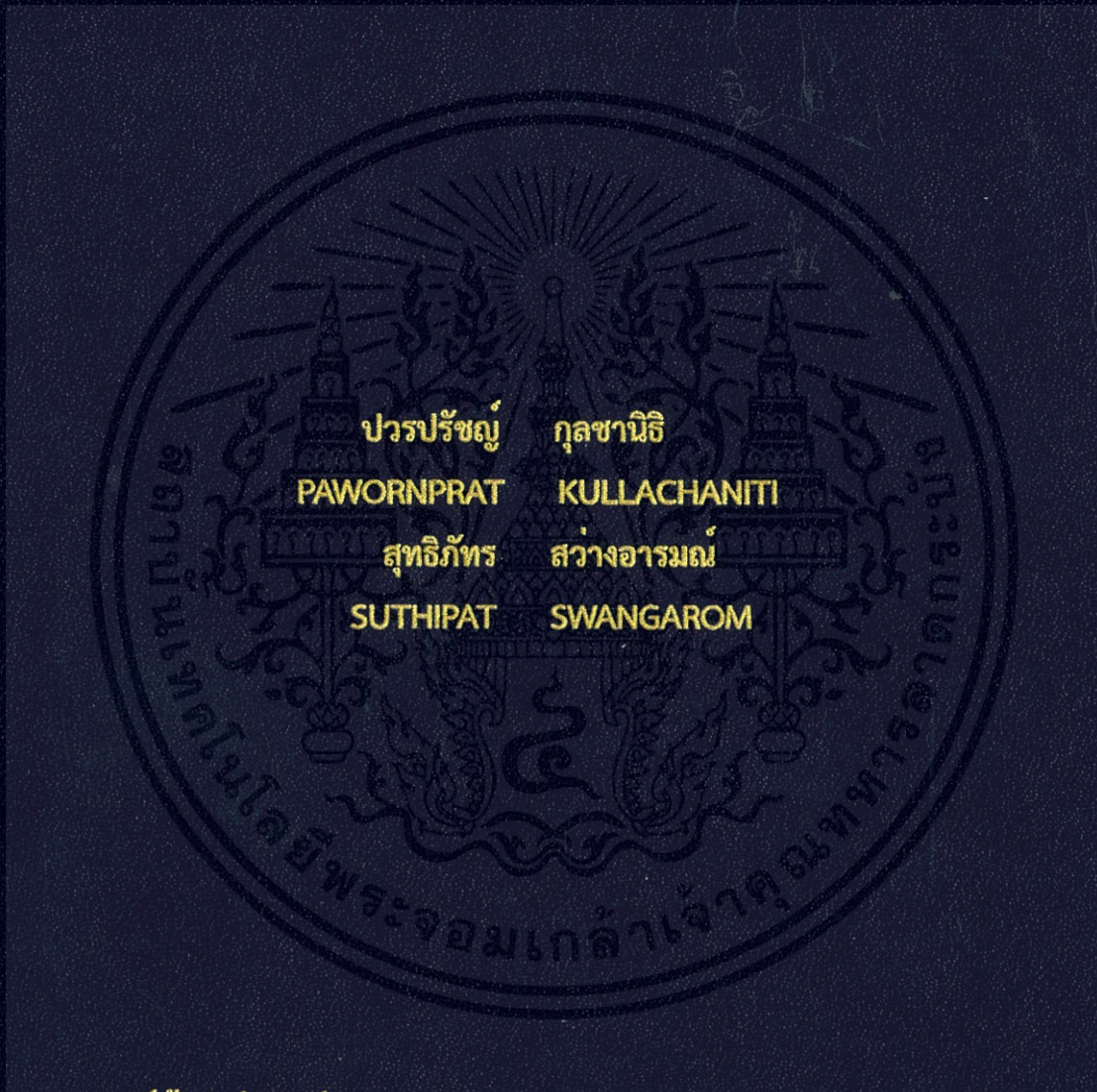


เครื่องให้อาหารสัตว์เลี้ยงอัตโนมัติผ่านระบบอินเทอร์เน็ต

AUTOMATIC HAPPY PET FEEDER



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2557

เครื่องให้อาหารสัตว์เลี้ยงอัตโนมัติผ่านระบบอินเทอร์เน็ต

AUTOMATIC HAPPY PET FEEDER



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการเรียนเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

พ.ศ.2557

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AUTOMATIC HAPPY PET FEEDER



PAWORNPRAT KULLACHANITI
SUTHIPAT SWANGAROM

THIS THESIS IS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
BACHELOR OF ENGINEERING IN INFORMATION ENGINEERING
FACULTY OF ENGINEERING
KING MONTKUT'S OF TECHNOLOGY LADKRABANG
ACADEMIC YEAR 2014

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2557

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ใบรับรองปริญญานิพนธ์

หัวข้อปริญญานิพนธ์

เครื่องให้อาหารสัตว์อัตโนมัติผ่านระบบอินเทอร์เน็ต

Thesis Title

AUTOMATIC HAPPY PET FEEDER

ชื่อนักศึกษา

นายปวรปรัชญ์ กุลขานิติ

นางสาวสุทธิภัทร สว่างอารมณ์

ระดับปริญญา

วิศวกรรมศาสตรบัณฑิต

สาขาวิชา

วิศวกรรมสารสนเทศ

ปริญญานิพนธ์ปีการศึกษา

2557

(.....)

ผศ.ไพศาล สิทธิโยภาสกุล

อาจารย์ที่ปรึกษาปริญญานิพนธ์

(.....)

ผศ.ดร.กฤตากร กล่อมการ

อาจารย์ที่ปรึกษาปริญญานิพนธ์ร่วม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VII
สารบัญรูป.....	VIII
บทที่ 1 บทนำ.....	1
1.1 แนวคิดและที่มาของปัญหา.....	1
1.2 วัตถุประสงค์ของการศึกษา.....	1
1.3 ขอบเขตของโครงการ.....	2
1.3.1 ขอบเขตด้านฮาร์ดแวร์.....	2
1.3.2 ขอบเขตด้านซอฟต์แวร์.....	2
1.4 ขั้นตอนการดำเนินงาน.....	2
1.5 ผลที่คาดว่าจะได้รับ.....	3
บทที่ 2 ทฤษฎีพื้นฐานที่ใช้.....	4
2.1 ไมโครคอนโทรลเลอร์อาร์ดูโนลีโอนาร์โด.....	4
2.1.1 ความจำของอาร์ดูโนลีโอนาร์โด.....	5
2.1.2 อินพุตและเอาต์พุต.....	5
2.1.3 การติดต่อสื่อสาร.....	6
2.1.4 การเขียนโปรแกรม.....	7
2.2 เซอร์โวมอเตอร์.....	7
2.2.1 ส่วนประกอบภายนอกเซอร์โวมอเตอร์.....	8
2.2.2 ส่วนประกอบภายในเซอร์โวมอเตอร์.....	9
2.2.3 หลักการทำงานเซอร์โวมอเตอร์.....	9
2.2.4 วิธีการควบคุมเซอร์โวมอเตอร์ด้วยอาร์ดูโน.....	10

สารบัญ (ต่อ)

	หน้า
2.3 โซลีนอยด์.....	11
2.3.1 หลักการทำงานของโซลีนอยด์.....	11
2.4 รีเลย์.....	13
2.4.1 คุณสมบัติของรีเลย์.....	15
2.4.2 ขาสัญญาณ.....	16
2.5 กล้องเว็บแคม.....	17
2.5.1 ความหมายของกล้องเว็บแคม.....	17
2.5.2 ชนิดของกล้องเว็บแคม.....	17
2.6 ภาษาซี.....	18
2.6.1 โครงสร้างของโปรแกรมภาษาซี.....	18
2.6.2 ตัวแปรในโปรแกรมภาษาซี.....	20
2.6.3 คำสงวนในโปรแกรมภาษาซี.....	21
2.6.4 ฟังก์ชันของโปรแกรมภาษาซี.....	21
2.6.5 ชนิดของข้อมูล.....	23
2.6.6 ตัวดำเนินการทางคณิตศาสตร์.....	23
2.6.7 ตัวดำเนินการเปรียบเทียบ.....	24
2.6.8 ตัวดำเนินการทางตรรกะ.....	25
2.6.9 นิพจน์ทางคณิตศาสตร์.....	25
2.7 ภาษาซีพลัสพลัส.....	26
2.7.1 รูปแบบของภาษาซีพลัสพลัส.....	26
2.8 IIS7.....	26
2.9 วิชวลเบสิก.....	27
2.9.1 จุดเด่นของภาษาวิชวลเบสิก.....	27
2.10 Twitch.....	27
2.11 xSplit.....	28
2.11.1 การตั้งค่าความละเอียด.....	29
2.11.2 การตั้งค่า Frame Per Second.....	29

2.11.3 การตั้งค่าสตรีมมิ่ง.....	29
บทที่ 3 การวิเคราะห์และออกแบบ.....	30
3.1 แนวคิดของระบบ.....	30
3.2 ภาพรวมของระบบ.....	31
3.3 ยูสเคสของระบบ.....	32
3.4 โพลชาร์ตของระบบ.....	32
3.4.1 โพลชาร์ตของระบบควบคุมการหมุนของล้อ.....	32
3.4.2 โพลชาร์ตของระบบการสั่งงานให้อาหารสัตว์โดยกำหนดเป็นวัน/เดือน/ปี ชั่วโมง (เวลา) และชั่วโมง(นาที)	33
3.4.3 โพลชาร์ตของระบบการสั่งงานให้อาหารสัตว์โดยกำหนดเป็นปริมาณอาหาร.....	34
3.5 การออกแบบเว็บไซต์.....	35
3.6 การเชื่อมต่อของวงจร.....	36
บทที่ 4 ผลการดำเนินงาน.....	37
4.1 ผลการใช้งานล้อเว็บแคม.....	37
4.2 ผลการใช้งานเครื่องให้อาหารสัตว์แบบตั้งค่าวัน/เดือน/ปี.....	38
4.3 ผลการใช้งานเครื่องให้อาหารสัตว์แบบกำหนดปริมาณอาหาร.....	39
บทที่ 5 สรุปผลการทดลอง.....	40
5.1 ปัญหาที่เกิดขึ้น.....	40
5.2 แนวทางการพัฒนาต่อ.....	40
บรรณานุกรม.....	41
ภาคผนวก.....	42

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้า
ตารางที่ 1.1 ขั้นตอนการดำเนินงาน.....	2
ตารางที่ 2.1 คุณสมบัติไมโครคอนโทรลเลอร์อาร์ดูโนลีโอนาร์โด.....	5
ตารางที่ 2.2 ขาที่ใช้ในการเชื่อมต่อรีเลย์4ช่องสัญญาณ.....	16
ตารางที่ 2.3 คำอธิบายโครงสร้างของโปรแกรมภาษาซี.....	19
ตารางที่ 2.4 รูปแบบการประกาศค่าตัวแปร.....	20
ตารางที่ 2.5 แสดงค่าสงวนของโปรแกรมภาษาซี.....	21
ตารางที่ 2.6 แสดงค่ารหัสสีที่ใช้ในโปรแกรมภาษาซี.....	22
ตารางที่ 2.7 แสดงตัวดำเนินการทางคณิตศาสตร์.....	24
ตารางที่ 2.8 แสดงตัวดำเนินการเปรียบเทียบ.....	24
ตารางที่ 2.9 แสดงนิพจน์ทางคณิตศาสตร์.....	25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

	หน้า
รูปที่ 2.1 โครงสร้างอาร์คิโนลีโอนาร์โด.....	4
รูปที่ 2.2 อาร์คิโนลีโอนาร์โด.....	7
รูปที่ 2.3 ส่วนประกอบภายนอกเซอร์โวมอเตอร์.....	8
รูปที่ 2.4 สายไฟของเซอร์โวมอเตอร์.....	8
รูปที่ 2.5 ส่วนประกอบภายในเซอร์โวมอเตอร์.....	9
รูปที่ 2.6 หลักการทำงานของเซอร์โวมอเตอร์.....	10
รูปที่ 2.7 การหมุนของเซอร์โวมอเตอร์.....	10
รูปที่ 2.8 กฎของมือขวา.....	12
รูปที่ 2.9 การไหลของแม่เหล็กและเส้นแรงแม่เหล็ก.....	12
รูปที่ 2.10 แสดงแกนเหล็กและเส้นแรงแม่เหล็ก.....	13
รูปที่ 2.11 สัญญาณวงจรไฟฟ้าในรีเลย์.....	13
รูปที่ 2.12 แสดงสวิตช์ของรีเลย์.....	14
รูปที่ 2.13 สภาวะของรีเลย์.....	14
รูปที่ 2.14 ข้อมูลบนตัวของรีเลย์.....	15
รูปที่ 2.15 แสดงขาที่ใช้ในการเชื่อมต่อของรีเลย์ 4 ช่องสัญญาณ.....	16
รูปที่ 2.16 กล้องเว็บแคมแบบมีสาย.....	17
รูปที่ 2.17 กล้องเว็บแคมแบบไร้สาย.....	18
รูปที่ 2.18 หน้าจอโปรแกรม xSplit.....	28
รูปที่ 2.19 หน้าจอการตั้งค่าสตรีมมิ่ง.....	29
รูปที่ 3.1 แสดงภาพรวมของระบบ.....	31
รูปที่ 3.2 แสดงยูสเคสของระบบ.....	32
รูปที่ 3.3 แสดงฟลิวชาร์ตของระบบควบคุมการหมุนของกล้อง.....	32
รูปที่ 3.4 แสดงฟลิวชาร์ตของระบบการสั่งงานให้อาหารโดยกำหนดเป็นวัน/เดือน/ปี.....	33
รูปที่ 3.5 แสดงฟลิวชาร์ตของระบบการสั่งงานให้อาหารโดยกำหนดเป็นปริมาณ.....	34
รูปที่ 3.6 แสดงหน้าเว็บไซต์.....	35
รูปที่ 3.7 แสดงการเชื่อมต่อของวงจร.....	36

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้อ่านเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

	หน้า
รูปที่ 4.1 แสดงหน้าต่างเมื่อระบบยังไม่เชื่อมต่อโปรแกรม.....	37
รูปที่ 4.2 แสดงหน้าต่างเมื่อระบบสามารถทำการสตรึมมิ่งสำเร็จ.....	38
รูปที่ 4.3 แสดงหน้าต่างเมื่อป้อนข้อมูลวัน/เดือน/ปี และเวลา.....	38
รูปที่ 4.4 แสดงข้อมูลที่เก็บในลิสต์บ็อก.....	39
รูปที่ 4.5 แสดงหน้าต่างการใช้งานแบบป้อนปริมาณอาหาร.....	39



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญาานิพนธ์ เครื่องให้อาหารสัตว์อัตโนมัติผ่านระบบอินเทอร์เน็ต
Thesis Title AUTOMATIC HAPPY PET FEEDER
ชื่อนักศึกษา นายปวรปรัชญ์ กุลขานิติ รหัสนักศึกษา 54010801
นางสาวสุทธิภัทร สว่างอารมณ์ รหัสนักศึกษา 54011394
ระดับปริญญา วิศวกรรมศาสตรบัณฑิต
สาขาวิชา วิศวกรรมสารสนเทศ
ปีการศึกษา 2557
อาจารย์ที่ปรึกษาปริญญาานิพนธ์ ผศ.ไพศาล สิทธิโยภาสกุล
อาจารย์ที่ปรึกษาปริญญาานิพนธ์ร่วม ผศ.ดร.กฤตดากร กลุ่มการ

บทคัดย่อ

เนื่องจากปัจจุบันผู้เลี้ยงสัตว์เลี้ยงมักเกิดปัญหาไม่สามารถให้อาหารสัตว์ได้ขณะไม่อยู่ในที่พักอาศัยและอาจก่อให้เกิดความเดือดร้อนถ้าหากมีความจำเป็นในการเดินออกไปนอกบ้านหรือต่างประเทศเป็นระยะเวลานาน ทางผู้จัดทำจึงคิดพัฒนาการให้อาหารสัตว์โดยการจัดทำเครื่องให้อาหารสัตว์อัตโนมัติขึ้นมาเพื่อให้ความสะดวกสบายแก่ผู้เลี้ยงสัตว์เลี้ยง โดยเครื่องให้อาหารสัตว์สามารถให้อาหารอัตโนมัติเมื่อได้รับคำสั่งในการให้อาหารพร้อมทั้งสามารถเรียกดูภาพจากกล้องเว็บแคม (Webcam) แบบสตรีมมิ่งได้ โดยการใช้งานทุกอย่างจะต้องผ่านการใช้งานระบบเครือข่ายอินเทอร์เน็ต(Internet) ซึ่งผู้จัดทำตระหนักว่าจะส่งประโยชน์ให้แก่ผู้เลี้ยงสัตว์เลี้ยงเป็นอย่างมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	Automatic happy pet feeder		
Student	Mr.Pawornprat Kullachaniti	Student ID.	54010801
	Miss.Suthipat Swangarom	Student ID.	54011394
Degree	Bachelor of Engineering		
Program	Information Engineering		
Academic Year	2557		
Thesis Advisor	Asst.Prof.Dr.Paisarn Sittiyopaskul		
Thesis Co-Advisor	Asst.Prof.Dr.Kitdakorn Klomkarn		

ABSTRACT

The goal of this project is the automatic pet feeder. We understand that feeding the pet is a big problem for who have pets at home and when they want to go outside they can not feed their pets. Our project is solve for this problem. We can use the automatic happy pet feeder everywhere that we have the internet and we can watch our pets by using webcam in real-time. We realize this product will be give some benefit to who love pets as well.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญาโทเล่มนี้สำเร็จได้ด้วยความกรุณาจากอาจารย์ที่ปรึกษา ผศ.ไพศาล สิทธิโยภาสกุล และ ผศ.ดร.กฤตากร กลุ่มการ ที่ท่านอาจารย์คอยให้ความช่วยเหลือ ให้คำชี้แนะช่วยแก้ปัญหาตลอดจนให้ความรู้และประการณ์ที่ดีแก่ผู้จัดทำ ทางผู้จัดทำขอขอบพระคุณเป็นอย่างสูงมา ณ ที่นี้

ขอบพระคุณคุณพ่อ คุณแม่ และครอบครัวที่คอยอยู่เบื้องหลังในการเรียน คอยให้กำลังใจแก่ผู้จัดทำ และสนับสนุนในการเรียน จนทุกอย่างสำเร็จไปได้ด้วยดีตลอดมา และสิ่งหนึ่งที่ขาดไม่ได้คือผู้จัดทำต้องขอบคุณเพื่อนๆทุกคนที่คอยช่วยเหลือ ให้คำแนะนำ ช่วยแก้ปัญหา และคอยให้กำลังใจกันและกัน จนทำให้โครงการออกมาสำเร็จลุล่วงไปได้ด้วยดี



