สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

หุ่นยนต์สำรวจและกู้ระเบิด

EXPLOSIVE SEARCH AND SALVATION ROBOT

T117391

ชนกฤษณ์ ธัญสุวรรณกุล

พุฒิ ศรีพิริยะวงศ์

มงคล พัทมาศรี

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เลขหมู่
เลขทะเบียน 117391
วัน,เดือน,ปี 1 ส.ค. 2554

b 12343274
i

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2553

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2553

สาขาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง หุ่นยนต์สำรวจและกู้ระเบิด

Explosive Search and Salvation Robot

ผู้จัดทำ

1. ชนกฤษณ์	ธัญสุวรรณกุล	รหัสนักศึกษา	50010293
2. มงคล	พัทมาศรี	รหัสนักศึกษา	50011204
3. พุฒิ	ศรีพิริยะวงศ์	รหัสนักศึกษา	50011131

_____อาจารย์ที่ปรึกษา

(ดร.ปกรณ์ วัฒนจตุรพร)

_____อาจารย์ที่ปรึกษา

(ดร.วัชระ ฉัตรวิริยะ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หุ่นยนต์สำรวจและกู้ระเบิด

นายชนกฤษณ์	ธัญสุวรรณกุล	50010293
นายพุฒิ	ศรีพิริยะวงศ์	50011131
นายมงคล	พัทธมาศรี	50011204
ดร.ปกรณ์	วัฒนจตุรพร	อาจารย์ที่ปรึกษา
ดร.วัชระ	ฉัตรวิริยะ	อาจารย์ที่ปรึกษาร่วม

ปีการศึกษา 2553

บทคัดย่อ

งานทางด้านการสำรวจเก็บกู้ระเบิด เป็นงานที่เสี่ยงต่อชีวิตมนุษย์เป็นอย่างมาก ทางกลุ่มของข้าพเจ้าจึงทำโครงงานหุ่นยนต์สำรวจและกู้ระเบิดขึ้น เพื่อให้หุ่นยนต์สามารถทำหน้าที่สำรวจและกู้ระเบิดได้โดยมนุษย์เป็นผู้ควบคุมระยะไกล โดยที่มนุษย์ควบคุมผ่านโทรศัพท์มือถือระบบปฏิบัติการแอนดรอย์ (Android) ผ่านการติดต่อสื่อสารแบบไวเลสแลน (Wireless LAN) ไปยังตัวหุ่นยนต์

ในการออกแบบและพัฒนานั้นได้แบ่งเป็นสามส่วนด้วยกัน ส่วนแรกคือ ส่วน ติดต่อกับผู้ใช้งาน ในส่วนนี้จะมีปุ่มสำหรับควบคุมหุ่นยนต์เพื่อควบคุมการเคลื่อนไหวของหุ่นยนต์ในแบบต่างๆ นอกจากนี้ ยังมีปุ่มสำหรับควบคุมกล้องไอพีแคม (IPCam) ซึ่งติดตั้งไว้ที่ตัวหุ่นยนต์ และยังมีส่วนสำหรับแสดงผลภาพจากกล้องไอพีแคม ทำให้สามารถเห็นระเบิด และสภาวะแวดล้อมในขณะนั้นได้ ส่วนที่สอง คือส่วนสำหรับการติดต่อ ในที่นี้เราใช้ยูดีพี (UDP) ในการติดต่อสื่อสารผ่าน Wireless Lan เนื่องจากต้องการความรวดเร็วในการควบคุมหุ่นยนต์ และต้องการที่จะสามารถควบคุมระยะไกลได้ ส่วนสุดท้ายเป็นส่วนของหุ่นยนต์ ซึ่งกลุ่มของข้าพเจ้าได้ใช้ไวเลสเราเตอร์ (Wireless Router) เป็นตัวรับข้อมูลจากส่วนที่หนึ่ง จากนั้นก็นำข้อมูลที่ได้รับจากส่วนที่หนึ่งส่งไปยังไมโครคอนโทรลเลอร์ (Microcontroller) เพื่อควบคุมหุ่นยนต์ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Explosive Search and Salvation Robot

Mr.Chanakrit	Thansuwankul	50010293
Mr.Putti	Srepiriyavongse	50011131
Mr.Mongkol	pattamasi	50011204
Dr.Pakorn	Watanachaturapol	Advisor
Dr.Watchara	Chatwiriya	Co-Advisor

Academic Year 2010

ABSTRACT

Explosive search and salvation is a risk business for human. An explosive search and salvation robot is developed to perform the tasks. The robot is remotely controlled by a human using an Android mobile phone. The phone is connected to the robot via a wireless LAN.

The design and development of the robot consists of three parts. The first part is the user interface; it contains buttons to control movement of the robot, control buttons for the IP camera on the robot, and the real time video display from the camera. The second part is a communication component. This design utilized a UDP protocol to communication via a wireless LAN to the robot so that the robot can be communicated and controlled remotely with only short delay. The last part is the robot itself. The microcontroller installed in the robot received commands from the user via a wireless router.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ทางกลุ่มของข้าพเจ้าขอขอบพระคุณคุณพ่อ และคุณแม่ เป็นอย่างสูงที่ได้ให้การสนับสนุนมาตลอดในทุกๆด้าน โดยเฉพาะในด้านการศึกษา ตลอดจนถึงระดับปริญญาตรี และ ปริญญานิพนธ์เล่มนี้

ปริญญานิพนธ์เล่มนี้จะสำเร็จไม่ได้เลยหากไม่ได้รับการคำปรึกษาจาก ดร.ปกรณ์ วัฒนจตุรพร และ ดร.วัชระ ฉัตรวิริยะ ผู้ที่ได้ให้คำปรึกษาทั้งหมดเกี่ยวกับวิทยานิพนธ์ ทั้งในด้านเทคนิค แนวทางการแก้ปัญหา รวมถึงคำแนะนำต่างๆ ตลอดมา ซึ่งกลุ่มของข้าพเจ้าขอขอบพระคุณเป็นอย่างสูง นอกจากนี้ความรู้พื้นฐานในการทำปริญญานิพนธ์นั้นเป็นสิ่งจำเป็นอย่างมาก กระผมจึงขอขอบพระคุณอาจารย์ที่ได้พร่ำสอนวิชาการต่างๆในด้านคอมพิวเตอร์ตลอดระยะเวลาสี่ปี รวมถึง วิชาการในด้านอื่นๆ ตั้งแต่กลุ่มของข้าพเจ้าได้มีโอกาสศึกษาตลอดมา

ชนกฤษณ์ ธัญสุวรรณกุล

พุฒิ ศรีพิริยะวงศ์

มงคล พัทธมาศรี

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย I

บทคัดย่อภาษาอังกฤษ II

กิตติกรรมประกาศ III

สารบัญ IV

สารบัญรูปภาพ VI

บทที่ 1 บทนำ

1.1 ที่มาและปัญหา 1

1.2 จุดประสงค์ของโครงง 2

1.3 ประโยชน์ที่คาดว่าจะได้รับ 2

1.4 ขอบเขตของโครงงาน 2

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง

2.1 ความหมายของหุ่นยนต์กู้ภัย 3

2.2 ตัวอย่างของหุ่นยนต์กู้ภัย 3

2.3 หลักการเชื่อมต่อแบบ TCP/IP 6

2.4 Encapsulation/Demultiplexing 6

2.5 โครงสร้าง TCP/IP 8

2.6 IP (Internet Protocol) 9

2.7 ICMP (Internet Control Message Protocol) 10

2.8 ชั้นสื่อสารนำส่งข้อมูล (Transport Layer) 11

2.9 Google Android 17

2.10 Wellcom A88 19

2.11 Arduino 20

บทที่ 3 การพัฒนาและการออกแบบ

3.1 ฮาร์ดแวร์ 22

3.2 การพัฒนา Android ด้วย Eclipse 38

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 Connection ระหว่าง Android กับหุ่นยนต์ 30
3.4 Connection ระหว่าง Android กับ IP Cam 32
3.5 การออกแบบส่วนต่างๆของหุ่นยนต์ 33
3.6 การออกแบบระบบ 34
3.7 ความต้องการระบบ 35
บทที่ 4 ผลการทดลอง
4.1 การทดลองระยะทางที่สามมารถควบคุมหุ่นยนต์ได้ 36
4.2 การทดลองเรื่องความสามารถในการหยิบจับวัตถุรูปทรงต่าง 36
บทที่ 5 สรุป
5.1 สรุป 38
5.2 ผลที่ได้รับ 38
5.3 ปัญหาและอุปสรรค 38
5.4 แนวทางการพัฒนาต่อ 38
บรรณานุกรม 39

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูป	หน้า
2.1 Robot PackBot	1
2.2 แสดงการเคลื่อนที่ของ PackBot	4
2.3 TALON-SWORDS	5
2.4 แสดง Chaos Robot	5
2.5 RAPTOR-II Robot	6
2.6 ขั้นตอนการ Encapsulation และ Demultiplexing	7
2.7 โครงสร้าง TCP/IP	8
2.8 IP Header	9
2.9 ICMP Header	11
2.10 UDP Header	12
2.11 TCP Header	13
2.12 สื่อสารของ TCP	14
2.13 การส่ง Reject	16
2.14 Google Android	17
2.15 แต่ละ Layer ของระบบปฏิบัติการAndroid	18
2.16 Wellcom A 88	20
2.17 Arduino Duemilanove	21
2.18 Arduino IDE	22
3.1 สถาปัตยากรรมของไมโครคอนโทรลเลอร์	24
3.2 AVR Architecture	26
3.3 Single level pipelining.	27
3.4 กลุ่มของ AVR	27
3.5 TinyAVR	28
3.6 MegaAVR-ATmega series	28
3.7 XMEGA-ATxmega series	29
3.8 Application specific AVR	29
3.9 Code ส่วนที่เป็น JAVA	30
3.10 Code ส่วนที่เป็น XML	31

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.11 การควบคุมหุ่นยนต์ได้จากระยะไกล 32
3.12 TCP/IP model 32
3.13 โครงสร้าง IPv4 33
3.14 UDP Datagram 33
3.15 Datagram 34
3.16 แขนกลจากการออกแบบบน Solidwork 35
3.17 ตัวหุ่นยนต์ 35
3.18 หุ่นยนต์ที่ประกอบแขนเสร็จแล้ว 36
3.19 การออกแบบระบบควบคุมหุ่นยนต์ 36

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตาราง หน้า

4.1 การทดสอบการควบคุมระยะไกลเปรียบเทียบระหว่างที่โล่งกับที่ที่มีอาคารหรือสิ่งกีดขวาง..... 35

4.2 ความสามารถในการหยิบวัตถุรูปทรงต่างๆ 35

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ที่มาและปัญหา

ในปัจจุบันถึงแม้ว่าจะมีประชากรมนุษย์มากมาย แต่ก็ยังคงมีงานในหลายๆประเภทที่จะให้มนุษย์ทำไม่ได้ เนื่องจากหลายๆสาเหตุเช่น ความอันตรายของงาน หรือ การที่มนุษย์อยากสะดวกสบาย จึงได้มีการคิดค้นหุ่นยนต์ขึ้นมาทำแทน

ความหมายของหุ่นยนต์

(A robot reprogrammable, multifunctional manipulator designed to move materials, parts, tools or specialized devices through various programmed motions for the performance of a variety of tasks.)

ความหมายของ "หุ่นยนต์" โดยสถาบันหุ่นยนต์อเมริกา (The Robotics Institute of America) ได้ให้ความหมายไว้ ดังนี้"หุ่นยนต์ คือเครื่องจักรใช้งานแทนมนุษย์ ที่ออกแบบให้สามารถตั้งลำดับการทำงาน การใช้งานได้หลากหลายหน้าที่ ใช้เคลื่อนย้ายวัสดุอุปกรณ์ ส่วนประกอบต่าง ๆ เครื่องมือหรืออุปกรณ์พิเศษ ตลอดจนการเคลื่อนที่ได้หลากหลาย ตามที่ตั้งลำดับการทำงาน เพื่อสำหรับใช้ในงานหลากหลายประเภท"

นิยามดังกล่าว อีกนัยหนึ่งก็คือ เครื่องจักรกลทุกชนิดที่สามารถปฏิบัติงานแทนมนุษย์ได้ทุกประเภท ทั้งทางตรงและทางอ้อม รวมทั้งในงานที่เสี่ยงอันตรายโดยที่มนุษย์ไม่สามารถปฏิบัติงานได้ ตลอดจนการทำงานที่เป็นอัตโนมัติโดยตนเองหรือถูกควบคุมโดยมนุษย์ และสามารถปรับเปลี่ยนรูปแบบการทำงานได้หลากหลาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 จุดประสงค์ของโครงงาน

1) ศึกษาความรู้เกี่ยวกับหุ่นยนต์สำรวจและเก็บกู้ระเบิด
2) ศึกษาความรู้เกี่ยวกับการพัฒนาบนระบบปฏิบัติการ Android
3) นำความรู้เกี่ยวกับการพัฒนาบนระบบปฏิบัติการ Android มาประยุกต์ใช้กับหุ่นยนต์สำรวจและเก็บกูระเบิด

1.3 ประโยชน์ที่คาดว่าจะได้รับ

1) ความรู้เรื่องโครงสร้างของหุ่นยนต์
2) ความรู้เรื่องการเขียน Application บน Android
3) ความรู้เรื่องการเขียนการเชื่อมต่อ ระหว่าง Android กับ หุ่นยนต์

1.4 ขอบเขตของโครงงาน

1) ศึกษาความรู้เกี่ยวกับหุ่นยนต์สำรวจและเก็บกู้ระเบิด
2) ออกแบบการเชื่อมต่อระหว่าง Android กับ หุ่นยนต์ได้
3) ออกแบบโครงสร้างของหุ่นยนต์
4) เขียน Interface บน Remote Control บน Android

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

2.1 ความหมาย ของหุ่นยนต์กู้ภัย (Robot Rescue)

หุ่นยนต์กู้ภัยเป็นหุ่นยนต์ที่ได้รับการออกแบบเพื่อวัตถุประสงค์ในการช่วยงานกู้ภัย เพื่อต้องการลดความเหนื่อยล้าของบุคลากร ลดความเสี่ยงที่จะเกิดขึ้นกับบุคลากร และเข้าถึงในพื้นที่ที่บุคลากรไม่สามารถเข้าถึงได้ โดยหุ่นยนต์ที่ผู้จัดทำสนใจนั้น เป็นหุ่นยนต์ที่มีใช้ในการกู้วัตถุระเบิด ซึ่งมีตัวอย่างของหุ่นยนต์กู้วัตถุระเบิดที่สามารถใช้งานได้จริงดังต่อไปนี้

2.2 ตัวอย่างของหุ่นยนต์กู้ภัย

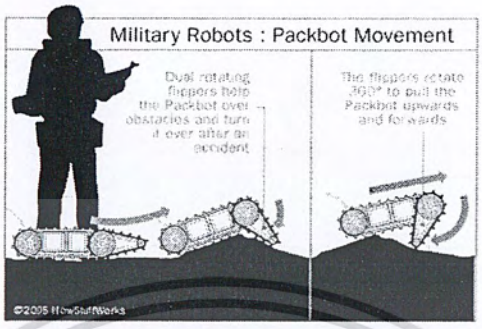
2.2.1 Pacbot

รูป 2.1 Robot PackBot

PackBot เป็นหุ่นยนต์เอนกประสงค์ของกองกำลังสหรัฐ มันถูกใช้งานครั้งแรกในอัฟกานิสถาน และใช้ต่อมาในสงครามอิรัก ทหารจะใช้มันเดินเข้าไปสำรวจถ้ำ บ้าน แหล่งหลบซ่อน เคลียร์ระเบิด ไปจนถึงกับติดปืนกลเพื่อออกไประดมยิงใส่ข้าศึกในที่หลบซ่อนไม่เห็นตัว มันมีน้ำหนักเพียง 18 กิโลกรัม ติด GPS เข็มทิศอิเล็กทรอนิกส์ เซ็นเซอร์วัดอุณหภูมิ และระบบคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

