

โครงการที่ 814

อาจารย์ที่ปรึกษา ผศ.ดร.อรรถพล สมุทรกุล

การพัฒนาระบบตรวจสอบตำแหน่งการวางชิ้นงานบนสายพาน

**DEVELOPMENT OF PART PLACEMENT CHECKING SYSTEM ON
CONVEYER BELT.**

ความสำคัญและที่มาของปัญหาที่ทำโครงการ

ในปัจจุบันโรงงานอุตสาหกรรมได้เล็งเห็นถึงความสำคัญของระบบอัตโนมัติซึ่งระบบอัตโนมัตินั้นสามารถเพิ่มกำลังการผลิตของบริษัทและเครื่องจักรอัตโนมัติสามารถควบคุมและช่วยลดความเสียหายจากการทำงานโดยแรงงานมนุษย์ มีการทำงานที่สะดวก ง่ายและสามารถผลิตสินค้าที่ได้มาตรฐาน อย่างไรก็ตามแขนกลอาจหยิบ หรือวางชิ้นส่วนของผลิตภัณฑ์ผิดตำแหน่งจึงทำให้เกิดของเสีย ซึ่งอาจเกิดจากการติดตั้งหุ่นยนต์ที่ไม่ได้มาตรฐาน การเกิดความร้อนที่สูงภายในตัวหุ่นยนต์ เกิดจากการใช้งานเป็นระยะเวลานาน หรือ อาจเกิดจากการหลุดตัวของอาคารทำให้เกิดการวางตำแหน่งที่เปลี่ยนไปจากตำแหน่งเดิม เพราะฉะนั้นจึงได้มีระบบเซนเซอร์ในการตรวจสอบ เพื่อบ่งบอกว่ามีความผิดปกติของตำแหน่งชิ้นงานมาแล้ว จากนั้นระบบจะทำการเก็บข้อมูลผ่านระบบ Internet of Things (IoT) ซึ่งจะเข้าไปช่วยแก้ปัญหาในกระบวนการที่มีผลกระทบต่อระบบการผลิตในโรงงานอุตสาหกรรม เนื่องจากระบบเซนเซอร์จะมีการติดตามตลอดเวลา และรายงานผลแบบออนไลน์ตลอด 24 ชั่วโมง ส่งผลให้กระบวนการผลิตสามารถทำงานได้เต็มประสิทธิภาพมากยิ่งขึ้น ลดการสูญเสียค่าใช้จ่าย ลดความเสียหายของผลิตภัณฑ์ หรือสินค้า โดยการใช้งานแขนกลสามารถส่งเสริมความปลอดภัยของกระบวนการผลิตได้เป็นอย่างดีด้วยระบบเซนเซอร์ตรวจจับการทำงาน ทดแทนแรงงานมนุษย์ในการปฏิบัติงานที่มีความเสี่ยงต่อสุขภาพ และความปลอดภัยของผู้ปฏิบัติงานในพื้นที่ โดยเฉพาะการปฏิบัติงานที่มีความเสี่ยงต่อความสูญเสียเป็นหลักไม่ว่าจะเป็นการยกของหนัก การขนย้ายวัสดุในพื้นที่จำกัด หรือการทำงานในพื้นที่ที่มีสภาพแวดล้อมอันตราย

วัตถุประสงค์

- 1 เพื่อประยุกต์ใช้เซนเซอร์ในการตรวจสอบตำแหน่งของชิ้นงาน
- 2 เพื่อสร้างระบบเตือนโดยผ่านระบบ Internet of Things (IoT) ให้แสดงข้อมูลความผิดปกติ

ขอบเขตการศึกษา

- 1 ศึกษากระบวนการหยิบกล่องว่างบนสายพาน
- 2 ศึกษาความผิดปกติตำแหน่งการว่างกล่อง
- 3 การประยุกต์ใช้ระบบ Internet of Things (IoT) ผ่านราสเบอร์รี่ พาย (Raspberry Pi)
- 4 การแสดงผลข้อมูลผ่านทางคอมพิวเตอร์

ประโยชน์ที่คาดว่าจะได้รับ

- 1 ความสะดวก และความแม่นยำในการเก็บข้อมูล
- 2 สามารถเข้าถึงข้อมูลได้ง่ายโดยผ่านระบบ Internet of Things (IoT)
- 3 สามารถนำไปประยุกต์ใช้กับระบบอื่น



วิธีและขั้นตอนในการ ทำงานวิจัย



ศึกษากระบวนการหยิบกล่องโดยใช้หุ่นยนต์



ศึกษากระบวนการทำงานของเซนเซอร์



ออกแบบกระบวนการทำงาน



ศึกษาระบบควบคุมการทำงาน DOBOT magician, บอร์ด Raspberry Pi, ระบบ IoT



สร้างแบบกระบวนการทำงาน



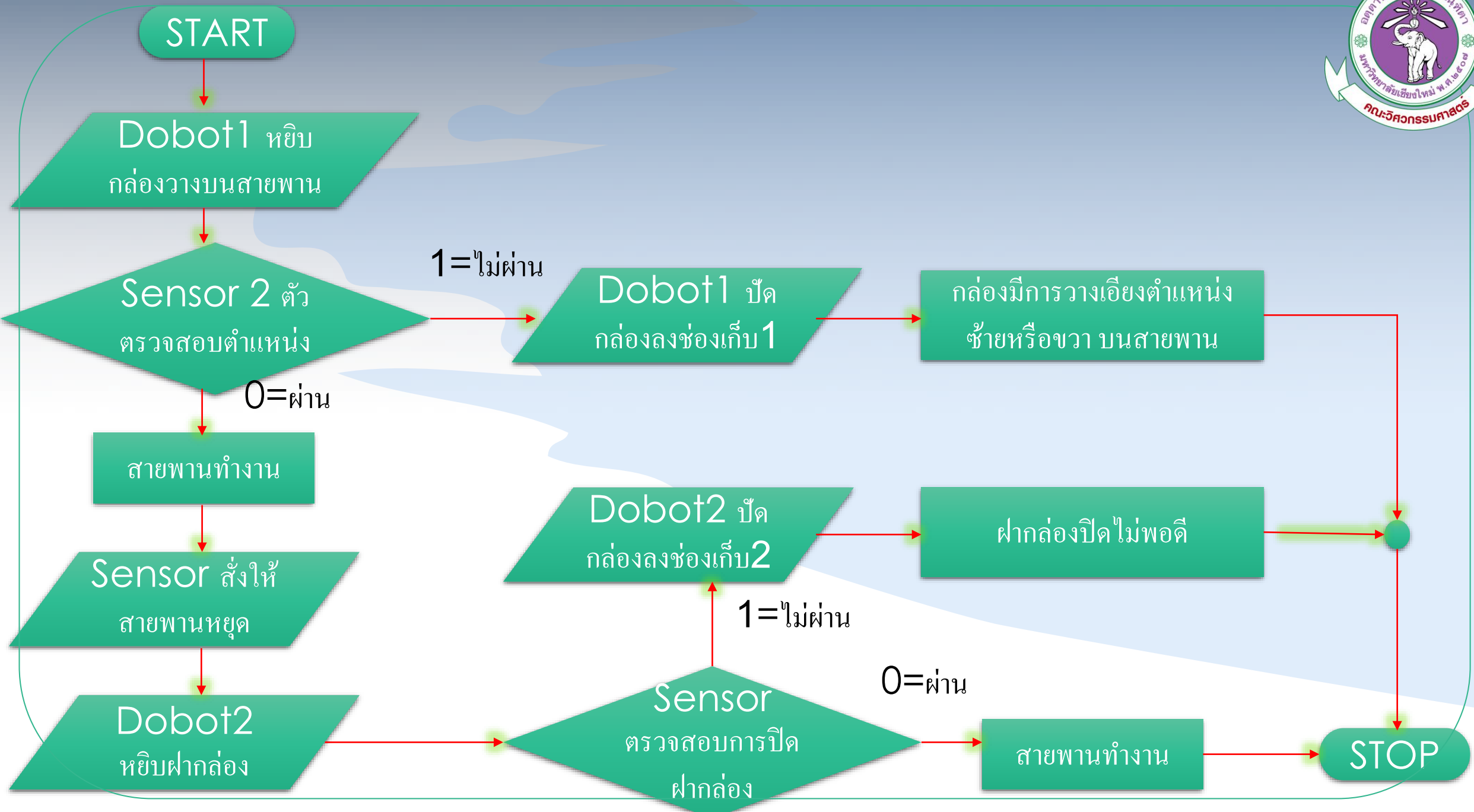
สรุปผลและจัดทำรูปเล่ม



ทดสอบและปรับปรุงผล

The background image shows a close-up of a laptop screen with PHP code. The code includes a loop to generate a gallery of images from a database. A semi-transparent light blue rounded rectangle is overlaid on the right side of the screen, containing the title text. The text is in Thai and has a green outline.