ปวรปรัชญ์ กุลขานิติ
สุทธิภัทร สว่างอารมณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 แนวคิดและที่มาของปัญหา

จากการเลี้ยงสัตว์เลี้ยงในปัจจุบันอาจเป็นปัญหาสำหรับผู้ที่ไม่สามารถอยู่เฝ้าดูแลสัตว์เลี้ยงที่บ้านตลอดเวลา เครื่องให้อาหารสัตว์อัตโนมัติผ่านสมาร์ทโฟน จึงถูกคิดค้นและทำขึ้นมาเพื่อให้คนรักสัตว์สามารถให้อาหารสัตว์เลี้ยงได้ตามความต้องการของเจ้าของ เนื่องจากผู้เลี้ยงอาจมีธุระหรือภารกิจที่ไม่สามารถอยู่เฝ้าดูแลสัตว์เลี้ยงได้ โดยที่เจ้าของยังสามารถให้อาหารสัตว์เลี้ยงได้เหมือนเดิมทุกวัน ในบางกรณีเจ้าของก็จะนำสัตว์เลี้ยงไปฝากญาติ หรือเพื่อนบ้าน ซึ่งบางครั้งสัตว์เลี้ยงของเรา ก็อาจจะก่อความรำคาญให้กับผู้ที่ดูแล

ทางคณะผู้จัดทำโครงการจึงศึกษาและจัดทำเครื่องให้อาหารสัตว์อัตโนมัติผ่านสมาร์ทโฟน เพื่อตอบสนองความต้องการของคนรักสัตว์ โดยที่เครื่องให้อาหารสัตว์อัตโนมัติผ่านสมาร์ทโฟนจะมีกล้องเพื่อสามารถสังเกตติดตามการรับประทานอาหารของสัตว์เลี้ยงว่ามีลักษณะการอยู่การกินอย่างไร ทำให้เพิ่มความสะดวกสบายให้กับผู้รักสัตว์เลี้ยง

1.2 วัตถุประสงค์

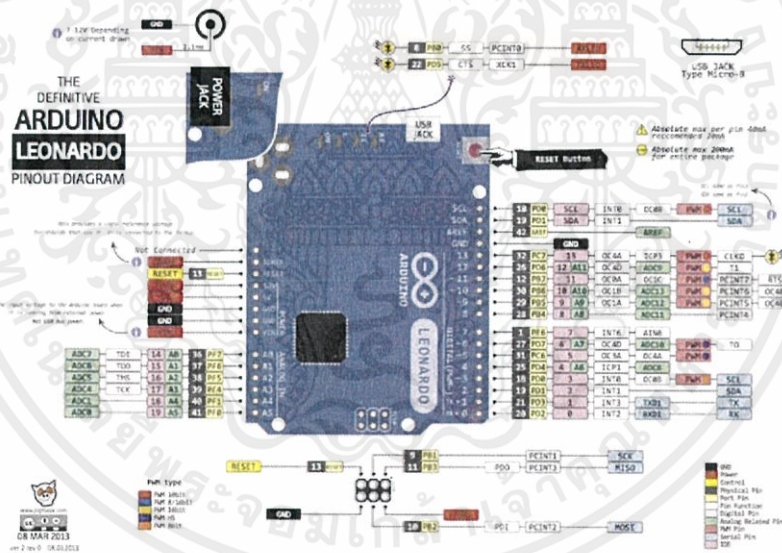
- 1.2.1 เพื่ออำนวยความสะดวกในการให้อาหารสัตว์เลี้ยง
- 1.2.2 เพื่อพัฒนาเครื่องให้อาหารสัตว์เลี้ยงให้มีความทันสมัยมากขึ้น
- 1.2.3 เพื่อนำความรู้ที่ได้ศึกษามาประยุกต์ใช้และพัฒนาอุปกรณ์ให้มีประโยชน์และใช้งานได้สะดวกสบายมากขึ้น
- 1.2.4 เพื่อศึกษาและออกแบบซอฟต์แวร์ (Software) และ ฮาร์ดแวร์ (Hardware) ที่ใช้ในการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2 ทฤษฎีพื้นฐานที่ใช้

2.1 ไมโครคอนโทรลเลอร์อาร์ดูโนลีโอนาร์โด (Arduino Leonardo)

Arduino Leonardo เป็นไมโครคอนโทรลเลอร์บนพื้นฐานของ ATMEGA32U4 ซึ่งมี 20 ขาดิจิตอลอินพุต / เอาต์พุต, สามารถบันทึกการทำงานที่ 16 MHz, การเชื่อมต่อ USB ขนาดเล็ก, แจ็คไฟหัว ICSP และปุ่มรีเซ็ต Arduino Leonardo แตกต่างจากบอร์ดทั้งหมดเนื่องจาก ATMEGA32U4 มีตัวกลางในการสื่อสารกับ USB จึงไม่จำเป็นต้องใช้หน่วยประมวลผลรอง ซึ่ง Arduino Leonardo มีคุณสมบัติตามตารางที่ 2.1



รูปที่ 2.1 โครงสร้างอาร์ดูโนลีโอนาร์โด

a(ที่มา: http://o.lnwfile.com/_/o/_raw/q1/lu/nn.png)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 คุณสมบัติไมโครคอนโทรลเลอร์อาร์ดูโนลีโอนาร์โด

คุณสมบัติ	Arduino Leonardo
Microcontroller	ATmega32u4
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	20
PWM Channels	7
Analog Input Channels	12
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega32u4) of which 4 KB used by bootloader
SRAM	2.5 KB (ATmega32u4)
EEPROM	1 KB (ATmega32u4)
Clock Speed	16 MHz

2.1.1 ความจำของอาร์ดูโนลีโอนาร์โด

ATMEGA32U4 มี 32 กิโลไบต์ (มี 4 กิโลไบต์ที่ใช้สำหรับ bootloader) นอกจากนี้ยังมี 2.5 กิโลไบต์ของ SRAM และ 1 กิโลไบต์ของ EEPROM

2.1.2 อินพุตและเอาต์พุต

ขา 20 ดิจิตอล I / O ของเลโอนาร์โดที่สามารถใช้เป็นอินพุตหรือเอาต์พุตโดยใช้ฟังก์ชัน `pinMode ()` `digitalWrite ()` และ `digitalRead ()` สามารถทำงานที่ 5 โวลต์ แต่ละขาสามารถให้หรือรับได้ไม่เกิน 40 mA และมีความต้านทานภายใน (ตัดการเชื่อมต่อโดยค่าเริ่มต้น) 20-50 kOhms นอกจากนี้ขาบางขามีฟังก์ชันพิเศษ เช่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Serial: 0 (RX) and 1 (TX) ใช้ในการได้รับ (RX) และส่ง (TX) TTL ข้อมูลแบบอนุกรมโดยใช้ฮาร์ดแวร์ ATMEGA32U4

- TWI: 2 (SDA) และ 3 (SCL) สนับสนุนการสื่อสาร TWI

- External Interrupts: 3 (interrupt 0), 2 (interrupt 1), 0 (interrupt 2), 1 (interrupt 3) and 7 (interrupt 4) พินเหล่านี้สามารถกำหนดค่าที่จะเรียกขัดจังหวะได้

- PWM: 3, 5, 6, 9, 10, 11, และ 13 ให้ออก PWM 8 บิตที่ ฟังก์ชัน analogWrite ()

- SPI: on the ICSP header ขาเหล่านี้สนับสนุนการสื่อสาร SPI ใช้ห้องสมุด SPI โปรดทราบว่าขา SPI ไม่ได้เชื่อมต่อใด ๆ ของขาดิจิตอล I / O ที่พวกเขาอยู่ใน Uno, พวกเขาที่อยู่เฉพาะในการเชื่อมต่อ ICSP ซึ่งหมายความว่าหากคุณมีโลที่ใช้ SPI แต่ไม่ได้มีการเชื่อมต่อ ICSP 6 ขาที่เชื่อมต่อกับเลโอนาร์โด 6 ขาส่วนหัว ICSP โลจะไม่ทำงาน

- LED: 13 มีไฟในตัว ซึ่ง LED เชื่อมต่อกับขาดิจิตอลพิน 13 เมื่อขาสูงค่าไฟ LED ติดเมื่อขาอยู่ในระดับต่ำก็ออก

- Analog Inputs: A0-A5, A6 - A11 (on digital pins 4, 6, 8, 9, 10, and 12) Leonardo มี 12 อนุาล็อกป้าย A0 ถึง A11 ซึ่งทั้งหมดนอกจากนี้ยังสามารถนำมาใช้เป็น Digital I / O ขา A0-A5 คล้ายคลึงกับ Uno

- AREF แรงดันอ้างอิงสำหรับอนุาล็อก ใช้กับ analogReference ()

- Reset เพื่อตั้งค่าไมโครคอนโทรลเลอร์

2.1.3 การติดต่อสื่อสาร

เลโอนาร์โดมีการอำนวยความสะดวกสำหรับการสื่อสารกับคอมพิวเตอร์หรืออาร์ดูโนตัวอื่น ๆ หรือตัวไมโครคอนโทรลเลอร์อื่น ๆ ซึ่ง ATMEGA32U4 จัดให้ UART TTL (5V) เป็นการสื่อสารแบบอนุกรมซึ่งมีอยู่ในขาดิจิตอล 0 (RX) และ 1 (TX) Mega32U4 ยังช่วยให้การติดต่อสื่อสารแบบอนุกรม (CDC) ผ่าน USB และปรากฏเป็นพอร์ต COM เหมือนกับซอฟต์แวร์บนเครื่องคอมพิวเตอร์ ชิปีมีความเร็วสูงสุดเหมือนกับ USB 2.0 โดยใช้ไดรเวอร์มาตรฐาน USB COM สำหรับการใช้งานกับ Windows เนื่องจากไฟล์ .inf จำเป็นต้องมีซอฟต์แวร์อาร์ดูโนและพอร์ตการตรวจสอบอนุกรมซึ่งช่วยให้สามารถส่งข้อมูลติดต่อสื่อสารกับบอร์ดอาร์ดูโนได้และส่งไปยังและออกจากบอร์ดอาร์ดูโน ไฟ LED RX และ TX บนกระดานจะกะพริบเมื่อข้อมูลที่จะถูกส่งผ่านการเชื่อมต่อ USB เข้ากับเครื่องคอมพิวเตอร์

ลีโอนาร์โดสามารถจำลองตนเองเสมือนเป็นแป้นพิมพ์และเมาส์ทั่วไปโดยสามารถตั้งโปรแกรมให้ควบคุมอุปกรณ์การป้อนข้อมูลโดยใช้แป้นพิมพ์และเมาส์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาด้านนี้ ไม่นอนุญาติให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.4 การเขียนโปรแกรม

ลีโอนาร์โดสามารถโปรแกรมด้วยซอฟต์แวร์อาร์ดูโน ATMEGA32U4 ในอาร์ดูโนมี bootloader ที่ช่วยให้สามารถอัปโหลดโปรแกรมใหม่ได้โดยไม่ต้องใช้โปรแกรมฮาร์ดแวร์ภายนอก โดย ATMEGA32U4 จะติดต่อสื่อสารโดยใช้โปรโตคอล AVR109



รูปที่ 2.2 อาร์ดูโนลีโอนาร์โด

(ที่มา: <http://digipak.org/zencart/images/ArduinoLeonardo.jpg>)

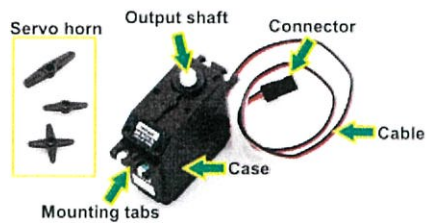
2.2 เซอร์โวมอเตอร์

เซอร์โวมอเตอร์ มีความหมายคือของที่เราสามารถสั่งงานหรือตั้งค่าแล้วตัวมอเตอร์จะหมุนไปยังตำแหน่งองศาที่เราสั่งได้เองอย่างถูกต้อง โดยใช้การควบคุมแบบป้อนกลับ(Feedback Control)

พีดแบคคอนโทรล คือ ระบบควบคุมที่มีการวัดค่าเอาต์พุตของระบบนำมาเปรียบเทียบกับค่าอินพุตเพื่อควบคุมและปรับแต่งให้ค่าเอาต์พุตของระบบให้มีค่าเท่ากับหรือใกล้เคียงกับค่าอินพุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1 ส่วนประกอบภายนอกเซอร์โวมอเตอร์



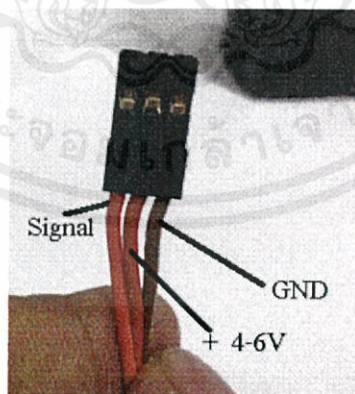
รูปที่ 2.3 ส่วนประกอบภายนอกเซอร์โวมอเตอร์

(ที่มา: <http://goo.gl/pwiXoC>)

เคสตัวถัง หรือ กรอบของตัวเซอร์โวมอเตอร์ ประกอบด้วย

- Mounting Tab ส่วนจับยึดตัว Servo กับชิ้นงาน
- Output Shaft เพลาส่งกำลัง
- Servo Horns ส่วนเชื่อมต่อกับ Output shaft เพื่อสร้างกลไก
- Cable สายเชื่อมต่อเพื่อ จ่ายไฟฟ้า และ ควบคุม Servo Motor จะประกอบด้วยสายไฟ 3 เส้น และ ใน RC Servo Motor จะมีสีของสายแตกต่างกันไปดังนี้
- สายสีแดง คือ ไฟเลี้ยง (4.8-6V)
- สายสีดำ หรือ น้ำตาล คือ กราวด์
- สายสีเหลือง (ส้ม ขาว หรือฟ้า) คือ สายส่งสัญญาณพัลส์ควบคุม (3-5V)

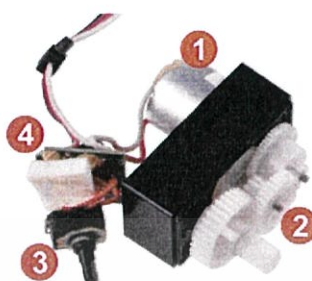
Connector จุดเชื่อมต่อสายไฟ



รูปที่ 2.4 สายไฟเซอร์โวมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.2 ส่วนประกอบภายในเซอร์โวมอเตอร์



รูปที่ 2.5 ส่วนประกอบภายในเซอร์โวมอเตอร์

(ที่มา: www.pololu.com)

หมายเลข 1 มอเตอร์เป็นส่วนของตัวมอเตอร์

หมายเลข 2 Gear Train หรือ Gearbox เป็นชุดเกียร์ทดแรง

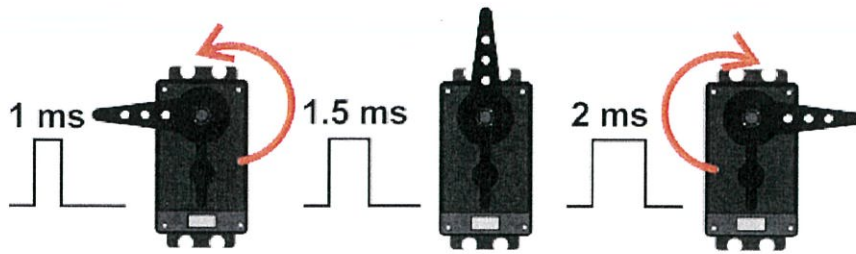
หมายเลข 3 Position Sensor เป็นเซ็นเซอร์ตรวจจับตำแหน่งเพื่อหาค่าองศาในการหมุน

หมายเลข 4 Electronic Control System เป็นส่วนที่ควบคุมและประมวลผล

2.2.3 หลักการทำงานของเซอร์โวมอเตอร์

เมื่อจ่ายสัญญาณพัลส์เข้ามายัง RC Servo Motor ส่วนวงจรควบคุม (Electronic Control System) ภายใน Servo จะทำการอ่านและประมวลผลค่าความกว้างของสัญญาณพัลส์ที่ส่งเข้ามา เพื่อแปลค่าเป็นตำแหน่งองศาที่ต้องการให้ Motor หมุนเคลื่อนที่ไปยังตำแหน่งนั้น แล้วส่งคำสั่งไปทำการควบคุมให้ Motor หมุนไปยังตำแหน่งที่ต้องการ โดยมี Position Sensor เป็นตัวเซ็นเซอร์คอยวัดค่ามุมที่ Motor กำลังหมุน เป็น Feedback กลับมาให้วงจรควบคุมเปรียบเทียบกับค่าอินพุตเพื่อควบคุมให้ได้ตำแหน่งที่ต้องการอย่างถูกต้องแม่นยำ มุมหรือองศาจะขึ้นอยู่กับความกว้างของสัญญาณพัลส์ ซึ่งโดยส่วนมากความกว้างของพัลส์ที่ใช้ใน RC Servo Motor จะอยู่ในช่วง 1-2 ms หรือ 0.5-2.5 ms ยกตัวอย่างเช่นหากกำหนดความกว้างของสัญญาณพัลส์ไว้ที่ 1 ms ตัว Servo Motor จะหมุนไปทางซ้ายสุด ในทางกลับกันหากกำหนดความกว้างของสัญญาณพัลส์ไว้ที่ 2 ms ตัว Servo Motor จะหมุนไปยังตำแหน่งขวาสุด แต่หากกำหนดความกว้างของสัญญาณพัลส์ไว้ที่ 1.5 ms ตัว Servo Motor ก็ จะหมุนมาอยู่ที่ตำแหน่งตรงกลางพอดี

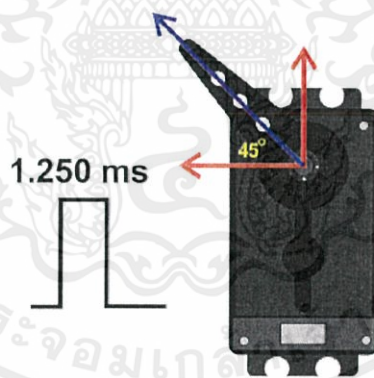
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาติให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 หลักการทำงานของเซอร์โวมอเตอร์

(ที่มา: <http://goo.gl/exCyd5>)

ดังนั้นสามารถกำหนดองศาการหมุนของ RC Servo Motor ได้โดยการเทียบค่า เช่น RC Servo Motor สามารถหมุนได้ 180 องศา โดยที่ 0 องศาใช้ความกว้างพัลส์เท่ากับ 1000 us ที่ 180 องศาความกว้างพัลส์เท่ากับ 2000 us เพราะฉะนั้นค่าที่เปลี่ยนไป 1 องศาจะใช้ความกว้างพัลส์ต่างกัน $(2000-1000)/180$ เท่ากับ 5.55 us จากการหาค่าความกว้างพัลส์ที่มุม 1 องศาข้างต้น หากต้องการให้ RC Servo Motor หมุนไปที่มุม 45 องศาจะหาค่าพัลส์ที่ต้องการได้จาก 5.55×45 เท่ากับ 249.75 us แต่ที่มุม 0 องศาเราเริ่มที่ความกว้างพัลส์ 1ms หรือ 1000 us เพราะฉะนั้นความกว้างพัลส์ที่ใช้กำหนดให้ RC Servo Motor หมุนไปที่ 45 องศา คือ $1000 + 249.75$ เท่ากับ ประมาณ 1250 us