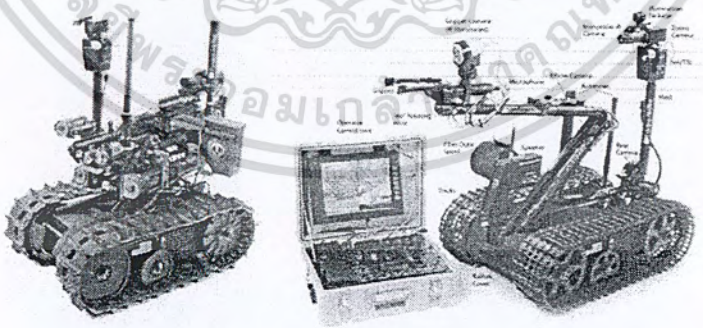
เพนเทียม ซึ่งต่อภายนอกด้วย USB และ Ethernet PackBot สามารถถูกโปรแกรมให้ทำงานเป็นฝูงด้วยตัวเองแบบกึ่งอัตโนมัติ มันจะทำงานประสานกัน คุยติดต่อกันเองและกับมนุษย์ เพื่อเข้าโอบล้อมเป้าหมายและโจมตีข้าศึก



รูป 2.2 แสดงการเคลื่อนที่ของ PackBot

2.2.2 Talon-Swirds

Talon-swords เป็นเป็นฐานเคลื่อนที่ด้วยสานพานคู่ มุ่งเน้นการตรวจสอบและปฏิบัติงานแทนคนในการค้นหาวัตถุอันตราย จึงได้ติดตั้งแขนกล 3 องศาอิสระพร้อมปากคีบ และกล้องไว้บนฐานเคลื่อนที่ เพื่อให้สามารถทำงานแทนคนในระยะไกลได้ นอกจากนั้นแล้ว TALON-SWORDS สามารถติดตั้งอุปกรณ์เพิ่มเติมได้ เช่น เครื่องฉายรังสี X-ray ปืนยิงน้ำแรงดันสูง สำหรับการเก็บกู้วัตถุระเบิด และปืนพก เป็นต้น TALON-SWORDS มีน้ำหนัก 55 kg และมีความเร็วสูงสุด 8 km/hr รูปที่ 13 แสดงลักษณะฐานเคลื่อนที่ TALON-SWORDS

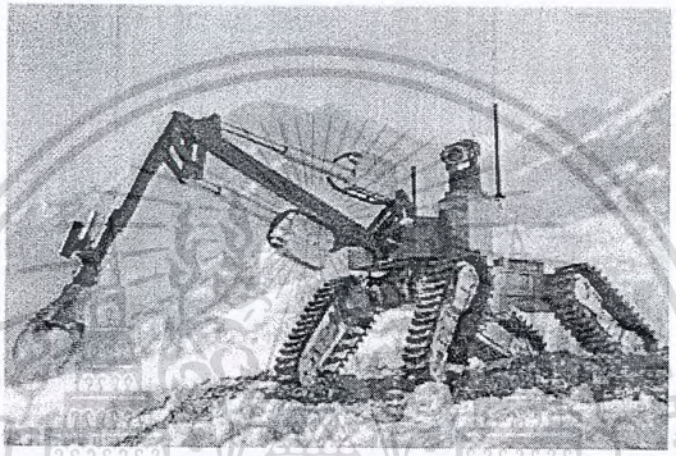


รูป 2.3 TALON-SWORDS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3 Chaos

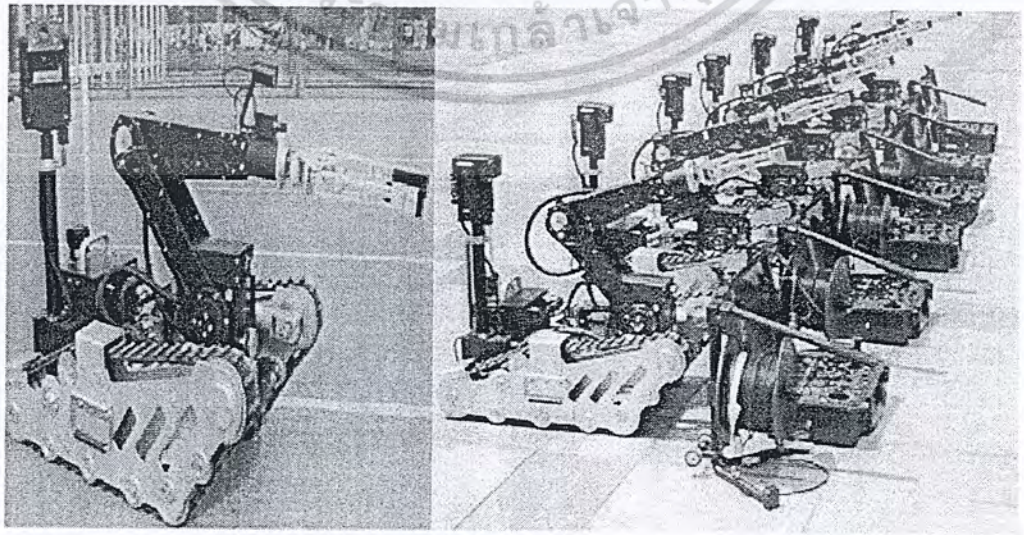
Chaos เป็นเป็นฐานเคลื่อนที่ด้วยสานพาน 4 ชุดอิสระจากกัน มุ่งเน้นการตรวจสอบและปฏิบัติงานแทนคนในการค้นหาวัตถุอันตราย จึงได้ติดตั้งแขนกล 3 องศาอิสระพร้อมปากคีบ และกล้องวงจรปิดไว้บนฐานเคลื่อนที่ เพื่อให้สามารถทำงานแทนคนในระยะไกลได้ นอกจากนั้นแล้ว ชุดขับเคลื่อนของฐานเคลื่อนที่แต่ละชุด สามารถปรับยกขึ้น-ลงได้อิสระจากกัน ทำให้หุ่นยนต์ Chaos สามารถเคลื่อนที่ให้พื้นที่ ที่มีสิ่งกีดขวางได้อย่างดีเยี่ยม Chaos มีน้ำหนัก 55 kg และมีความเร็วสูงสุด 8 km/hr รูปที่ 14 แสดงลักษณะฐานเคลื่อนที่ Chaos



รูป 2.4 แสดง Chaos Robot

2.2.4 Raptor-II

Rapter-II เป็นฐานเคลื่อนที่ด้วยสานพานคู่ และมีแขนกล 4 องศาอิสระพร้อมปากคีบเป็นอุปกรณ์มาตรฐานติดตั้งอยู่บนฐานเคลื่อนที่ มีน้ำหนัก 80 kg และมีความเร็วสูงสุด 1.44 km/hr รูปที่ 9 แสดงลักษณะฐานเคลื่อนที่ RAPTOR-II



รูป 2.5 RAPTOR-II Robot

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 หลักการการเชื่อมต่อแบบ TCP/IP Protocol

TCP/IP (Transmitsion Control Protocol/Internet Protocol) เป็นชุดของโปรโตคอลที่ถูกใช้ในการสื่อสารผ่านเครือข่ายอินเทอร์เน็ต โดยมีวัตถุประสงค์เพื่อให้สามารถใช้สื่อสารจากต้นทางข้ามเครือข่ายไปยังปลายทางได้ และสามารถหาเส้นทางที่จะส่งข้อมูลไปได้เองโดยอัตโนมัติ ถึงแม้ว่าในระหว่างทางอาจจะผ่านเครือข่ายที่มีปัญหา โปรโตคอลก็ยังคงหาเส้นทางอื่นในการส่งผ่านข้อมูลไปให้ถึงปลายทางได้ ชุดโปรโตคอลนี้ได้รับการพัฒนามาตั้งแต่ปี 1960 ซึ่งถูกใช้เป็นครั้งแรกในเครือข่าย ARPANET ซึ่งต่อมาได้ขยายการเชื่อมต่อไปทั่วโลกเป็นเครือข่ายอินเตอร์เน็ต ทำให้ TCP/IP เป็นที่ยอมรับอย่างกว้างขวางจนถึงปัจจุบัน

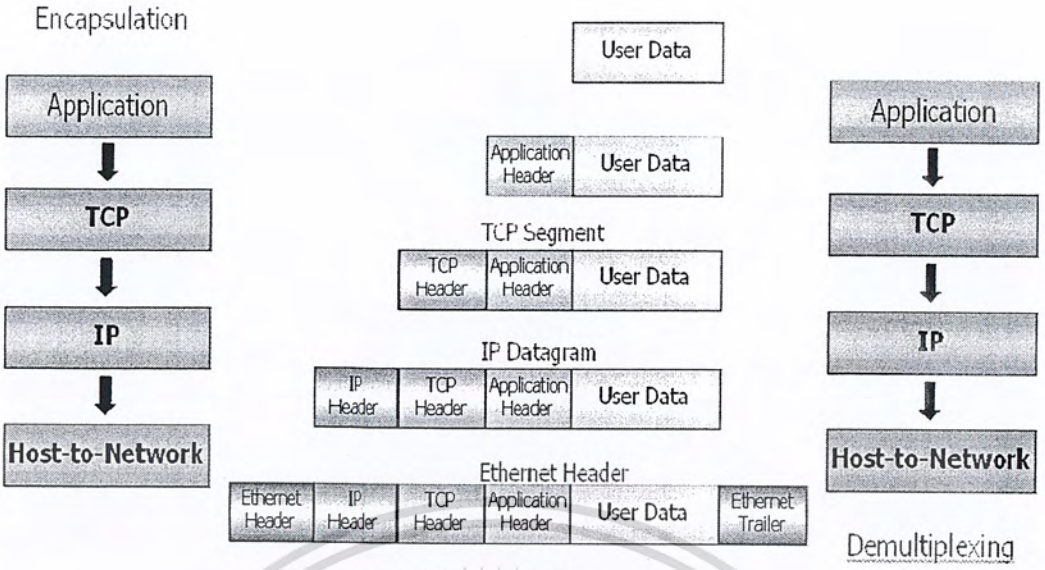
TCP/IP มีจุดประสงค์ของการสื่อสารตามมาตรฐาน สามประการคือ

1) เพื่อใช้ติดต่อสื่อสารระหว่างระบบที่มีความแตกต่างกัน
2) ความสามารถในการแก้ไขปัญหาที่เกิดขึ้นในระบบเครือข่าย เช่นในกรณีที่ผู้ส่งและผู้รับยังคงมีการติดต่อกันอยู่ แต่โหนดกลางที่ใช้เป็นผู้ช่วยรับ-ส่งเกิดเสียหายใช้การไม่ได้ หรือสายสื่อสารบางช่วงถูกตัดขาด กฎการสื่อสารนี้จะต้องสามารถจัดหาทางเลือกอื่นเพื่อทำให้การสื่อสารดำเนินต่อไปได้โดยอัตโนมัติ
3) มีความคล่องตัวต่อการสื่อสารข้อมูลได้หลายชนิดทั้งแบบที่ไม่มีความเร่งด่วน เช่น การจัดส่งแฟ้มข้อมูล และแบบที่ต้องการรับประกันความเร่งด่วนของข้อมูล เช่น การสื่อสารแบบ real-time และทั้งการสื่อสารแบบเสียง (Voice) และข้อมูล (data)

2.4 Encapsulation/Demultiplexing

การส่งข้อมูลผ่านในแต่ละเลเยอร์ แต่ละเลเยอร์จะทำการประกอบข้อมูลที่ได้รับมา กับข้อมูลส่วนควบคุมซึ่งถูกนำมาไว้ในส่วนหัวของข้อมูลเรียกว่า Header ภายใน Header จะบรรจุข้อมูลที่สำคัญของโปรโตคอลที่ทำการ Encapsulate เมื่อผู้รับได้รับข้อมูล ก็จะเกิดกระบวนการทำงานย้อนกลับคือ โปรโตคอลเดียวกัน ทางฝั่งผู้รับก็จะได้รับข้อมูลส่วนที่เป็น Header ก่อนและนำไปประมวลและทราบว่าข้อมูลที่ตามมามีลักษณะอย่างไร ซึ่งกระบวนการย้อนกลับนี้เรียกว่า Demultiplexing

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.6 ขั้นตอนการ Encapsulation และ Demultiplexing

2.4.1 ข้อมูลที่ผ่านการ Encapsulate ในแต่ละเลเยอร์มีชื่อเรียกแตกต่างกัน ดังนี้

1) ข้อมูลที่มาจาก User หรือก็คือข้อมูลที่ User เป็นผู้ป้อนให้กับ Application เรียกว่า User Data

2) เมื่อแอพพลิเคชั่นได้รับข้อมูลจาก user ก็จะนำมาประกอบกับส่วนหัวของแอพพลิเคชั่น เรียกว่า Application Data และส่งต่อไปยังโปรโตคอล TCP

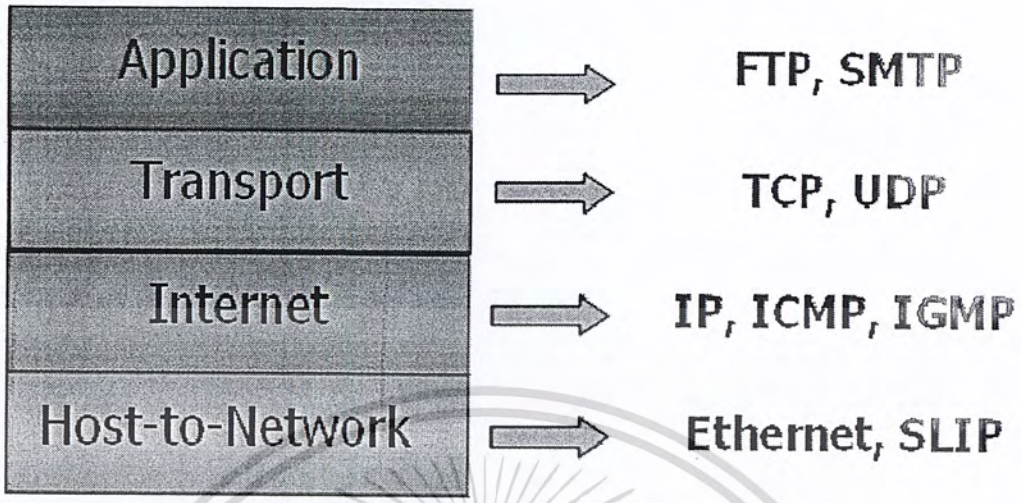
3) เมื่อโปรโตคอล TCP ได้รับ Application Data ก็จะนำมารวมกับ Header ของ โปรโตคอล TCP เรียกว่า TCP Segment และส่งต่อไปยังโปรโตคอล IP

4) เมื่อโปรโตคอล IP ได้รับ TCP Segment ก็จะนำมารวมกับ Header ของ โปรโตคอล IP เรียกว่า IP Datagram และส่งต่อไปยังเลเยอร์ Host-to-Network Layer

5) ในระดับ Host-to-Network จะนำ IP Datagram มาเพิ่มส่วน Error Correction และ flag เรียกว่า Ethernet Frame ก่อนจะแปลงข้อมูลเป็นสัญญาณไฟฟ้า ส่งผ่านสายสัญญาณที่เชื่อมโยงอยู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในแต่ละเลเยอร์ของโครงสร้าง TCP/IP สามารถอธิบายได้ดังนี้



รูป 2.7 โครงสร้าง TCP/IP

2.5 โครงสร้าง TCP/IP

2.5.1. ชั้นโฮสต์-เครือข่าย (Host-to-Network Layer)

โพรโตคอลสำหรับการควบคุมการสื่อสารในชั้นนี้เป็นสิ่งที่ไม่มีการกำหนดรายละเอียดอย่างเป็นทางการ หน้าที่หลักคือการรับข้อมูลจากชั้นสื่อสาร IP มาแล้วส่งไปยังโหนดที่ระบุไว้ในเส้นทางเดินข้อมูลทางด้านผู้รับก็จะทำงานในทางกลับกัน คือรับข้อมูลจากสายสื่อสารแล้วนำส่งให้กับโปรแกรมในชั้นสื่อสาร

