWriter("C:\XMLLogTest.txt", True)
        TestStep.WriteLine(DateTime.Now.ToString("yyyy-MM-dd HH:mm:ss :") +
ex.ToString)
    End Using
End Try

'Attachment Path
Try
    If IsPwrChkMode Then
        'PowerCheck
        Dim TestDataDirName As String = "PowerCheck"

```

```

Dim PowerCheckPath As String =
String.Format("{0}\{1}_{2}_{3:yyyyMMddHHmmss}.txt", ServerAttachment,
Dut.ProductData.SerialNumber, TestDataDirName, Dut.ProductData.StartDate)

writer.WriteStartElement("Data")
writer.WriteAttributeString("CompOperator", GELE)
writer.WriteAttributeString("DataType", "Attachment")
writer.WriteAttributeString("Name", "TestData")
writer.WriteAttributeString("Status", "")
writer.WriteAttributeString("Units", "NA")
writer.WriteAttributeString("Value", PowerCheckPath)
writer.WriteEndElement()
Else
'WaveBer
Dim TestDataDirName As String = "WaveBer"
Dim WaveBerPath As String =
String.Format("{0}\{1}_{2}_{3:yyyyMMddHHmmss}.txt", ServerAttachment,
Dut.ProductData.SerialNumber, TestDataDirName, Dut.ProductData.StartDate)
writer.WriteStartElement("Data")
writer.WriteAttributeString("CompOperator", GELE)
writer.WriteAttributeString("DataType", "Attachment")
writer.WriteAttributeString("Name", "TestData")
writer.WriteAttributeString("Status", "")
writer.WriteAttributeString("Units", "NA")
writer.WriteAttributeString("Value", WaveBerPath)
writer.WriteEndElement()

'Berplot
Dim BerPlotDataDirName As String = "BerPlot"
Dim BerPlotPath As String =
String.Format("{0}\{1}_{2}_{3:yyyyMMddHHmmss}.txt", ServerAttachment,
Dut.ProductData.SerialNumber, BerPlotDataDirName, Dut.ProductData.StartDate)

```



```

writer.WriteStartElement("Data")
writer.WriteAttributeString("CompOperator", GELE)
writer.WriteAttributeString("DataType", "Attachment")
writer.WriteAttributeString("Name", "BERPlot_Data")
writer.WriteAttributeString("Status", "")
writer.WriteAttributeString("Units", "NA")
writer.WriteAttributeString("Value", BerPlotPath)
writer.WriteEndElement()

'Waveform
Dim Waveform_count As Int32 = 0
For Waveform As Integer = 0 To Waveform_list.Count - 1
    Dim FileWaveform As String = Waveform_list.Item(Waveform)
    Waveform_count = Waveform_count + 1

    writer.WriteStartElement("Data")
    writer.WriteAttributeString("CompOperator", GELE)
    writer.WriteAttributeString("DataType", "Image")
    writer.WriteAttributeString("Name", String.Format("Waveform_Data_{0}",
Waveform_count.ToString))
    writer.WriteAttributeString("Status", "")
    writer.WriteAttributeString("Units", "NA")
    writer.WriteAttributeString("Value", String.Format("{0}\{1}", ServerImage,
FileWaveform))

    writer.WriteEndElement()

Next

'Rom
Dim CRomDirName As String = "CustomerRom"

```

```

Dim CRomPath As String =
String.Format("{0}\{1}_{2}_{3:yyyyMMddHHmmss}.txt", ServerAttachment,
Dut.ProductData.SerialNumber, CRomDirName, Dut.ProductData.StartDate)

writer.WriteStartElement("Data")
writer.WriteAttributeString("CompOperator", GELE)
writer.WriteAttributeString("DataType", "Attachment")
writer.WriteAttributeString("Name", "Rom_Data_Customer")
writer.WriteAttributeString("Status", "")
writer.WriteAttributeString("Units", "NA")
writer.WriteAttributeString("Value", CRomPath)
writer.WriteEndElement()

```

```

Dim VRomDirName As String = "VendorRom"
Dim VRomPath As String =
String.Format("{0}\{1}_{2}_{3:yyyyMMddHHmmss}.txt", ServerAttachment,
Dut.ProductData.SerialNumber, VRomDirName, Dut.ProductData.StartDate)

writer.WriteStartElement("Data")
writer.WriteAttributeString("CompOperator", GELE)
writer.WriteAttributeString("DataType", "Attachment")
writer.WriteAttributeString("Name", "Rom_Data_Vendor")
writer.WriteAttributeString("Status", "")
writer.WriteAttributeString("Units", "NA")
writer.WriteAttributeString("Value", VRomPath)
writer.WriteEndElement()

```