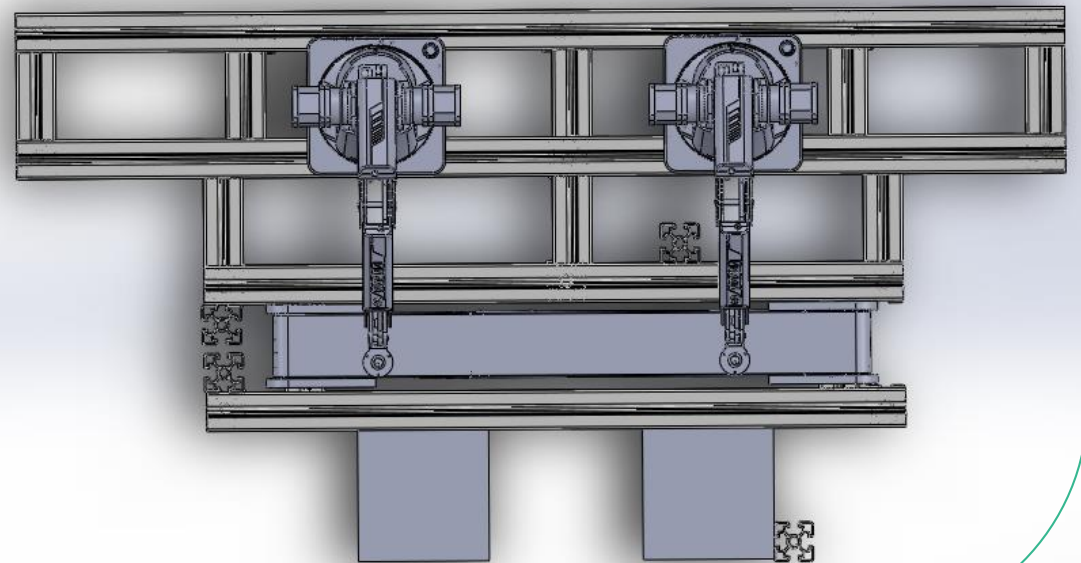
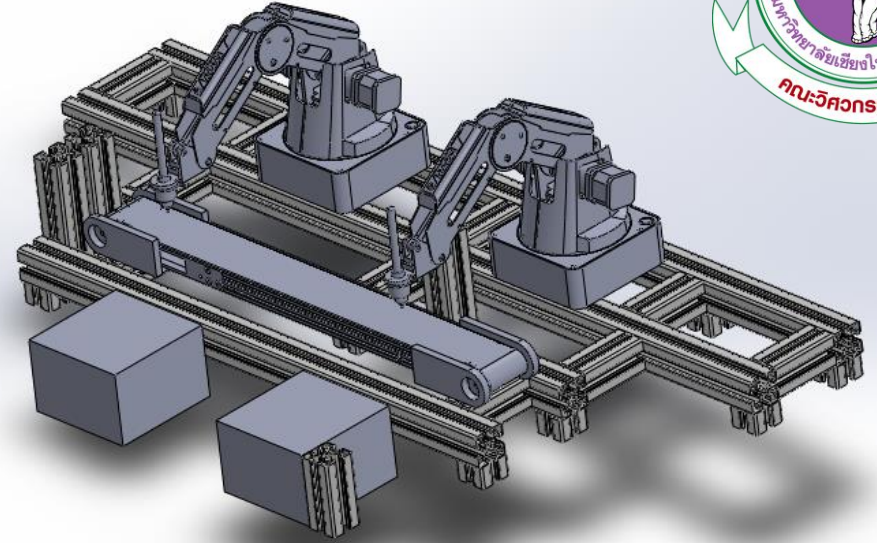
การออกแบบกระบวนการด้วยแผนผังการทำงาน



isometric view

การออกแบบเครื่องจักรด้วยโปรแกรมโซลิดเวิร์ค
(Solidwork)

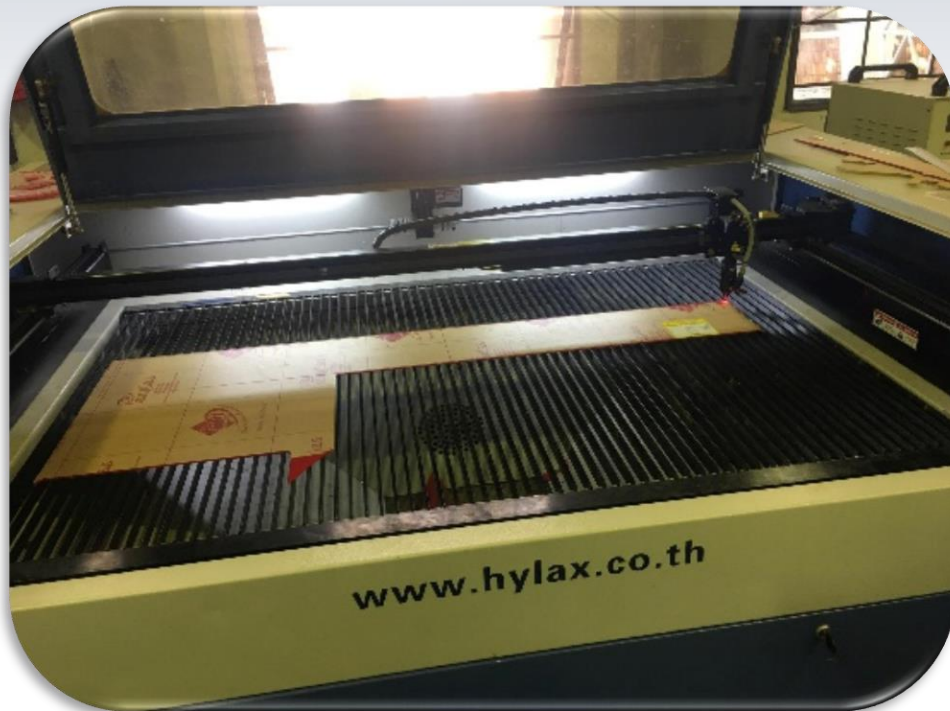
Top view



ประกอบอตุมิเนียนโปรไฟล์เมื่อเสร็จ



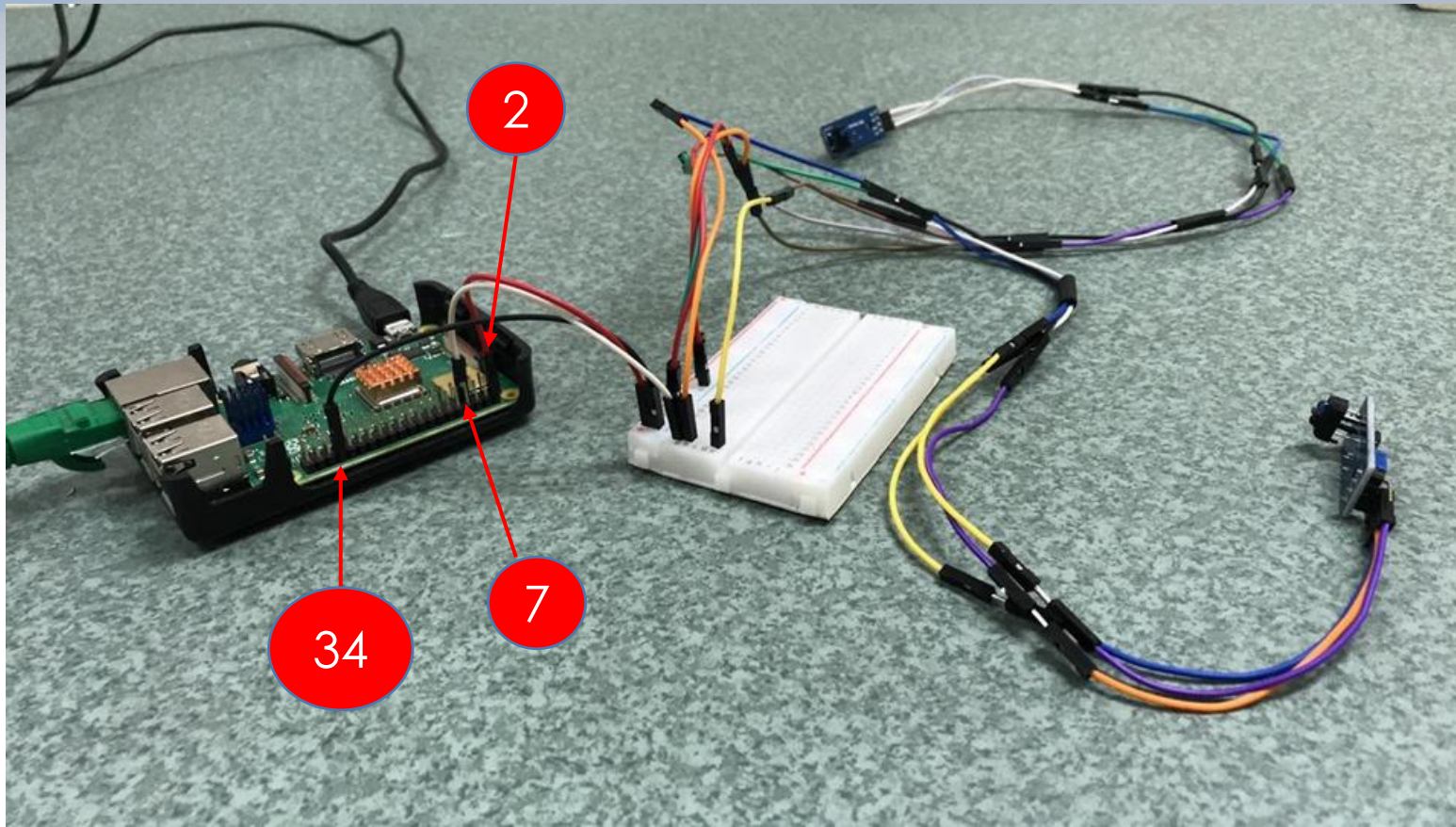
ทำการตัดแผ่นอะคริลิก



ประกอบแผ่นอะคริลิกเข้ากับอลูมิเนียมโปรไฟล์

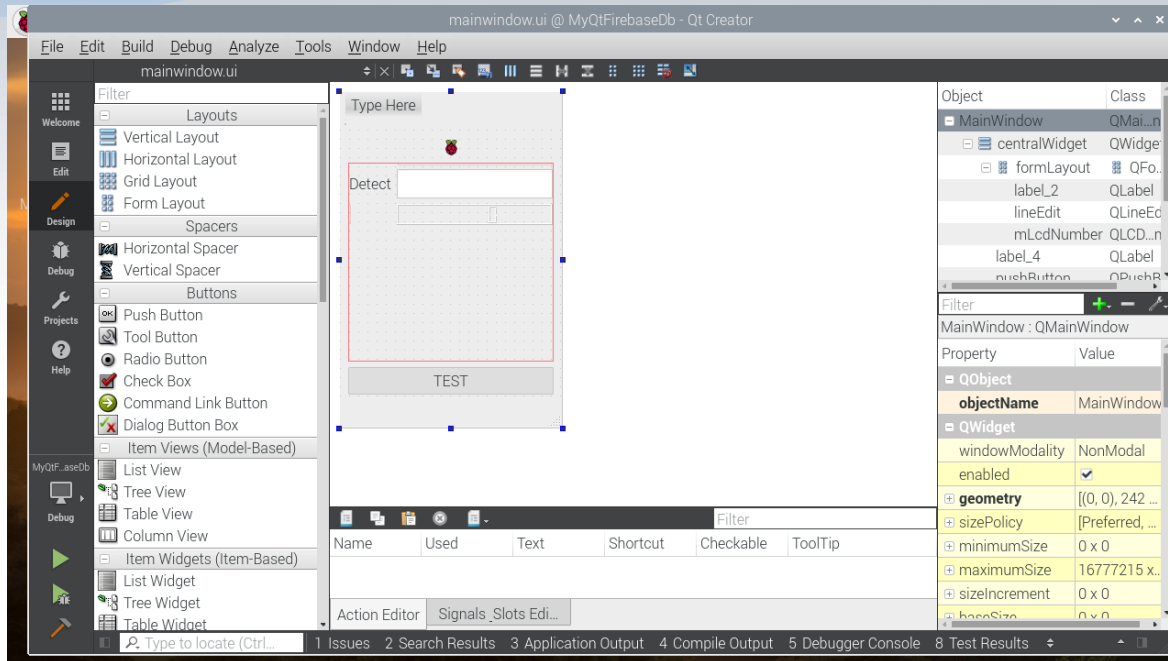


ทำการเชื่อมต่อวงจร เซนเซอร์อินฟราเรด (Infrared Reflectance Sensor TCRT5000) กับ ราสเบอร์รี่ พาย(Raspberry Pi)