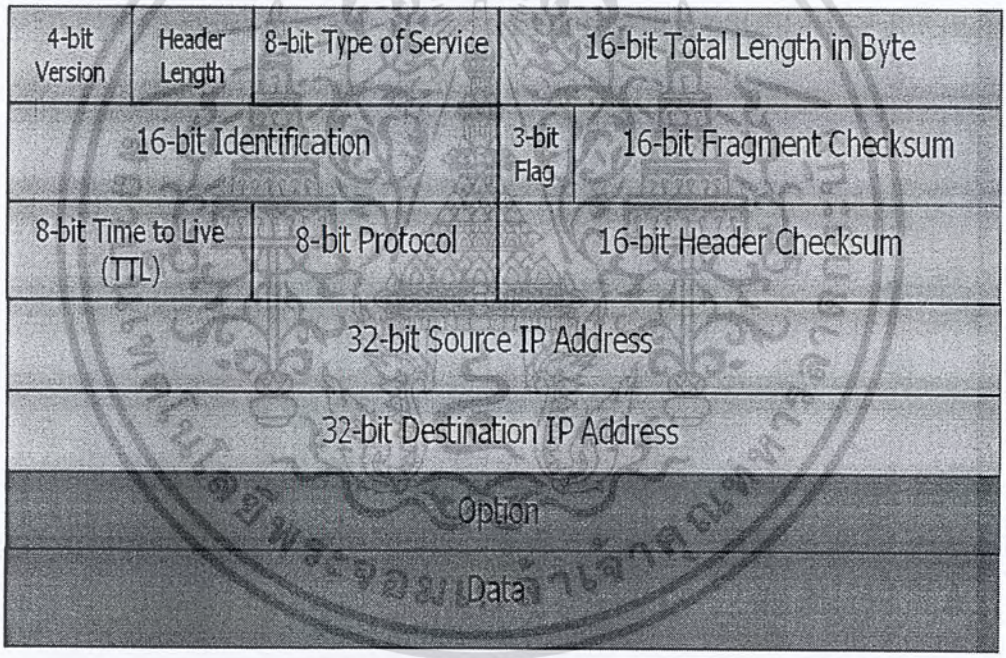
2.5.2. ชั้นสื่อสารอินเทอร์เน็ต (The Internet Layer)

ใช้ประเภทของระบบการสื่อสารที่เรียกว่า ระบบเครือข่ายแบบสลับช่องสื่อสารระดับแพ็กเก็ต (packet-switching network) ซึ่งเป็นการติดต่อแบบไม่ต่อเนื่อง (Connectionless) หลักการทำงานคือการปล่อยให้ข้อมูลขนาดเล็กที่เรียกว่า แพ็กเก็ต (Packet) สามารถไหลจากโหนดผู้ส่งไปตามโหนดต่างๆ ในระบบจนถึงจุดหมายปลายทางได้โดยอิสระ หากว่ามีการส่งแพ็กเก็ตออกมาเป็นชุดโดยมีจุดหมายปลายทางเดียวกันในระหว่าง การเดินทางในเครือข่าย แพ็กเก็ตแต่ละตัวในชุดนี้ก็จะเป็นอิสระแก่กันและกัน ดังนั้น แพ็กเก็ตที่ส่งไปถึงปลายทางอาจจะไม่เป็นไปตามลำดับก็ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 IP (Internet Protocol)

IP เป็นโปรโตคอลในระดับเน็ตเวิร์คเลเยอร์ ทำหน้าที่จัดการเกี่ยวกับแอดเดรสและข้อมูล และควบคุมการส่งข้อมูลบางอย่างที่ใช้ในการหาเส้นทางของแพ็กเก็ต ซึ่งกลไกในการหาเส้นทางของ IP จะมีความสามารถในการหาเส้นทางที่ดีที่สุด และสามารถเปลี่ยนแปลงเส้นทางได้ในระหว่างการส่งข้อมูล และมีระบบการแยกและประกอบดาต้าแกรม (datagram) เพื่อรองรับการส่งข้อมูลระดับ data link ที่มีขนาด MTU (Maximum Transmission Unit) ทีแตกต่างกัน ทำให้สามารถนำ IP ไปใช้บนโปรโตคอลอื่นได้หลากหลาย เช่น Ethernet ,Token Ring หรือ Apple Talkการเชื่อมต่อของ IP เพื่อทำการส่งข้อมูล จะเป็นแบบ connectionless หรือเกิดเส้นทางการเชื่อมต่อในทุกๆครั้งของการส่งข้อมูล 1 ดาต้าแกรม โดยจะไม่ทราบถึงข้อมูลดาต้าแกรมที่ส่งก่อนหน้าหรือส่งตามมา แต่การส่งข้อมูลใน 1 ดาต้าแกรม อาจจะเกิดการส่งได้หลายครั้งในกรณีที่มีการแบ่งข้อมูลออกเป็นส่วนย่อยๆ (fragmentation) และถูกนำไปรวมเป็นดาต้าแกรมเดิมเมื่อถึงปลายทาง



รูป 2.8 IP Header

เฮดเดอร์ของ IP โดยปกติจะมีขนาด 20 bytes ยกเว้นในกรณีที่มีการเพิ่ม option บางอย่าง ฟิลด์ของเฮดเดอร์ IP จะมีความหมายดังนี้

Version : หมายเลขเวอร์ชันของโปรโตคอล ที่ใช้งานในปัจจุบันคือ เวอร์ชัน 4 (IPv4) และเวอร์ชัน 6 (IPv6)

Header Length : ความยาวของเฮดเดอร์ โดยทั่วไปถ้าไม่มีส่วน option จะมีค่าเป็น 5 (5*32 bit)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Type of Service (TOS) : ใช้เป็นข้อมูลสำหรับเราเตอร์ในการตัดสินใจเลือกการเราต์ข้อมูลในแต่ละดาต้าแกรม แต่ในปัจจุบันไม่ได้มีการนำไปใช้งานแล้ว

Length : ความยาวทั้งหมดเป็นจำนวนไบต์ของดาต้าแกรม ซึ่งด้วยขนาด 16 บิตของฟิลด์ จะหมายถึงความยาวสูงสุดของดาต้าแกรม คือ 65535 byte (64k) แต่ในการส่งข้อมูลจริง ข้อมูลจะถูกแยกเป็นส่วนๆตามขนาดของ MTU ที่กำหนดในลิงค์เลเยอร์ และนำมารวมกันอีกครั้งเมื่อส่งถึงปลายทาง แอพพลิเคชั่นส่วนใหญ่จะมีขนาดของดาต้าแกรมไม่เกิน 512 byte

Identification : เป็นหมายเลขของดาต้าแกรมในกรณีที่มีการแยกดาต้าแกรมเมื่อข้อมูลส่งถึงปลายทางจะนำข้อมูลที่มี identification เดียวกันมารวมกัน

Flag : ใช้ในกรณีที่มีการแยกดาต้าแกรม

Fragment offset : ใช้ในการกำหนดตำแหน่งข้อมูลในดาต้าแกรมที่มีการแยกส่วน เพื่อให้สามารถนำกลับมาเรียงต่อกันได้อย่างถูกต้อง

Time to live (TTL) : กำหนดจำนวนครั้งที่มากที่สุดที่ดาต้าแกรมจะถูกส่งระหว่าง hop (การส่งผ่านข้อมูลระหว่างเน็ตเวิร์ค) เพื่อป้องกันไม่ให้เกิดการส่งข้อมูลโดยไม่สิ้นสุด โดยเมื่อข้อมูลถูกส่งไป 1 hop จะทำการลดค่า TTL ลง 1 เมื่อค่าของ TTL เป็น 0 และข้อมูลยังไม่ถึงปลายทาง ข้อมูลนั้นจะถูกยกเลิก และเราเตอร์สุดท้ายจะส่งข้อมูล ICMP แจ้งกลับมายังต้นทางว่าเกิด time out ในระหว่างการส่งข้อมูล

Protocol : ระบุโปรโตคอลที่ส่งในดาต้าแกรม เช่น TCP ,UDP หรือ ICMP

Header checksum : ใช้ในการตรวจสอบความถูกต้องของข้อมูลในเฮดเดอร์

Source IP address : หมายเลข IP ของผู้ส่งข้อมูล

Destination IP address : หมายเลข IP ของผู้รับข้อมูล

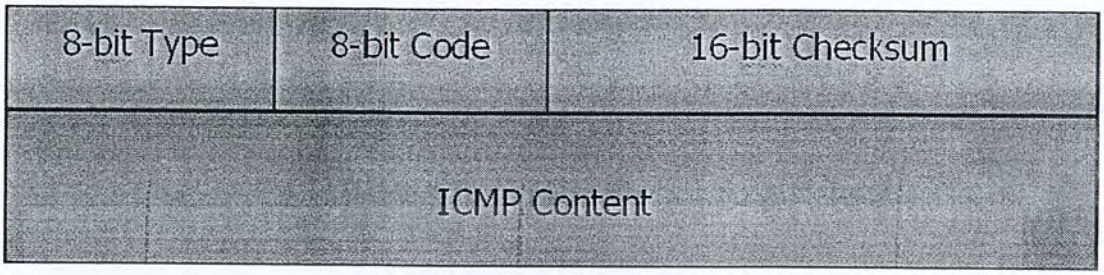
Data : ข้อมูลจากโปรโตคอลระดับบน

2.7 ICMP (Internet Control Message Protocol)

ICMP เป็นโปรโตคอลที่ใช้ในการตรวจสอบและรายงานสถานภาพของดาต้าแกรม (Datagram) ในกรณีที่เกิดปัญหากับดาต้าแกรม เช่น เราเตอร์ไม่สามารถส่งดาต้าแกรมไปถึงปลายทางได้ ICMP จะถูกส่งออกไปยังโฮสต้นทางเพื่อรายงานข้อผิดพลาด ที่เกิดขึ้น อย่างไรก็ดี ไม่มีอะไรรับประกันได้ว่า ICMP Message ที่ส่งไปจะถึงผู้รับจริงหรือไม่ หากมีการส่งดาต้าแกรมออกไปแล้วไม่มี ICMP Message ฟ้อง Error กลับมา ก็แปลความหมายได้สองกรณีคือ ข้อมูลถูกส่งไปถึงปลายทางอย่างเรียบร้อย หรืออาจจะมีปัญหา ในการสื่อสารทั้งการส่งดาต้าแกรม และ ICMP Message ที่ส่งกลับมาก็มีปัญหาระว่างทางก็ได้ ICMP จึงเป็นโปรโตคอลที่ไม่มีความน่าเชื่อถือ (unreliable) ซึ่งจะเป็นหน้าที่ของ โปรโตคอลในระดับสูงกว่า Network Layer ในการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จัดการให้การสื่อสารนั้นๆ มีความน่าเชื่อถือ ในส่วนของ ICMP Message จะประกอบด้วย Type ขนาด 8 บิต Checksum ขนาด 16 บิต และส่วนของ Content ซึ่งจะมีขนาดแตกต่างกันไปตาม Type และ Code ดังรูป



รูป 2.9 ICMP Header

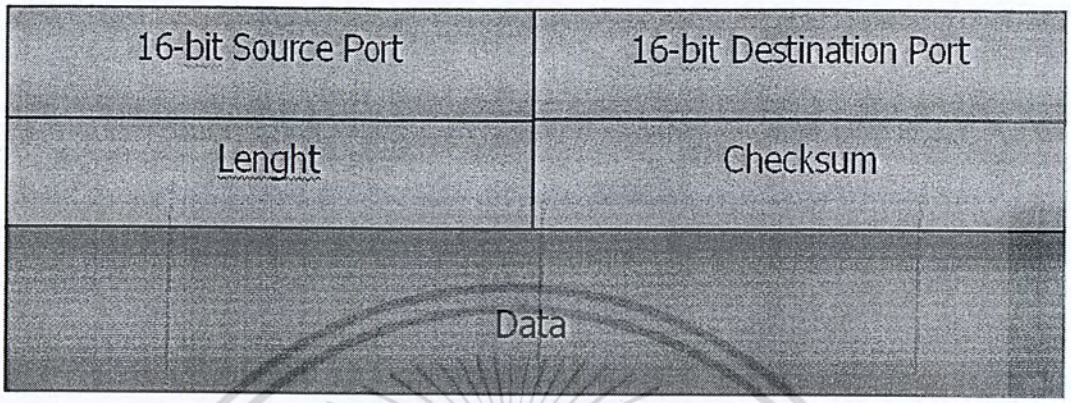
2.8 ชั้นสื่อสารนำส่งข้อมูล (Transport Layer)

แบ่งเป็นโพรโตคอล 2 ชนิดตามลักษณะ ลักษณะแรกเรียกว่า Transmission Control Protocol (TCP) เป็นแบบที่มีการกำหนดช่วงการสื่อสารตลอดระยะเวลาการสื่อสาร (connection-oriented) ซึ่งจะยอมให้มีการส่งข้อมูลเป็นแบบ Byte stream ที่ไว้ใจได้โดยไม่มีข้อผิดพลาด ข้อมูลที่มีปริมาณมากจะถูกแบ่งออกเป็นส่วนเล็กๆ เรียกว่า message ซึ่งจะถูกส่งไปยังผู้รับผ่านทางชั้นสื่อสารของอินเทอร์เน็ต ทางฝ่ายผู้รับจะนำ message มาเรียงต่อกันตามลำดับเป็นข้อมูลตัวเดิม TCP ยังมีความสามารถในการควบคุมการไหลของข้อมูลเพื่อป้องกันไม่ให้ผู้ส่ง ส่งข้อมูลเร็วเกินกว่าที่ผู้รับจะทำงานได้ทันอีกด้วยโปรโตคอลการนำส่งข้อมูลแบบที่สองเรียกว่า UDP (User Datagram Protocol) เป็นการติดต่อแบบไม่ต่อเนื่อง (connectionless) มีการตรวจสอบความถูกต้องของข้อมูลแต่จะไม่มีการแจ้งกลับไปยังผู้ส่ง จึงถือได้ว่าไม่มีการตรวจสอบความถูกต้องของข้อมูล อย่างไรก็ตาม วิธีการนี้มีข้อดีในด้านความรวดเร็วในการส่งข้อมูล จึงนิยมใช้ในระบบผู้ให้และผู้ใช้บริการ (client/server system) ซึ่งมีการสื่อสารแบบ ถาม/ตอบ (request/reply) นอกจากนั้นยังใช้ในการส่งข้อมูลประเภทภาพเคลื่อนไหวหรือการส่งเสียง (voice) ทางอินเทอร์เน็ต

2.8.1 UDP:(UserDatagramProtocol)เป็นโปรโตคอลที่อยู่ใน Transport Layer เมื่อเทียบกับโมเดล OSI โดยการส่งข้อมูลของ UDP นั้นจะเป็นการส่งครั้งละ 1 ชุดข้อมูล เรียกว่า UDP datagram ซึ่งจะไม่มีความสัมพันธ์กันระหว่างดาต้าแกรมและจะไม่มีกลไกการตรวจสอบความ สำเร็จในการรับส่งข้อมูล กลไกการตรวจสอบโดย checksum ของ UDP นั้นเพื่อเป็นการป้องกันข้อมูลที่อาจจะถูกแก้ไข หรือมีความผิดพลาดระหว่างการส่ง และหากเกิดเหตุการณ์ดังกล่าวปลายทางจะได้รู้ว่ามีข้อผิดพลาดเกิดขึ้น แต่มันจะเป็นการตรวจสอบเพียงฝ่ายเดียวเท่านั้น โดยในข้อกำหนดของ UDP หากพบว่า Checksum Error ก็ให้ผู้รับปลายทางทำการทิ้งข้อมูลนั้น แต่จะไม่มีการแจ้งกลับไปยังผู้ส่งแต่อย่างใด การรับส่งข้อมูลแต่ละครั้งหากเกิดข้อผิดพลาดในระดับ IP เช่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่งไม่ถึง, หมดเวลา ผู้ส่งจะได้รับ Error Message จากระดับ IP เป็น ICMP Error Message แต่เมื่อข้อมูลส่งถึงปลายทางถูกต้อง แต่เกิดข้อผิดพลาดในส่วนของ UDP เอง จะไม่มีการยืนยัน หรือแจ้งให้ผู้ส่งทราบแต่อย่างใด