รูปที่ 2.7 การหมุนของเซอร์โวมอเตอร์

(ที่มา: <http://goo.gl/cXhHI9>)

2.2.4 วิธีการควบคุมเซอร์โวมอเตอร์ด้วยอาร์ดูโน

อาร์ดูโนมีไลบรารีสำหรับสั่งงานเซอร์โวมอเตอร์มาให้ใช้งานอยู่แล้วเป็นฟังก์ชันสำเร็จรูปและใช้งานได้ง่าย เซอร์โวลิบรารีของอาร์ดูโนสามารถสั่งงานเซอร์โวได้ทั้งแบบหมุนไปกลับได้ 0-180 องศา และแบบต่อเนื่องที่หมุนครบรอบได้เรียกว่าเป็น Continuous Rotation Servo ที่มีการนำไปใช้

ฟังก์ชันภายใน Servo Library

- attach() คือ ฟังก์ชันที่ใช้ในการกำหนดขาสัญญาณที่ Servo Motor ต่อกับ Arduino และกำหนดความกว้างของพัลส์ที่ 0 องศาและ 180 องศา
- write() คือ ฟังก์ชันที่ใช้ควบคุมตำแหน่งที่ต้องการให้ Servo Motor หมุนไปยังองศาที่กำหนดสามารถกำหนดเป็นค่าองศาได้เลย คือ 0-180 องศา แต่ใน Servo Motor ที่เป็น Full Rotation คำสั่ง write จะเป็นการกำหนดความเร็วในการหมุน โดย ค่าเท่ากับ 90 คือคำสั่งให้ Servo Motor หยุดหมุน ค่าเท่ากับ 0 คือการหมุนด้วยความเร็วสูงสุดในทิศทางหนึ่ง ค่าเท่ากับ 180 คือการหมุนด้วยความเร็วสูงสุดในทิศทางตรงกันข้าม
- writeMicroseconds() คือ คือฟังก์ชันที่ใช้ควบคุมตำแหน่งที่ให้ Servo Motor หมุนไปยังตำแหน่งองศาที่กำหนดโดยกำหนดเป็นค่าความกว้างของพัลส์ในหน่วย us ซึ่งปกติแล้ว RC Servo Motor จะใช้ความกว้างของพัลส์อยู่ที่ 1000-2000 us ตามที่ได้กล่าวไปข้างต้นแล้ว แต่ RC Servo Motor บางรุ่นหรือบางยี่ห้อไม่ได้ใช้ ช่วงความกว้างของพัลส์ตามที่ได้กล่าวเอาไว้ อาจจะใช้ช่วง 700-2300 แทนก็สามารถใช้ฟังก์ชัน writeMicroseconds นี้เพื่อกำหนดความกว้างพัลส์ได้เอง

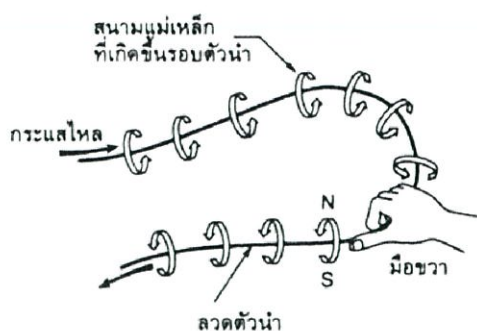
การใช้ฟังก์ชัน writeMicroseconds สามารถกำหนดค่าได้อิสระ ต้องระวังในการใช้งานหากสั่งงานเซอร์โวมอเตอร์แบบ 0 - 180 องศา จนหมุนไปเกินจุดสิ้นสุดคือเกินทั้งฝั่ง 0 หรือ 180 องศา จะทำให้เกิดเสียงครางดังจากการหมุนไปต่อไม่ได้และมอเตอร์จะกินกระแสสูงขึ้นด้วยในเวลาเดียวกันนั้น ซึ่งอาจทำให้เซอร์โวมอเตอร์นั้นเกิดความเสียหายได้

- read() คือ คือฟังก์ชันอ่านค่าองศาที่ส่งเข้าไปด้วยฟังก์ชัน write() เพื่อให้รู้ว่าตำแหน่งองศาสุดท้ายที่เราสั่งเข้าไปนั้นมีค่าเท่าไรซึ่งค่าที่อ่านออกมานั้นจะมีค่าอยู่ในช่วง 0 - 180
- attached() คือ คือฟังก์ชันตรวจสอบว่าเซอร์โวที่เราต้องการใช้กำลังต่ออยู่กับสัญญาณของอาร์ดูโนอยู่หรือไม่
- detach() คือ คือฟังก์ชันคืนสถานะของขาที่เรากำหนดให้เป็นขาควบคุมเซอร์โวมอเตอร์ด้วยคำสั่ง attached() ให้กลับคือสู่การใช้งานปกติ

2.3 โขลินอยด์

2.3.1 หลักการทำงานของโขลินอยด์

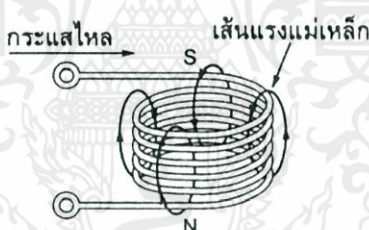
เมื่อมีกระแสไฟฟ้าไหลในขดลวดตัวนำใดๆก็ตามจะเกิดสนามแม่เหล็กขึ้นรอบๆตัวนำนั้นและตามหลักการของกฎมือขวาเมื่อดูทิศทางเส้นแรงแม่เหล็กด้วย ถ้าหากนำมือขวากำรอบเส้นลวด โดยนิ้วหัวแม่มือแทนทิศทางกระแสไหล นิ้วที่เหลือทั้งหมดจะแสดงทิศทางเส้นแรง แม่เหล็กจากขั้วใต้ไปขั้วเหนือ



รูปที่ 2.8 กฎของมือขวา

(ที่มา: <http://goo.gl/U99HDD>)

เมื่อเรานำเส้นลวดแบบดัดกล้วมาขดเป็นวงหลายวง จะเกิดลักษณะของขดลวดขึ้น (ดังภาพที่ 3) สนามแม่เหล็กที่เกิดจากขดลวดแต่ละขดจะอยู่ในทิศทางที่เสริมกันและก่อกำเนิดเป็นเส้นแรงของสนามแม่เหล็กถาวรแท่งหนึ่งซึ่งพร้อมที่จะดูดสารแม่เหล็กทันทีแต่เนื่องจากสภาพรอบๆ ขดลวดอาจเป็นอากาศจึงส่งผลให้เส้นแรงแม่เหล็กบริเวณดังกล่าวจึงมีความเข้มข้นไม่มากนัก

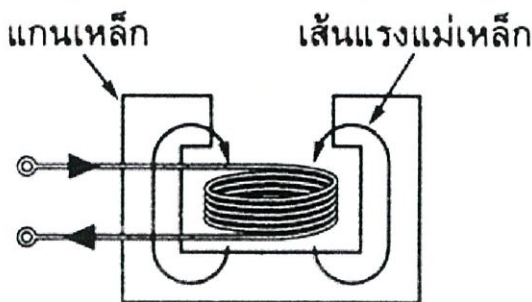


รูปที่ 2.9 กระแสไหลของแม่เหล็กและเส้นแรงแม่เหล็ก

(ที่มา: <http://goo.gl/qOgoKK>)

เพื่อที่จะไม่ให้สนามแม่เหล็กที่เกิดขึ้นกระจัดกระจายจึงใส่แกนเหล็กอ่อนรูปตัว C เข้ามารอบๆขดลวดเพื่อให้สนามแม่เหล็กมากขึ้น(ดังรูปที่ 4) ถ้าเอาแกนกระทงเข้ามาใส่ในตำแหน่งตรงกลางขดลวดในตำแหน่งที่ 1 แกนกระทงจะถูกดูดให้ลึกลงเข้ามาจนสนิทในตำแหน่งที่ 2 ยิ่งระยะทางไกลมากเท่าไร แรงดูดก็จะมากขึ้นเท่านั้น

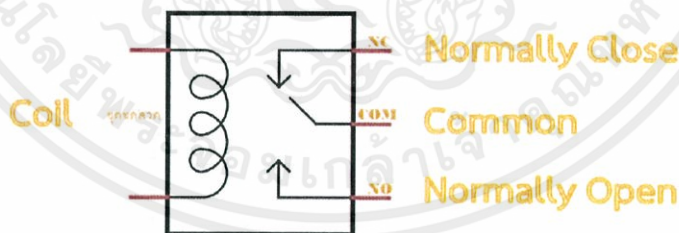
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.10 แกนเหล็กและเส้นแรงแม่เหล็ก
(ที่มา: <http://goo.gl/2Gchxj>)

2.4 รีเลย์ (Relay)

รีเลย์เป็นอุปกรณ์ไฟฟ้าชนิดหนึ่ง ซึ่งทำหน้าที่ตัดต่อวงจรแบบเดียวกับสวิตช์ โดยควบคุมการทำงานด้วยไฟฟ้า รีเลย์มีหลายประเภท ตั้งแต่รีเลย์ขนาดเล็กที่ใช้ในงานอิเล็กทรอนิกส์ทั่วไป จนถึงรีเลย์ขนาดใหญ่ที่ใช้ในงานไฟฟ้าแรงสูง โดยมีรูปร่างหน้าตาแตกต่างกันออกไป แต่มีหลักการทำงานที่คล้ายคลึงกัน สำหรับการนำรีเลย์ไปใช้งาน จะใช้ในการตัดต่อวงจร ทั้งนี้รีเลย์ยังสามารถเลือกใช้งานได้หลากหลายรูปแบบ

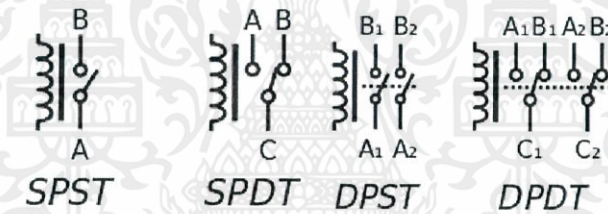


รูปที่ 2.11 สัญญาณวงจรไฟฟ้าในรีเลย์
(ที่มา: <http://goo.gl/SMcJW>)

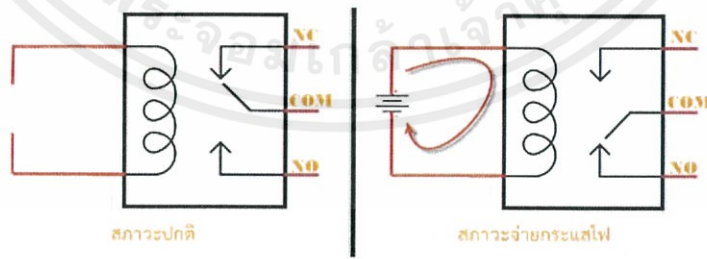
เอกสารนี้เป็นเอกสารลิขสิทธิ์ของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น หน้าสัมผัส NC (Normally Close) เป็นหน้าสัมผัสสปกติปิด โดยในสภาวะปกติหน้าสัมผัสนี้ จะต่อเข้ากับขา COM (Common) และจะลดยหรือไม่สัมผัสกันเมื่อมีกระแสไฟฟ้าไหลผ่านขดลวด

- หน้าสัมผัส NO (Normally Open) เป็นหน้าสัมผัสปกติเปิด โดยในสภาวะปกติจะลอยอยู่ไม่ถูกต้องกับขา COM (Common) แต่จะเชื่อมต่อกันเมื่อมีกระแสไฟไหลผ่านขดลวด
- ขา COM (Common) เป็นขาที่ถูกใช้งานร่วมกันระหว่าง NC และ NO ขึ้นอยู่กับว่าขณะนั้นมีกระแสไฟฟ้าไหลผ่านขดลวดหรือไม่ หน้าสัมผัสใน Relay 1 ตัวอาจมีมากกว่า 1 ชุด ขึ้นอยู่กับผู้ผลิตและลักษณะของงานที่ถูกนำไปใช้ จำนวนหน้าสัมผัสถูกแบ่งออกดังนี้

สวิตช์จะถูกแยกประเภทตามจำนวน Pole และจำนวน Throw ซึ่งจำนวน Pole (SP-Single Pole, DP-Double Pole, 3P-Triple Pole, etc.) จะบอกถึงจำนวนวงจรที่ทำการเปิด-ปิด หรือ จำนวนของขา COM นั้นเอง และจำนวน Throw (ST, DT) จะบอกถึงจำนวนของตัวเลือกของ Pole ตัวอย่างเช่น SPST- Single Pole Single Throw สวิตช์จะสามารถเลือกได้เพียงอย่างเดียวโดยจะเป็นปกติเปิด (NO-Normally Open) หรือปกติปิด (NC-Normally Close) แต่ถ้าเป็น SPDT- Single Pole Double Throw สวิตช์จะมีหนึ่งคู่เป็นปกติเปิด (NO) และอีกหนึ่งคู่เป็นปกติปิดเสมอ (NC) ดังรูปด้านล่าง



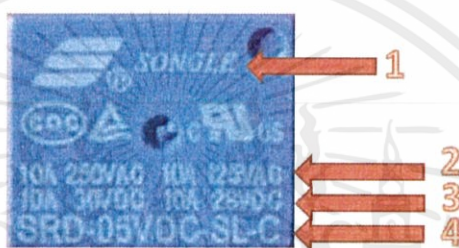
รูปที่ 2.12 สวิตช์ของรีเลย์
(ที่มา: <http://goo.gl/SMcCJW>)



รูปที่ 2.13 สภาวะของรีเลย์
(ที่มา: <http://goo.gl/SMcCJW>)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการทดลองเครื่องให้อาหารสัตว์อัตโนมัติเราจะใช้งานรีเลย์แบบ SPDT (Single Pole Double Throw) หลักการทำงานของรีเลย์นั้น ในส่วนของขดลวดเมื่อมีกระแสไฟฟ้าไหลผ่านจะทำให้ขดลวดเกิดการเหนี่ยวนำและทำหน้าที่เสมือนแม่เหล็กไฟฟ้า ส่งผลให้ขา COM ที่เชื่อมต่ออยู่กับหน้าสัมผัส NC (ในสถานะที่ยังไม่เกิดการเหนี่ยวนำ) ย้ายกลับเชื่อมต่อกับหน้าสัมผัส NO แทน และปล่อยให้ขา NC ลอย เมื่อมองที่ขา NC กับ COM และ NO กับ COM แล้วจะเห็นว่ามีการทำงานติด-ดับลักษณะคล้ายการทำงานของสวิทช์ เรานำคุณสมบัตินี้ไปประยุกต์ใช้งานในการซักเปิดปิดโซลินอยด์ในการปล่อยให้อาหารสัตว์



รูปที่ 2.14 ข้อมูลบนตัวรีเลย์
(ที่มา: <http://goo.gl/SMcCJW>)

- หมายเลข 1 ยี่ห้อ รุ่นของผู้ผลิต (แบรนด์) รวมถึงสัญลักษณ์มาตรฐานต่างๆ
 หมายเลข 2 รายละเอียดของไฟฟ้ากระแสสลับที่รองรับการทำงานได้ (VAC)
 หมายเลข 3 รายละเอียดของไฟฟ้ากระแสตรงที่รองรับการทำงานได้ (VDC)
 หมายเลข 4 โมเดล ระดับแรงดันฝั่งขดลวด ชนิดและโครงสร้าง และข้อมูลด้าน Coil Sensitivity

2.4.1 คุณสมบัติของรีเลย์ 4 ช่องสัญญาณ (รีเลย์เอาต์พุตแบบ SPDT จำนวน 4 ช่อง)

2.4.1.1 สั่งงานด้วยระดับแรงดัน TTL

2.4.1.2 CONTACT OUTPUT ของรีเลย์รับแรงดันได้สูงสุด 250 VAC 10 A, 30

VDC 10 A

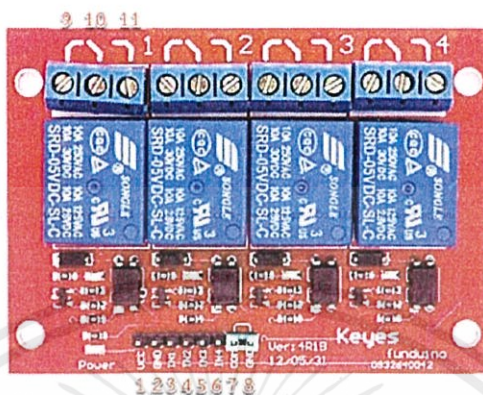
2.4.1.3 มี LED แสดงสถานะ การทำงานของรีเลย์และแสดงสถานะของบอร์ด

2.4.1.4 มีจัมป์เปอร์สำหรับเลือกว่าจะใช้กราวด์ร่วมหรือแยก

2.4.1.5 มี OPTO-ISOLATED เพื่อแยกกราวด์ส่วนของสัญญาณควบคุมกับไฟฟ้าที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
 ขั้วรีเลย์ออกจากกัน

2.4.2 ขาสัญญาณ (Pin Definition)



รูปที่ 2.15 ขาที่ใช้ในการเชื่อมต่อของรีเลย์ 4 ช่องสัญญาณ
(ที่มา: <http://goo.gl/SMcJW>)

ตารางที่ 2.2 แสดงขาที่ใช้ในการเชื่อมต่อของรีเลย์ 4 ช่องสัญญาณ

ขาที่	คำอธิบาย
1	+VCC ขาไฟ 5VDC
2	GND
3	ขาสัญญาณอินพุตรีเลย์1 (IN1)
4	ขาสัญญาณอินพุตรีเลย์2 (IN2)
5	ขาสัญญาณอินพุตรีเลย์3 (IN3)
6	ขาสัญญาณอินพุต4 (IN4)
7	COM (คอมมอนของ OPTO)
8	GND (กราวด์ของบอร์ดเป็นกราวด์เดียวกันกับขาที่ 2)
9	NC (Normal Close) หน้าสัมผัสแบบปกติปิด
10	COM (Common) ที่จะตัดหรือต่อวงจรจากขา NC, NO
11	NO (Normal Open) หน้าสัมผัสแบบปกติเปิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 กล้องเว็บแคม (Web Camera)