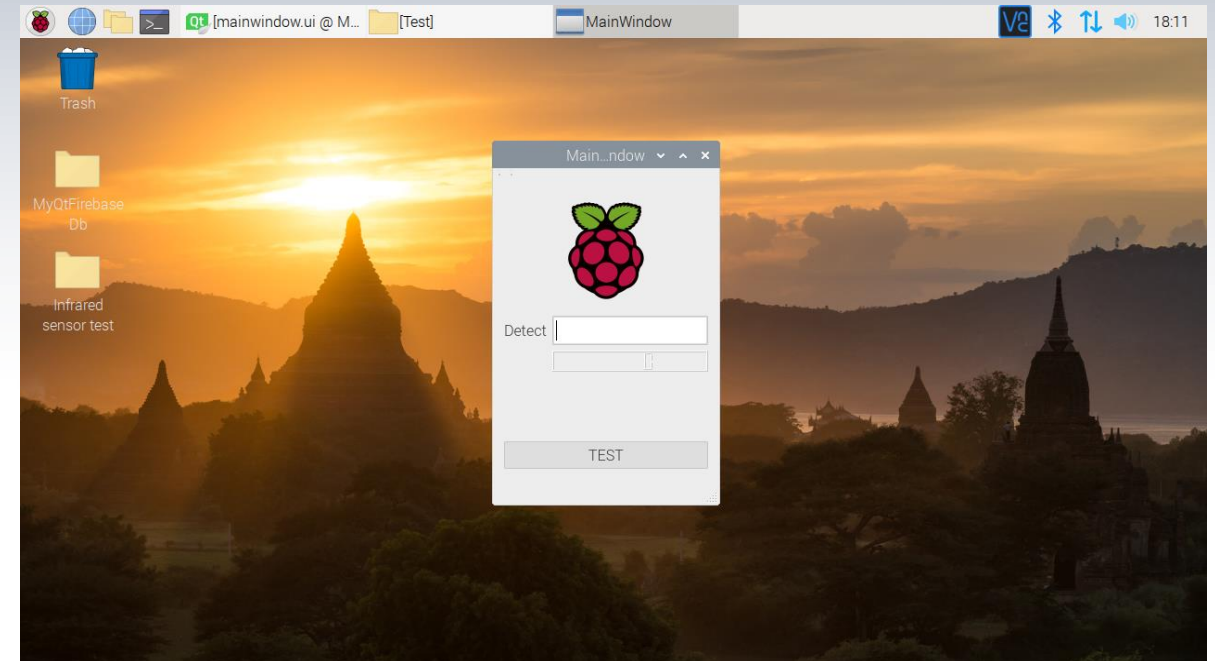


- ขาที่ 2 จะเป็นขาที่จ่ายแรงดันไฟฟ้า 5 โวลต์ ที่ส่งไปยังตัวเซนเซอร์อินฟราเรด
- ขาที่ 34 คือการวัด
- ขาที่ 7 จะเป็น GPIO4 ที่รับค่า input จากเซนเซอร์อินฟราเรด

การโปรแกรม sensor เพื่อตรวจจับกล่องที่ถูกปฏิเสธ (Reject) แล้วส่งข้อมูลแบบตอบสนองทันที (Real Time) ด้วยภาษา C++



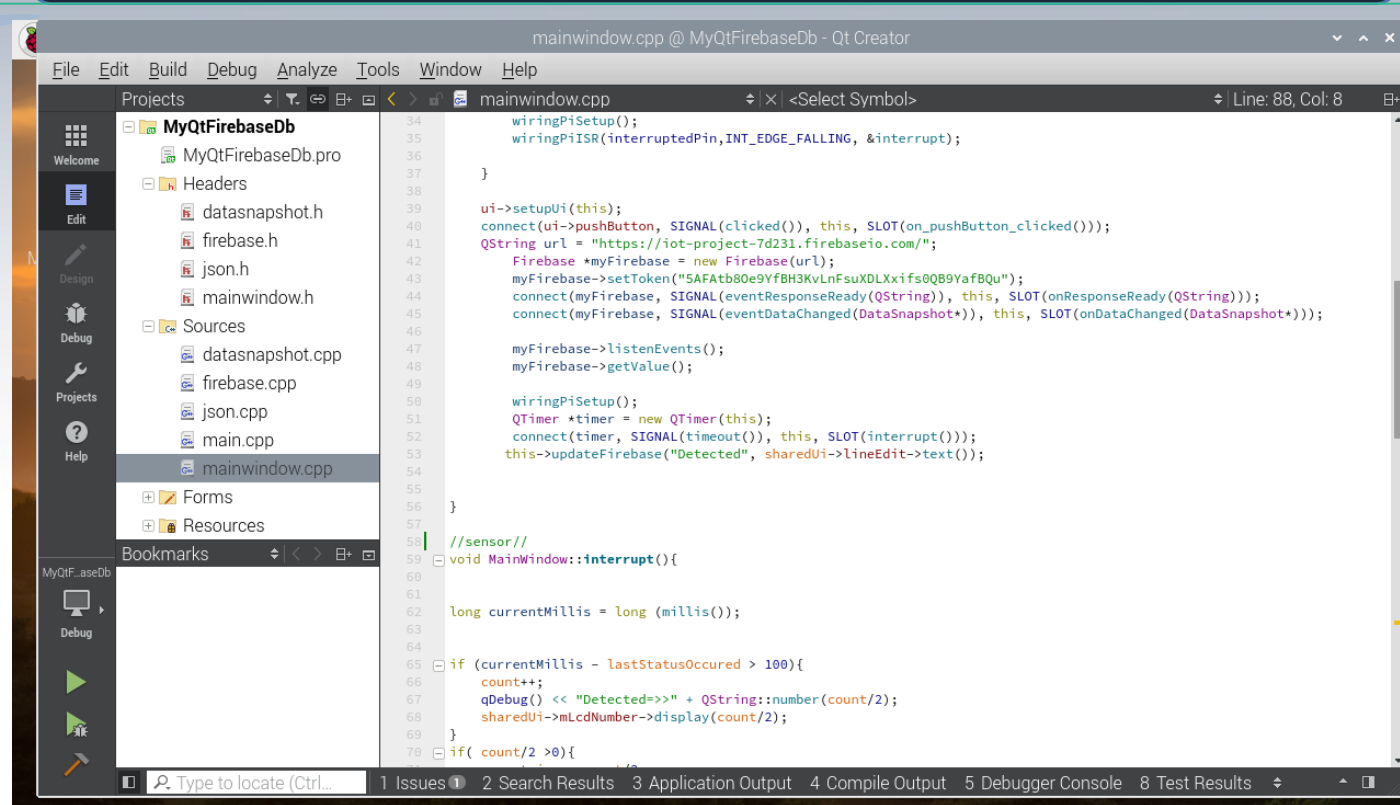
การออกแบบ Graphical User Interface



Graphical User Interface (GUI)

เมื่อออกแบบเรียบร้อยแล้ว

การโปรแกรมเซนเซอร์ (Sensor) ด้วยโปรแกรม Qt Creator



```
mainwindow.cpp @ MyQtFirebaseDb - Qt Creator
File Edit Build Debug Analyze Tools Window Help
Projects mainwindow.cpp <Select Symbol> Line: 88, Col: 8
MyQtFirebaseDb
  MyQtFirebaseDb.pro
  Headers
    datasnapshot.h
    firebase.h
    json.h
    mainwindow.h
  Sources
    datasnapshot.cpp
    firebase.cpp
    json.cpp
    main.cpp
    mainwindow.cpp
  Forms
  Resources
Bookmarks
//sensor//
void MainWindow::interrupt(){
    long currentMillis = long (millis());
    if (currentMillis - lastStatusOccured > 100){
        count++;
        qDebug() << "Detected=>" + QString::number(count/2);
        sharedUi->mLcdNumber->display(count/2);
    }
    if (count/2 > 0){
        ui->setupUi(this);
        connect(ui->pushButton, SIGNAL(clicked()), this, SLOT(on_pushButton_clicked()));
        QString url = "https://iot-project-7d231.firebaseio.com/";
        Firebase *myFirebase = new Firebase(url);
        myFirebase->setToken("5AFAtb80e9YfBH3KvLnFsuXDLXxf0QB9YafBQu");
        connect(myFirebase, SIGNAL(eventResponseReady(QString)), this, SLOT(onResponseReady(QString)));
        connect(myFirebase, SIGNAL(eventDataChanged(DataSnapshot*)), this, SLOT(onDataChanged(DataSnapshot*)));
        myFirebase->listenEvents();
        myFirebase->getValue();
        wiringPiSetup();
        QTimer *timer = new QTimer(this);
        connect(timer, SIGNAL(timeout()), this, SLOT(interrupt()));
        this->updateFirebase("Detected", sharedUi->lineEdit->text());
    }
}
```

อย่างไรก็ตามเมื่อทำการเขียนโปรแกรมให้เซนเซอร์ทำงานแล้วต้องเขียนโปรแกรมให้ส่งแบบ Realtime โดยจะทำการนำ URL (Uniform Resource Locator) คือ ที่อยู่ (Address) ของข้อมูลต่างๆในinternet และToken คือ สิ่งที่ใช้แทน username และ password มาเขียนลงในโปรแกรม Qt Creator โดยจะส่งข้อมูลผ่านเว็บไซต์ Firebase Google