รูป 2.10 UDP Header

มีรายละเอียด ดังนี้

Source Port Number : หมายเลขพอร์ตต้นทางที่ส่งดาต้าแกรมนี้

Destination Port Number : หมายเลขพอร์ตปลายทางที่จะเป็นผู้รับดาต้าแกรม

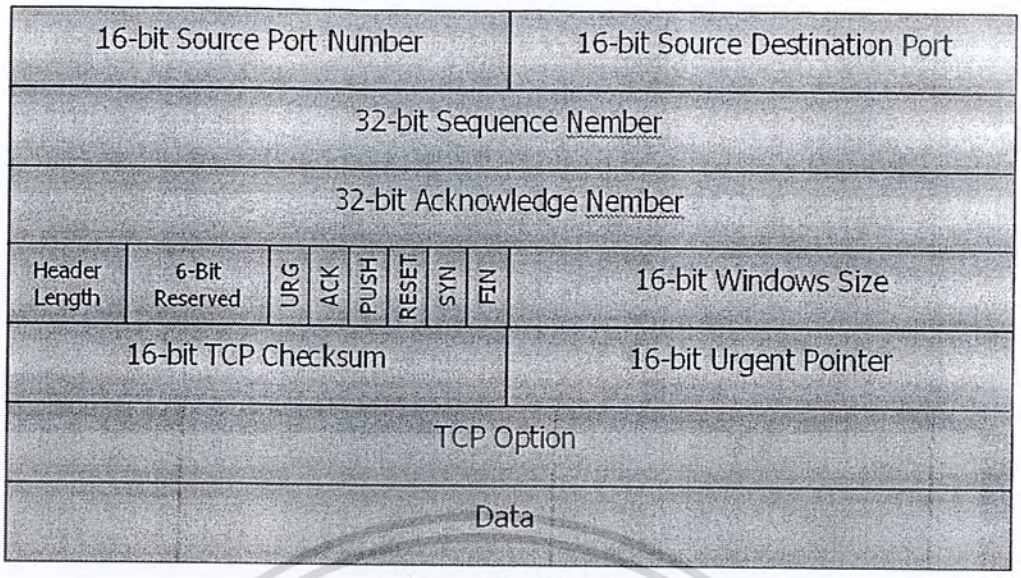
UDP Length : ความยาวของดาต้าแกรม ทั้งส่วน Header และ data นั่นหมายความว่าค่าที่น้อยที่สุดในฟิลด์นี้คือ 8 ซึ่งเป็นขนาดของ Header

Checksum : เป็นตัวตรวจสอบความถูกต้องของ UDP datagram และจะนำข้อมูลบางส่วนใน IP Header มาคำนวณด้วย

2.8.2 TCP : (Transmission Control Protocol)

อยู่ใน Transport Layer เช่นเดียวกับ UDP ทำหน้าที่จัดการและควบคุมการรับส่งข้อมูล ซึ่งมีความสามารถและรายละเอียดมากกว่า UDP โดยดาต้าแกรมของ TCP จะมีความสัมพันธ์ต่อเนื่องกัน และมีกลไกควบคุมการรับส่งข้อมูลให้มีความถูกต้อง (reliable) และมีการสื่อสารอย่างเป็นกระบวนการ (connection-oriented)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.11 TCP Header

มีรายละเอียด ดังนี้

Source Port Number : หมายเลขพอร์ตต้นทางที่ส่งดาต้าแกรมนี้

Destination Port Number : หมายเลขพอร์ตปลายทางที่จะเป็นผู้รับดาต้าแกรม

Sequence Number : ฟิลด์ที่ระบุหมายเลขลำดับอ้างอิงในการสื่อสารข้อมูลแต่ละครั้ง เพื่อใช้ในการแยกแยะว่าเป็นข้อมูลของชุดใด และนำมาจัดลำดับได้ถูกต้อง

Acknowledgment Number : ทำหน้าที่เช่นเดียวกับ Sequence Number แต่จะใช้ในการตอบรับ

Header Length : โดยปกติความยาวของเฮดเดอร์ TCP จะมีความยาว 20 ไบต์ แต่อาจจะมากกว่านั้น ถ้ามีข้อมูลในฟิลด์ option แต่ต้องไม่เกิน 60 ไบต์

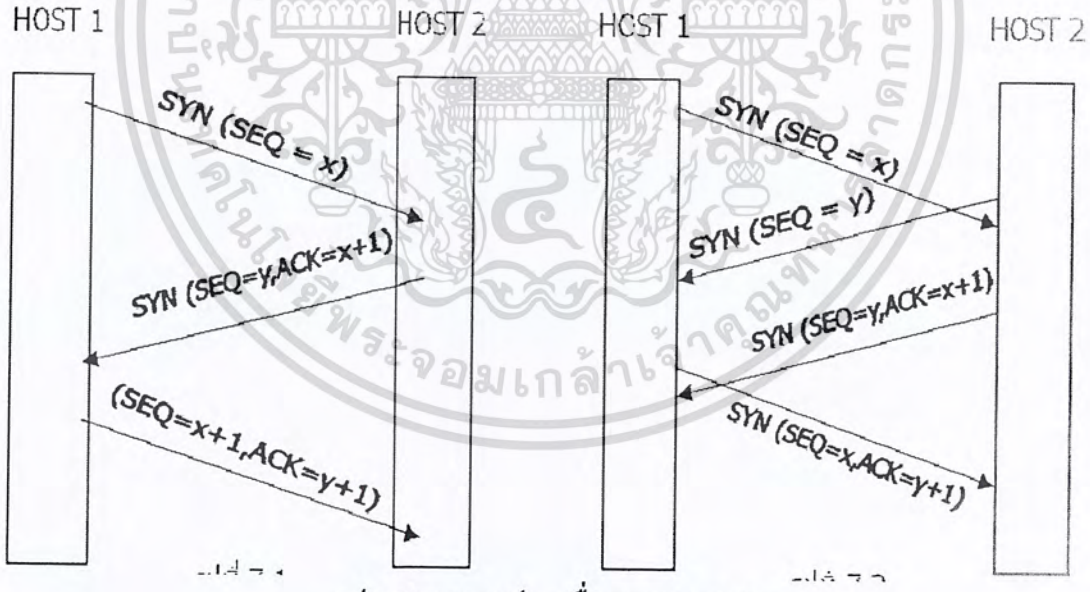
Flag : เป็นข้อมูลระดับบิตที่อยู่ในเฮดเดอร์ TCP โดยใช้เป็นตัวบอกคุณสมบัติของแพ็กเก็ต TCP ขณะนั้นๆ และใช้เป็นตัวควบคุมจังหวะการรับส่งข้อมูลด้วย ซึ่ง Flag มีอยู่ทั้งหมด 6 บิต แบ่งได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางFlag ที่มีอยู่ทั้ง6 บิต

Type	Description
URG	ใช้บอกความหมายว่าเป็นข้อมูลด่วน และมีข้อมูลพิเศษมาด้วย (อยู่ใน Urgent pointer)
ACK	แสดงว่าข้อมูลในฟิลด์ Acknowledge Number นำมาใช้งานได้
DSH	เป็นการแจ้งให้ผู้รับข้อมูลทราบว่าควรจะส่งข้อมูล Segment นี้ไปยัง Application ที่กำลังรออยู่โดยเร็ว
RST	ยกเลิกการติดต่อ (reset) เนื่องจากในกรณีที่เกิดการสับสนขึ้นด้วยเหตุผลต่างๆ เช่น โฮสต์มีปัญหา ให้เริ่มต้นสื่อสารกันใหม่
SYN	ใช้ในการเริ่มต้นขอติดต่อกับปลายทาง
FIN	ใช้ส่งเพื่อแจ้งให้ปลายทางทราบว่ายุติการติดต่อ

Flag ในเฮดเดอร์ของ TCP มีความสำคัญในการกำหนดการทำงานของ TCP segment เนื่องจากข้อมูลในเฮดเดอร์ของ TCP จะมีข้อมูลครบถ้วนทั้งการรับและการส่งข้อมูล ซึ่งในการส ทำงานแต่ละอย่างจะมีการใช้งานฟิลด์ไม่เหมือนกัน flag จะเป็นตัวกำหนดว่าให้ใช้งานฟิลด์ไหน เช่น ฟิลด์ Acknowledgment number จะไม่ถูกใช้ในขั้นตอนการเริ่มต้นการเชื่อมต่อ แต่จะมีข้อมูลในฟิลด์ ซึ่งเป็นข้อมูลที่ไม่มีความหมายใดๆ ซึ่งถ้าไม่มี flag เป็นตัวกำหนดก้ออาจจะมีการนำข้อมูลมาใช้ และก่อให้เกิดความผิดพลาดได้



รูป 2.12 แสดงรูปการสื่อสารของ TCP

เมื่อเซกเมนต์ CONNECT (SYN = "1" และ ACK = "0") เดินทางมาถึง Entity TCP ที่โฮสต์ปลายทางจะค้นหาโพรเซสตามหมายเลขพอร์ตที่กำหนดในเขตข้อมูล Destination port ซึ่งถ้าหากไม่พบก็จะตอบปฏิเสธด้วยเซกเมนต์ที่มี RST = "1" กลับไปยังผู้ส่ง เซกเมนต์ CONNECT ของผู้ส่งจะถูกส่งต่อไปยังโพรเซส ตามพอร์ตที่ระบุซึ่งอาจจะตอบรับหรือตอบปฏิเสธก็ได้ ถ้าโพรเซสนั้น

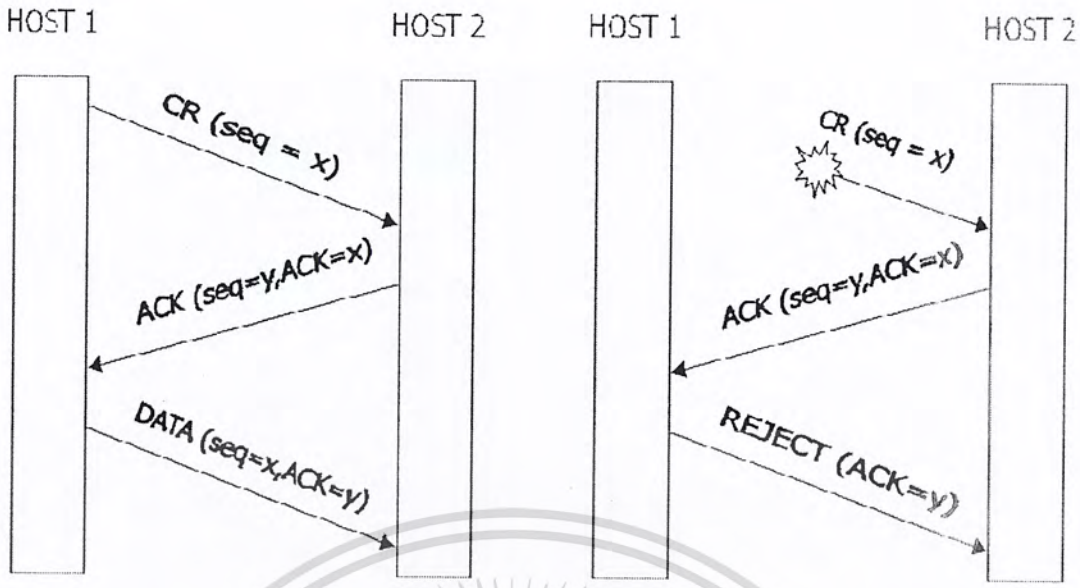
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต้องการสื่อสารด้วยก็จะส่งเซกเมนต์ตอบรับกลับไป รูปที่ 6-1 แสดงลำดับขั้นตอนการส่ง TCP เซกเมนต์ในการสร้างการเชื่อมต่อในสภาวะปกติระหว่างผู้ส่งและผู้รับในกรณีที่โฮสต์สองแห่งพยายามสร้างการเชื่อมต่อระหว่างซ็อคเก็ตคู่เดียว กันจะเกิดเป็นลำดับขั้นตอนแสดงในรูปที่ 6-2 ผลสุดท้ายจะมีการเชื่อมต่อเกิดขึ้นเพียงหนึ่งช่องทางเท่านั้นเนื่องจากการ เชื่อมต่อในแต่ละช่องทางจะถูกกำหนดขึ้นโดยใช้หมายเลขซ็อคเก็ตผู้ส่งและผู้ รับ ถ้าการเชื่อมต่อลำดับแรกสำเร็จก็จะถูกบันทึกไว้ในตารางการสื่อสาร เช่น (x, y) ถ้าการเชื่อมต่อลำดับที่สองสำเร็จในเวลาต่อมา ข้อมูลนี้ก็จะถูกบันทึกไว้ที่เดียวกันคือ (x, y)ขั้นตอนในการสร้างการเชื่อมต่อและการยกเลิกสามารถเขียนอธิบายด้วยไฟไนท์สเตทแมชชีนที่มีการทำงาน 11 สถานะ ดังแสดงในตารางข้างล่าง ในแต่ละสถานะจะมีเหตุการณ์บางอย่างที่เป็นไปได้ซึ่งจะได้รับการตอบสนองด้วยการกระทำที่เหมาะสม ในทางตรงกันข้าม เหตุการณ์ที่เป็นไปไม่ได้จะกลายเป็นข้อผิดพลาดที่จะต้องรายงานให้ทราบการเชื่อมต่อเริ่มต้นจากสถานะ CLOSED เมื่อเรียกใช้บริการ LISTEN หรือ CONNECT ก็จะมีการเปลี่ยนสถานะไปจากเดิม และถ้าอีกฝ่ายตองการเชื่อมต่อด้วย การเชื่อมต่อก็จะเกิดขึ้นและย้ายไปอยู่ในสถานะ ESTABLISHED คือการเชื่อต่อสมบูรณ์ และเมื่อยกเลิกการติดต่อก็จะกลับไปสู่สถานะ CLOSED อย่างเดิม

2.8.1.1 การเริ่มต้นการสื่อสารของ TCP แบบ Three-way handshake

Three-way Handshake เป็นวิธีการส่งแพ็กเก็ตที่สามารถช่วยแก้ปัญหาในเรื่องแพ็กเก็ตซ้ำซ้อนได้ดี แต่วิธีนี้จำเป็นจะต้องสร้างช่องสื่อสารให้ได้ก่อนที่จะเริ่มรับ-ส่งข้อมูล อย่างไรก็ตาม แพ็กเก็ตควบคุมที่ใช้ในการต่อรองค่าตัวแปรสำหรับการสื่อสารต่างๆ อาจเกิดการตกค้างอยู่ในระบบได้ ทำให้การกำหนดค่าหมายเลขลำดับมีปัญหาไปด้วย เช่นการสร้างช่องสื่อสารระหว่างโฮสต์1 และ โฮสต์2 เริ่มจาก โฮสต์1 ขอเริ่มการเชื่อต่อด้วยการส่งแพ็กเก็ต CR (Connection Request) ไปยังโฮสต์2 ซึ่งจะมีค่าตัวแปรต่างๆสำหรับการสื่อสารรวมทั้งหมายเลขลำดับและหมายเลขช่อง สื่อสารไปด้วย ผู้รับคือโฮสต์2 ก็จะส่ง ACK (Acknowledge) กลับมายังโฮสต์1 แต่ถ้าแพ็กเก็ต จากผู้ส่งเกิดสูญหายระหว่างทางและสำเนาแพ็กเก็ตที่ยังตกค้างอยู่ระบบเกิด เดินทางไปถึงผู้รับในภายหลังก็จะทำให้การสร้างช่องสื่อสารใช้การไม่ได้ เนื่องจากมีค่าตัวแปรต่างๆไม่ตรงกัน การใช้ Three-way handshake เป็นการไม่บังคับให้ผู้ส่งและผู้รับข้อมูลจะต้องกำหนดค่าเริ่มต้นของหมายเลขลำดับเป็นเลขเดียวกัน ทำให้สามารถนำวิธีนี้มาใช้ร่วมกับวิธีการจัดจังหวะการทำงานให้พร้อมกัน (Synchronization) แบบต่างๆได้ แทนที่จะเป็นการใช้วิธีการบันทึกเวลา ดังรูปที่ 7-1 แสดงขั้นตอนการเริ่มต้นการทำงานจากโฮสต์ 1 ไปยังโฮสต์ 2 สมมุติให้โฮสต์ 1 เลือกหมายเลขลำดับเป็น “x” และส่งแพ็กเก็ต CONNECTION REQUEST ไปยังโฮสต์ 2 โฮสต์ 2 ตอบรับด้วยแพ็กเก็ต CONNECTION ACCEPTED ซึ่งจะยอมรับหมายเลขลำดับ “x” พร้อมกับประกาศหมายเลขลำดับ “y” ที่เป็นของตนเอง จากนั้นโฮสต์ 1 ก็จะตอบรับค่าตัวเลือกของโฮสต์ 2 ผ่านทางเขตข้อมูลสำหรับการควบคุมในแพ็กเก็ตข้อมูลแรกที่ส่งมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.13 แสดงการส่ง Reject