2.5.1 ความหมายของกล้องเว็บแคม

เว็บแคมเป็นอุปกรณ์พื้ที่สามารถจับภาพเคลื่อนไหวของสิ่งที่ตัวกล้องจับภาพอยู่ไปปรากฏในหน้าจอคอมพิวเตอร์ และสามารถส่งภาพเคลื่อนไหวนี้ผ่านระบบเครือข่ายเพื่อให้คอมพิวเตอร์อีกจุดๆ หนึ่งสามารถเห็นภาพเคลื่อนไหวได้เหมือนอยู่ต่อหน้า ถือว่าเป็นอุปกรณ์ที่มีประโยชน์อีกตัวหนึ่ง

2.5.2 ชนิดของกล้องเว็บแคม

กล้องเว็บแคม (Webcam) แบ่งออกได้ 2 ชนิด คือ แบบมีสาย และแบบไร้สาย โดยแต่ละชนิดมีความแตกต่างกันดังนี้

กล้องเว็บแคมแบบมีสาย จะมีความยุ่งยากในเรื่องการใช้สายต่อพ่วงเข้ากับเครื่องคอมพิวเตอร์ แต่จะมีราคาถูกกว่าแบบไร้สายมาก ทำให้คนส่วนใหญ่นิยมซื้อกล้องเว็บแคมแบบมีสายมาใช้งาน ข้อเสียของกล้องเว็บแคมแบบมีสาย คือ ทำให้ไม่สามารถวางตัวกล้องได้ไกลจากเครื่องคอมพิวเตอร์ ทำให้กล้องไม่สามารถจับภาพเคลื่อนไหวในระยะไกล ๆ ได้เหมือนแบบไร้สาย



รูปที่ 2.16 ตัวอย่างกล้องเว็บแคมแบบมีสาย

(ที่มา: <http://goo.gl/VrNOOo>)

กล้องเว็บแคม (Webcam) แบบไร้สาย จะมีราคาค่อนข้างแพงมากเมื่อเทียบกับแบบมีสาย เนื่องจากตัวกล้อง ต้องใช้เทคโนโลยีแบบไร้สายที่เรียกว่า Wireless WiFi หรือ IEEE 802.11 ที่ค่อนข้างมีต้นทุนสูง จึงส่งผลให้ตัวกล้องมีราคาแพงจึงไม่ค่อยได้รับความนิยมนัก จุดเด่นของกล้องเว็บแคมแบบไร้สาย คือ สามารถนำไปติดตั้งที่จุดใดก็ได้ โดยไม่ต้องคำนึงระยะห่างระหว่างตัวกล้องกับเอกสารนี้คอมพิวเตอร์ ส่วนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.17 ตัวอย่างกล้องเว็บแคมแบบไร้สาย
(ที่มา: <http://goo.gl/iHv0V1>)

2.6 ภาษาซี

ภาษาซี (C Programming Language) คือ ภาษาคอมพิวเตอร์ใช้สำหรับพัฒนาโปรแกรมทั่วไป ถูกพัฒนาครั้งแรกเพื่อใช้เป็นภาษาสำหรับพัฒนาระบบปฏิบัติการยูนิกซ์ (Unix Operating System) แทนภาษาแอสเซมบลี ซึ่งเป็นภาษาระดับต่ำที่สามารถกระทำในระบบฮาร์ดแวร์ได้ด้วยความรวดเร็ว แต่จุดอ่อนของภาษาแอสเซมบลีก็คือความยุ่งยากในการโปรแกรม ความเป็นเฉพาะตัว และความแตกต่างกันไปในแต่ละเครื่อง

ภาษาซีจัดเป็นภาษาระดับกลางที่มีลักษณะเป็นภาษาโครงสร้าง สามารถประยุกต์ใช้ได้กับงานในลักษณะต่างๆ เป็นภาษาที่ใกล้เคียงกับภาษาแอสเซมบลี ผู้เขียนโปรแกรมสามารถเขียนโปรแกรมได้อย่างคล่องตัวโดยไม่มีข้อจำกัดในการวางตำแหน่งฟังก์ชันในโปรแกรม ภาษาซีจึงเป็นภาษาที่ง่ายต่อการเข้าใจและการนำไปใช้งาน การสร้างโปรแกรมภาษาซีจะเริ่มจากการเขียนโปรแกรมต้นกำเนิด แล้วนำไปทำการแปลด้วยตัวแปลภาษาซีเกิดเป็นโปรแกรมประสมค์ หลังจากนั้นจึงนำโปรแกรมประสมค์ไปทำการเชื่อมโยงเพื่อให้เกิดเป็นโปรแกรมทำการที่สามารถทำงานได้อย่างรวดเร็ว

2.6.1 โครงสร้างของโปรแกรมภาษาซี

โปรแกรมภาษาซีมีองค์ประกอบที่สำคัญ 2 ส่วนคือไฟล์ส่วนหัวโปรแกรม และไฟล์โปรแกรม ไฟล์ส่วนหัวโปรแกรมเป็นไฟล์ที่ใช้เก็บไลบรารีเพื่อใช้รวม (include) ในการคอมไพล์โปรแกรมซึ่งจะมีส่วนขยายเป็น *.h มีชื่อเรียกว่า Compiler Directive ไฟล์โปรแกรมจะเริ่มต้นด้วยฟังก์ชัน main() และตามด้วยเครื่องหมายปีกกาเปิด เพื่อเริ่มต้นเขียนโปรแกรม การเขียนโปรแกรมจะต้องเขียนด้วยอักษรภาษาอังกฤษตัวพิมพ์เล็กเสมอ และเมื่อจบประโยคคำสั่ง จะใช้เครื่องหมายเซมิโคลอน (;) ในการปิดคำสั่งแต่ละคำสั่ง ภายในโปรแกรมจะประกอบด้วยฟังก์ชันและส่วนของคำอธิบาย เมื่อเขียนคำสั่งไปใช้เสร็จจะปิดท้ายโปรแกรมด้วยเครื่องหมายปีกกาปิดเสมอ

2.6.1.1 ตัวอย่างโครงสร้างโปรแกรมภาษาซี

```

#include<library>    /* ไฟล์ส่วนหัวโปรแกรม*/

void main(void)     /*ฟังก์ชันหลักของโปรแกรม*/
{
    /* เริ่มต้นการเขียนโปรแกรมด้วยเครื่องหมายปีกกาเปิด*/

    variable declaration; /*การประกาศค่าตัวแปรที่ใช้ในโปรแกรม*/

    program statement; /*ประโยคคำสั่งในโปรแกรม*/

}                   /*จบการเขียนโปรแกรมด้วยเครื่องหมายปีกกาปิด*/

```

ตารางที่ 2.3 คำอธิบายโครงสร้างของโปรแกรมภาษาซี

#include<library>	เป็นส่วนหัวโปรแกรมที่จะต้องเขียนไว้เพื่อให้ใช้งานฟังก์ชันต่างๆ ในกรณีที่ต้องการทราบว่าฟังก์ชันใดถูกนิยามไว้ที่ใดให้ทำแถบสีที่ฟังก์ชันดังกล่าวและกดปุ่ม Ctrl+f1
main	เป็นฟังก์ชันหลักของโปรแกรม
()	ภายในวงเล็บเป็นค่าพารามิเตอร์ที่จะส่งผ่านไปทำงานยังฟังก์ชันอื่นๆ ถ้าไม่มีการใส่ค่าแสดงว่าไม่ต้องการมีค่าพารามิเตอร์
{	ปีกกาเปิดแสดงการเริ่มต้นการเขียนโปรแกรม
variable declarations	ประกาศตัวแปร
program statement	การเขียนประโยคคำสั่ง
}	ปีกกาปิดแสดงการจบการเขียนโปรแกรม
/*ข้อความ*/	คำอธิบายโปรแกรม ใช้ในการอธิบายประโยชน์การใช้งาน คำอธิบายของคำสั่งหรือสิ่งที่ต้องการเขียนไปใช้

	ไว้กันลืมจะไม่มีผลใดๆกับโปรแกรม แต่การเขียนจะต้องเริ่มต้นด้วยเครื่องหมาย /* และจบด้วยเครื่องหมาย*/
--	--

2.6.2 ตัวแปรในโปรแกรมภาษาซี (Variable)

ตัวแปร (variable) คือชื่อที่ผู้เขียนโปรแกรมตั้งขึ้น เพื่อใช้เก็บค่าที่ต้องการนำมาใช้งานในการเขียนโปรแกรม เพื่อทำการประมวลผลข้อมูล โดยมีกฎในการตั้งชื่อตัวแปรดังนี้

2.6.2.1 ต้องเขียนขึ้นต้นด้วยตัวอักษรซึ่งตัวต่อไปอาจเป็นตัวอักษรหรือตัวเลขก็ได้

2.6.2.2 ห้ามใช้สัญลักษณ์อื่นใด ยกเว้นเครื่องหมายสตริงก์ (String) และเครื่องหมายขีดล่าง (Underscore)

2.6.2.3 ตัวแปรอักษรภาษาอังกฤษตัวพิมพ์เล็กและตัวพิมพ์ใหญ่มีความหมายต่างกัน

2.6.2.4 ห้ามเว้นวรรคระหว่างตัวแปร

2.6.2.5 ห้ามตั้งชื่อซ้ำกับคำสงวนในภาษาซี

ก่อนที่จะนำตัวแปรภาษาซีไปใช้งานจะต้องมีการประกาศค่าตัวแปรให้สอดคล้องกับข้อมูลที่จะนำไปใช้โดยมีรูปแบบดังนี้

ตารางที่ 2.4 รูปแบบการประกาศค่าตัวแปร

รูปแบบ	คำอธิบาย
Type	ชนิดของตัวแปร ซึ่งอาจจะป็น char, int, float, double หรือตัวแปรชนิดอื่นๆ เป็นต้น
variable name	ชื่อของตัวแปร ถ้ามีมากกว่า 1 ตัวให้ใช้เครื่องหมายคอมม่าคั่น

ตัวอย่างการประกาศค่าตัวแปร

```
char n; //ประกาศค่าตัวแปรชื่อ n เป็นข้อมูลชนิด character
```

```
float a,b,c; //ประกาศค่าตัวแปรชื่อ a,b,c เป็นข้อมูลชนิด float
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งยังมีให้เปลี่ยนแปลงเนื้อหา และห้องเรียนจึงขอสงวนสิทธิ์ในการนำไปใช้