การส่งข้อมูลจาก Raspberry Pi ไป Firebase Google

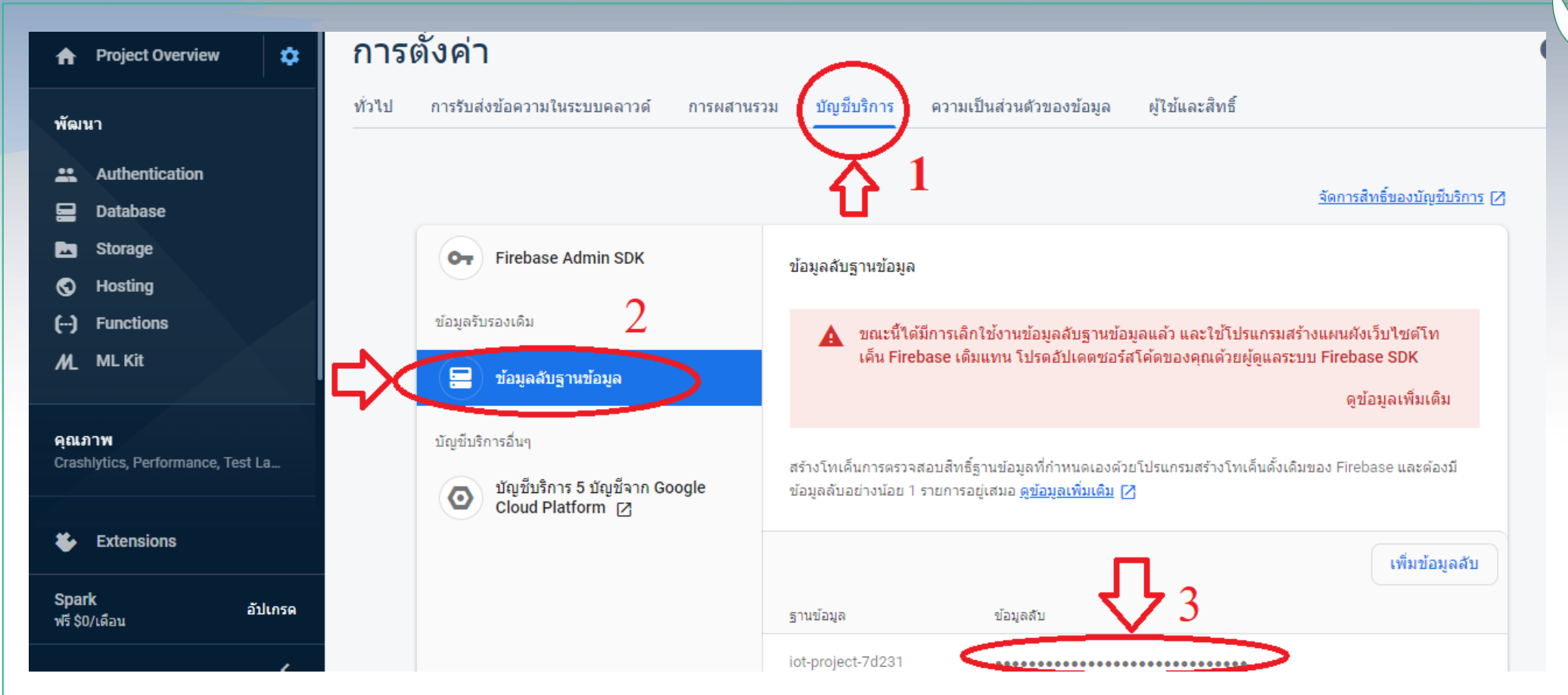


The screenshot shows the Firebase console interface for an 'IoT Project'. The left sidebar contains the following menu items:

- Project Overview (with a settings gear icon)
- การพัฒนา (Development)
 - Authentication
 - Database
 - Storage
 - Hosting
 - Functions
 - ML Kit
- คุณภาพ (Quality)
 - Crashlytics, Performance, Test La...
- Extensions
- Spark
 - ฟรี \$0/เดือน (Free \$0/month)
 - อัปเกรด (Upgrade)

The main content area displays the 'IoT Project' settings:

- ชื่อโปรเจกต์ (Project Name): IoT Project
- รหัสโปรเจกต์ (Project ID): `iot-project-7d231`
- ตำแหน่งทรัพยากร Google Cloud Platform (GCP) (Google Cloud Platform Region): `asia-south1`
- คีย์ API ของเว็บ (Web API Key): `AIzaSyAGVG_IqC-IE8BVQxGOJpdnq4kpAhZISp8`
- การตั้งค่าความเป็นสาธารณะ (Public Settings)
 - การตั้งค่าเหล่านี้เป็นการควบคุมอินสแตนซ์ของโปรเจกต์ที่แสดงสู่สาธารณะ (These settings control the public visibility of your project instances)
 - ชื่อที่เปิดเผยต่อสาธารณะ (Public Name): `project-333580002083`
 - อีเมลสนับสนุน (Support Email): ไม่ได้กำหนดค่า (Not set)



การตั้งค่า

ทั่วไป การรับส่งข้อความในระบบคลาวด์ การผสมรวม **บัญชีบริการ** ความเป็นส่วนตัวของข้อมูล ผู้ใช้และสิทธิ์

จัดการสิทธิ์ของบัญชีบริการ [\[?\]](#)

Firebase Admin SDK

ข้อมูลรับรองเดิม

ข้อมูลพื้นฐานข้อมูล

บัญชีบริการอื่นๆ

บัญชีบริการ 5 บัญชีจาก Google Cloud Platform [\[?\]](#)

ข้อมูลพื้นฐานข้อมูล

ข้อมูลลับ

iot-project-7d231

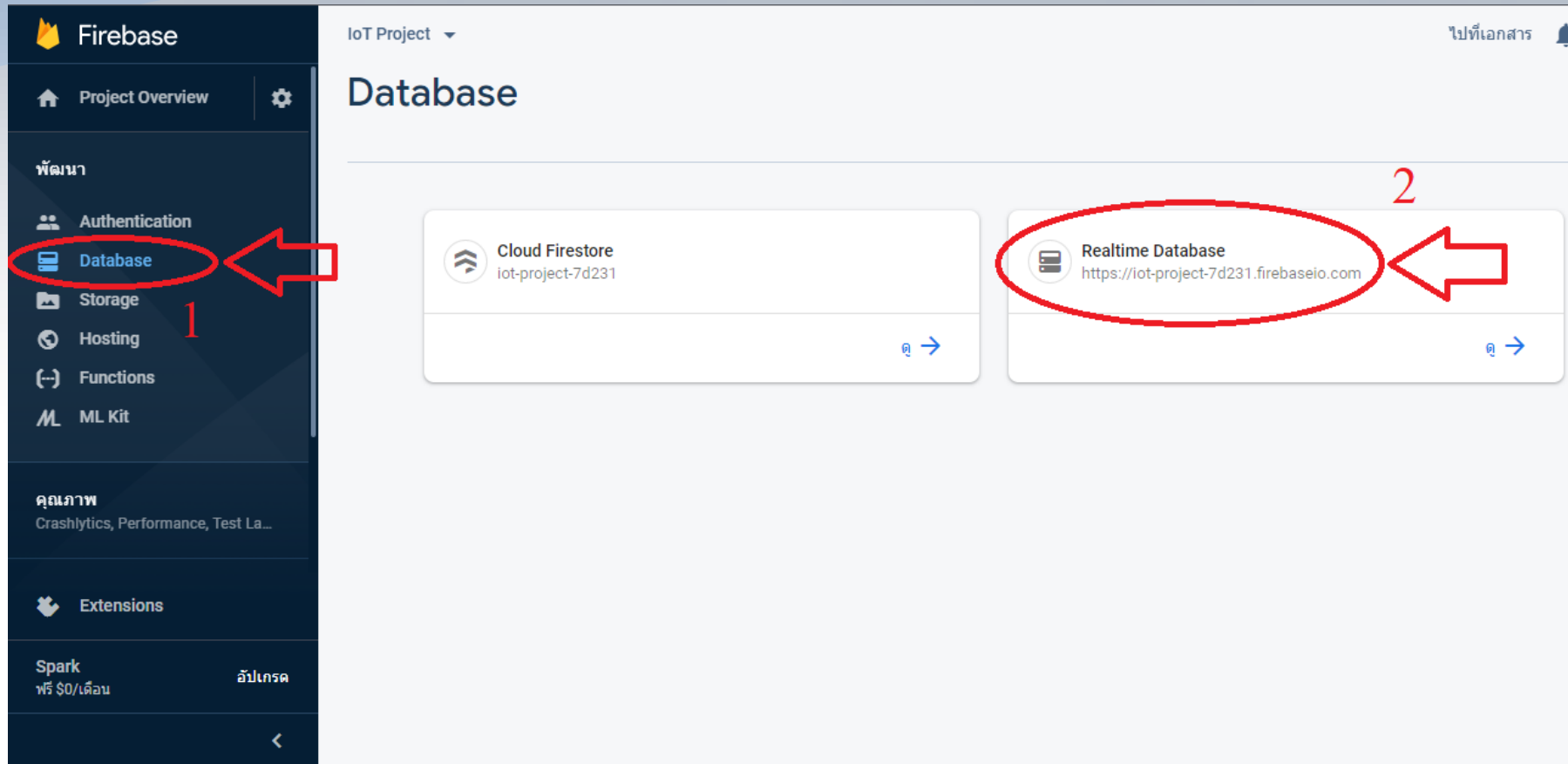
คุณจะได้มีการเลิกใช้งานข้อมูลพื้นฐานข้อมูลแล้ว และใช้โปรแกรมสร้างแผนผังเว็บไซต์ที่เห็น Firebase เติมนาม โปรเจกต์เดสก์ท็อปของคุณด้วยผู้ดูแลระบบ Firebase SDK

ดูข้อมูลเพิ่มเติม

สร้างโทเค็นการตรวจสอบสิทธิ์พื้นฐานข้อมูลที่กำหนดเองด้วยโปรแกรมสร้างโทเค็นดั้งเดิมของ Firebase และต้องมีข้อมูลลับอย่างน้อย 1 รายการอยู่เสมอ [ดูข้อมูลเพิ่มเติม](#)