สมมติว่าได้เกิดปัญหาการสูญหายของแพ็กเก็ตในขณะที่สำเนาแพ็กเก็ตที่ค้างในระบบเดินทางไปถึงผู้รับแทน รูปที่7-2 แสดงเหตุการณ์ที่แพ็กเก็ตTPDU (ตัวแรกในรูป) เป็นสำเนาแพ็กเก็ตเก่าที่พึ่งจะเดินทางไปถึงโฮสต์ 2 โดยที่โฮสต์ 1 ไม่ทราบ โฮสต์ 2 ก็จะทำงานตามปกติคือจะตอบรับด้วยการส่งแพ็กเก็ต CONNECTION ACCEPTED TPDU กลับมา ที่โฮสต์ 1 ซึ่งโฮสต์1 จะสามารถตรวจสอบได้ว่า หมายเลขลำดับโฮสต์2 ตอบกลับมานั้นเป็นหมายเลขลำดับที่ได้เลิกใช้ไปแล้ว จึงมีการส่งแพ็กเก็ต REJECTกลับมายังโฮสต์ 2 เพื่อบอกยกเลิกการทำงาน จะเห็นว่าวิธีการนี้อาศัยการสื่อสารผ่านแพ็กเก็ต 3 ตัวซึ่งเป็นที่มาของคำว่า "การจับมือร่วมสามขั้นตอน" ผลสุดท้าย ทั้งโฮสต์ 1 และโฮสต์ 2 ก็จะไม่มีการสร้างช่องสื่อสารขึ้นมาจากข้อมูลในสำเนาแพ็กเก็ตเก่าแต่อย่างใด

2.8.1.2. ชั้นสื่อสารการประยุกต์ (Application Layer)

มีโพรโตคอลสำหรับสร้างจอเทอร์มินัลเสมือน เรียกว่า TELNET โพรโตคอลสำหรับการจัดการแฟ้มข้อมูล เรียกว่า FTP และโพรโตคอลสำหรับการให้บริการจดหมายอิเล็กทรอนิกส์ เรียกว่า SMTP โดยโพรโตคอลสำหรับสร้างจอเทอร์มินัลเสมือนช่วยให้ผู้ใช้สามารถติดต่อกับ เครื่องโฮสต์ที่อยู่ไกลออกไปโดยผ่านอินเทอร์เน็ต และสามารถทำงานได้เสมือนกับว่ากำลังนั่งทำงานอยู่ที่เครื่องโฮสต์นั้น โพรโตคอลสำหรับการจัดการแฟ้มข้อมูลช่วยในการคัดลอกแฟ้มข้อมูลมาจากเครื่อง อื่นที่อยู่ในระบบเครือข่ายหรือส่งสำเนาแฟ้มข้อมูลไปยังเครื่องใดๆก็ได้ โพรโตคอลสำหรับให้บริการจดหมายอิเล็กทรอนิกส์ช่วยในการจัดส่งข้อความไปยัง ผู้ใช้ในระบบหรือรับข้อความที่มีผู้ส่งเข้ามา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

2.9 Google Android

รูป 2.14 Google Android

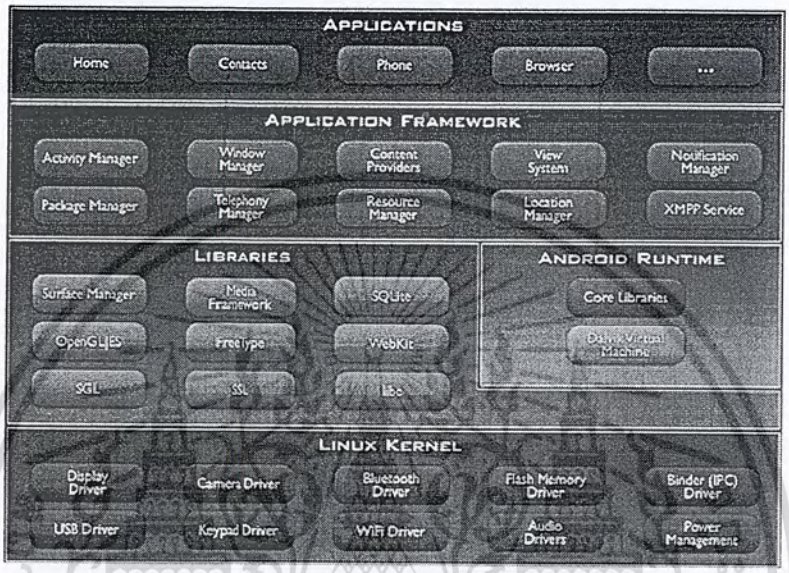
กูเกิลแอนดรอยด์ (Google Android) คือ ระบบปฏิบัติการ ที่เป็นซอฟต์แวร์แพลตฟอร์มบนมือถือ สร้างขึ้นมาจากระบบปฏิบัติการลีนุกซ(Powered by the Linux kernel) พัฒนาขึ้นมาโดยกูเกิล กูเกิลแอนดรอยด์นั้นได้เปิดให้นักพัฒนา สามารถเข้ามาจัดการเขียนโค๊ตต่างๆได้ด้วยภาษาจาวา และเขียนควบคุมอุปกรณ์ต่างๆ ผ่านทางจาวาไลบลารี่ที่ทางกูเกิลพัฒนาขึ้นโดยเฉพาะ (Google-developed Java libraries) โปรแกรมต่างๆ ที่รันบนกูเกิลแอนดรอย์สามารถเขียนได้ด้วยภาษาซี(C) และภาษาอื่น ส่วนการพัฒนาผ่านการคอมไพล์ด้วย สถาปัตยกรรมแบบ ARM Native Code(32bit) นั้นยังไม่ได้รับการสนับสนุนจากทางกูเกิลแต่อย่างใด

กูเกิลแอนดรอยด์ ได้เปิดตัวเป็นครั้งแรกในวันที่ 5 พฤศจิกายน พ.ศ. 2550 โดยทางกูเกิลได้ประกาศก่อตั้ง Open Handset Alliance กลุ่มบริษัทฮาร์ดแวร์, ซอฟต์แวร์ และการสื่อสาร 48 แห่ง และได้ออกมาให้ยนโฉมตัวจริงกันในช่วงปี 2551 ที่ผ่านมา ลิขสิทธิ์ของ กูเกิลแอนดรอยด์นั้น จะอยู่ในลักษณะของฟรีซอฟต์แวร์และ โอ เพ็นซอร์จ โดยอยู่ภายใต้สิทธิบัตรของ ครีเอทีพ คอมมอนส์ แอทรีบิว 2.5 ซึ่งทำให้ผู้ใช้นั้นสามารถดาวโหลดซอฟต์แวร์ของกูเกิลแอนดรอยด์ไป ใช้ได้ฟรี และยังสามารถนำซอฟต์แวร์ที่ได้ไปแชร์แจกต่อได้แต่ไม่อนุญาตให้แก้ไขโดยการนำเอาชื่อผู้เขียนซอฟต์แวร์ หรือรายการสิทธิบัตรของซอฟต์แวร์นั้นออกตัวโปรแกรม

Android คือ ระบบปฏิบัติการ (OS) หรือแพลตฟอร์ม ที่จะใช้ควบคุมการทำงานบนอุปกรณ์อีเล็คทรอนิกส์ต่างๆ สำหรับโทรศัพท์มือถือ และอุปกรณ์พกพา โดยมี กูเกิล อิงก์, ที-โมบาย, เอชทีซี, ควอลคอมม์, โมโตโรลา และบริษัทชั้นนำอีกมากมายร่วมพัฒนาโปรเจ็กต์แอนดรอยด์ ผ่านกลุ่มพันธมิตรเครื่องมือสื่อสารระบบเปิด (Open Handset Alliance) ซึ่งเป็นกลุ่มพันธมิตรชั้นนำระดับนานาชาติด้านเทคโนโลยีและเครื่องมือสื่อ สารเคลื่อนที่ ซึ่ง Android ประกอบด้วยระบบปฏิบัติการ ไลบรารีเฟรมเวิร์ค และซอฟต์แวร์อื่นๆ ที่จำเป็นในการพัฒนา ซึ่งเทียบเท่ากับ Windows Moble, Palm OS, Symbian, OpenMoko และ Maemo ของโนเกีย โดยใช้องค์ประกอบที่เป็นโอเพนซอร์สหลายอย่าง เช่น Linux Kernel, SSL, OpenGL, FreeType, SQLite, WebKit และเขียนไลบรารีเฟรมเวิร์คของตัวเองเพิ่มเติม ซึ่งทั้งหมดจะโอเพนซอร์ส ใช้ (Apache License)ความร่วมมือครั้งนี้มี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
117391

เป้าหมายในการส่งเสริมนวัตกรรมบนเครื่องมือสื่อสาร เพื่อให้ได้รับประสบการณ์ที่เหนือกว่าแพลตฟอร์มโมบายทั่วไปที่มีอยู่ใน ปัจจุบัน ทั้งนี้ การนำเสนอมิติใหม่ของแพลตฟอร์มระบบเปิดให้แก่นักพัฒนาจะทำช่วยให้กลุ่มคน เหล่านี้ทำงานร่วมกันได้มีประสิทธิภาพยิ่งขึ้น โดยแอนดรอยด์จะช่วยเร่งและผลักดันบริการระบบสื่อสารรูปแบบใหม่ไปสู่ผู้บริโภคได้อย่างที่ ไม่เคยเกิดขึ้นมาก่อน



รูป 2.15 แสดงแต่ละ Layer ของระบบปฏิบัติการAndroid

2.6.1เหตุผลที่ต้องเป็น Android

1) มี Program และ Application ให้ Download ได้อย่างมากมาย กว่า 10,000 แอปพลิเคชั่นใน Android Market
2) สามารถจัดการ Application ต่างๆ ได้ด้วยตัวเอง เช่น การติดตั้งลงโปรแกรม ลบโปรแกรม และอัพเดทโปรแกรม โดยที่ไม่ต้องผ่าน Computer หรือ Notebook อีก
3) ใช้งานได้อย่างง่ายดาย โดยไม่ต้องพึ่งคู่มือ เพียงคุณได้ลองเล่น ก็สามารถเรียนรู้การใช้งานได้อย่างรวดเร็ว
4) สามารถปรับเปลี่ยน หน้าจอและวิทเจ็ตต่างๆ ของคุณเองได้ เพื่อบ่งบอก Lifestyle ในตัวคุณ
5) รองรับการใช้งาน Gmail, Google Talk, Google Map อย่างเต็มรูปแบบ ข้อมูลของคุณจะถูกเก็บเข้าที่ Google Account โดยอัตโนมัติ ทำให้หมดความกังวลว่าข้อมูลที่สำคัญของคุณจะสูญหายอีกต่อไป
6) Web browser รองรับ Java application ในการแสดงผล Website ได้อย่างเต็มรูปแบบ
7) ความสามารถในการ "ค้นหา" และ "เข้าถึง" ฐานข้อมูลของ Google ได้อย่างง่ายดาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และรวดเร็ว เช่น การค้นหาเว็บเพจ การค้นหาเพลง การค้นหาวีดีโอ เป็นต้น

8) มีพื้นฐานมาจาก Linux ซึ่งเป็นระบบปฏิบัติการที่ได้รับการยอมรับว่ามีประสิทธิภาพสูงสุด และเสถียรที่สุด ทำให้ Android เป็นระบบปฏิบัติการที่เสถียรและทำงานได้อย่างรวดเร็ว

9) การเป็นระบบเปิด ทำให้มีผู้ผลิตหลายค่าย นำ Android ไปใช้ได้ ผลิตโทรศัพท์ออกมาให้เลือกเลือกใช้กัน ตอบสนองตามความต้องการของผู้ใช้มากขึ้น

10) การพัฒนาอย่างต่อเนื่อง ด้วยจำนวนนักพัฒนาทั่วโลกที่เพิ่มขึ้นอย่างต่อเนื่อง ทำให้ผู้ใช้ได้รับความแปลกใหม่อย่างไม่รู้จบ

2.10 Wellcom A88 (เวลคอม เอ 88)

ลักษณะเด่นประจำรุ่น

1) ส่งผ่านข้อมูล (Data Transfer)WiFi 802.11b/g, WLAN (Wireless LAN)

2) ใช้งานอินเตอร์เน็ต xHTML, HTML, WAP 2.0 Browser

3) จอสัมผัส TFT-LCD - 320 x 480 พิกเซล (3.2") หมุนภาพอัตโนมัติด้วยเทคโนโลยี G-sensor

4) ระบบปฏิบัติการ Android เวอร์ชั่น 2.1 เอแคร์

5) CPU : Qualcomm® MSM7227™, 600 MHz

6) หน่วยความจำ ROM 512 MB - RAM 256 MB

7) ระบบ Quadband (GSM 850/900/1800/1900 MHz) UMTS 850/1900/2100 MHz, HSDPA7.2 Mbps

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูป 2.16 Wellcom A88

2.11 Arduino

Arduinoเป็นโมดูลไมโครคอนโทรลเลอร์สำเร็จรูป ซึ่งใช้ไมโครคอนโทรลเลอร์ตระกูล AVR โดยตัวของมันเองสามารถรับและส่งค่าจากเซนเซอร์ต่างๆได้ เช่น มอเตอร, เซนเซอร์แสง เป็นต้น โดยที่ตัวไมโครคอนโทรลเลอร์สามารถโปรแกรมได้โดยใช้Arduino programming language ผ่านทางสายในที่นี้ได้เลือกใช้ ArduinoDuemilanove

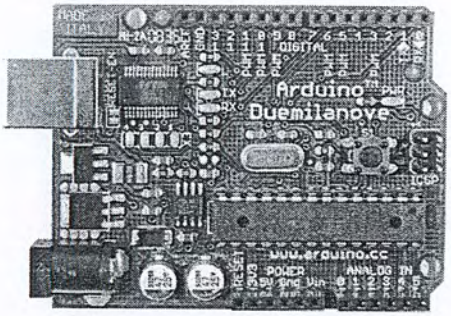
ArduinoDuemilanoveได้ใช้ไมโครคอนโทรเลอร์ ATmega328 ซึ่งมีขาดิจิตอลขาเข้าและออกจำนวน 14 ขา ในจำนวนนี้มี 6 ขาที่สามารถส่งสัญญาณขาออกแบบ PWM, ขารับสัญญาณแอนนาลอก 6 ขา, 16MHz crystal oscillator, หัวต่อ USB, หัวรับไฟฟ้า, ICSP header และปุ่มรีเซ็ท โดยรายละเอียดของขาที่ได้ใช้มีดังต่อไปนี้

Serial: 0 (RX) และ 1(TX) ใช้รับและส่งสัญญาณผ่านสาย Serial

PWM : 3, 5, 6, 9, 10, และ11ใช้ส่งสัญญาณ PWM ได้ตามขาเหล่านี้

Digital Input and output: 1 ถึง 13 ใช้สัญญาณดิจิตอลได้ทั้งขาเข้าและออก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.17 Arduino Duemilanove

ในการเขียนโปรแกรมสามารถใช้IDE ของ Arduinoซึ่งแสดงในรูป XX โดยสามารดาวน์โหลดได้จาก http://www.arduino.cc/en/Main/Softwareหลังจากนั้นทำการติดตั้ง และเลือกรุ่นให้เหมาะสมกับ Arduino Board ที่เรามี หลังจากนั้นทำการเขียนโปรแกรมโดยใช้ภาษา C เพื่อให้มันทำตามคำสั่งได้ดังที่เราต้องการ หลังจากนั้น เขียนโปรแกรมลง Arduino Board

ในการเขียนโปรแกรมของ Arduinoจะไม่มีฟังก์ชั่น main() เหมือนภาษา C ทั่วไปๆ แต่จะมีอยู่สองฟังก์ชั่นหลัก คือ setup() ใช้สำหรับ กำหนดค่าเริ่มต้นทุกอย่าง เช่น กำหนดว่าขาที่ใช้เป็นขาเข้า หรือ ออก, กำหนดลักษณะการส่งสัญญาณของสาย Serial เป็นต้น