```
int number=1; //ประกาศค่าตัวแปรชื่อ number เป็นข้อมูลชนิด
integer และ กำหนดให้ตัวแปร count มีค่าเท่ากับ 1
char name[15]; //ประกาศตัวแปรชื่อ name เป็นลักษณะตัวแปรชุดเก็บ
ชื่อยาวไม่ เกิน 15 ตัวอักษร
```

2.6.3 คำสงวนในโปรแกรมภาษาซี (Reserved Words)

ตารางที่ 2.5 แสดงคำสงวนของโปรแกรมภาษาซี

Auto	Break	Case	Char
Const	Default	Do	Double
Else	Enum	Short	Signed
Sizeof	Extern	Float	For
Goto	If	Int	Long
Return	Register	Continue	While
Static	Struct	Switch	Typeof
Unon	Unsigned	Void	Volatile

2.6.4 ฟังก์ชันของโปรแกรมภาษาซี (Function)

ฟังก์ชัน clrscr(); เป็นฟังก์ชันที่ใช้ในการลบจอภาพ

ฟังก์ชัน printf(); เป็นฟังก์ชันที่ใช้ในการแสดงผลข้อมูลที่อยู่ในตัวแปร
ค่าคงที่และนิพจน์ออกจอภาพ

ฟังก์ชัน scanf(); เป็นฟังก์ชันที่ใช้ในการรับข้อมูลจากแป้นพิมพ์เข้ามาเก็บ
ไว้ในตัวแปร

ฟังก์ชัน getch(); เป็นฟังก์ชันที่ใช้ในการรอรับการกดแป้นพิมพ์หนึ่งครั้ง
โดยไม่ต้องกดEnter และตัวอักษรที่ป้อนเข้ามาจะไม่ปรากฏบนจอภาพ
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชัน `getchar()`; เป็นฟังก์ชันที่ใช้ในการรับข้อมูลเข้ามาทางแป้นพิมพ์ทีละ 1 ตัวอักษรแล้วกด Enter 1 ครั้ง ข้อมูลที่ป้อนจะแสดงบนจอภาพ

ฟังก์ชัน `gets()`; เป็นฟังก์ชันที่ใช้ในการรับข้อมูลที่เป็นข้อความจากแป้นพิมพ์เข้ามาเก็บไว้ในตัวแปรแบบอาร์เรย์ การใช้ฟังก์ชัน `gets()`; จะต้องมีการประกาศตัวแปรแบบอาร์เรย์ และกำหนดจำนวนตัวอักษรที่ต้องการป้อน โดยคอมพิวเตอร์จะจองพื้นที่ไว้ตามจำนวนตัวอักษร แต่จะป้อนได้น้อยกว่าที่จองไว้ 1 ตัว เพื่อให้ตัวแปรเก็บ 0 อีก 1 ตัว

ฟังก์ชัน `textcolor()`; เป็นฟังก์ชันที่ใช้ในการกำหนดสีตัวอักษร โดยจะต้องใช้ร่วมกับฟังก์ชัน `cprintf` ซึ่งมีสีต่างๆ ให้เลือก ตัวเลขค่าสีอาจจะพิมพ์เป็นตัวเลขหรือชื่อสีเป็นภาษาอังกฤษตัวพิมพ์ใหญ่ก็ได้

ตารางที่ 2.6 แสดงค่ารหัสสีที่ใช้ในโปรแกรมภาษาซี

ตัวเลขค่าสี	สีที่ปรากฏ
0	ดำ
1	น้ำเงิน
2	เขียว
3	ฟ้า
4	แดง
5	ม่วง
6	น้ำตาล
7	เทาสว่าง
8	เทาดำ
9	น้ำเงินสว่าง
10	เขียวสว่าง
11	ฟ้าสว่าง
12	แดงสว่าง
13	ม่วงสว่าง

14	เหลือง
15	ขาว

ฟังก์ชัน `cprintf()`; เป็นฟังก์ชันที่ใช้ในการพิมพ์ข้อความเหมือนฟังก์ชัน `printf` แต่จะแสดงเป็นสีต่างๆ ตามที่กำหนดไว้ในฟังก์ชัน `textcolor` การใช้ฟังก์ชัน `cprintf` ต้องกำหนดสีของตัวอักษรในฟังก์ชัน `textcolor` ก่อน

ฟังก์ชัน `textbackground()`; เป็นฟังก์ชันที่ใช้ในการกำหนดสีพื้นให้กับตัวอักษร

2.6.5 ชนิดของข้อมูล (Data Type)

ภาษาซีเป็นอีกภาษาหนึ่งที่มีชนิดของข้อมูลให้ใช้งานหลายอย่างด้วยกัน ซึ่งชนิดของข้อมูลแต่ละอย่างมีขนาดเนื้อที่ที่ใช้ในหน่วยความจำที่แตกต่างกัน และเนื่องจากการที่มีขนาดที่แตกต่างกันนั้นเอง ทำให้มีความสามารถในการจัดเก็บข้อมูลแต่ละประเภทแตกต่างกันไป ดังนั้นในการเลือกงานประเภทข้อมูลก็ควรจะคำนึงถึงความจำเป็นในการใช้งานด้วย สำหรับประเภทของข้อมูลมีดังนี้คือ

- ข้อมูลชนิดตัวอักษร คือ ข้อมูลที่เป็นรหัสแทนตัวอักษรหรือค่าจำนวนเต็ม ได้แก่ ตัวอักษร ตัวเลขและกลุ่มตัวอักษรพิเศษใช้พื้นที่ในการเก็บข้อมูล 1 ไบต์
- ข้อมูลชนิดจำนวนเต็ม คือ ข้อมูลที่เป็นเลขจำนวนเต็ม ได้แก่ จำนวนเต็มบวก จำนวนเต็มลบ และศูนย์ ข้อมูลชนิดจำนวนเต็มใช้พื้นที่ในการเก็บข้อมูล ขนาด 2 ไบต์
- ข้อมูลชนิดจำนวนเต็มที่มีขนาด 2 เท่า (Long Integer) คือ ข้อมูลที่เป็นเลขจำนวนเต็มใช้พื้นที่ในการเก็บเป็น 2 เท่าของ Integer คือมีขนาด 4 ไบต์
- ข้อมูลชนิดเลขทศนิยม (Float) คือ ข้อมูลที่เป็นเลขทศนิยม ขนาด 4 ไบต์
- ข้อมูลชนิดเลขทศนิยมอย่างละเอียด (Double) คือ ข้อมูลที่เป็นเลขทศนิยม ใช้พื้นที่ในการเก็บข้อมูลเป็น 2 เท่าของ float คือมีขนาด 8 ไบต์

2.6.6 ตัวดำเนินการทางคณิตศาสตร์

ในโปรแกรมภาษาซีจะใช้ตัวดำเนินการทางคณิตศาสตร์เป็นตัวเชื่อมในการเขียนโปรแกรม เพื่อหาผลลัพธ์จากการคำนวณซึ่งสามารถกระทำกับข้อมูลได้หลายรูปแบบ เช่น บวก ลบ คูณ หาร เปอร์เซ็นต์ ดังตัวอย่างในตารางข้างล่างนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.7 แสดงตัวดำเนินการทางคณิตศาสตร์

ตัวดำเนินการ	ความหมาย	ตัวอย่าง
+	บวก	$X+Y$
-	ลบ	$X-Y$
*	คูณ	$X*Y$
/	หาร	X/Y
++	เพิ่มค่าครั้งละ 1	$X++$
--	ลดค่าครั้งละ 1	$X--$
%	หารเอาผลลัพธ์เฉพาะเศษ	$X\%Y$

2.6.7 ตัวดำเนินการเปรียบเทียบ

ตัวดำเนินการเปรียบเทียบในโปรแกรมภาษาซีคือเครื่องหมายที่ใช้ในการเปรียบเทียบในทางคณิตศาสตร์ ผลลัพธ์จะมี 2 กรณีคือ ถ้าผลลัพธ์ถูกต้องหรือเป็นจริงจะมีค่าเป็น 1 ถ้าผลลัพธ์ผิดหรือเป็นเท็จจะมีค่าเป็น 0 ผลลัพธ์ของการเปรียบเทียบมีชื่อเรียกอีกอย่างหนึ่งว่าค่าคงที่บูลีน (Boolean Constant)

ตารางที่ 2.8 แสดงตัวดำเนินการเปรียบเทียบ

ตัวดำเนินการ	ความหมาย	ตัวอย่าง
>	มากกว่า	$X>Y$
<	น้อยกว่า	$X<Y$
>=	มากกว่าหรือเท่ากับ	$X>=Y$
<=	น้อยกว่าหรือเท่ากับ	$X<=Y$
==	เท่ากับ	$X==Y$
!=	ไม่เท่ากับ	$X!=Y$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.8 ตัวดำเนินการทางตรรกะ

ตัวดำเนินการทางตรรกะในโปรแกรมภาษาซี คือเครื่องหมายที่ใช้เชื่อมเงื่อนไข 2 เงื่อนไข หรือมากกว่า 2 เงื่อนไข เพื่อให้การเปรียบเทียบมีความละเอียดมากขึ้น ใช้สัญลักษณ์แทนในแต่ละเครื่องหมาย ดังต่อไปนี้

1. && จะให้ผลลัพธ์เป็นจริงเมื่อเงื่อนไขทั้งสองเป็นจริง แต่ถ้าเงื่อนไขใดเป็นเท็จ หรือทั้งสองเงื่อนไขเป็นเท็จจะทำให้ผลลัพธ์เป็นเท็จ
2. || จะให้ผลลัพธ์เป็นจริงเมื่อเงื่อนไขใดเงื่อนไขหนึ่งเป็นจริง หรือเป็นจริงทั้งสองเงื่อนไข แต่ถ้าเป็นเท็จทั้งสองเงื่อนไขจะทำให้ผลลัพธ์เป็นเท็จ
3. ! จะให้ผลลัพธ์เป็นจริงเมื่อเงื่อนไขหลัง not เป็นเท็จ แต่ถ้าเงื่อนไขหลัง not เป็นจริง จะทำให้ผลลัพธ์เป็นเท็จ
4. || หรือ (or) `Score<0||score>100`
5. ! มีความหมายว่า ไม่

2.6.9 นิพจน์ทางคณิตศาสตร์

ลำดับการประมวลผลของนิพจน์คณิตศาสตร์จะทำการประมวลผลในส่วนของวงเล็บก่อนในกรณีที่มีวงเล็บ จากนั้นจะคำนวณไปตามลำดับของการประมวลดังตารางข้างล่างนี้ และหากมีเครื่องหมายที่อยู่ในลำดับการประมวลผลเดียวกันจะทำการคำนวณจากด้านซ้ายไปด้านขวา

ตารางที่ 2.9 แสดงนิพจน์ทางคณิตศาสตร์

เครื่องหมาย	ลำดับการประมวลผล
()	1
++ --	2
* / %	3
+ -	4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 ภาษาซีพลัสพลัส (C++)

ภาษาซีพลัสพลัสเป็นภาษาโปรแกรมคอมพิวเตอร์เนกประสงค์ มีโครงสร้างภาษาที่มีการจัดชนิดข้อมูลแบบสแตติก (statically typed) และสนับสนุนรูปแบบการเขียนโปรแกรมที่หลากหลาย (multi-paradigm language) ได้แก่ การโปรแกรมเชิงกระบวนการ, การนิยามข้อมูล, การโปรแกรมเชิงวัตถุ, และการโปรแกรมแบบเจเนริก (generic programming) ภาษาซีพลัสพลัสเป็นภาษาโปรแกรมเชิงพาณิชย์ที่นิยมมากภาษาหนึ่งนับตั้งแต่ช่วงทศวรรษ 1990

2.7.1 รูปแบบของภาษาซีพลัสพลัส

- ภาษาซีพลัสพลัสได้ถูกออกแบบมาเพื่อเป็นภาษาสำหรับการเขียนโปรแกรมทั่วไป สามารถรองรับการเขียนโปรแกรมในระดับภาษาเครื่องได้ เช่นเดียวกับภาษาซี
- ในทางทฤษฎี ภาษาซีพลัสพลัสควรมีความเร็วเทียบเท่าภาษาซี แต่ในการเขียนโปรแกรมจริงนั้น ภาษาซีพลัสพลัสเป็นภาษาที่มีการเปิดกว้างให้โปรแกรมเมอร์ (Programmer) เลือกรูปแบบการเขียนโปรแกรม ซึ่งทำให้มีแนวโน้มที่โปรแกรมเมอร์อาจใช้รูปแบบที่ไม่เหมาะสม ทำให้โปรแกรมที่เขียนมีประสิทธิภาพต่ำกว่าที่ควรจะเป็น และภาษาซีพลัสพลัสนั้นเป็นภาษาที่มีความซับซ้อนมากกว่าภาษาซี จึงทำให้มีโอกาสเกิดขณะคอมไพล์มากกว่า
- ภาษาซีพลัสพลัสได้รับการออกแบบเพื่อเข้ากันได้กับภาษาซีในเกือบทุกกรณี
- มาตรฐานของภาษาซีพลัสพลัส ถูกออกแบบมาเพื่อไม่ให้มีการเจาะจงแพลตฟอร์มคอมพิวเตอร์
- ภาษาซีพลัสพลัสถูกออกแบบมาให้รองรับรูปแบบการเขียนโปรแกรมที่หลากหลาย (multi-paradigm)

2.8 IIS7

IIS ย่อมาจาก Internet Information Service เป็นโปรแกรมสำหรับการจำลองเครื่องคอมพิวเตอร์ของเราให้กลายเป็นเครื่องเว็บเซิร์ฟเวอร์ ซึ่งมีไว้ให้บริการด้านเซิร์ฟเวอร์ในรูปแบบต่างๆ ของอินเทอร์เน็ต เช่น Web server , FTP Server , SMTP Server ฯลฯ ในระบบปฏิบัติการวินโดวส์ ถูกพัฒนาโดยบริษัทไมโครซอฟท์ ซึ่งในวินโดวส์เซิร์ฟเวอร์ 2003 นั้น เวอร์ชันของ IIS จะเป็นเวอร์ชัน 6.0 (IIS 6.0) ซึ่งทางไมโครซอฟท์ได้ทำการออกแบบโปรแกรมใหม่ทั้งหมด โดยเน้นในเรื่องความปลอดภัยเป็นพิเศษ เนื่องจากในเวอร์ชันก่อนหน้านั้นคือ IIS 5.0 ในวินโดวส์เซิร์ฟเวอร์ 2000 จะมีช่องโหว่ความปลอดภัยค่อนข้างมาก และที่สำคัญคือความมั่นคงจะถูกติดตั้งโดยดีฟอลท์พร้อมกับระบบปฏิบัติการ ซึ่งทำให้เกิดปัญหาด้านความปลอดภัยและเป็นช่องทางการระบาดของไวรัสต่างๆ เช่น Code Red และ Nimda ดังนั้น บนวินโดวส์เซิร์ฟเวอร์ 2003 นั้น IIS 6.0 จะไม่ทำการติดตั้งโดย

ดีฟอลท์พร้อมกับระบบปฏิบัติการแต่ผู้ใช้ต้องทำการติดตั้งเองเมื่อต้องการใช้งาน และนอกจากนี้ IIS 6.0 ยังได้รับการพัฒนาให้มีประสิทธิภาพการทำงานที่ดีขึ้น ทำให้สามารถรองรับการใช้งานต่างๆ ได้ดียิ่งขึ้น และล่าสุดบริษัทไมโครซอฟท์ ได้ออกเวอร์ชันใหม่นั้นคือ IIS 7.0

2.9 วิชาลเบสิก (Visual basic)

โปรแกรมภาษาวิชาลเบสิกเป็นโปรแกรมแบบ GUI พัฒนามาจากภาษาเบสิก(Basic) โดยบริษัทไมโครซอฟท์(Microsoft) เป็นเครื่องมือที่ช่วยให้การพัฒนาแอปพลิเคชัน (Application) บนระบบปฏิบัติการวินโดว (Windows) ซึ่งสามารถใช้งานได้ง่าย ภาษานี้เป็นหนึ่งในภาษาโปรแกรมยอดนิยมสำหรับโปรแกรมที่ใช้ในด้านธุรกิจ

2.9.1 จุดเด่นของภาษาวิชาลเบสิก

- มีโครงสร้างภาษาที่ใกล้เคียงกับภาษามนุษย์ทำให้สามารถเรียนรู้ได้รวดเร็วและง่าย
- โปรแกรมมีการรวบรวมเครื่องมือที่ช่วยพัฒนาแอปพลิเคชันทำให้สะดวกต่อการใช้งาน
- สามารถสร้างไฟล์ .EXE ที่สามารถทำงานได้ด้วยตัวเอง
- สามารถออกแบบส่วนหน้าจอติดต่อผู้ใช้ (Form) ได้ทันทีโดยไม่ต้องรอการเขียนรหัสโปรแกรม
- ในโปรแกรมการบันทึกส่วนติดต่อผู้ใช้ (Form) และส่วนของรหัสโปรแกรม (Code) แยกกัน ทำให้สามารถนำส่วนติดต่อผู้ใช้เดิมไปปรับปรุงใช้กับโปรแกรมอื่นๆ ได้โดยไม่ต้องทำการสร้างใหม่
- สามารถพัฒนาแอปพลิเคชันได้หลายแบบ เช่น โปรแกรมด้านธุรกิจ สื่อการเรียนการสอน เกมส์ มัลติมีเดีย การจัดการฐานข้อมูล อินเทอร์เน็ต และ เว็บแอปพลิเคชัน เป็นต้น
- ตัวโปรแกรมสามารถจัดการกับหน่วยความจำได้ดีมาก
- สามารถสร้างแอปพลิเคชันบนมือถือได้

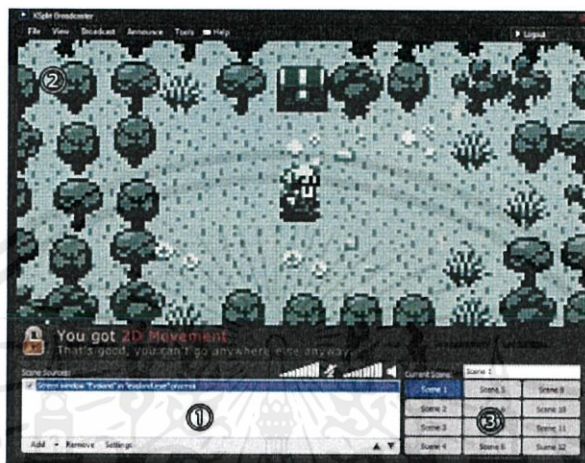
2.10 Twitch

Twitch เป็นโปรแกรมที่ใช้ในการทำการสตรีมมิ่ง เป็นบริการถ่ายทอดสดลักษณะเดียวกัน แต่จับกลุ่มตลาดเฉพาะคือกลุ่มคนเล่นเกมส์ (Game) ทั้งการแข่งขันเกมส์ หรือบันทึกการเล่นเกมของผู้เล่นคนใดๆ และนำมาเผยแพร่ในวงกว้าง และหารายได้จากการโฆษณาบนสตรีมวิดีโอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.11 xSplit

Xsplit เป็นโปรแกรมที่ทำมาเพื่อใช้ในการสตรีมมิ่ง โดยเฉพาะ เครื่องมือในตัวโปรแกรมนั้นมีอยู่ไม่มากนัก แต่ก็ยังเป็นเครื่องมือที่แทบจะต้องใช้ทั้งหมดสำหรับผู้ใช้งานการสตรีมมิ่ง



รูปที่ 2.18 ตัวอย่างหน้าจอโปรแกรม xSplit
(ที่มา: <http://goo.gl/hKRjQN>)

คำอธิบายรูปภาพ

1. Screen Source คือ เลเยอร์ในการเรียกออพเจ็ค ไม่ว่าจะเป็นภาพ วิดีโอ ที่ต้องการจะนำมาแสดงบนช่องในหมายเลข 2 ในส่วนนี้ หากต้องการเรียกภาพ เช่น กล้องเว็บแคม หรือ ใส่ตัวหนังสือบนจอ และครอป (Crop) ภาพบนหน้าจอคอม ขึ้นมาให้กดที่ Add ด้านซ้ายล่าง

- Add Camera คือ ในส่วนที่ใช้เรียก Device ของกล้องเว็บแคมขึ้นมา