เพิ่มข้อมูลลับ

หมายเลข3จะเป็นรหัส Token ที่นำไปใช้ในการเขียนโปรแกรม Qt Creator



Firestore

IoT Project

ไปที่เอกสาร

Database

พัฒนา

- Authentication
- Database**
- Storage
- Hosting
- Functions
- ML Kit

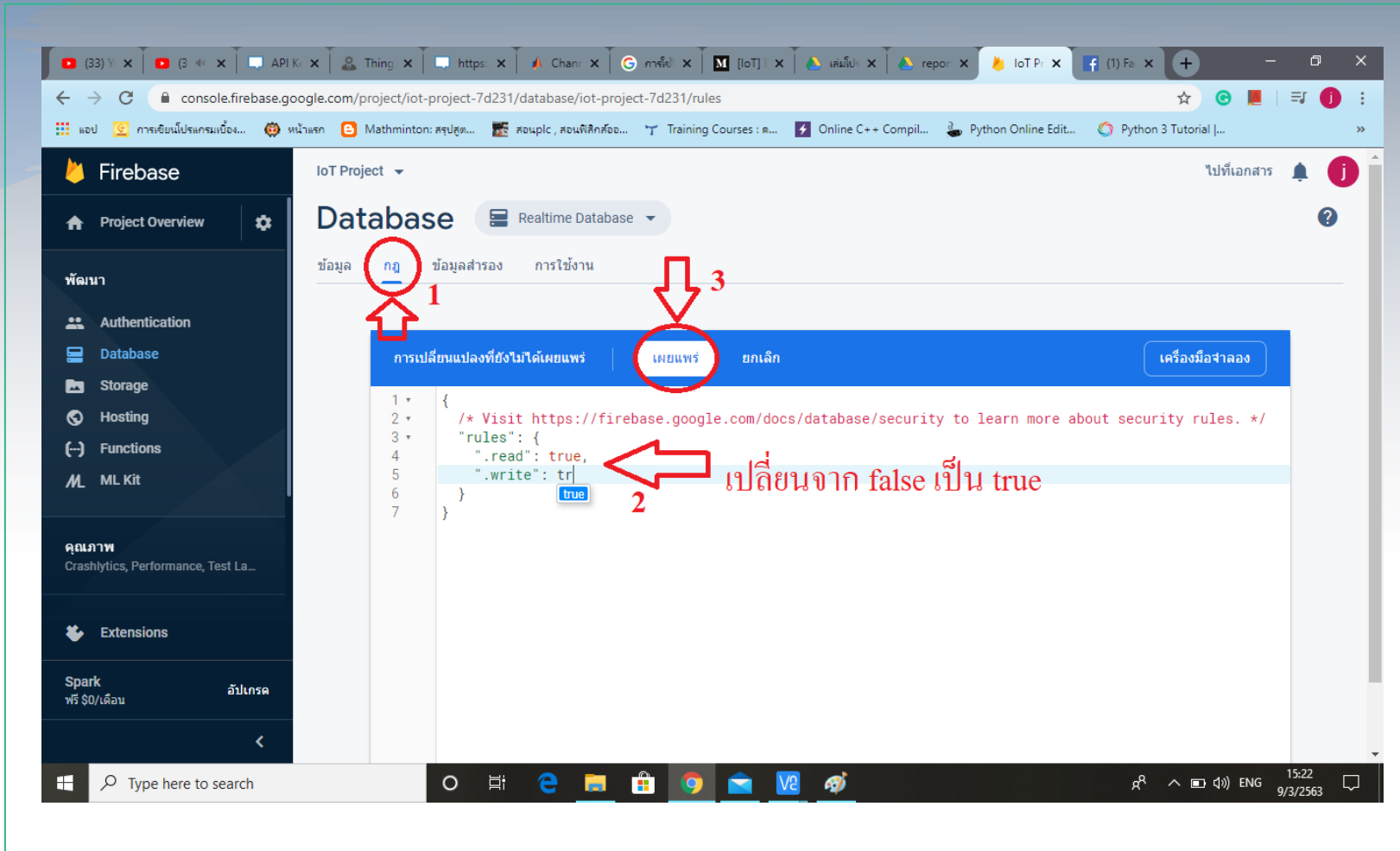
คุณภาพ
Crashlytics, Performance, Test La...

Extensions

Spark
ฟรี \$0/เดือน อัปเดต

Cloud Firestore
iot-project-7d231

Realtime Database
<https://iot-project-7d231.firebaseio.com>



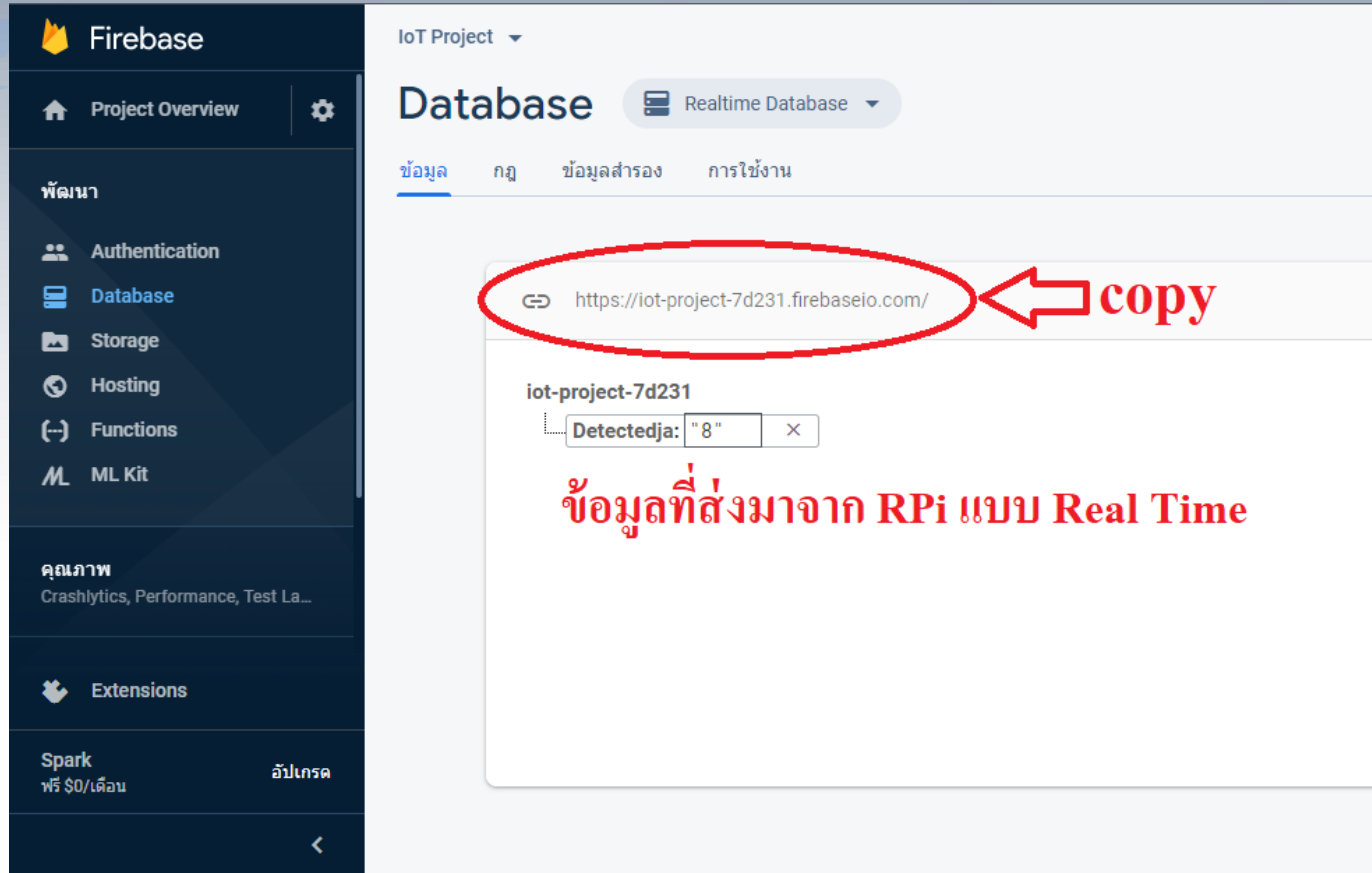
The screenshot shows the Firebase console for an IoT Project. The 'Database' section is active, and the 'Rules' tab is selected. The code editor shows the following rules:

```
1 {  
2   /* Visit https://firebase.google.com/docs/database/security to learn more about security rules. */  
3   "rules": {  
4     ".read": true,  
5     ".write": true  
6   }  
7 }
```

Red annotations indicate the steps to enable rules:

1. Click the 'Rules' tab.
2. Click the 'true' value next to the '.write' rule.
3. Click the 'Preview' button.

หมายเลข 3 เผยแพร่เพื่อที่จะสามารถแก้ไข .read และ .write ในข้อมูล Database ได้



Firestore

Project Overview

พัฒนา

- Authentication
- Database
- Storage
- Hosting
- Functions
- ML Kit

คุณภาพ

Crashlytics, Performance, Test La...

Extensions

Spark

ฟรี \$0/เดือน

อัปเดต

IoT Project

Database

Realtime Database

ข้อมูล กฎ ข้อมูลสำรอง การใช้งาน

<https://iot-project-7d231.firebaseio.com/>

iot-project-7d231

Detectedja: "8"

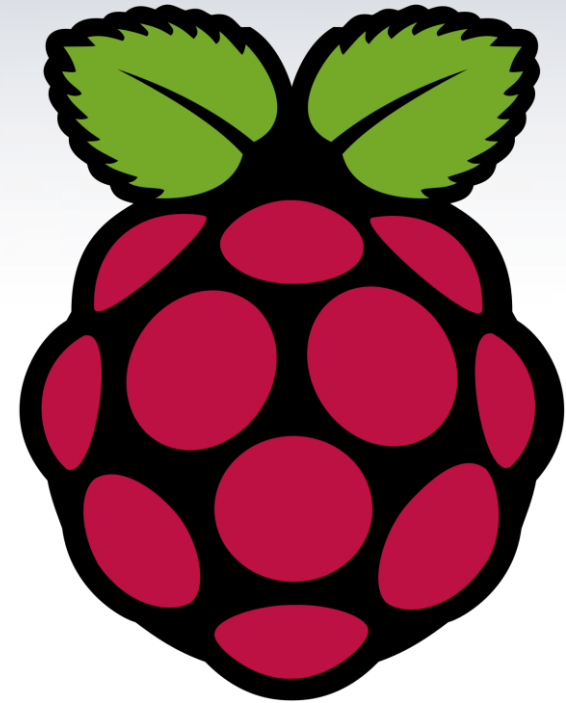
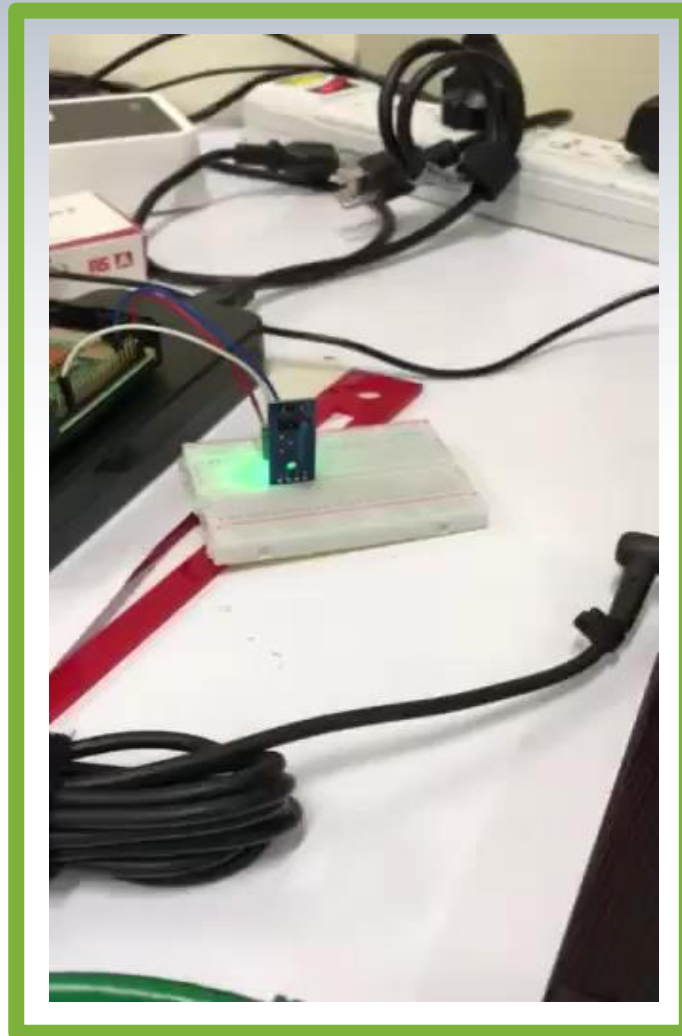
ข้อมูลที่ส่งมาจาก RPi แบบ Real Time