ฟังก์ชั่นที่สองคือ loop() ใช้สำหรับใส่คำสั่งที่ต้องการให้มันทำงานตลอด โดย Board จะอ่านคำสั่งที่อยู่ในฟังก์ชั่นนี้ไปตลอด ซึ่งตัวอย่างคำสั่งที่ใช้ มีดังต่อไปนี้

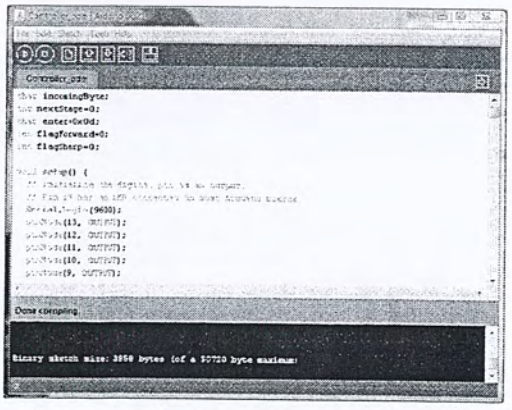
delay(1000) ใช้เพื่อหน่วงเวลาโดยมีหน่วยเป็น ms

digitalWrite(9,HIGH) ใช้เพื่อให้ส่งไปยังขาที่ 9 สัญญาณ HIGH

Serial.read()ใช้เพื่ออ่านข้อมูลจากสาย Serial

Serial.write()ใช้เพื่อเขียนข้อมูลเพื่อส่งไปยังSerial

analogWrite(9,255) ใช้เพื่อส่งสัญญาณไปยังขาที่รองรับ PWM (ขาที่ 9) โดยมีแรงดันไฟฟ้าค่าต่างๆ (0-255)



รูป 2.18 ArduinoIDE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบและพัฒนา

ในการพัฒนาหุ่นยนต์สำรวจและกู้ระเบิด การพัฒนาจะแบ่งเป็นสองส่วนใหญ่ๆ คือส่วนที่เป็นฮาร์ดแวร์และส่วนที่เป็นซอฟแวร์ ฮาร์ดแวร์จะอยู่บนตัวหุ่นยนต์ประกอบด้วย

3.1 ฮาร์ดแวร์

3.1.1 Microcontroller

ไมโครคอนโทรลเลอร์ คือ อุปกรณ์ควบคุมขนาดเล็ก ซึ่งบรรจุความสามารถที่คล้ายคลึงกับระบบคอมพิวเตอร์ โดยในไมโครคอนโทรลเลอร์ได้รวมเอาซีพียู, หน่วยความจำ และพอร์ต ซึ่งเป็นส่วนประกอบหลักสำคัญของระบบคอมพิวเตอร์เข้าไว้ด้วยกัน โดยทำการบรรจุเข้าไว้ในตัวถังเดียวกัน

โครงสร้างโดยทั่วไป ของไมโครคอนโทรลเลอร์นั้น สามารถแบ่งออกมาได้เป็น 5 ส่วนใหญ่ๆ ดังต่อไปนี้

3.1.1.1 หน่วยประมวลผลกลางหรือซีพียู (CPU : Central Processing Unit)

3.1.1.2 หน่วยความจำ (Memory) สามารถแบ่งออกเป็น 2 ส่วน คือ หน่วยความจำที่มีไว้สำหรับเก็บโปรแกรมหลัก (Program Memory) เปรียบเสมือนฮาร์ดดิสก์ของ เครื่องคอมพิวเตอร์ตั้งโต๊ะ คือข้อมูลใดๆ ที่ถูกเก็บไว้ในนี้จะไม่สูญหายไปแม้ไม่มีไฟเลี้ยง อีกส่วนหนึ่งคือหน่วยความจำข้อมูล (Data Memory) ใช้เป็นเหมือนกกระดาษทดในการคำนวณของซีพียู และเป็นที่พักข้อมูลชั่วคราวขณะทำงาน แต่หากไม่มีไฟเลี้ยง ข้อมูลก็จะหายไปคล้ายกับหน่วยความแรม (RAM) ในเครื่องคอมพิวเตอร์ทั่วๆ ไป แต่สำหรับไมโครคอนโทรลเลอร์สมัยใหม่ หน่วยความจำข้อมูลจะมีทั้งที่เป็นหน่วยความจำแรม ซึ่งข้อมูลจะหายไปเมื่อไม่มีไฟเลี้ยง และเป็นอีอีพรอม (EEPROM : Erasable Electrically Read-Only Mempry) ซึ่งสามารถเก็บข้อมูลได้แม้ไม่มีไฟเลี้ยง

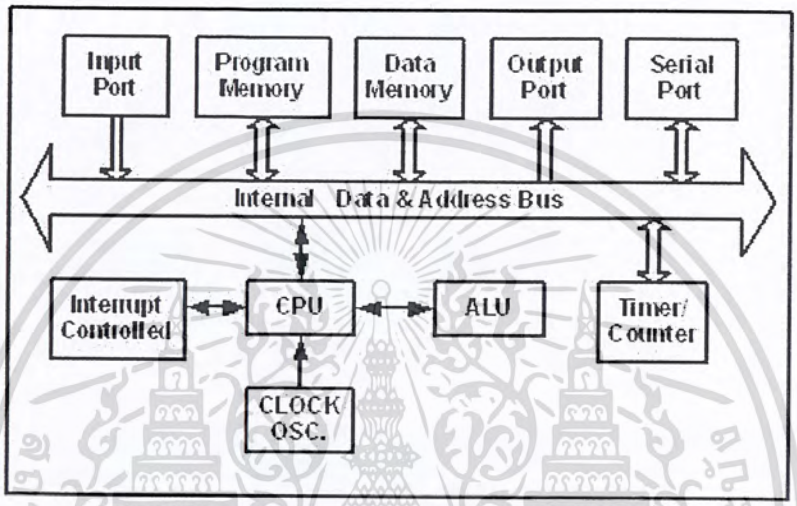
3.1.1.3 ส่วนติดต่อกับอุปกรณ์ภายนอก หรือพอร์ต (Port) มี 2 ลักษณะคือ พอร์ตอินพุต (Input Port) และพอร์ตส่งสัญญาณหรือพอร์ตเอาต์พุต (Output Port) ส่วนนี้จะใช้ในการเชื่อมต่อกับอุปกรณ์ภายนอก ถือว่าเป็นส่วนที่สำคัญมาก ใช้ร่วมกันระหว่างพอร์ตอินพุต เพื่อรับสัญญาณอาจจะด้วยการกดสวิตช์ เพื่อนำไปประมวลผลและส่งไปพอร์ตเอาต์พุต เพื่อแสดงผลเช่น การติดสว่างของหลอดไฟ เป็นต้น

3.1.1.4 ช่องทางเดินของสัญญาณ หรือบัส (BUS) คือเส้นทางการแลกเปลี่ยนสัญญาณข้อมูลระหว่าง ซีพียู หน่วยความจำและพอร์ต เป็นลักษณะของสายสัญญาณ จำนวนมากอยู่ภายใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวไมโครคอนโทรลเลอร์ โดยแบ่งเป็นบัสข้อมูล (Data Bus) , บัสแอดเดรส (Address Bus) และบัสควบคุม (Control Bus)

3.1.1.5 วงจรกำเนิดสัญญาณนาฬิกา นับเป็นส่วนประกอบที่สำคัญมากอีกส่วนหนึ่ง เนื่องจากการทำงานที่เกิดขึ้นในตัวไมโครคอนโทรลเลอร์ จะขึ้นอยู่กับการกำหนดจังหวะ หากสัญญาณนาฬิกามีความถี่สูง จังหวะการทำงานก็จะสามารถทำได้ถี่ขึ้นส่งผลให้ไมโครคอนโทรลเลอร์ตัวนั้น มีความเร็วในการประมวลผลสูงตามไปด้วย



รูป 3.1 สถาปัตยากรรมของไมโครคอนโทรลเลอร์

3.1.2 AVR Microcontroller

ไมโครคอนโทรลเลอร์ AVR เป็นไอซีไมโครคอนโทรลเลอร์ของบริษัท Atmel มีสถาปัตยกรรมภายในเป็นแบบ RISC (reduced instruction set computer) โดยใช้สัญญาณนาฬิกาเพียง 1 ลูกในการปฏิบัติงานใน 1 คำสั่ง โดยจะประกอบด้วยหน่วยความจำโปรแกรมภายในที่เป็นแบบแฟลช์ โปรแกรมข้อมูลได้แบบ In-System programmable และในบางเบอร์ยังสามารถมีการกำหนดตำแหน่งของหน่วยความจำที่สร้างเป็นบูตโหลด เดอร์ (เขียนโปรแกรมเพื่อติดต่อกับ PC หรือไอซีตัวอื่นๆ และยังสามารถโปรแกรมให้กับตัวเองได้) มีขนาดของหน่วยความจำตามเบอร์ของไอซีแต่ละตัว ซึ่งส่วนมากมีคุณสมบัติดังต่อไปนี้

1) มีโครงสร้างภายในแบบ RISC(RISC คือการให้ซีพียูทำงาน ที่มีไซเกิลแน่นอน โดยลดจำนวนคำสั่งลงให้เหลือคำสั่งพื้นฐานมากที่สุด แล้ว ใช้หลักการไปป์ไลน์ (pipeline) คือการนำเอาคำสั่งมาเรียงการทำงานให้เป็นแบบ ขนานเหลื่อมกัน หรือเข้าทำงาน ในแต่ละตำแหน่งตามลำดับเรียงกันไป ทุกตำแหน่งงานจะมีการงานทำงานตลอดเวลา จึงเป็นการลดจำนวนคำสั่งลงนั้นเอง ซึ่งเรียกการทำงานของซีพียูแบบนี้เป็นประเภท RISC หรือ Reduce Instruction set Computer สถาปัตยกรรมที่มี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงสร้างเป็นแบบ RISC จึงทำงานได้เร็ว และ เป็นกลไก ที่สามารถเพิ่มขีดความสามารถโดยรวมได้หนึ่งคำสั่งใช้เวลาหนึ่ง ลูกของสัญญาณ นั่นคือถ้าเป็น 10 MHz ก็ทำได้ 10 ล้านคำสั่งในเวลา 1 วินาที ทำให้สามารถใช้คำสั่งง่ายขึ้น ไม่ยุ่งยากซับซ้อน)

2) มีคำสั่งควบคุมการทำงานไมโครคอนโทรลเลอร์ 130 คำสั่ง คำสั่งส่วนมากจะทำสำเร็จในรอบสัญญาณนาฬิกาเดี่ยว

3) เขียน /ลบ ได้ถึง 10,000 ครั้ง สำหรับหน่วยความจำแบบ Flash และ 100,000 สำหรับหน่วยความจำแบบ EEPROM

4) กำหนดการ Boot Code Section ในตำแหน่งต่างๆ และ Lock Bits ได้ (ทำ boot loader)

5) Programming Lock for Software Security ป้องกันข้อมูล

6) Timer/Counters ขนาด 8-bit 2 ตัว และมี Separate Prescaler โหมด Compare อีก 1 ตัว

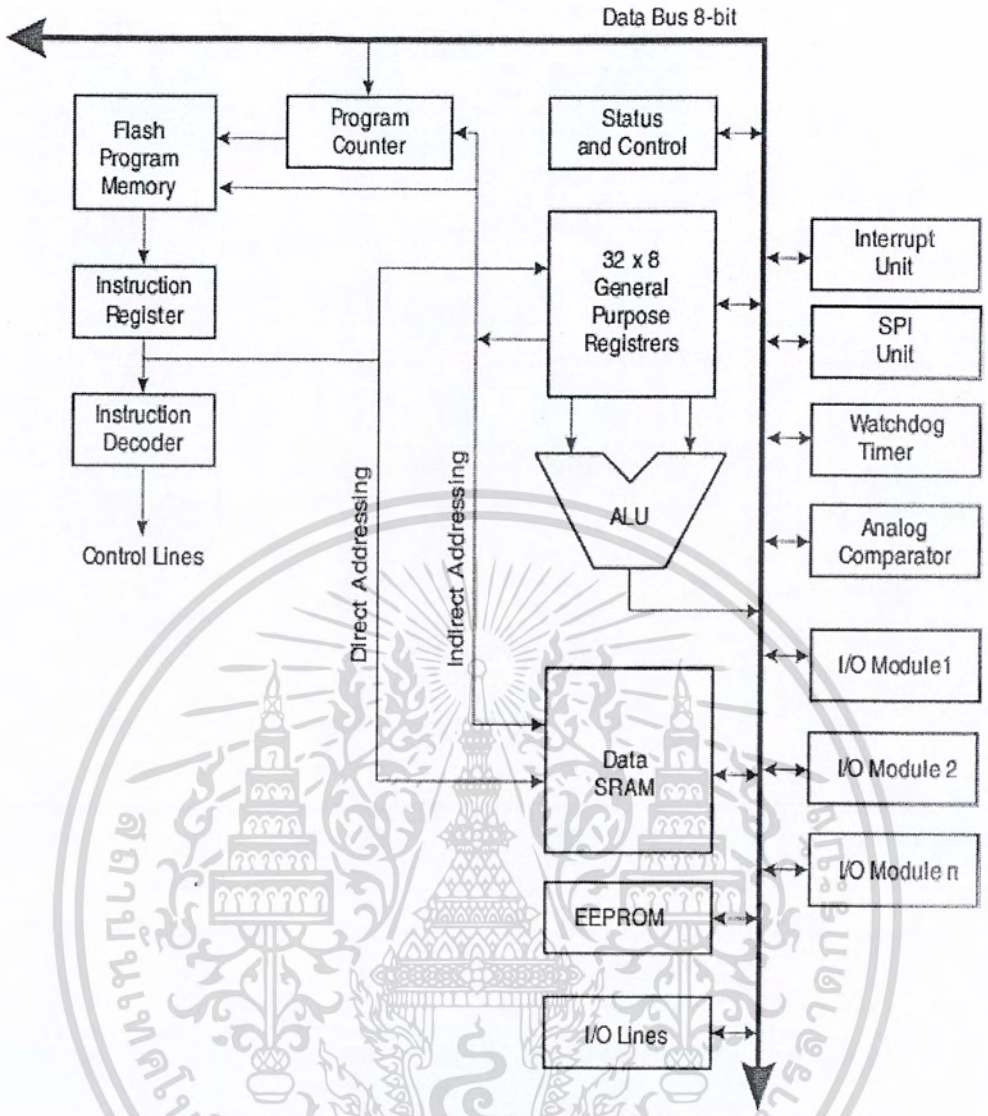
7) One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture

8) มี PWM 3 ช่อง

9) มีการติดต่อแบบ Master/Slave SPI Serial Interface

10) ใช้งาน RC Oscillator ภายในไอซี และภายนอกไอซีได้

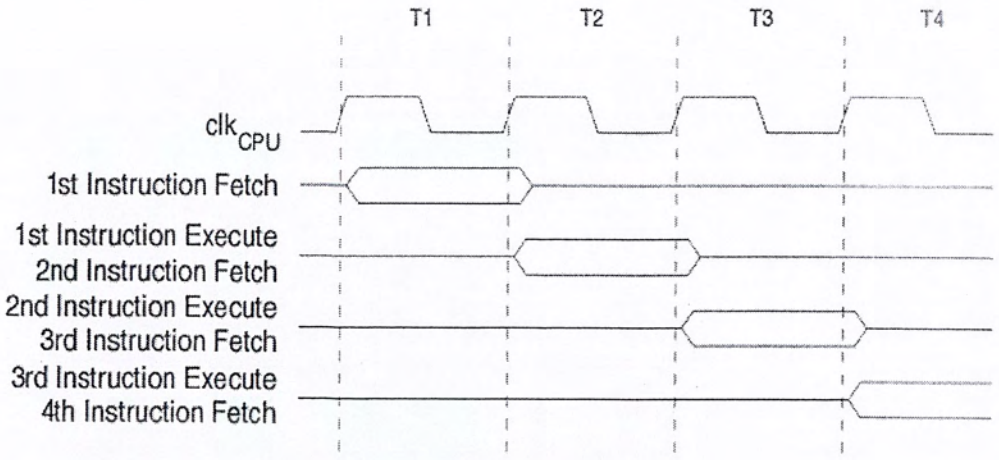
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.2 AVR Architecture