- Add Screen Region คือ เมื่อเราทำการคลิกแล้ว โปรแกรมจะให้เราครอปจุดที่ต้องการให้แสดงภาพขึ้นมา

- Add Title คือ ในส่วนที่ใช้สำหรับใส่ตัวหนังสือลงไปบนหน้าจอ

2. Screen คือ ในส่วนของหน้าจอ ใช้สำหรับในการจัดเรียงออพเจ็คต่างๆจากช่อง 1 ซึ่งตรงหน้านี้ ถ้าถูกจัดวางอย่างไร ผู้รับชมการสตรีมก็จะเห็นอย่างนั้นเช่นกัน

3. Current Scene คือ ในส่วนหน้าจอที่ใช้สำหรับการเสนอภาพ ตัวอย่างเช่น เมื่อเวลาเรากำลังนำเสนองาน เราสามารถจัดเรียงออพเจ็คจากช่อง 1 ไว้หลายๆหน้า แล้วกดเรียกขึ้นมาได้เลยโดยไม่ต้องมาจัดวางใหม่ เพียงแต่กด Scene 1 2 3 ไปเรื่อยๆ

2.11.1 การตั้งค่าความละเอียด

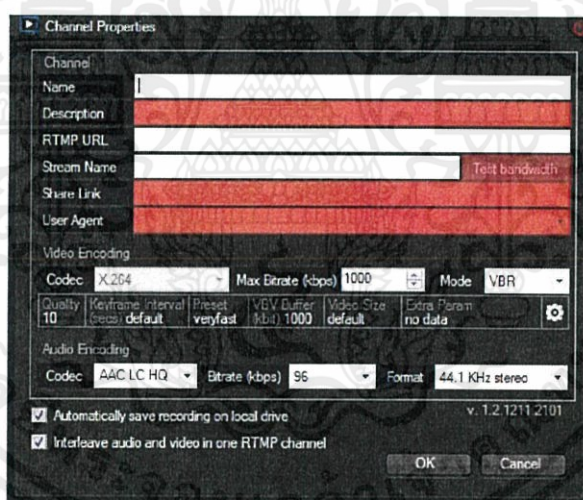
View > Resolution > แล้วทำเลือกขนาดที่ต้องการ หากไม่มีขนาดที่ต้องการ ให้เข้าไปที่ Edit Resolution แล้ว Add ขนาดที่ต้องการเองได้เลย

2.11.2 การตั้งค่า Frame Per Second (FPS)

ไปที่ View > Frame Rate แล้วเลือก FPS ที่ต้องการ ในการใช้งานโปรแกรมฟรีจะมีการจำกัดค่า FPS อยู่แค่เพียง 25 เท่านั้น

2.11.3 การตั้งค่าสตรีมมิ่ง

ใน Xsplit นั้น จะมีการรองรับของเว็บสตรีมมิ่งต่างๆทั่วโลก เช่น Ustream Niconico Twitch Justin เป็นต้น จะทำให้โปรแกรมสามารถเลือกเข้าระบบ(Login) ผ่านยูเซอร์ของเว็บนั้นๆ ได้ แล้วทำการตั้งค่า โดยเริ่มจาก ไปที่ Broadcast > Edit Channel > Add Channel



รูปที่ 2.19 หน้าจอการตั้งค่าการสตรีมมิ่ง

(ที่มา: <http://goo.gl/LFFgLZ>)

คำอธิบายรูป 2.19

1. Name คือ ชื่อของช่องสัญญาณที่เราต้องการ
2. RTMP URL : คือ URL ของเซิร์ฟเวอร์ ที่เราต้องการจะเชื่อมต่อเข้าไป

เอกสารนี้เป็นเอกสารที่ 3. Stream Name หรือ Stream Key คือ รหัสสำหรับการเชื่อมต่อให้การสตรีมของเราขึ้น การค้าไม่ว่ากรณีไปแสดงภาพได้ถูกต้อง หากใส่รหัสผิดจะไม่สามารถเชื่อมต่อได้ เจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. Video Codec คือ ทางโปรแกรมบังคับให้ตั้งค่าเท่ากับ X.264 (ไม่สามารถปรับค่าได้)

บทที่ 3

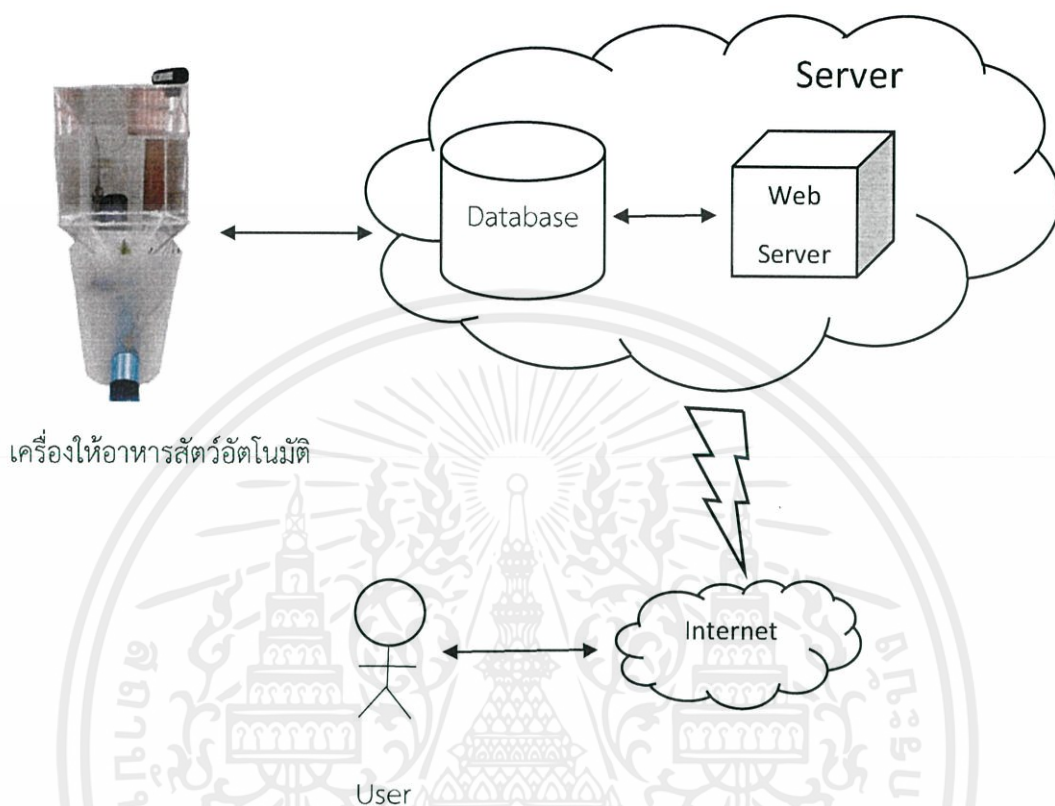
การวิเคราะห์และออกแบบระบบ

3.1 แนวคิดของระบบ

เครื่องให้อาหารสัตว์อัตโนมัติผ่านระบบอินเทอร์เน็ตได้ออกแบบไว้จะประกอบด้วย 2 ส่วนใหญ่คือ ส่วนของฮาร์ดแวร์ และ ส่วนของซอฟต์แวร์ ในส่วนของฮาร์ดแวร์จะติดต่อสื่อสารกับซอฟต์แวร์ เพื่อให้อุปกรณ์ต่างต่างนั้นทำงานตามที่ผู้เขียนโปรแกรมได้ทำการเขียนไว้ซึ่งในด้านฮาร์ดแวร์จะประกอบด้วยเครื่องให้อาหารสัตว์ที่จะต้องทำการปล่อยอาหารสัตว์ลงมาเพื่อมีคำสั่ง, กล้องเว็บแคมใช้ตรวจจับภาพสัตว์, บอร์ดไมโครคอนโทรลเลอร์อาร์ดูโนลีโอนาร์โดซึ่งใช้สำหรับการประมวลผลและติดต่อสื่อสารระหว่างฮาร์ดแวร์และตัวโปรแกรม, เซอร์โวที่ใช้ในการหมุนกล้องเว็บแคมและโซลินอยด์ที่ใช้ในการชักการเปิดปิดในการปล่อยอาหารสัตว์ลงมา และทางผู้จัดทำใช้ตัวเครื่องคอมพิวเตอร์พกพาในการทำเสมือนเป็นเซิร์ฟเวอร์ ในส่วนของซอฟต์แวร์จะมีการแบ่งเป็น 3 ส่วนใหญ่หลักๆ คือ การเรียกดูภาพจากกล้องเว็บแคม การเขียนโปรแกรมการปล่อยอาหารสัตว์โดยการกำหนดปริมาณอาหาร และการเขียนโปรแกรมการปล่อยอาหารสัตว์ตามวันเดือนปีตามที่ผู้ใช้งานต้องการให้อาหาร ทั้งนี้การใช้งานเครื่องให้อาหารสัตว์อัตโนมัตินั้นต้องสามารถใช้งานผ่านสมาร์ตโฟนได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 ภาพรวมของระบบ

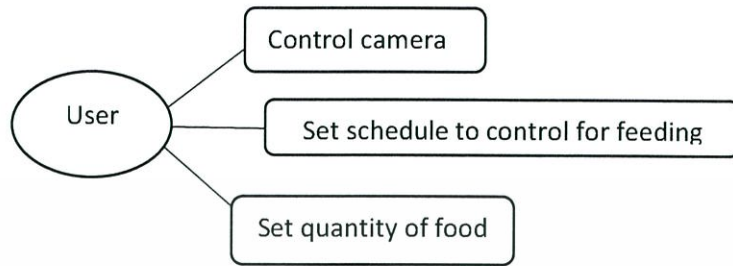


รูปที่ 3.1 ภาพรวมของระบบ

ในส่วนของการรับข้อมูลรับข้อมูลจากยูเซอร์ทำการป้อนข้อมูลผ่านเครือข่ายอินเทอร์เน็ตที่เชื่อมต่อกับระบบเซิร์ฟเวอร์อยู่ จากนั้นระบบเซิร์ฟเวอร์จะส่งข้อมูลผ่านซีเรียลพอร์ทซึ่งเป็นสัญญาณคำสั่งไปยังบอร์ดอาร์ดูโนลีโอนาร์โดในการสั่งงานให้เซอร์โวมอเตอร์หมุนเพื่อทำการควบคุมองศาของกล้อ และควบคุมโซลินอยด์ให้สามารถทำงานปล่อยอาหารตามคำสั่งของยูเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

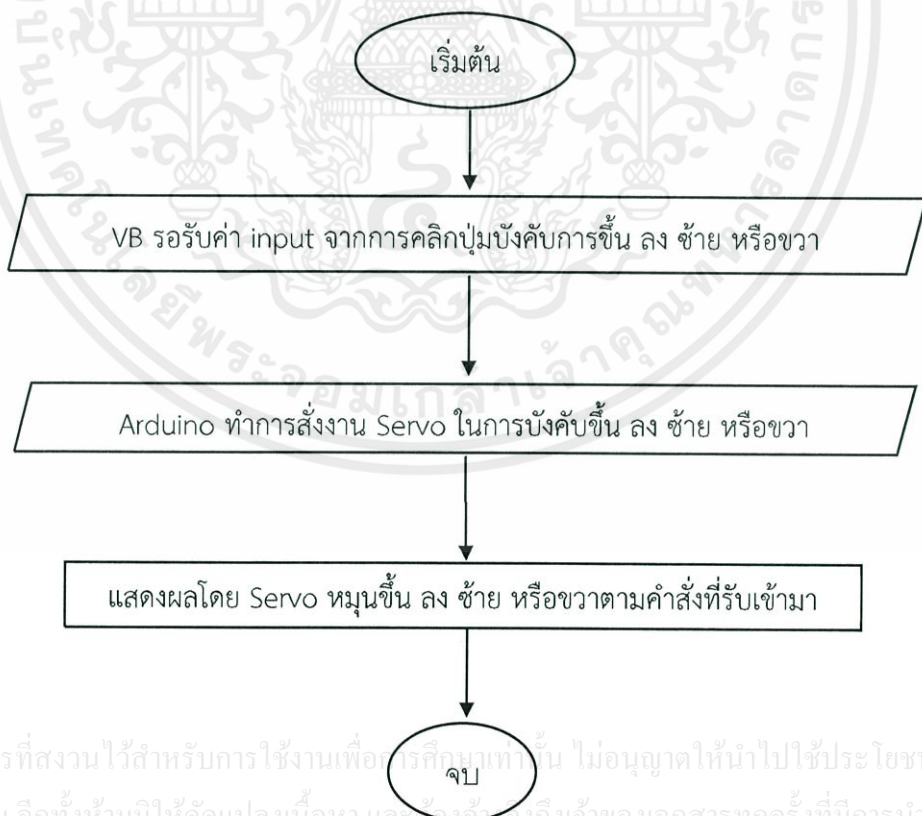
3.3 ยูสเคสของระบบ



รูปที่ 3.2 ยูสเคสของระบบ

3.4 โฟลวชาร์ตของระบบ

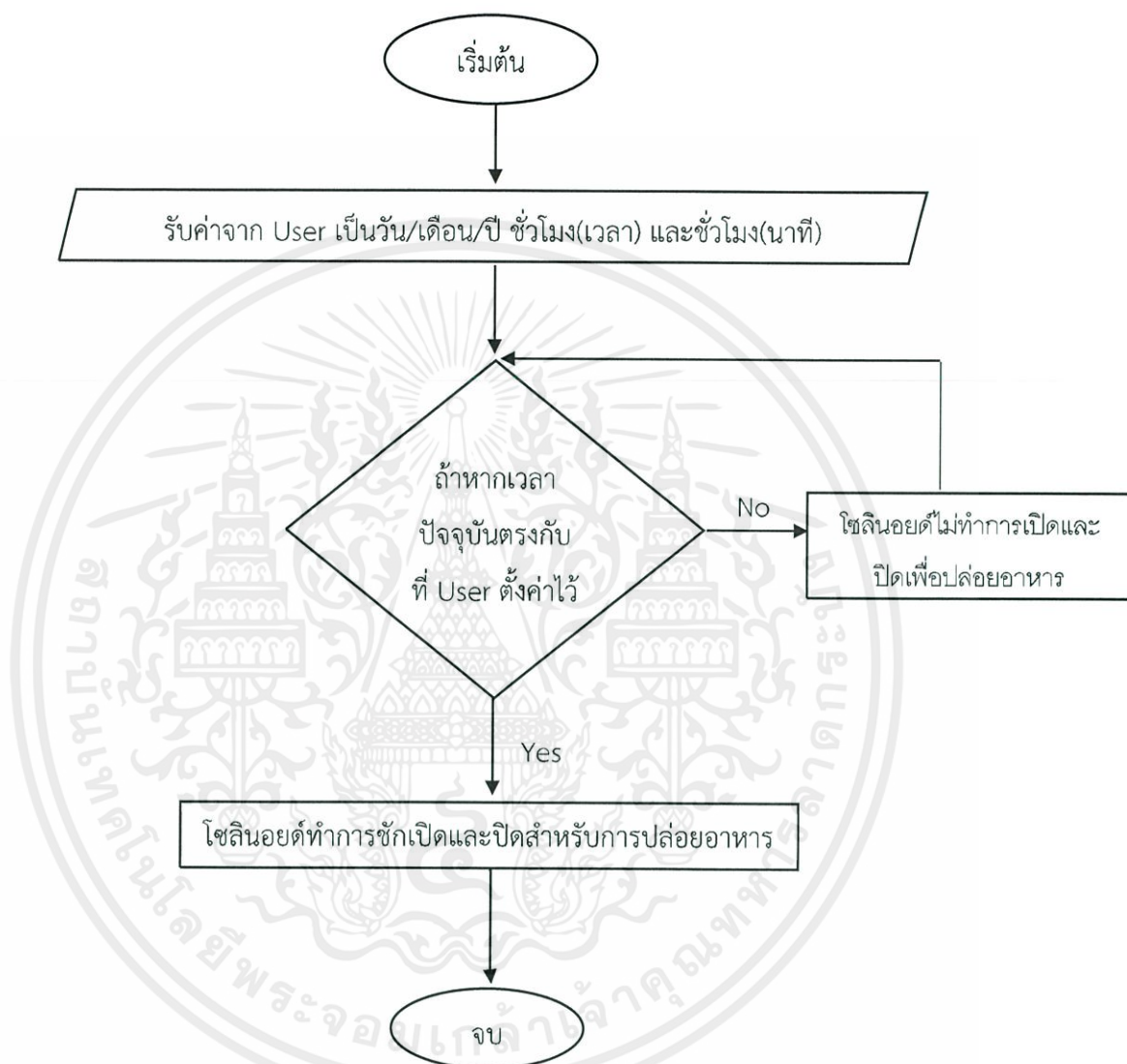
3.4.1 โฟลวชาร์ตของระบบควบคุมการหมุนของกล้อง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และเผยแพร่ไปยังเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.3 โฟลว์ชาร์ตของระบบควบคุมการหมุนของกล้อง

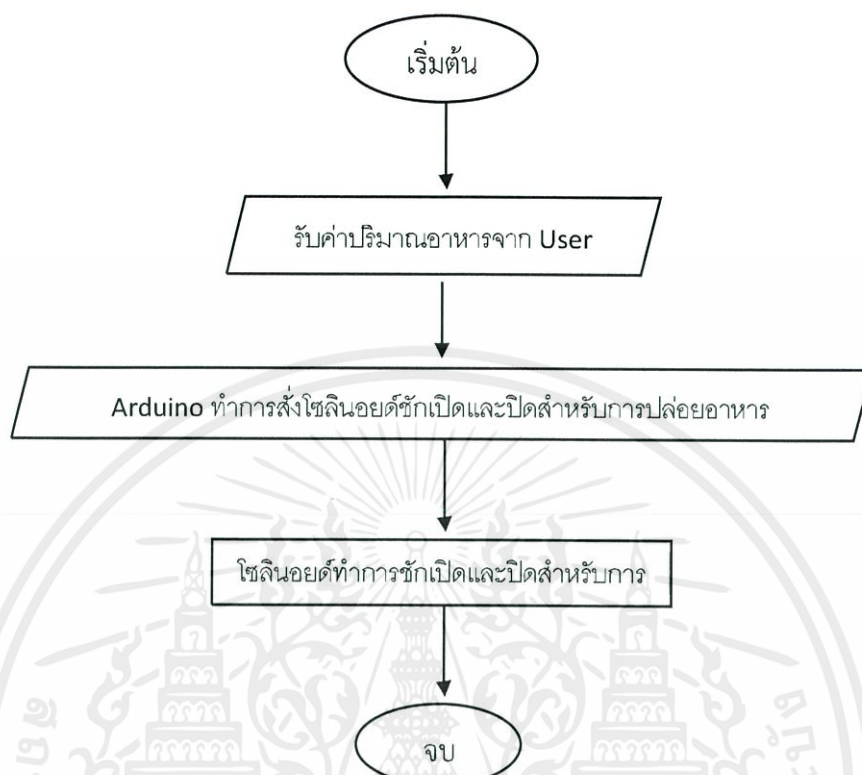
3.4.2 โฟลวชาร์ตของระบบการสั่งงานให้อาหารสัตว์โดยกำหนดเป็นวัน/เดือน/ปี ชั่วโมง (เวลา) และชั่วโมง(นาที)



รูปที่ 3.4 โฟลวชาร์ตของระบบการสั่งงานให้อาหารสัตว์โดยกำหนดเป็นวัน/เดือน/ปี ชั่วโมง(เวลา) และชั่วโมง(นาที)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

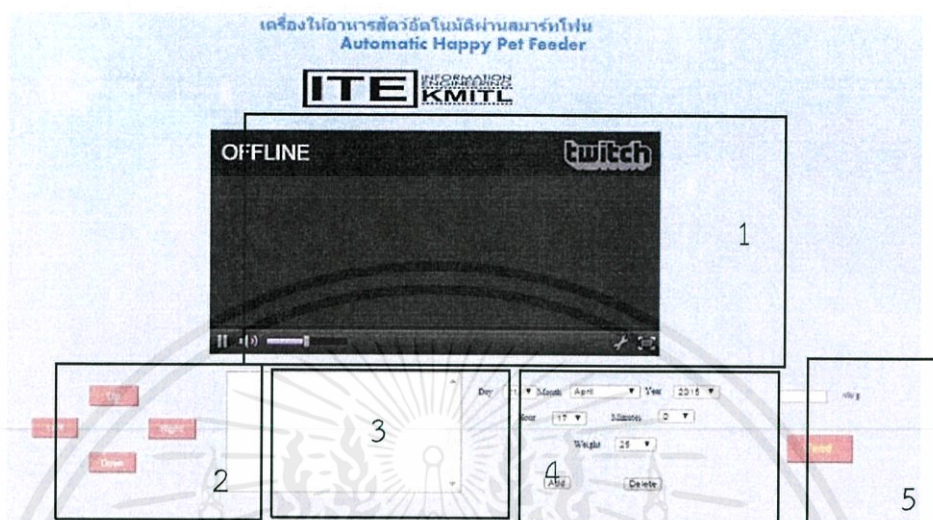
3.4.3 โพลวชาร์ตของระบบการสั่งงานให้อาหารสัตว์โดยกำหนดโดยปริมาณอาหาร



รูปที่ 3.5 โพลวชาร์ตของระบบการสั่งงานให้อาหารสัตว์โดยกำหนดเป็นวัน/เดือน/ปี ชั่วโมง(เวลา) และชั่วโมง(นาที)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 การออกแบบเว็บไซต์

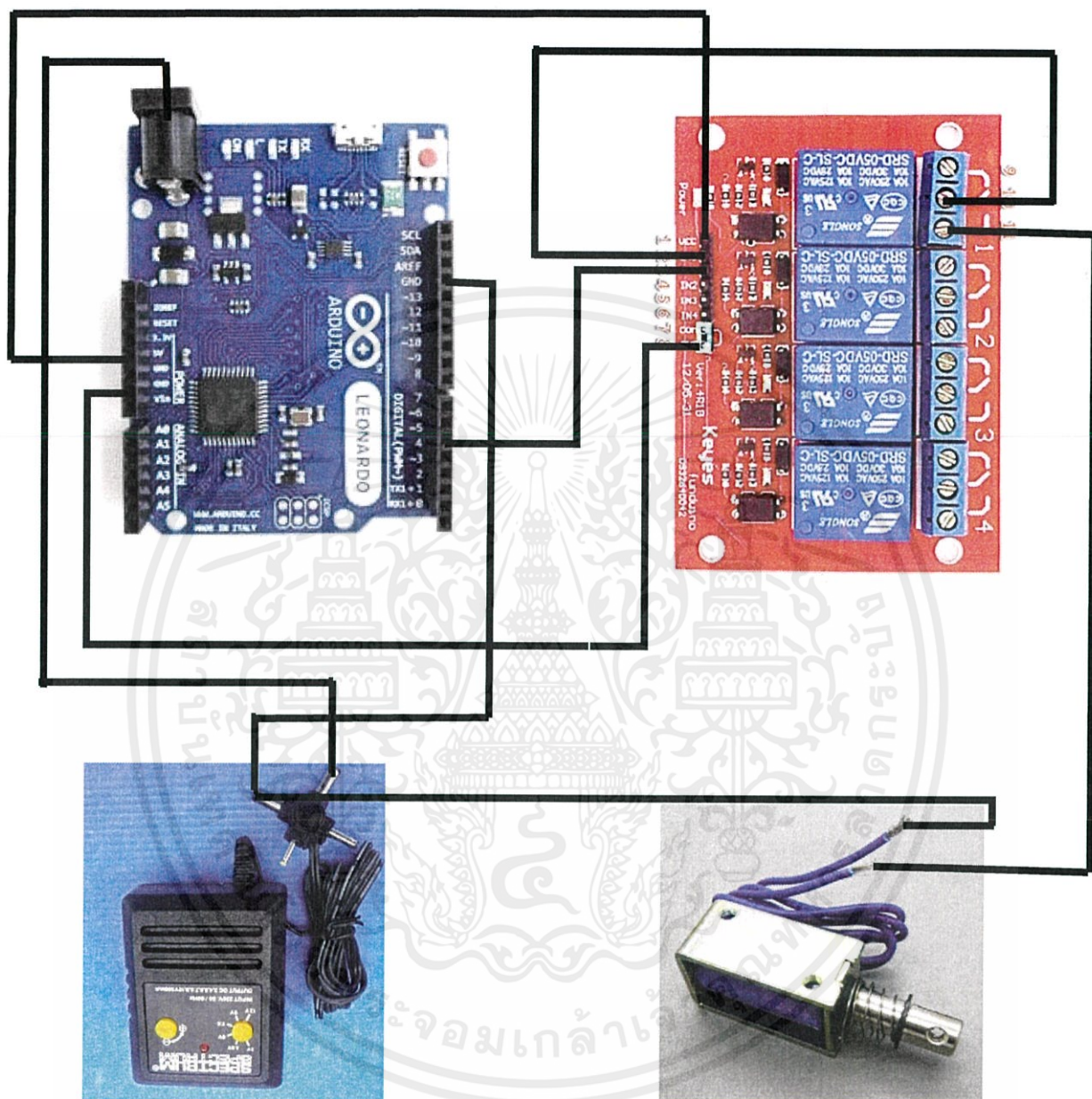


รูปที่ 3.6 หน้าเว็บไซต์

- หมายเลข 1 คือ ส่วนของหน้าจอที่ใช้สำหรับการดูสัตว์เลี้ยงจากกล้องเว็บแคม
- หมายเลข 2 คือ ส่วนของปุ่มสำหรับการกดบังคับการหมุนของเซอร์โวมอเตอร์ที่ใช้สำหรับการบังคับขึ้น ลง ซ้าย และขวา
- หมายเลข 3 คือ ส่วนของลิสต์บ็อก (List box) ที่ใช้สำหรับการเก็บข้อมูลการให้อาหารสัตว์เลี้ยงย้อนหลัง
- หมายเลข 4 คือ ส่วนของการให้ยูเซอร์กำหนดวัน เดือน ปี เวลา(ชั่วโมง) และ เวลา(นาที) ในการให้ เครื่องให้อาหารสัตว์เลี้ยงปล่อยอาหารโดยอัตโนมัติเมื่อถึงเวลานั้นๆ
- หมายเลข 5 คือ ส่วนของการรับข้อมูลในการให้อาหารสัตว์เลี้ยงโดยกำหนดเป็นปริมาณอาหาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6 การเชื่อมต่อของวงจร



รูปที่ 3.7 แสดงการเชื่อมต่อโดยรวมของวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการดำเนินงาน

การทดลองแบ่งเป็น 3 ส่วนใหญ่ๆคือ ส่วนของการใช้งานกล้องเว็บแคม, ส่วนของการใช้งานเครื่องให้อาหารสัตว์อัตโนมัติโดยการตั้งค่าวัน/เดือน/ปี และเวลาในการให้อาหาร และ ส่วนของการใช้งานเครื่องให้อาหารสัตว์โดยการกำหนดปริมาณอาหาร

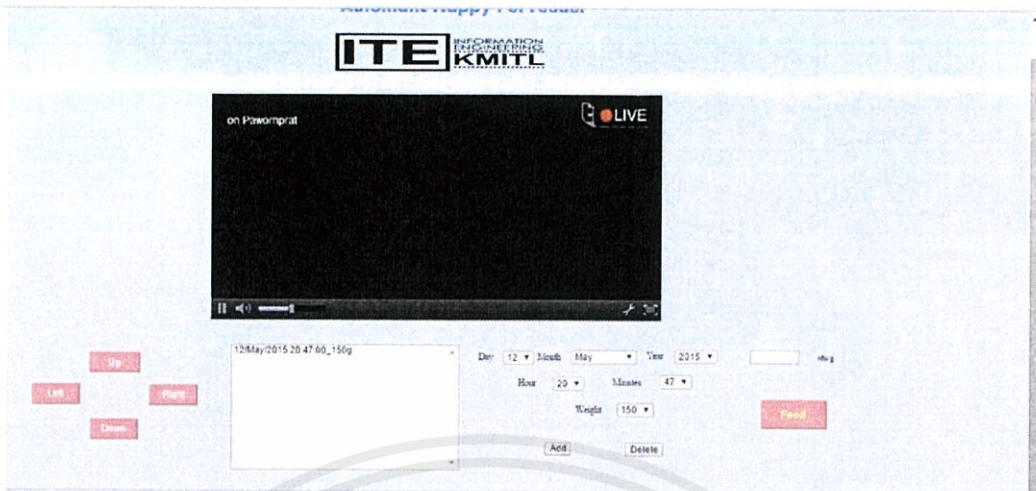
4.1 ผลการใช้งานกล้องเว็บแคม

การใช้งานกล้องเว็บแคมต้องทำการเชื่อมต่อโปรแกรม 2 โปรแกรมเข้าด้วยกัน คือ โปรแกรม Twitch และ โปรแกรม xSplit ซึ่งในตอนแรกหากเรายังไม่ทำการเชื่อม 2 โปรแกรมดังกล่าวเข้าด้วยกันหน้าจจะปรากฏดังภาพ (รูปที่ 4.1)



รูปที่ 4.1 หน้าต่างเมื่อระบบยังไม่เชื่อมต่อโปรแกรม xSplit และ Twitch

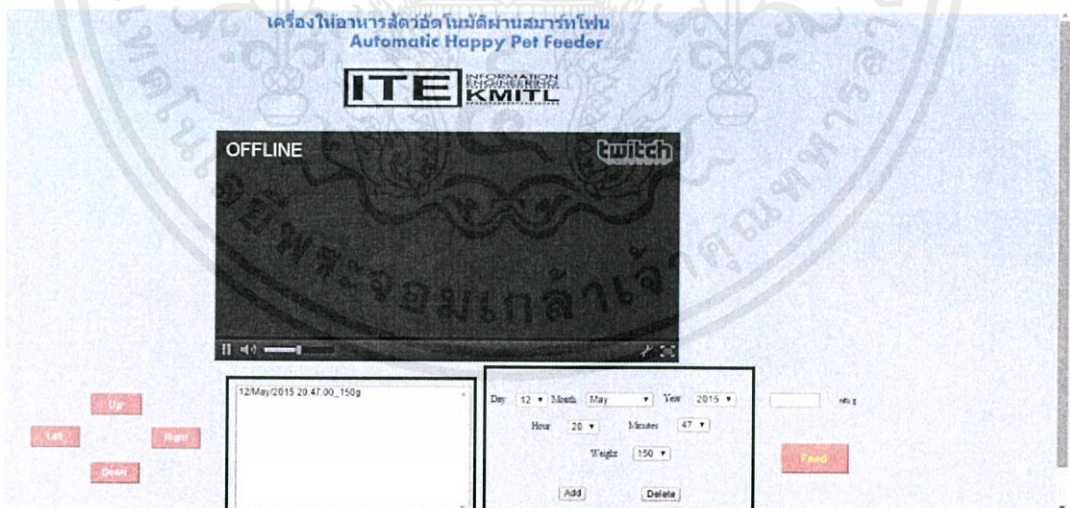
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 หน้าต่างเมื่อระบบสามารถทำการสตรีมมิ่งสำเร็จ

เมื่อระบบทำการเชื่อมต่อสำเร็จจะสังเกตเห็นคำว่า LIVE ขึ้นโชว์ที่มุมบนขวามือ เนื่องจากความเร็วอินเทอร์เน็ตที่ใช้งานไม่มีความเร็วที่พอสมควรและเซิร์ฟเวอร์ที่ใช้ในการทดลองเลือกเซิร์ฟเวอร์จากประเทศสิงคโปร์ส่งผลให้เกิดความดีเลย์ในการรับ-ส่งข้อมูล

4.2 ผลการใช้งานเครื่องให้อาหารสัตว์อัตโนมัติเมื่อทำการตั้งค่าวัน/เดือน/ปี เวลา (ชั่วโมง) และ เวลา(นาที)



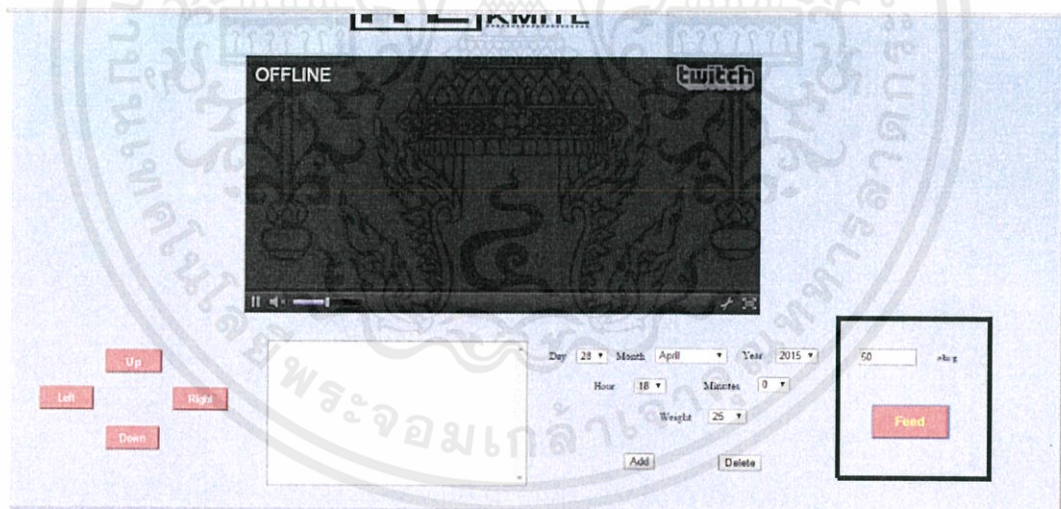
รูปที่ 4.3 หน้าต่างเมื่อป้อนข้อมูลวัน/เดือน/ปี และเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.4 ข้อมูลที่ป้อนและข้อมูลที่เก็บในลิสต์บ็อก

เมื่อทำการป้อนข้อมูลวัน/เดือน/ปี เวลา(ชั่วโมง) และเวลา(นาที) ในกล่องทางด้านขวามือ ระบบจะส่งข้อมูลไปเก็บยังฐานข้อมูลและส่งกลับมายังลิสต์บ็อก(List box) และเมื่อถึงเวลาตามที่กำหนดเครื่องให้อาหารสัตว์จะทำการปล่อยอาหารโดยอัตโนมัติตามปริมาณที่ระบุไว้

4.3 ผลการใช้งานเครื่องให้อาหารสัตว์อัตโนมัติเมื่อทำการป้อนค่าข้อมูลเป็นปริมาณอาหารที่ต้องการให้



รูปที่ 4.5 หน้าต่างการใช้งานแบบป้อนปริมาณอาหาร

เมื่อทำการกรอกปริมาณอาหารในหน่วยของกรัมที่ต้องการให้อาหารสัตว์เลี้ยง เครื่องให้อาหารสัตว์ทำการปล่อยอาหารสัตว์ทันทีโดยจะปล่อยอาหารลงมาโดย 1 วินาทีจะปล่อยอาหารลงมาประมาณ 37 กรัม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการทดลอง

เครื่องให้อาหารสัตว์อัตโนมัติผ่านระบบอินเทอร์เน็ตที่ทางผู้จัดทำทำการจัดทำขึ้นนั้น เป็นการใช้งานผ่านระบบเครือข่ายอินเทอร์เน็ตไร้สาย หรือ เครือข่ายอินเทอร์เน็ตโทรศัพท์เคลื่อนที่ โดยสามารถใช้ได้กับทั้งสมาร์ตโฟน แท็บเล็ต(Tablet) และคอมพิวเตอร์

โดยการใช้งานหลักๆจะมี 3 ส่วนคือ

1. การใช้งานกล้อง การใช้งานสามารถทำการส่งบังคับให้เซอร์โวหมุนขึ้น ลง ซ้าย หรือขวาได้ตามที่ยูเซอร์สั่งงาน และสามารถเรียกใช้งานเว็บแคมผ่านระบบอินเทอร์เน็ตเพื่อดูสภาพแวดล้อมของสัตว์เลี้ยงขณะนั้นได้
2. การให้อาหารสัตว์โดยการตั้งค่าวัน/เดือน/ปี ชั่วโมง(เวลา)และชั่วโมง(นาที) เมื่อยูเซอร์ทำการตั้งค่าเวลาที่ต้องการจะให้อาหารสัตว์และเมื่อถึงเวลานั้นเครื่องให้อาหารสัตว์อัตโนมัติจะทำการปล่อยอาหารสัตว์แบบอัตโนมัติเอง
3. การให้อาหารสัตว์โดยการกำหนดปริมาณอาหารที่ต้องการจะให้ เครื่องให้อาหารสัตว์อัตโนมัติจะทำการปล่อยอาหารสัตว์ตามปริมาณที่ยูเซอร์ทำการป้อนข้อมูลเข้ามา

5.1 ปัญหาที่เกิดขึ้น

- อุปกรณ์ที่สามารถนำมาใช้งานนั้นมีประสิทธิภาพไม่เพียงพอ
- อาหารสัตว์แต่ละเม็ดมีปริมาณน้ำหนักที่ไม่เท่ากันจึงยากต่อการคำนวณปริมาณน้ำหนักต่อการกำหนดปริมาณน้ำหนักของอาหารที่จะให้แต่ละครั้ง
- เนื่องจากระบบอินเทอร์เน็ตไม่มีความเสถียรจึงส่งผลให้เกิดดีเลย์ในการใช้งาน

5.2 แนวทางการพัฒนาต่อ

- พัฒนาในส่วนของตัวเครื่องให้มีประสิทธิภาพมากยิ่งขึ้น
- ปรับแต่งรูปแบบเว็บไซต์ให้มีความสะดวกและน่าใช้งานมากขึ้น
- แก้ไขปัญหาที่เกี่ยวกับความดีเลย์ให้มีความน่าเชื่อถือมากยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] Arduino. [ออนไลน์].

เข้าถึงได้จาก : <http://www.arduino.cc/en/Main/ArduinoBoardLeonardo> (23 เมษายน 2558)

[2] บทความ “ตัวอย่างการควบคุม RC Servo Motor ด้วย Arduino”. [ออนไลน์].

เข้าถึงได้จาก : <http://goo.gl/a8NBHs> (23 เมษายน 2558)

[3] ฤทธิ์ ชีระโกเมน. (ม.ป.ป.). โซลินอยด์.

เข้าถึงได้จาก : <http://goo.gl/29PQuj> (23 เมษายน 2558)

[4] Relay. [ออนไลน์]. เข้าถึงได้จาก : <http://goo.gl/SMcCJW> (23 เมษายน 2558)

[5] Doungjai Khuntasamon. (2556). อุปกรณ์คอมพิวเตอร์.

เข้าถึงได้จาก : <http://goo.gl/kjXTOJ> (24 เมษายน 2558)

[6] ภาษา C. (2556). เข้าถึงได้จาก : <http://guru.sanook.com/6394/> (24 เมษายน 2558)

[7] ภาษาซีพลัสพลัส. [ออนไลน์]. เข้าถึงได้จาก : <http://goo.gl/Bqa5Ja> (24 เมษายน 2558)

[8] IIS คืออะไร. (2557). เข้าถึงได้จาก : <http://goo.gl/q2oCL3> (25 เมษายน 2558)

[9] ISAREEYA KEATWUTTIKAN. (2557). โปรแกรม ภาษา VISUAL BASIC คืออะไร.

เข้าถึงได้จาก : <http://goo.gl/TjrAbl> (26 เมษายน 2558)

[10] รู้จัก Twitch บริการถ่ายทอดสด “เกมส์” สู่โลกออนไลน์. (2556).

เข้าถึงได้จาก : <https://www.blognone.com/node/49398> (27 เมษายน 2558)

[11] การใช้งาน xSplit. [ออนไลน์].

เข้าถึงได้จาก : http://www.men9ch.com/?page_id=561 (27 เมษายน 2558)

[12] การเขียนโปรแกรมคอมพิวเตอร์. [ออนไลน์].

เข้าถึงได้จาก : <http://www.lks.ac.th/anchalee/cindex-n.htm>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

โค้ดที่ใช้ในการรันโปรแกรมสำหรับการสร้างหน้าสื่อคอนเทนต์