Copy เพื่อนำมาใส่ url ในการเขียนโปรแกรม Qt Creator

VDO การส่งข้อมูลไปยังเว็บไซต์ Firebase Google



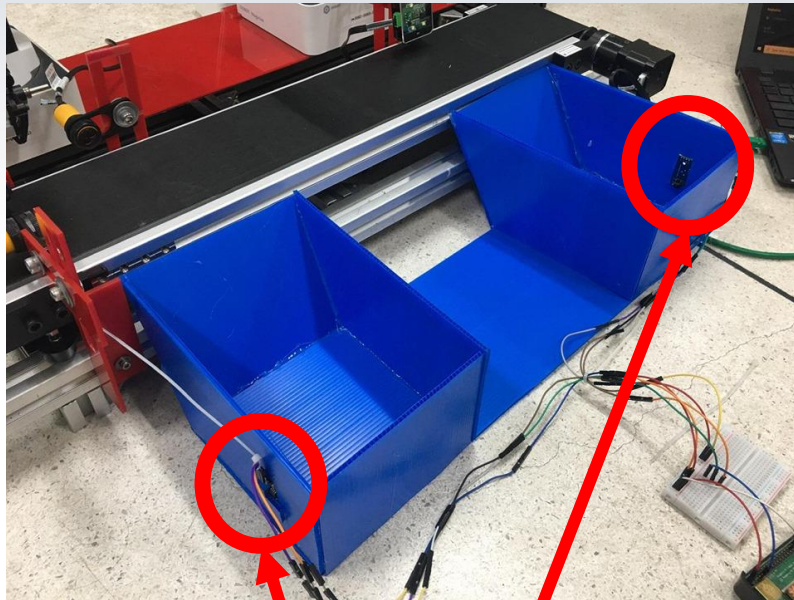
Firebase



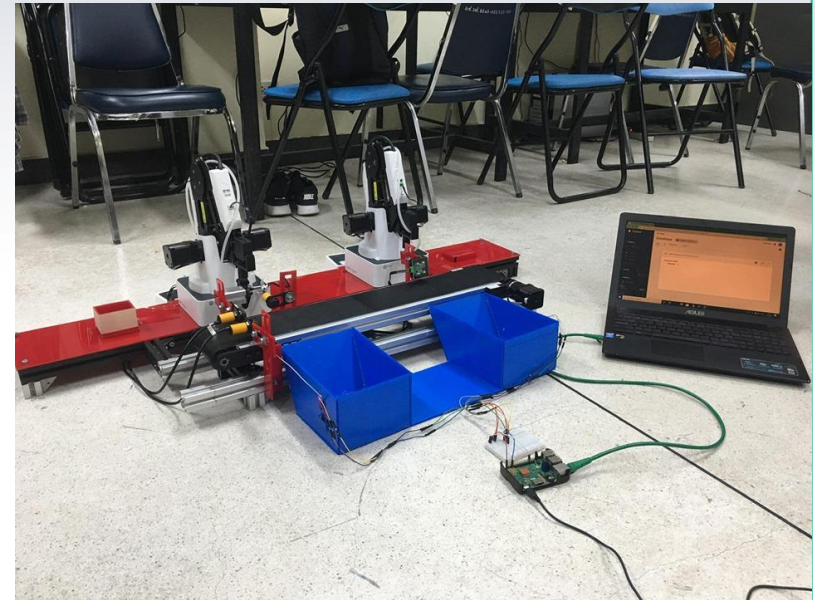
เมื่อนำ Dobot และ Raspberry Pi นำมาประกอบเข้าด้วยกัน



Infrared sensor

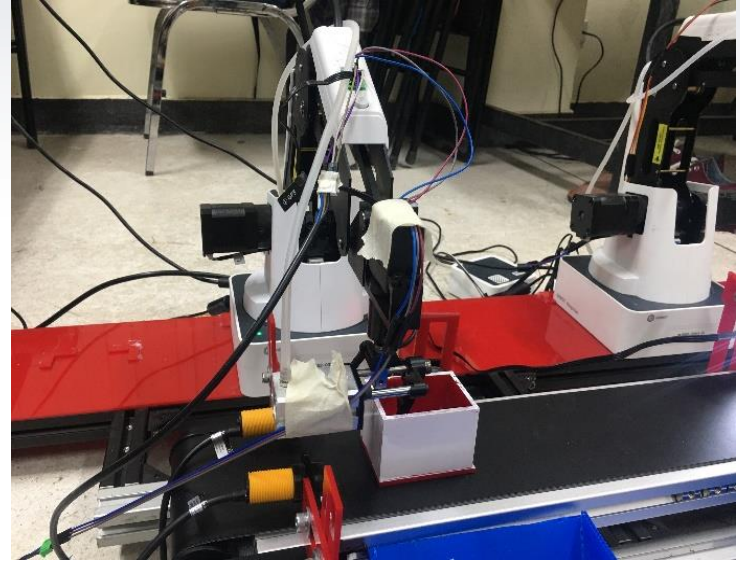
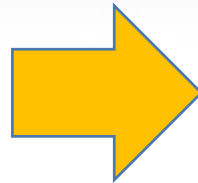
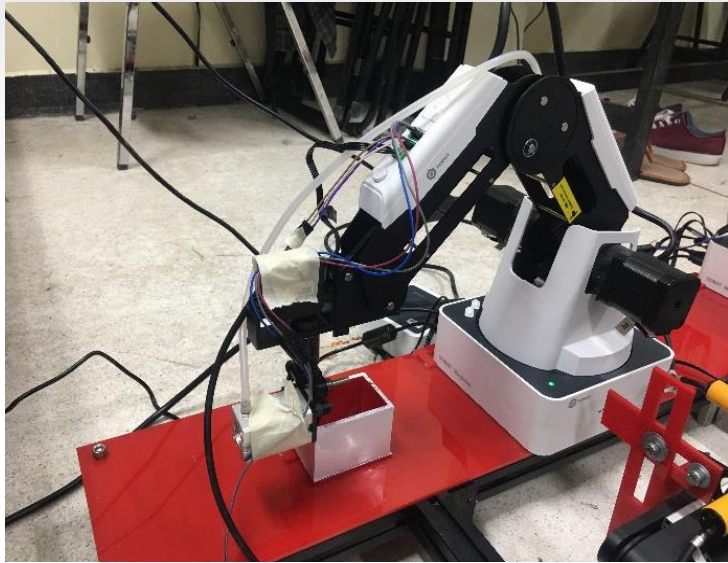


ตำแหน่งที่ติดตั้ง



การทำงานของ DOBOT

ตัวที่ 1

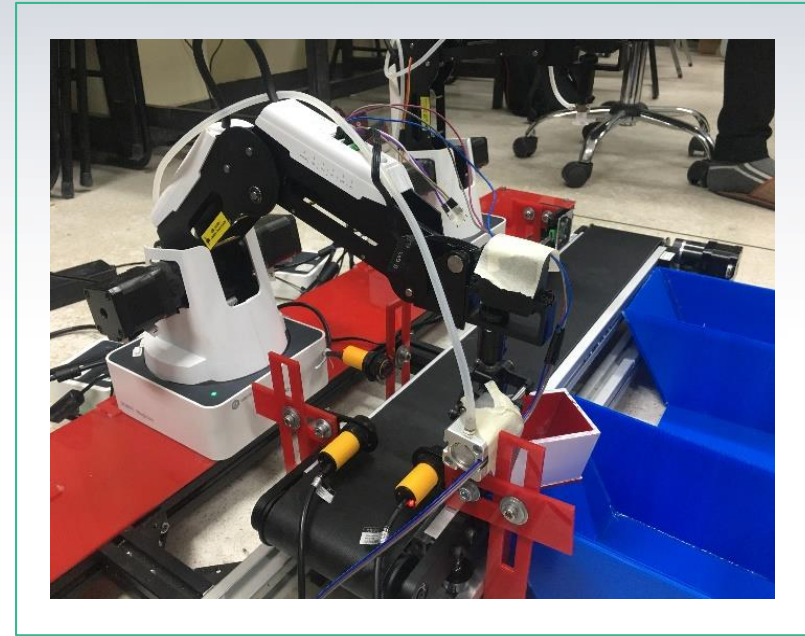
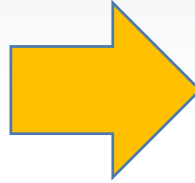


DOBOT ตัวที่ 1 ทำการหยิบกล่องจากตำแหน่งที่ 1

หยิบกล่องนำไปวางบนสายพาน



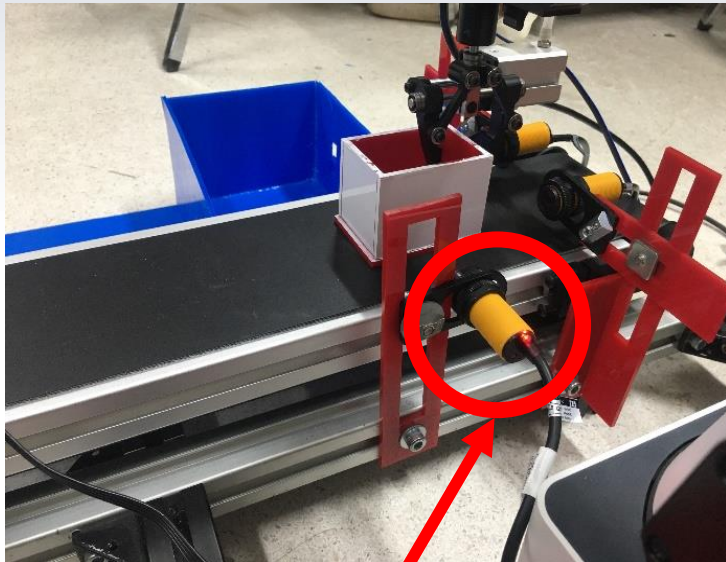
โฟโตอิเล็กทริกเซ็นเซอร์ (Photoelectric Sensor) ตรวจพบ