```

'Adjust
Dim AdjustLogDirName As String = "AdjustLog"
Dim AdjustPath As String =
String.Format("{0}\{1}_{2}_{3:yyyyMMddHHmmss}.txt", ServerAttachment,
Dut.ProductData.SerialNumber, AdjustLogDirName, Dut.ProductData.StartDate)

```

```

writer.WriteStartElement("Data")
writer.WriteAttributeString("CompOperator", GELE)
writer.WriteAttributeString("DataType", "Attachment")
writer.WriteAttributeString("Name", "AdjustLog_Data")
writer.WriteAttributeString("Status", "")
writer.WriteAttributeString("Units", "NA")
writer.WriteAttributeString("Value", AdjustPath)
writer.WriteEndElement()

```

If DutObj.ProductData.IsAdjust Then

```

    Dim AdjustDataDirName As String = "Adjust"
    Dim AdjustDataPath As String =
String.Format("{0}\{1}_{2}_{3:yyyyMMddHHmmss}.txt", ServerAttachment,
Dut.ProductData.SerialNumber, AdjustDataDirName, Dut.ProductData.StartDate)
    writer.WriteStartElement("Data")
    writer.WriteAttributeString("CompOperator", GELE)
    writer.WriteAttributeString("DataType", "Attachment")
    writer.WriteAttributeString("Name", "Adjust_Data")
    writer.WriteAttributeString("Status", "")
    writer.WriteAttributeString("Units", "NA")
    writer.WriteAttributeString("Value", AdjustDataPath)
    writer.WriteEndElement()

```

```

    Dim ChannelDataDirName As String = "Channel"

```

```

    Dim AdjustChanelPath As String =
String.Format("{0}\{1}_{2}_{3:yyyyMMddHHmmss}.txt", ServerAttachment,
Dut.ProductData.SerialNumber, ChannelDataDirName, Dut.ProductData.StartDate)
    writer.WriteStartElement("Data")
    writer.WriteAttributeString("CompOperator", GELE)
    writer.WriteAttributeString("DataType", "Attachment")
    writer.WriteAttributeString("Name", "Adjust_Chanel_Data")

```

```

        writer.WriteString("Status", "")
        writer.WriteString("Units", "NA")
        writer.WriteString("Value", AdjustChanelPath)
        writer.WriteEndElement()

    End If

End If

Catch ex As Exception

    Using Attachment As New IO.StreamWriter("C:\XMLLogTest.txt", True)

        Attachment.WriteLine(DateTime.Now.ToString("yyyy-MM-dd HH:mm:ss :") +
ex.ToString)

    End Using

End Try

writer.WriteEndElement() 'end TestStep element.
writer.WriteStartElement("TestEquipment") 'start TestEquipment element.
writer.WriteEndElement() 'end TestEquipment element.

writer.WriteStartElement("ErrorData") 'start ErrorData element.
writer.WriteEndElement() 'end ErrorData element.

writer.WriteStartElement("FailureData") 'start FailureData element.
writer.WriteEndElement() 'end FailureData element.

writer.WriteStartElement("Configuration") 'start Configuration element.
writer.WriteEndElement() 'end Configuration element.

writer.WriteEndElement() 'end Result element.

writer.WriteEndElement() 'end Root.

writer.WriteEndDocument()

End Using

'copy file to Test Kaizen TDS server
If CopyImageEnabling = "TRUE" Then

```

```
My.Computer.FileSystem.CopyFile("C:\TsfpLogsTest\" + FileName,  
TDSServerLocation + FileName)  
End If  
Catch ex As Exception  
Using writer As New IO.StreamWriter("C:\XMLLogTest.txt", True)  
    writer.WriteLine(DateTime.Now.ToString("yyyy-MM-dd HH:mm:ss :") + ex.ToString)  
End Using  
End Try  
End Sub
```



## ประวัติผู้จัดทำ



ชื่อ-นามสกุล : นางสาววิลาวัลย์ เชื้อนิจ  
รหัสนักศึกษา : 6004000005  
ภาควิชา : วิศวกรรมคอมพิวเตอร์  
คณะ : วิศวกรรมศาสตร์  
ที่อยู่ปัจจุบัน : 1/42 ซ.เอกชัย 83 ถนนเอกชัย เขตบางบอน  
แขวงบางบอน จังหวัดกรุงเทพมหานคร  
10150