```
Imports System.Data.OleDb
Imports System.Web.Configuration
Public Class WebForm1
Inherits System.Web.UI.Page
Dim con As New
OleDbConnection(ConfigurationManager.ConnectionStrings("AccessConne
ctionString").ConnectionString)
'Dim cmd As OleDbCommand
'Dim reader As OleDbDataReader
Dim dbProvider As String
Dim dbSource As String
Dim selectString As String = "select * from Authen"
Dim da As OleDb.OleDbDataAdapter
Dim ds As New DataSet
Dim dt As New DataTable

Sub connect()
dbProvider = "PROVIDER = Microsoft.ACE.OLEDB.12.0;"
dbSource = "Data Source = " & AppDomain.CurrentDomain.BaseDirectory &
"App_Data\HappyPetFeeder.accdb;"
'con.ConnectionString =
con.Open()

ds.Tables.Add(dt)
da = New OleDbDataAdapter(selectString, con)
da.Fill(dt)
con.Close()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น ผู้ที่นำเอกสารนี้ไปให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End Sub
```

```
Protected Sub Page_Load(ByVal sender As Object, ByVal e As  
System.EventArgs) Handles Me.Load  
connect()  
End Sub
```

```
Protected Sub Button1_Click(sender As Object, e As EventArgs) Handles  
Button1.Click  
Dim userstr As String  
Dim passstr As String  
Dim countRow As Integer  
Dim CheckLogin As Boolean = False  
If (TextBox1.Text.ToString <> vbEmpty & TextBox2.Text.ToString <>  
vbEmpty) Then  
userstr = TextBox1.Text.ToString  
passstr = TextBox2.Text.ToString  
For countRow = 0 To dt.Rows.Count - 1  
If userstr = dt.Rows(countRow)("User") And passstr =  
dt.Rows(countRow)("Password") Then  
CheckLogin = True  
End If  
Next
```

```
If CheckLogin = False Then  
MsgBox("User or Password invalid.")  
End If
```

```
Elseif (TextBox1.Text <> vbEmpty & TextBox2.Text = vbEmpty) Then
```

```
MsgBox("Please enter password.")
```

```
Elseif (TextBox1.Text = vbEmpty & TextBox2.Text <> vbEmpty) Then
```

```
MsgBox("Please enter user name.")
```

```
Else  
MsgBox("Please enter user name and password.")  
End If
```

```
If CheckLogin = True Then  
Response.Redirect("Page.aspx")  
End If
```

```
End Sub
```

```
Protected Sub Button2_Click(sender As Object, e As EventArgs) Handles  
Button2.Click  
Response.Redirect("Default.aspx")  
End Sub  
End Class
```

โค้ดสำหรับการเขียนหน้าเว็บไซต์ด้วยโปรแกรม Visual Studio 2012

```
Imports System  
Imports System.IO  
Imports System.IO.Ports  
Imports System.Web.UI.UpdatePanel  
Imports System.Web.UI.Control  
Imports System.Web.UI.Page  
Imports System.Math  
Imports System.Drawing  
Imports System.Timers  
Imports System.Web.UI.Timer  
Imports System.Threading  
Imports System.Data.OleDb
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น กรุณาแจ้งให้ทราบก่อน และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Public Class _Default  
Inherits System.Web.UI.Page  
Shared _continue As Boolean
```

```

Shared _serialPort As SerialPort
Dim Serialport1 As New IO.Ports.SerialPort
Dim x As Double
Dim y As Double
Dim rcb(256) As Byte
Dim rcb1(255) As Byte
Dim a As Integer
Dim rssi As Integer
Dim count As Integer = 0
Dim z As Integer
Dim num As Integer
Dim clear As Integer
Dim rs1 As Integer
Dim distance1 As Double
Dim rs2 As Integer
Dim distance2 As Double
Dim rs3 As Integer
Dim distance3 As Double
Dim rs4 As Integer
Dim distance4 As Double
Dim Distance As Double
Dim DL_End1 As Integer = Convert.ToInt32("8D", 16) 'last Byte of DL(end_dev1):
8D
Dim DL_End2 As Integer = Convert.ToInt32("60", 16) 'last Byte of DL(end_dev2):
08
Dim DL_End3 As Integer = Convert.ToInt32("F1", 16) 'last Byte of DL(end_dev3):
33
Dim DL_End4 As Integer = Convert.ToInt32("FB", 16) 'last Byte of DL(end_dev4):
FB
Private trd As Thread
Dim con As New OleDbConnection(ConfigurationManager.ConnectionStrings("AccessConnection
String").ConnectionString)

```

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น

```
Dim dbProvider As String
Dim dbSource As String
Dim selectString As String = "select * from Schedule"
Dim da As OleDb.OleDbDataAdapter
Dim ds As New DataSet
Dim dt As New DataTable
```

```
Sub connect()
On Error Resume Next
```

```
da = New OleDbDataAdapter
ds = New DataSet
dt = New DataTable
dbProvider = "PROVIDER = Microsoft.ACE.OLEDB.12.0;"
dbSource = "Data Source = " & AppDomain.CurrentDomain.BaseDirectory &
"App_Data\HappyPetFeeder.accdb;"
'con.ConnectionString = dbProvider & dbSource
con.Open()

ds.Tables.Add(dt)
da = New OleDbDataAdapter(selectString, con)
da.Fill(dt)

con.Close()
```

```
End Sub
```

```
Sub AddItem(dtime As String, qt As String)
```

```
On Error Resume Next
```

```
'Dim conStr As String = "PROVIDER = Microsoft.ACE.OLEDB.12.0;Data Source = "
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้ทำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น "App_Data\HappyPetFeeder.accdb;" อย่างอึงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Dim con As New OleDbConnection(ConfigurationManager.ConnectionStrings("AccessConnection  
String").ConnectionString)  
con.Open()
```

'Create the command.