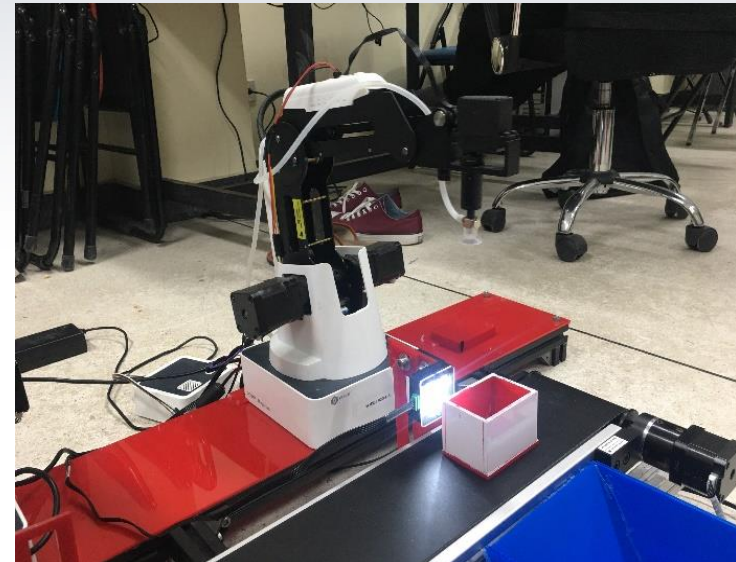
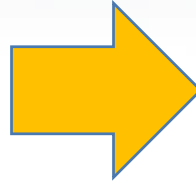
ทำการดันกล่องออกจากสายพาน

การทำงานของ DOBOT

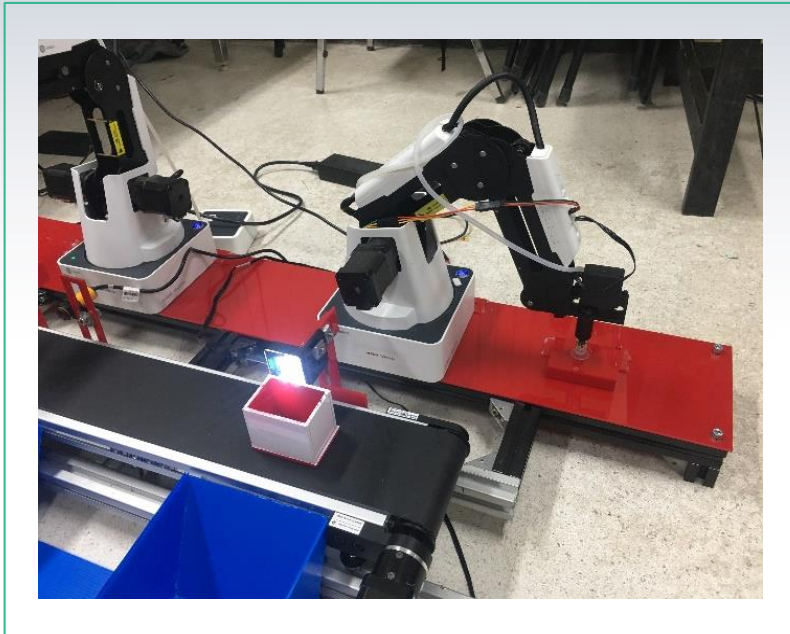
ตัวที่ 2



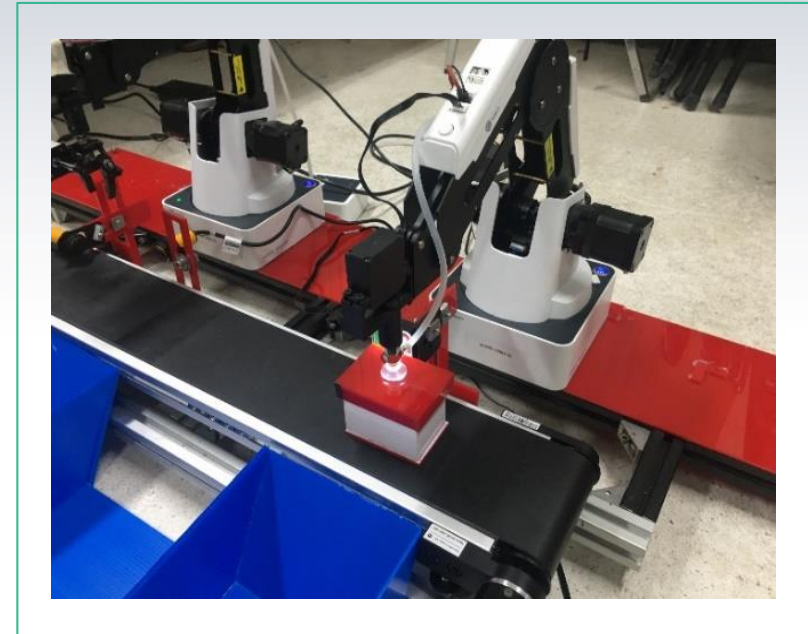
โฟโตอิเล็กทริกเซ็นเซอร์ (Photoelectric Sensor)
ทำการตรวจพบวัตถุ



สายพานเคลื่อนกล่อมายังตำแหน่งที่ 2

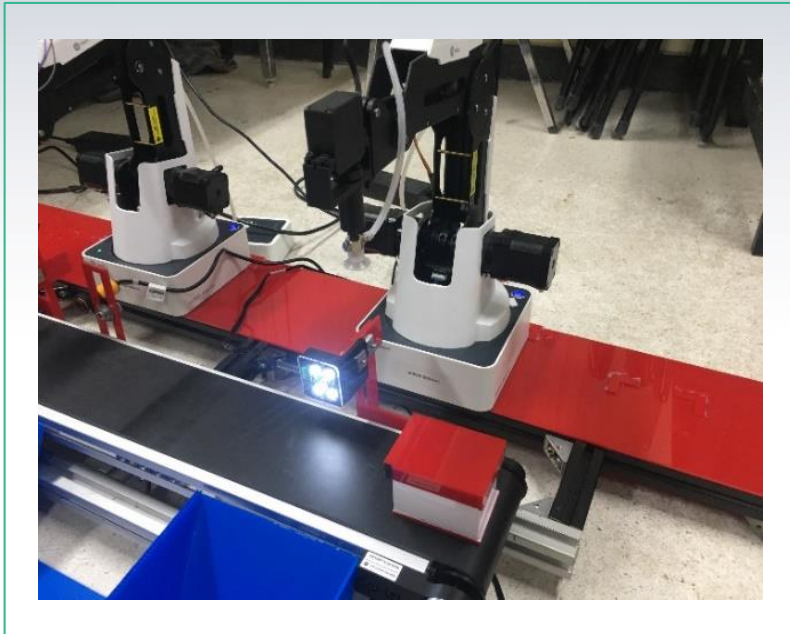


ทำการหยิบฟากล่องมาปิด



คัดเลอร์ตีเพคเตอร์เซ็นเซอร์ (Color Detector Sensor)

ทำการตรวจสอบว่ากล่องถูกปิดฝา



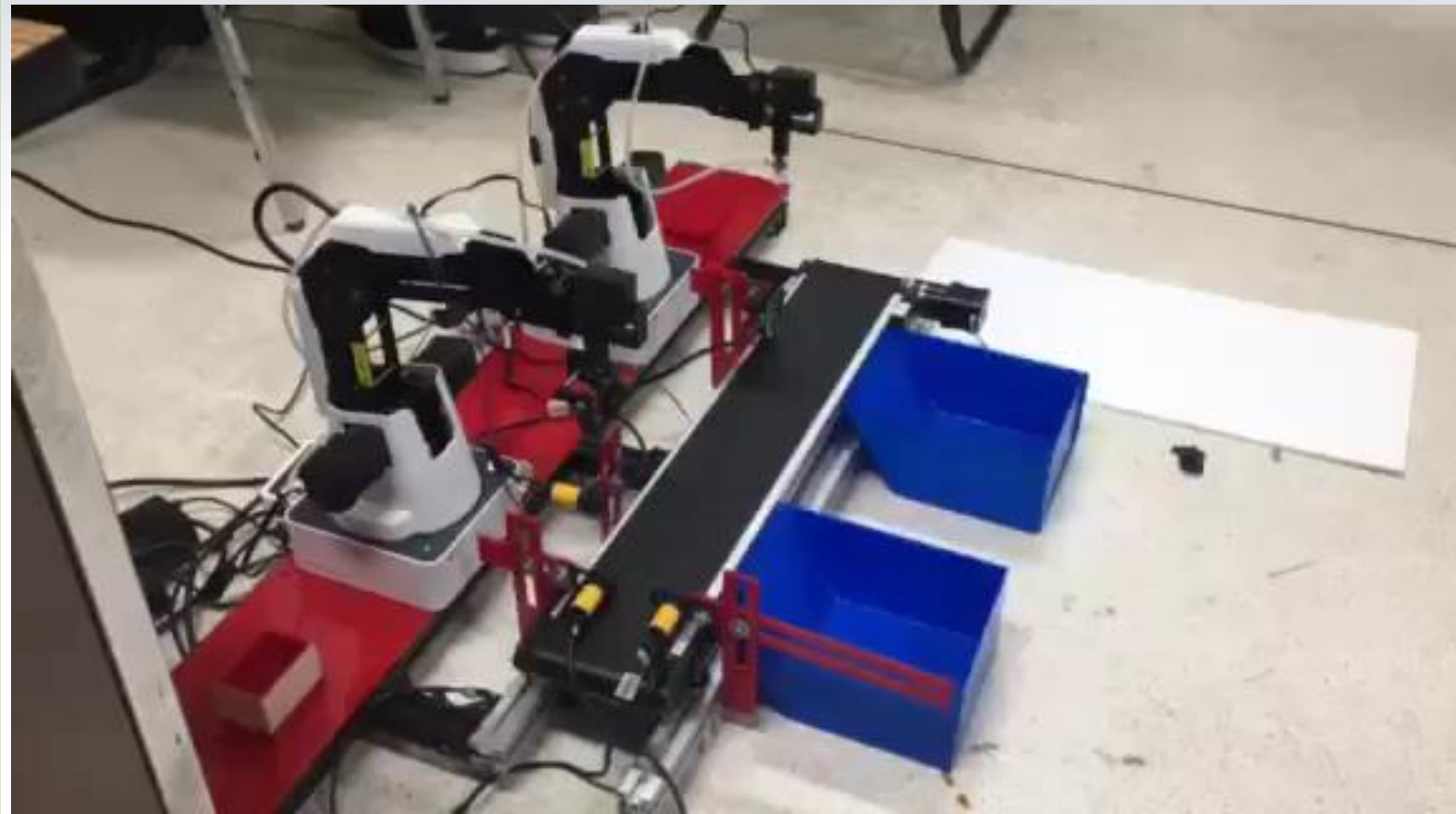
กล้องตรวจจับสี (Color Detector Sensor)
ตรวจพบสีแดง