AVR มีโครงสร้างเป็นแบบ Harvard ซึ่งจะแยกหน่วยความโปรแกรม (Program memory) ออกจากหน่วยความจำข้อมูล (Data memory) การ Fetch และ Execute คำสั่งของ AVR สามารถทำเสร็จภายใน 1-cycle ทำให้ AVR มีความเร็ว 1-MIPS (Mega Instruction Per Second) ต่อ 1-MHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.3 Single level pipelining

3.1.3 ไมโครคอนโทรลเลอร์ AVR

ได้แบ่งออกเป็น 4 กลุ่มใหญ่ๆ ดังนี้

รูป 3.4 กลุ่มของ AVR

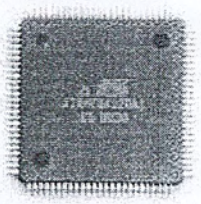
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูป 3.5 **TinyAVR**

1) TinyAVR — ATtiny series **เช่นเบอร์ Tiny13 , Tiny2313 ...**
   มีหน่วยความจำโปรแกรมขนาด 1–8 kB
   มีจำนวนขาใช้งาน 6–32-ขา
   มีส่วนของอุปกรณ์เสริมที่ค่อนข้างจำกัด

รูป 3.6 MegaAVR — ATmega series

2) MegaAVR — ATmega series เช่นเบอร์ ATmega8 ,ATmega16, ATmega32, ATmega64
   มีหน่วยความจำโปรแกรมขนาด 4 – 256 kB
   มีจำนวนขาใช้งาน 28–100 ขา
   มีชุดคำสั่งที่สามารถจัดการกับหน่วยความจำที่มีขนาดใหญ่มากขึ้น
   มีส่วนของอุปกรณ์เสริมมากในตัวไอซี

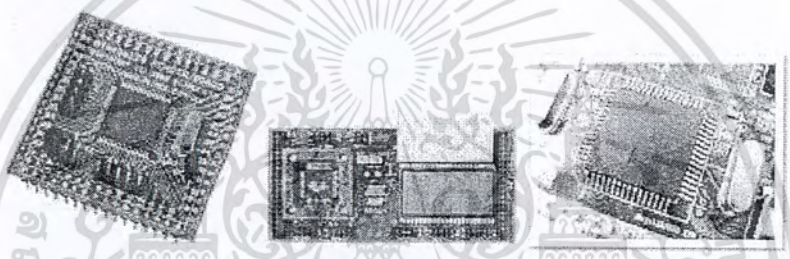


รูป 3.7 XMEGA — ATxmega series

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) XMEGA — ATxmega series เช่นเบอร์ ATxmega64A1, ATxmega128A1...
มีหน่วยความจำโปรแกรมขนาด 16–384 kB
มีจำนวนขาใช้งาน 44–64–100 ขา
มีชุดคำสั่งที่สามารถจัดการกับระบบ DMA และการเข้าถึงเหตุการณ์ได้เร็วมากขึ้น โดยใช้การสื่อสารในแบบต่างๆ ได้หลายรูปแบบ มีส่วนของอุปกรณ์เสริมมากในตัวไอซีโดยการใช้งานกับ Digital-to-analog converter (DACs) และยังสามารถเขียนรหัสเฉพาะ โดยเข้ากันได้กับไฟล์แบบ AES และ DES (เช่นใช้สำหรับการกำหนดรหัสส่วนตัว)

4) Application specific AVR เป็นไอซีที่สร้างเพื่อใช้งานเฉพาะ เช่น CAN AVR (ATmega64C1), LCD AVR (ATmega3290P/V), USB AVR (AT90USB1287)



รูป 3.8 Application specific AVR

ในส่วนที่เป็นซอฟแวร์ฝั่งอยู่บน Remote Control จะรันอยู่บนระบบปฏิบัติการแอนดรอย ในที่นี้จะใช้แอนดรอยที่รันบนโทรศัพท์มือถือ T-Mobile G1 ระบบปฏิบัติการ Android 2.2 froyo โดยติดต่อกับหุ่นยนต์ผ่าน Wireless Router ซึ่งจะได้นำแนะนำในส่วนต่อไปดังนี้

3.2 การพัฒนา Android ด้วย Eclipse

ในการพัฒนา Android ใน Eclipse ภาษาที่ใช้พัฒนาเป็นจาวา แต่เอามาเฉพาะภาษา (Java programming language) ไม่ได้ใช้แพลตฟอร์มของซันเลย โดย Android มีรันไทม์ของตัวเองชื่อว่า Dalvik ซึ่งแทนที่เราจะคอมไพล์เป็นไบต์โค้ด ก็ใช้ฟอร์แมต Dalvik Executable (.dex) แทน การสร้าง GUI ใช้เป็นไฟล์ XML (สไตล์เดียวกับ XAML หรือ XUL)

ไลบรารีที่มีให้เรียกก็เป็นของ Android ทั้งหมด (import android.*) ไม่เกี่ยวกับ J2ME เลย อารมณ์ว่ายืม syntax ของจาวามาใช้เฉยๆ แต่อย่างอื่นกูเกิลทำเองเกือบหมด เครื่องมือพัฒนา Android ก็ไม่ใช่อื่นไกล มีปลั๊กอินสำหรับ Eclipse ตามสมัยนิยม (ชื่อ Android Development Tools - ADT) ใช้ Apache Ant สำหรับ build สำหรับส่วนของ SDK นั้นก็ให้เครื่องมือมาครบ ทั้งตัวดีบั๊ก สารพัดชนิด และ emulator ของ GPhone สำหรับรันแอพพลิเคชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
package android.client.ctrl;

//import java.util.concurrent.BlockingQueue;

import android.app.Activity;
//import android.os.Handler;
//import android.os.Message;
//import android.util.Log;
//import android.view.Menu;
//import android.view.MenuItem;
//import android.view.MotionEvent;
//import android.view.View;
//import android.view.Window;
//import android.view.WindowManager;
//import android.view.View.OnClickListener;
//import android.widget.Button;
//import android.widget.ImageView;
//import android.content.Intent;
public class Console extends Activity {
    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.console);
    }