```
Dim insertSQL As String = "INSERT INTO [Schedule] ([DateTime], [Quantity]) "  
'insertSQL &= "(DateTime,Quantity) "  
insertSQL &= "VALUES (@dt, @qt)"
```

```
Dim cmd As New OleDbCommand(insertSQL, con)  
cmd.Parameters.AddWithValue("@dt", dtime)  
cmd.Parameters.AddWithValue("@qt", qt)
```

```
cmd.ExecuteNonQuery()
```

```
con.Close()
```

```
End Sub
```

```
Sub DeletetItem(dtime As String, qt As String)
```

```
On Error Resume Next
```

```
'Dim conStr As String = "PROVIDER = Microsoft.ACE.OLEDB.12.0;Data Source = "  
& AppDomain.CurrentDomain.BaseDirectory &  
"App_Data\HappyPetFeeder.accdb;"
```

```
Dim con As New OleDbConnection(ConfigurationManager.ConnectionStrings("AccessConnection  
String").ConnectionString)  
con.Open()
```

'Create the command.

```
Dim DeleteSQL As String = "DELETE FROM Schedule "  
'insertSQL &= "(DateTime,Quantity) "
```

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของสำนักงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น Dim DeleteSQL As String = "DELETE FROM Schedule" เอกสารทุกครั้งที่มีการนำไปใช้

```
DeleteSQL &= "WHERE DateTime=@dt AND Quantity=@qt"
```

```
Dim cmd As New OleDbCommand>DeleteSQL, con)
```

```
cmd.Parameters.AddWithValue("@dt", dtime)
```

```
cmd.Parameters.AddWithValue("@qt", qt)
```

```
'Try
```

```
cmd.ExecuteNonQuery()
```

```
'Catch ex As Exception
```

```
'End Try
```

```
con.Close()
```

```
End Sub
```

```
Sub DeleteAllItem()
```

```
On Error Resume Next
```

```
'Dim conStr As String = "PROVIDER = Microsoft.ACE.OLEDB.12.0;Data Source = "
```

```
& AppDomain.CurrentDomain.BaseDirectory &
```

```
"App_Data\HappyPetFeeder.accdb;"
```

```
Dim con As New
```

```
OleDbConnection(ConfigurationManager.ConnectionStrings("AccessConnection
```

```
String").ConnectionString)
```

```
con.Open()
```

```
'Create the command.
```

```
Dim DeleteSQL As String = "DELETE * FROM Schedule "
```

```
'insertSQL &= "(DateTime,Quantity) "
```

```
'DeleteSQL &= "WHERE DateTime=@dt AND Quantity=@qt"
```

```
Dim cmd As New OleDbCommand>DeleteSQL, con)
```

```
'cmd.Parameters.AddWithValue("@dt", dtime)
```

```
'cmd.Parameters.AddWithValue("@qt", qt)
```

```
'Try
```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
cmd.ExecuteNonQuery()
```

```
'Catch ex As Exception
```

```
'End Try
```

```
con.Close()
```

```
End Sub
```

```
Sub GetDataToListbox()
```

```
On Error Resume Next
```

```
connect()
```

```
Listbox1.Items.Clear()
```

```
Dim countRow As Integer
```

```
For countRow = 0 To dt.Rows.Count - 1
```

```
'Dim teststr As String = dt.Rows(0)("DateTime")
```

```
Listbox1.Items.Add(dt.Rows(countRow)("DateTime").ToString & " " &  
dt.Rows(countRow)("Quantity").ToString & "g")
```

```
Next
```

```
End Sub
```

```
Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs)
```

```
Handles Me.Load
```

```
With Serialport1
```

```
.PortName = "COM7"
```

```
.BaudRate = 9600
```

```
.Parity = Parity.None
```

```
.DataBits = 8
```

```
.StopBits = StopBits.One
```

```
End With
```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการเรียนการสอนเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น DropDownList2.SelectedIndex = 3 ้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
'DropDownList3.SelectedIndex = 0
```

```
'DropDownList4.SelectedIndex = 18
Dim test As Integer = DropDownList1.SelectedIndex
Response.CacheControl = "no-cache"
Response.AddHeader("Pragma", "no-cache")
Response.Expires = -1
```

```
trd = New Thread(AddressOf testt)
trd.IsBackground = True
trd.Start()
```

```
GetDataToListBox()
```

```
End Sub
```

```
Protected Sub testt()
```

```
Dim count As Integer = 0
```

```
Dim stopwatch As Stopwatch
```

```
While (1)
```

```
CheckDateTime()
```

```
stopwatch = stopwatch.StartNew
```

```
Thread.Sleep(45000)
```

```
stopwatch.Stop()
```

```
End While
```

```
End Sub
```

```
Protected Sub Button1_Click(sender As Object, e As EventArgs) Handles
```

```
Button1.Click
```

```
Serialport1.Open()
```

```
Serialport1.Write("3")
```

```
Serialport1.Close()
```

```
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Protected Sub Button2_Click(sender As Object, e As EventArgs) Handles  
Button2.Click  
Serialport1.Open()  
Serialport1.Write("2")  
Serialport1.Close()  
End Sub
```

```
Protected Sub Button3_Click(sender As Object, e As EventArgs) Handles  
Button3.Click  
Serialport1.Open()  
Serialport1.Write("0")  
Serialport1.Close()  
End Sub
```

```
Protected Sub Button4_Click(sender As Object, e As EventArgs) Handles  
Button4.Click  
Serialport1.Open()  
Serialport1.Write("1")  
Serialport1.Close()  
End Sub
```

```
Protected Sub Button5_Click(sender As Object, e As EventArgs) Handles  
Button5.Click  
Dim Weight As String  
Weight = TextBox2.Text  
Solinoide(Weight)
```

```
End Sub
```

```
Protected Sub Button6_Click(sender As Object, e As EventArgs) Handles  
Button6.Click
```

```
Dim Day As String  
Dim Month As String  
Dim Year As String
```

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น ลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Dim Hour As String
Dim Minutes As String
Dim Qua As String
Dim TempStr As String
Dim DTStr As String
Dim QStr As String
```

```
Day = DropDownList1.SelectedValue
Month = DropDownList2.SelectedValue
Year = DropDownList3.SelectedValue
Hour = DropDownList4.SelectedValue
Minutes = DropDownList5.SelectedValue
Qua = DropDownList6.SelectedValue
```

```
TempStr = Day + "/" + Month + "/" + Year + " " + Hour + ":" + Minutes + ":" +
"00" + "_" + Qua + "g"
```

```
DTStr = Day + "/" + Month + "/" + Year + " " + Hour + ":" + Minutes + ":" + "00"
```

```
QStr = Qua
```

```
'ListBox1.Items.Add(TempStr)
```

```
AddItem(DTStr, QStr)
```

```
GetDataToListBox()
```

```
End Sub
```

```
Protected Sub Button7_Click(sender As Object, e As EventArgs) Handles
Button7.Click
```

```
'ListBox1.Items.Remove(ListBox1.SelectedItem)
```

```
'If ListBox1.SelectedIndex.ToString > -1 Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น

```
'Dim gh As String = ListBox1.Items(2).ToString
```

```
'ListBox1.SelectedIndex = 3
```

```

'Dim test As Integer = ListBox1.SelectedIndex

'Dim valStr As String = ListBox1.SelectedValue.ToString()
' valStr = "22/4/2015 17:21:00_25g"
'Dim DT As String = Mid(valStr, 1, (InStr(valStr, "_") - 1))
'Dim QT As String = Mid(valStr, InStr(valStr, "_") + 1, InStr(valStr, "g") - InStr(valStr,
"_) - 1)
DeleteAllItem()
ListBox1.Items.Clear()
'GetDataToListBox()
'End If

End Sub

Private Sub Solinoid(Weight As String)
Dim Time = Str(Int(Int(Weight) / 38))
Dim String1 = "4:" + Time
Serialport1.Open()
Serialport1.Write(String1)
'Serialport1.Write("4")
Serialport1.Close()
End Sub

Private Sub ReadDateTime()
Dim dtmtest As Date
Dim timetest As Date
dtmtest = DateValue(Now)
timetest = TimeValue(Now)
End Sub

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น Private Sub CheckDateTime() และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

On Error Resume Next

```
Dim DateNow As Date
Dim TimeNow As Date
Dim IntHour As Integer
Dim IntMinute As Integer
Dim TestDate As Date
Dim TestTime As Date
Dim valueinlistbox As String
Dim temp As String
Dim Check As String
Dim valStr As String
```

```
DateNow = DateValue(Now)
TimeNow = TimeValue(Now)
IntHour = Hour(Now)
IntMinute = Minute(Now)

For i As Integer = 0 To ListBox1.Items.Count - 1
    valueinlistbox = ListBox1.Items(i).ToString
    'Check = Mid(valueinlistbox, 2, 1)
    'If Check = "N" Then
    'temp = Mid(valueinlistbox, InStr(valueinlistbox, "]") + 2,
    InStr(InStr(valueinlistbox, "_") + 1, valueinlistbox, "_") - InStr(valueinlistbox, "]") -
    2)
    temp = Mid(valueinlistbox, 1, InStr(valueinlistbox, " ") - 1)
    TestDate = CDate(temp)
    If TestDate = DateNow Then
    'temp = Mid(valueinlistbox, InStr(InStr(valueinlistbox, "_") + 1, valueinlistbox, "_")
    + 1, InStr(InStr(InStr(valueinlistbox, "_") + 1, valueinlistbox, "_") + 1,
    valueinlistbox, "_") - InStr(InStr(valueinlistbox, "_") + 1, valueinlistbox, "_") - 1)
    temp = Mid(valueinlistbox, InStr(valueinlistbox, " ") + 1, InStr(valueinlistbox, "_")
    - InStr(valueinlistbox, " ") - 1)
    TestTime = Format(CDate(temp), "HH:mm")
    TimeNow = Format(TimeNow, "HH:mm")
```

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น

```

If TestTime = TimeNow Then
'เปลี่ยนจาก N เป็น Y แล้วใส่ลงไปแทนที่อันเดิม
'valueinlistbox = valueinlistbox.Replace("N", "Y")
'ListBox1.Items.RemoveAt(i)
'ListBox1.Items.Insert(i, valueinlistbox)
'ListBox1.DataSource = valueinlistbox
'ตัดประโยคเอาจำนวนกรั้มมาใส่แทน 22 ข้างล่างเนี่ย
'temp = Mid(valueinlistbox, InStr(InStr(InStr(valueinlistbox, "_" ) + 1,
valueinlistbox, "_" ) + 1, valueinlistbox, "_" ) + 1, InStr(valueinlistbox, "g") -
InStr(InStr(InStr(valueinlistbox, "_" ) + 1, valueinlistbox, "_" ) + 1, valueinlistbox,
"_" ) - 1)
temp = Mid(valueinlistbox, InStr(valueinlistbox, "_" ) + 1, InStr(valueinlistbox, "g")
- (InStr(valueinlistbox, "_" ) + 1))
Solinoide(temp)
valStr = ListBox1.Items(i).ToString
Dim DT As String = Mid(valStr, 1, (InStr(valStr, "_" ) - 1))
Dim QT As String = Mid(valStr, InStr(valStr, "_" ) + 1, InStr(valStr, "g") - InStr(valStr,
"_" ) - 1)
DeleteItem(DT, QT)
GetDataToListBox()

'Response.Redirect("Default.aspx")
Response.Redirect(HttpContext.Current.Request.Url.ToString(), True)
Response.Redirect(Request.RawUrl)
Response.Redirect(Request.Url.AbsoluteUri)
Response.Redirect("Page.aspx")

```

```

Response.CacheControl = "no-cache"
Response.AddHeader("Pragma", "no-cache")

```

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์ของมหาวิทยาลัยราชภัฏวชิราวุธวิทยาลัย
 เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์ของมหาวิทยาลัยราชภัฏวชิราวุธวิทยาลัย
 ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Dim Day As String
Dim Month As String
Dim Year As String
Dim Hour As String
Dim Minutes As String
Dim Qua As String
Dim TempStr As String
```

```
Day = DropDownList1.SelectedValue
```

```
Month = DropDownList2.SelectedValue
```

```
Year = DropDownList3.SelectedValue
```

```
Hour = DropDownList4.SelectedValue
```

```
Minutes = DropDownList5.SelectedValue
```

```
Qua = DropDownList6.SelectedValue
```

```
'ListBox1.Items.Remove(ListBox1.SelectedItem.Value)
```

```
'TempStr = "[Y_" + Day + "/" + Month + "/" + Year + "_" + Hour + ":" + Minutes  
+ "_" + Qua + "g"
```

```
'ListBox1.Items.Add(TempStr)
```

```
'threading
```

```
'authen
```

```
End If
```

```
'Elseif TestDate < dtmtest Then
```

```
End If
```

เอกสารนี้เป็นเอกสารตัวอย่างไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Next i
```

End Sub

Protected Sub ListBox1_SelectedIndexChanged(sender As Object, e As EventArgs) Handles ListBox1.SelectedIndexChanged

End Sub

End Class

โค้ดที่ใช้เขียนในโปรแกรมอาร์ดูโนเพื่อบังคับเซอร์โวและโซลินอยด์และเชื่อมต่อกับโปรแกรม Visual Studio 2012 ผ่าน Serial Port

```
#include <Servo.h>
#include <stdlib.h>
#include <stdio.h>

boolean swOnOff = true;
Servo myservo;
Servo myservo1;
int contador = 511;
int contador1 = 511; // analog pin used to connect the potentiometer
int val;
int val1; // variable to read the value from the analog pin
String inData;
int inc;
int i;
int a;

void setup()
{
  pinMode(5, OUTPUT);
  myservo.attach(9);
  myservo1.attach(8); // attaches the servo on pin 9 to the servo object
  Serial.begin(9600);
}
```

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อใช้ในการศึกษาและวิจัยเท่านั้น ไม่สามารถนำเอกสารนี้ไปใช้ในการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น

เปล่งเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

void loop()
{
  digitalWrite(5, HIGH);
  while (Serial.available()>0)
  {
    //      for (i=0; i<5; i++ )
    //      {
    //          inData[i] = Serial.read();
    //      }

```

```

char c = Serial.read();
char d = Serial.read();
char e = Serial.read();
char f = Serial.read();
char g = Serial.read();

char teststr = '9';
//      inc = atoi(teststr) * 100 ;
//inc = (f-48)*1000;
if (g < 48)
{
  inc = (f-48)*1000;
}else{
  inc = ((f-48)*10000) + ((g-48)*1000);
}

//          inc = inc * 10;
//          inData = c+"000";

```

```

//if (inData [0] == '4')

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น //อีกทั้งห้ามมิให้ (d != ':') เนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

//          {

```

```
//          check = ((int)c*10) + ((int)d);
//          }
```

```
if (c == '4')
{
//if (inData[1]==':')
//          if (d== ':')
//          {
//          for (a=2; a<3; a++)
//          {
//          if (inData [a] != NULL)
//          {
//          //a = (int)inData[a] ;
//          a = (int)inData[a] *100;
//          }
//          }
digitalWrite(5, LOW);
//c*=1000;
delay(inc);
//          delay(7000);
swOnOff = false;
//          }
//else if (inData[0] == '0') //D
} else
{ if (c == '0')
{
contador = contador - 50;
(contador < 50) contador = 50;
```

```
// Establecemos el valor de la posicion a 50°
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น

```
val = map(contador, 0, 1023, 0, 179); // Establecemos la relación entre los  
grados de giro y el PWM
```

```

/* Con el mapa de valores anterior establecemos una relacion proporcional */
/* entre el pulso minimo 0 con el grado minimo, 0 tambien y el pulso maximo*/
/* 1023 con el grado maximo, 179*/
myservo.write(val); // Escribimos la posicion con el mapa de valores al servo
delay(150);

//else if (inData[0] == '1') //TOP
} else if (c == '1') {
contador = contador + 50; // Establecemos el valor de la posicion a 10°
if (contador > 1500) contador = 1500;
val = map(contador, 0, 1023, 0, 179); // Establecemos la relacion entre los
grados de giro y el PWM
/* Con el mapa de valores anterior establecemos una relacion proporcional */
/* entre el pulso minimo 0 con el grado minimo, 0 tambien y el pulso maximo*/
/* 1023 con el grado maximo, 179*/
myservo.write(val); // Escribimos la posicion con el mapa de valores al servo
delay(150);

//else if (inData[0] == '2') // L
} else if (c == '2') {
contador1 = contador1 - 50; // Establecemos el valor de la posicion a 10°
if (contador1 < 50) contador1 = 50;
val1 = map(contador1, 0, 1023, 0, 179); // Establecemos la relacion entre los
grados de giro y el PWM
/* Con el mapa de valores anterior establecemos una relacion proporcional */
/* entre el pulso minimo 0 con el grado minimo, 0 tambien y el pulso maximo*/
/* 1023 con el grado maximo, 179*/
myservo1.write(val1); // Escribimos la posicion con el mapa de valores al servo
delay(150);

```

เอกสารนี้เป็นเอกสาร //else if (inData[0] == '3') // R การศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น } else if (c == '3') { แปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
contador1 = contador1 + 50; // Establecemos el valor de la posicion a 10°

```

if (contador1 > 1500) contador1 = 1500;
val1 = map(contador1, 0, 1023, 0, 179); // Establecemos la relacion entre los
grados de giro y el PWM
/* Con el mapa de valores anterior establecemos una relacion proporcional */
/* entre el pulso minimo 0 con el grado minimo, 0 tambien y el pulso maximo*/
/* 1023 con el grado maximo, 179*/
myservo1.write(val1); // Escribimos la posicion con el mapa de valores al servo
delay(150);
}
}
//      else if (c == '4'+Time') // โสลินอยด์
//      {
//          //int val = Serial.read();
//          // digitalWrite(5, HIGH);
//          // swOnOff = false;
//          digitalWrite(5, LOW);
//          c*=1000;
//          delay(inc);
////          delay(3000);
//          swOnOff = false;
//      }
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้