ตรวจไม่พบ DOBOT ตัวที่ 2 ทำการดันกล่อง
ออกจากสายพาน

VDO กระบวนการทำงานของ Dobot และ Raspberry Pi

กรณีศึกษาที่ 1 Photoelectric Sensor ตรวจพบการวางตำแหน่งของกล่องไม่อยู่ตรงกลางสายพานในตำแหน่ง Dobot1



กรณีที่ 2 Color Detector Sensor ตรวจพบการวางฝากล่องที่ไม่เข้าพอดีกับตัวกล่อง



กรณีที่3 กระบวนการที่สมบูรณ์แบบ



การดำเนินการทดสอบ

Before

Dobot

After

ตาราง 4.1 ตรวจสอบการทำงานของ DOBOT

จำนวน ครั้ง	DOBOT ตัวที่ 1		DOBOT ตัวที่ 2		สาเหตุของความ ผิดพลาด
	ทำงาน ปกติ	ทำงาน ผิดปกติ	ทำงาน ปกติ	ทำงาน ผิดปกติ	
1	/			/	1. DOBOT มีความคลาดเคลื่อน
2	/		/		
3	/			/	2. ผู้ปฏิบัติงานวางฝาผิดตำแหน่งทำให้ DOBOT หยิบผิดตำแหน่ง
4	/		/		
5	/		/		3. ตำแหน่ง Color Detector Sensor อยู่ต่ำเกินไป
6	/		/		
7	/		/		4. DOBOT มีการใช้งานอย่างต่อเนื่องทำให้ตัว DOBOT เกิดความร้อน จึงทำให้เกิดความคลาดเคลื่อน
8	/			/	
9	/			/	
10	/			/	
11	/		/		
12	/		/		
13	/			/	
14	/			/	
15	/		/		
16	/		/		
17	/		/		
18	/			/	
19	/			/	
20	/		/		
รวม	20	0	11	9	

ตาราง 4.3 ตรวจสอบการทำงานของ DOBOT หลังการปรับปรุง

จำนวน ครั้ง	DOBOT ตัวที่ 1		DOBOT ตัวที่ 2		สาเหตุของความ ผิดพลาด
	ทำงาน ปกติ	ทำงาน ผิดปกติ	ทำงาน ปกติ	ทำงาน ผิดปกติ	
1	/			/	
2	/		/		
3	/		/		
4	/		/		
5	/		/		
6	/		/		
7	/		/		
8	/		/		
9	/			/	
10	/		/		
11	/		/		
12	/		/		
13	/		/		
14	/		/		
15	/		/		
16	/		/		
17	/			/	
18	/		/		
19	/		/		
20	/		/		
รวม	20	0	11	9	

Before

Raspberry Pi

After

ตาราง 4.2 ตรวจสอบการทำงานของเซนเซอร์นับจำนวนของเสีย

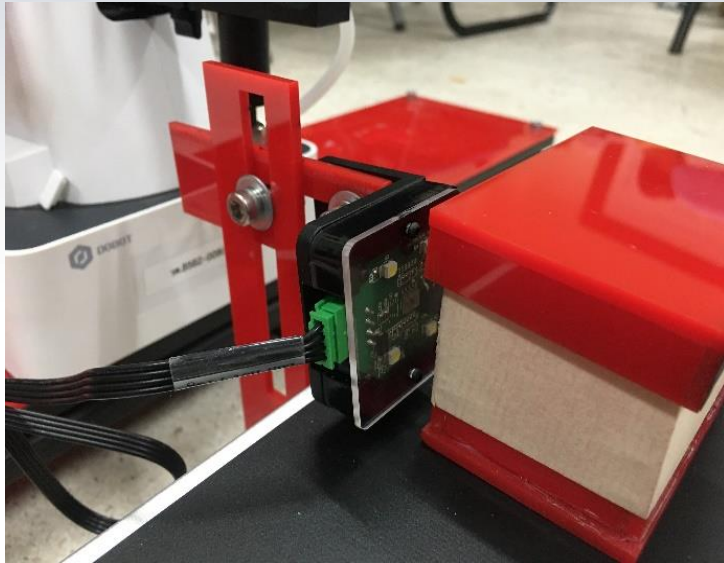
จำนวนครั้ง	การตรวจเซนเซอร์แบบ monitor		การตรวจเซนเซอร์แบบ real time		สาเหตุของความผิดพลาด
	ทำงาน	ไม่ทำงาน	ส่งข้อมูล	ไม่ส่งข้อมูล	
1		/		/	1. ตำแหน่งเซนเซอร์อยู่ห่างจากชิ้นงานมากเกินไป
2	-	-	-	-	
3	/		/		2. ตัวกล้องมีสีทึบมากเกินไปทำให้เซนเซอร์ไม่สามารถตรวจพบ
4	-	-	-	-	
5	-	-	-	-	3. อินเตอร์เน็ตเกิดความขัดข้อง
6	-	-	-	-	
7	-	-	-	-	
8	/		/	/	
9	/		/		
10	/		/		
11	-	-	-	-	
12	-	-	-	-	
13		/		/	
14	/		/		
15	-	-	-	-	
16	-	-	-	-	
17	-	-	-	-	
18	/		/		
19	/		/		
20	-	-	-	-	
รวม	7	2	6	3	

ตาราง 4.4 ตรวจสอบการทำงานของเซนเซอร์นับจำนวนของเสียหลังการปรับปรุง

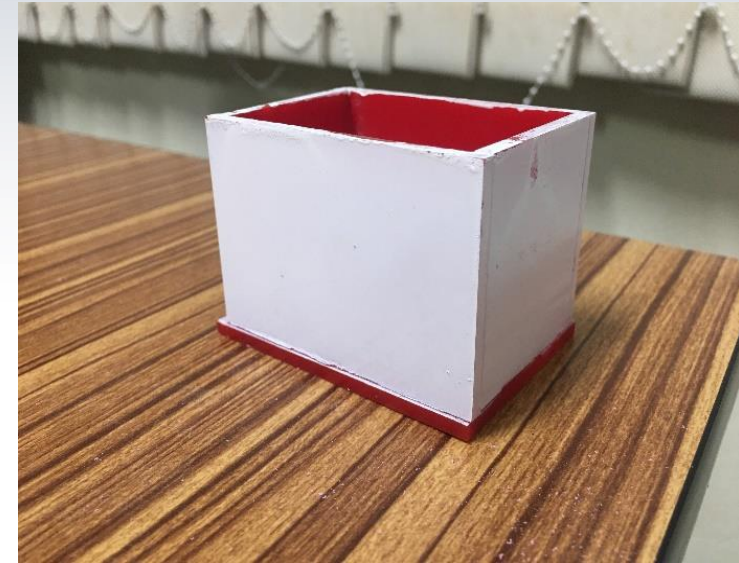
จำนวนครั้ง	การตรวจเซนเซอร์แบบ monitor		การตรวจเซนเซอร์แบบ real time		สาเหตุของความผิดพลาด
	ทำงาน	ไม่ทำงาน	ส่งข้อมูล	ไม่ส่งข้อมูล	
1		/		/	
2	-	-	-	-	
3	-	-	-	-	
4	-	-	-	-	
5	-	-	-	-	
6	-	-	-	-	
7	-	-	-	-	
8	-	-	-	-	
9	/		/		
10	-	-	-	-	
11	-	-	-	-	
12	-	-	-	-	
13	-	-	-	-	
14	-	-	-	-	
15	-	-	-	-	
16	-	-	-	-	
17	/	-	/	-	
18	-	-	-	-	
19	-	-	-	-	
20	-	-	-	-	
รวม	2	1	2	1	

ทำการปรับปรุงเพื่อลดความผิดพลาดของการทำงาน

การปรับตำแหน่งเซนเซอร์ให้เหมาะสม



พ่นสีกล่องให้เป็นสีขาว



เนื่องจากตอนแรกกล่องถูกทำให้เป็นสีดำ sensor จึง
ตรวจจับได้ไม่ดี อาจเกิดจากการที่ sensor ส่งสัญญาณ
ออกไปหากล่อง แล้วถูกความเข้มของสีที่ดูคลืน
สัญญาณ จึงไม่สามารถสะท้อนกลับมาหาตัวรับของ
sensor ได้

สรุปผลการทดลอง

ในการเดินเครื่องครั้งแรกเช่น ขั้นตอนการทำงานที่ไม่เป็นไปตามการออกแบบปัญหาความคลาดเคลื่อนของ DOBOT ในเรื่องของตำแหน่งการหยิบกล่อง และการทำงานของเซนเซอร์ต่างๆที่มีการทำงานที่ไม่คงที่ จากนั้นผู้วิจัยได้ทำการปรับปรุง หลังการปรับปรุงผลที่ได้ออกมาพึงพอใจมาก กระบวนการสามารถทำงานได้ มีความผิดพลาดในการทำงานน้อยมาก และลงข้อมูลแบบตอบสนองทันที (Real Time) ได้

- เนื่องจากโรงงานอุตสาหกรรมในยุคปัจจุบันได้มีการนำกระบวนการตรวจสอบภายในระบบอัตโนมัติและส่งข้อมูลแบบ Realtime ที่มีความเสถียรและมีราคาที่สูงอย่างมาก กลุ่มของข้าพเจ้าจึงได้ทำโครงการขึ้นนี้ขึ้นมาเพื่อ ศึกษากระบวนการในระดับที่เล็กลงจากของจริง เพื่อที่จะได้เป็นตัวต้นแบบในการเรียนรู้ระบบ INDUSTRIAL ROBOTICS ,MANUFACTURING AUTOMATION SYSTEM และพัฒนาในอนาคตของภาควิชาวิศวกรรมอุตสาหการต่อไป



THANK YOU