}
```

รูป 3.9 Code ส่วนที่เป็น .Java

```
<?xml version="1.0" encoding="utf-8"?>
<AbsoluteLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent" android:background="@drawable/black">
    <TextView
        android:id="@+id/server_label"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content" android:text="IP Address ::" android:t
    <EditText
        android:id="@+id/server_field"
        android:layout_height="wrap_content"
        android:background="@android:drawable/editbox_background"
        android:text="@string/default_server"
        android:layout_below="@id/server_label"
        android:layout_width="200px"
        android:layout_x="65px" android:layout_y="45px"/>
    <EditText
        android:id="@+id/server_bot"
        android:layout_height="wrap_content"
        android:background="@android:drawable/editbox_background"
        android:text="@string/default_server"
        android:layout_below="@id/server_label"
        android:layout_width="200px"
        android:layout_x="65px" android:layout_y="105px"/>
<ImageButton
    android:id="@+id/ImageButton01"
    android:background="@drawable/cams"
    android:layout_x="5dip"
    android:layout_height="45dip"
```

รูป 3.10 Code ส่วนที่เป็น .xml

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.1 ซอฟต์แวร์ที่ใช้

1) Eclipse เป็น IDE ที่ใช้ในการเขียนโปรแกรม และต้องเป็น version 3.3 ขึ้นไป ในที่นี่จะใช้ EclipseVersion: 3.5.2 (Galileo) Android Plugin ADT Download Plugin ที่ http://dl-ssl.google.com/android/ADT-0.9.1.zip

2) Android SDK Download ได้ที่ http://developer.android.com/sdk/1.5_r3/index.html ซึ่งในที่นี้จะใช้ Android 2.2

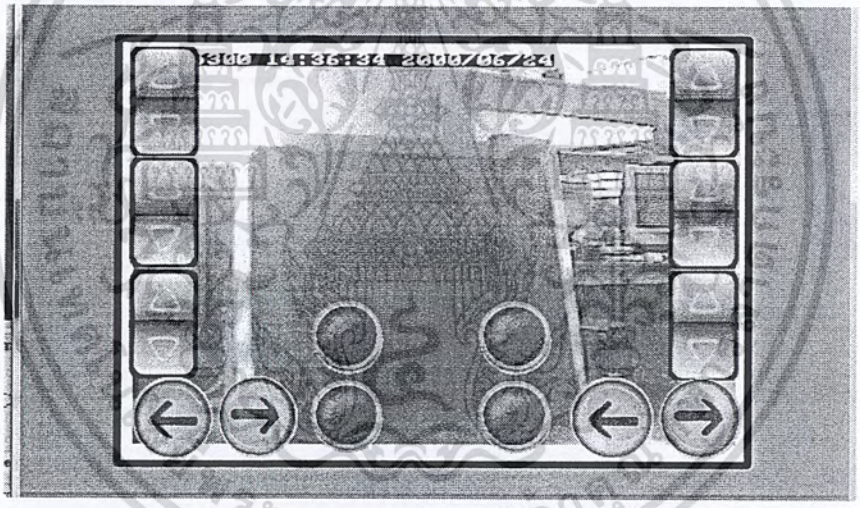
3) Java SDK ดาว์นโหลดได้จาก http://java.sun.com (jdk-6-windows-i586.exe)

3.2.2 Interface บน Remote Android

ในการทำรีโมทในการควบคุมหุ่นสำรวจและเก็บกู้วัตถุระเบิดนั้นจะประกอบด้วย UI สองส่วนนั่นคือ

1) ติดต่อ IP Cam จะทำหน้าที่ส่งข้อมูลภาพจากกล้องมายังหน้าจอบน Mobile

2) หน้าต่างการควบคุมจากระยะไกล ซึ่งจะมีปุ่มควบคุมส่วนต่างๆบนหุ่นยนต์

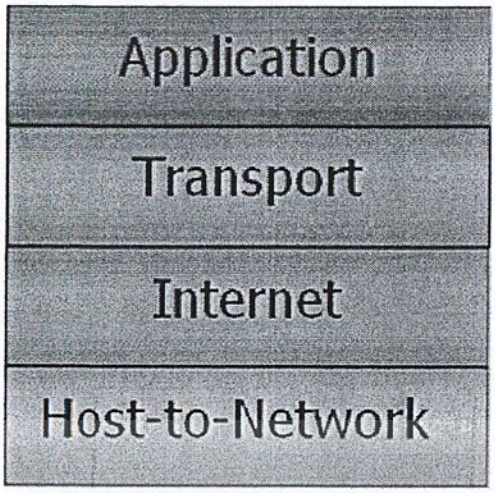


รูป 3.11 การควบคุมหุ่นยนต์ได้จากระยะไกล

3.3 Connection ระหว่าง Android กับ หุ่นยนต์

นอกจาก UI ในการควบคุมหุ่นยนต์แล้วยังมีส่วนที่เป็นชั้นของการทำ Connection ระหว่าง Android กับ หุ่นยนต์สำรวจ ผ่านWireless Router ซึ่งการติดต่อจะทำการเชื่อมต่อจาก Android ไปยัง Wireless Router เลย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.12 TCP/IP model

จากการออบแบบและพัฒนาการติดต่อสื่อสารระหว่าง Android กับ Wireless Router สามารถอธิบายการทำงานของแต่ละ Layer ได้ดังต่อไปนี้

3.3.1 Host-to-Network Layer

Wireless LAN ในการสื่อสาร

3.3.2 Internet Layer

ใช้ Ipv4 ในการสื่อสาร

bit offset	0–3	4–7	8–13	14-15	16–18	19–31
0	Version	Header Length	Differentiated Services Code Point	Explicit Congestion Notification		Total Length
32	Identification				Flags	Fragment Offset
64	Time to Live		Protocol			Header Checksum
96	Source IP Address					
128	Destination IP Address					
160	Options (if Header Length > 5)					
160 or 192+	Data					

รูป 3.13 โครงสร้าง Ipv4

3.3.3. Transport Layer

ใช้ UDP ในการติดต่อ เนื่องจากต้องการความรวดเร็วในการควบคุมหุ่นยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

bits	0 – 15	16 – 31
0	Source Port Number	Destination Port Number
32	Length	Checksum
64	Data	

รูป 3.14 UDP Datagram

3.3.4 Application Layer

ในส่วนของ Android Mobile ได้ใช้ Java Socket ในการพัฒนา

ในส่วนของ Wireless Router ได้ใช้ C Socket ในการพัฒนา

ซึ่งได้ออกแบบ Datagram ไว้ดังรูป

Length (8 bit)	Data (24 bit)	ChkSum (8 bit)

รูป 3.15 Datagram

Length คือ ความยาวของ Frame ทั้งหมด

Dataคือ คำสั่งที่ใช้ในการควบคุม Robot

ChkSum คือ ตรวจสอบความถูกต้องของ Frame

3.4. การติดต่อสื่อสารระหว่าง Android กับ Ipcam

จากการออบแบบและพัฒนาการติดต่อสื่อสารระหว่าง Android กับ IPcam สามารถอธิบายการทำงานของแต่ละ Layer ได้ดังต่อไปนี้

3.4.1 Host-to-Network Layer

Wireless LAN ในการสื่อสาร

3.4.2 Internet Layer

ใช้ Ipv4 ในการสื่อสารโดยใช้ IP แยกจากส่วนควบคุมต่างหาก

3.4.3 Transport Layer

ใช้ UDP ในการติดต่อ เนื่องจากต้องการความรวดเร็วในการแสดงผล

3.4.4 Application Layer

จะนำภาพ VDO มาแสดงผลบน Android Application

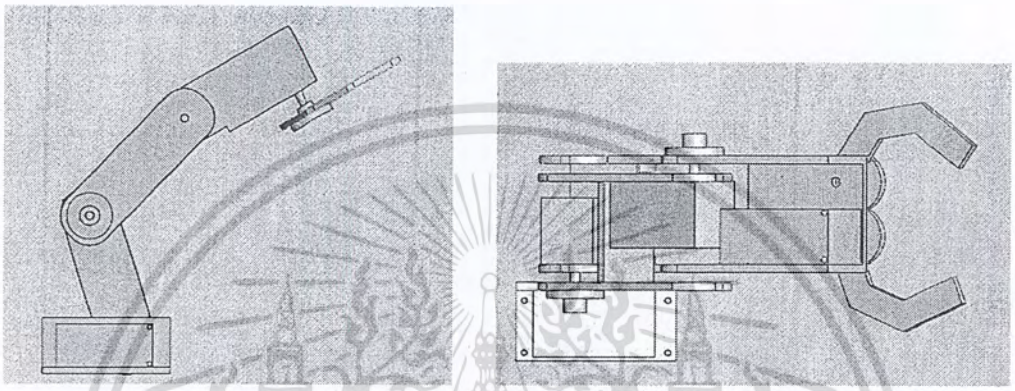
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 การออกแบบส่วนต่างๆของหุ่นยนต์

การออกแบบส่วนต่างๆของหุ่นยนต์นั่นได้ใช้โปรแกรม Solidwork 2010 โดยจะออกแบบเป็นส่วนๆประกอบดังต่อไปนี้

3.5.1 แขนกล

แขนเป็นส่วนที่ใช้ในการหยิบจับวัตถุ จะมีอยู่ด้วยกัน 4 ส่วนด้วยกัน แต่ละส่วนจะมีมอเตอร์ควบคุมการเคลื่อนไหวของข้อแขนอยู่



ก) ข)

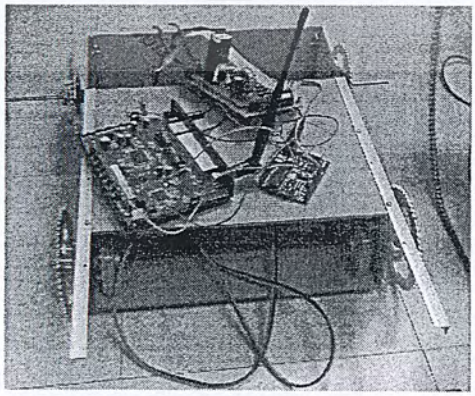
รูป 3.16 แขนกลจากการออกแบบบน Solidwork

ก) แขนกลมุมมองด้านข้าง

ข) แขนกลมุมมองด้านบน

3.5.2 ตัวหุ่นยนต์

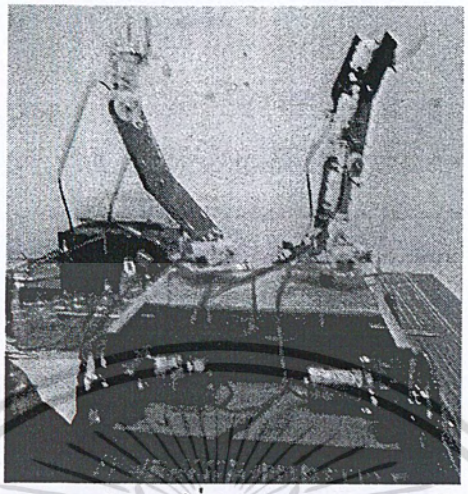
ตัวหุ่นยนต์จะประกอบด้วย ตัวหุ่นใช้ในการวางวงจรควบคุม กล้อง และWireless Router และล้อของหุ่นใช้ในการเคลื่อนหุ่นยนต์ตรงล้อจะมีมอเตอร์ควบคุมซ้ายขวา และจะมีสายพานด้วยดังรูป



รูป 3.17 ตัวหุ่นยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

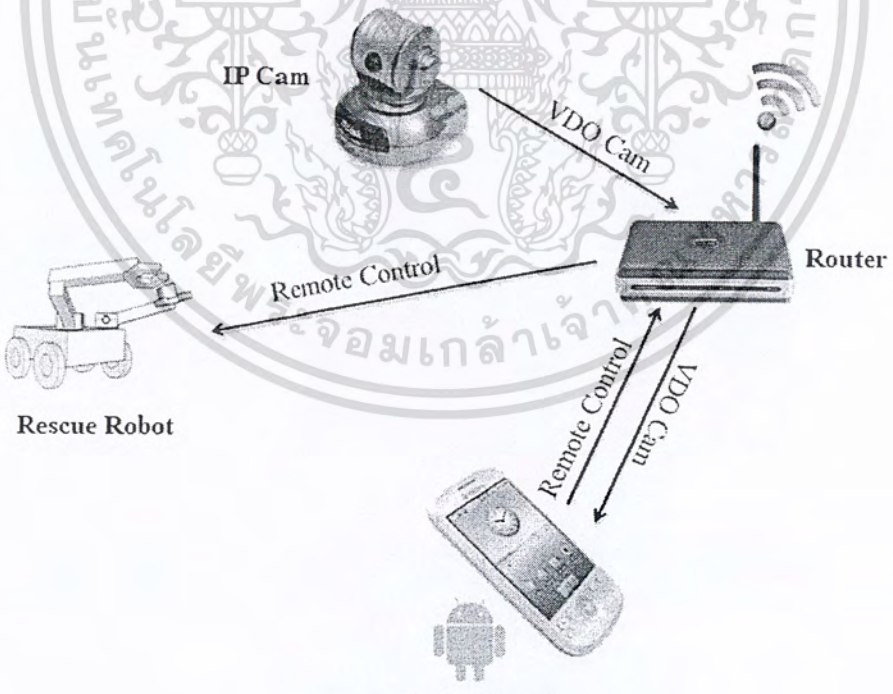
เมื่อนำทั้งสองส่วนมาประกอบเข้าด้วยกันโดยแขนทั้งสองข้างมาติดบนตัวหุ่นทั้งสองฝั่ง และวางสายพานให้กับล้อเรียบจะได้ดังรูป 3.18



รูป 3.18 หุ่นยนต์ที่ประกอบแขนเสร็จแล้ว

3.6 การออกแบบระบบ

การออกแบบระบบควบคุมหุ่นยนต์



รูป 3.19 การออกแบบระบบควบคุมหุ่นยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7 ความต้องการระบบ

3.7.1 ข้อกำหนดด้านฟังก์ชัน

1) หุ่นยนต์สามารถสำรวจและเก็บกู้ระเบิดได้

2) หุ่นสามรถควบคุมจากระยะไกลได้

3) หุ่นสามรถถ่ายทอดภาพจาก IP Cam ได้จากระยะไกล

4) หุ่นยนต์สามรถเคลื่อนที่ไปยังจุดต่างๆได้

5) การควบคุมการเคลื่อนที่ของหุ่นยนต์ใช้ G-Sensor ในการควบคุม

3.7.2 ข้อกำหนดที่ไม่ใช่ฟังก์ชัน

1) การทำรีโมทควบคุมจากระยะไกล พัฒนาบนระบบปฏิบัติการ Android

2) การควบคุมจากระยะไกลผ่าน wi-fi

3) สัญญาณ wi-fi อาจโดนสิ่งขวางจากสิ่งแวดล้อมทำให้ระยะในการควบคุมน้อยลง

4) ใช้ Arduino เป็น โมดูลไมโครคอนโทรลเลอร์สำเร็จรูป

5) ในRouter ใช้ระบบปฏิบัติการ Open WRT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทดลอง

4.1 การทดลองระยะทางที่สามมารถควบคุมหุ่นยนต์ได้

เนื่องจากการควบคุมจากระยะไกลผ่าน wi-fi นั้น จะมีข้อจำกัดอยู่คือเรื่องความไกลของสัญญาณที่ส่งออกไป จึงได้มีการทดสอบว่าหุ่นยนต์สามารถควบคุมผ่าน wi-fi ได้ไกลเท่าไร โดยจำลองสถานการณ์ขึ้นมาก 2 กรณี คือ แบบในที่โล่งแจ้ง ไม่มีสิ่งกีดขวาง กับในสถานที่ที่มีสิ่งกีดขวางจริงโดยทดลองกับอาคารจริง โดยใช้ตึก ECC เป็นสถานที่จำลองผลได้ดังนี้

ตาราง 4.1 การทดสอบการควมคุมระยะไกลเปรียบเทียบระหว่างที่โล่งกับที่ที่มีอาคารหรือสิ่งกีดขวาง

สถานที่	ระยะที่สามารถควบคุมได้ (m)
ที่โล่ง	50
ภายในอาคาร	30

4.2 การทดลองเรื่องความสามารถในการหยิบจับวัตถุรูปทรงต่าง

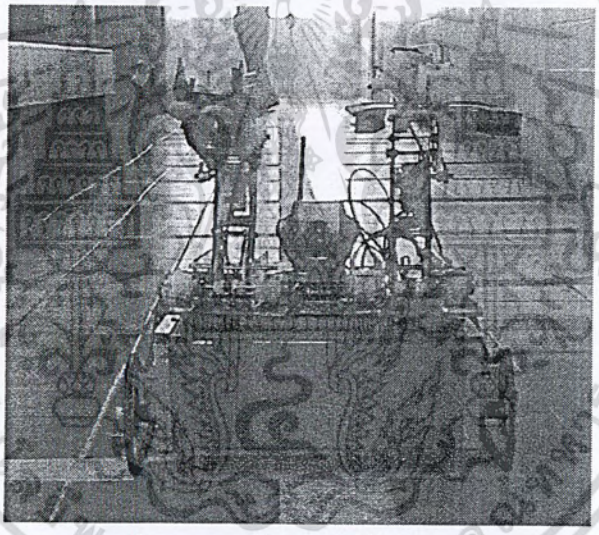
การหยิบจับวัตถุรูปทรงต่างๆนั้นได้ทำการทดลองเอาวัตถุรูปทรงต่างมาแล้วให้แขนกลควบคุมในการหยิบวัตถุ ในการหยิบวัตถุได้คือสามารถยกวัตถุขึ้นมาได้แต่รัศมีของวัตถุได้ไม่เกิน 7 เซนติเมตร ซึ่งได้ทดลองกับวัตถุต่างดังตารางนี้

ตาราง 4.2 ความสามารถในการหยิบวัตถุรูปทรงต่างๆ

วัตถุรูปทรงต่างๆ	สามารถหยิบได้(ได้/ไม่ได้)
ทรงกลม	ไม่ได้
สี่เหลี่ยมลูกบาศก์	ได้
ทรงกระบอก	ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูป 4.1 ภาพแขนกลหยิบขวดน้ำ



รูป 4.2 ภาพหุ่นตอนเสร็จสมบูรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทสรุป

5.1 สรุป

จากการทดลองแล้ว หุ่นยนต์จากจะสามารถเก็บกู้ระเบิดได้จากแขนทั้งสองข้าง และมองเห็นจากกล้อง IP Cam ติดต่อผ่าน wi-fi โดยมี Router เป็นตัวกลางในการแลกเปลี่ยนข้อมูล

ซึ่งเมื่อถูกควมคุมผ่านระบบปฏิบัติการแอดดรอยโดยติดต่อผกันผ่าน Router แบบ UDP ถ่ายทอดภาพและการทำงานกลับมาแบบ Real Time ก็จะยิ่งผ่ามความสามรถให้กับหุ่นยนต์สำรวจและกู้ระเบิดได้มากขึ้น การควบคุมระยะไกลผ่าน wi-fi นั้นในที่โล่งสามารถทำได้ถึง 30-50 เมตร แต่ถ้าในระยะที่มีสิ่งกีดขวางหรือภายในอาคารสัญญาณก็จะลดลงมาอีก ส่วนการหยิบวัตถุรูปทรงต่างๆนั่นสามารถทำได้แต่รัศมีของวัตถุนั้นต้องมีขนาดไม่เกิน 7 เซนติเมตร

5.2 ผลที่ได้รับ

1) ได้ทราบถึงกาพัฒนาแอพพลิเคชันบน Android Plaform
2) ได้หุ่นยนต์สำรวจและเก็บกู้ระเบิด
3) ได้ทราบถึงแนวทางการสร้างและพัฒนาหุ่นยนต์

5.3 ปัญหาและอุปสรรค

1) คุณภาพของภาพ IP Cam ที่ส่งกลับมายังไม่เร็วพอ ภาพยังคงล้าช้า ภาพยังไม่ชัด

5.4 แนวทางการพัฒนาต่อ

1) พัฒนาระยะในการควบคุมจากระยะไกล
2) พัฒนาแขนกลให้สามารถหมุน ได้อิสระมากขึ้น
3) เพิ่มระบบความปลอดภัยในการเชื่อมต่อให้มากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

Ed Burnette. December 2008.**Hello, Android - Introducing Google's Mobile Development Platform**

Oreilly. May.2009.**Application Development**.

Mark L. Murphy.2 Mar 2010.**Beginning.Android**

ThaiCERT.2548. **ความรู้พื้นฐานเกี่ยวกับ โปรโตคอล TCP/IP**.

[Online].Available: http://www.thaicert.org

Institute of Field Robotics.2552.**การออกแบบหุ่นยนต์.[Online]**.

Available: http://fibo.kmutt.ac.th

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

วิธีลง Android SDK ใน Eclipse

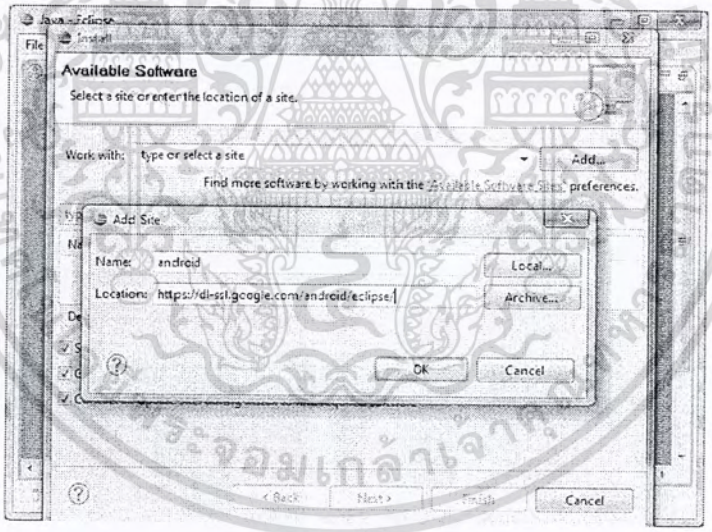
การพัฒนาจะเป็น Eclipse สามารถดาวน์โหลดได้ที่ http://www.eclipse.org/downloads/ แนะนำให้ดาวน์โหลด Eclipse Galileo ซึ่งการลง Android SDK สามารถลงได้ดังนี้

1) ดาวน์โหลดตัว SDK จาก http://developer.android.com/sdk/index.html โดยเลือกให้ตรงกับ platform ที่เราใช้พัฒนา หลังจากดาวน์โหลดมาแล้วให้แตกไฟล์ออกมาจะได้เป็นชื่อ android-sdk-<machine-platform> ในที่นี้ของเราใช้ windows ชื่อโฟลเดอร์ที่ได้ก็จะเป็น android-sdk-windows

2) เมื่อเปิด Eclipse ขึ้นมาไปที่เมนู เลือก Help -> Install New Software จะมีหน้าต่างใหม่ขึ้นมากดปุ่ม Add จะมีช่องให้กรอกข้อมูล

Name: ใส่ Android Plugin (จริงๆใส่อะไรก็ได้ แต่ให้อ่านแล้วเข้าใจก็พอ)

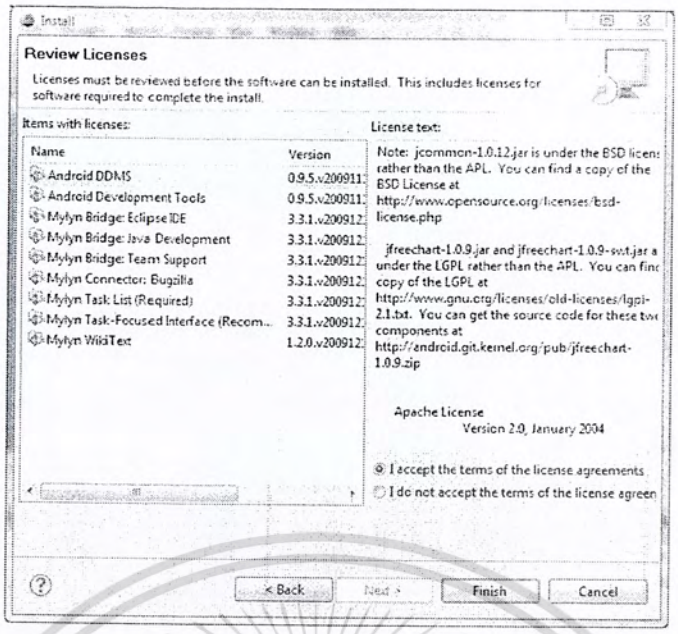
Location: ใส่ https://dl-ssl.google.com/android/eclipse/



รูป ก.1 การใส่ plugin

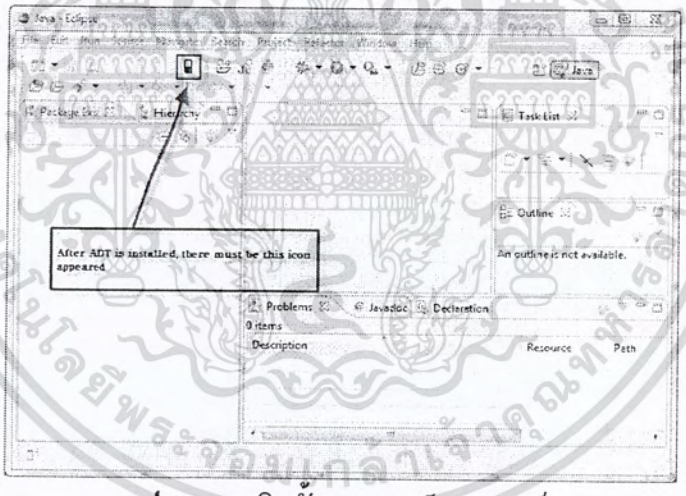
3) จากนั้นกด OK แล้วหน้าต่าง Available Software ในพื้นที่ว่างๆตรงกลางจะมี Developer Tool โผล่ขึ้นมา ให้ลองกดลูกศรหน้า Developer Tools จะมีอีกสองอันโผล่มาคือ Android DDMSกับ Android Development Tools ให้เลือกทั้งหมดแล้วกด Next รอมันจนกว่าจะมีให้กด Next อีกครั้ง พอกดแล้วหน้าต่างจะแบ่งออกเป็น 2 ฝั่ง ฝั่งซ้ายเป็นรายของ Software ที่เราจะติดตั้งด้านขวาเป็น Agreement ให้เราเลือก Accept แล้วกด Finish รอจนมัน Download และ Install เสร็จ มันจะให้ Restart Eclipse ใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป ก.2 การInstall ADT

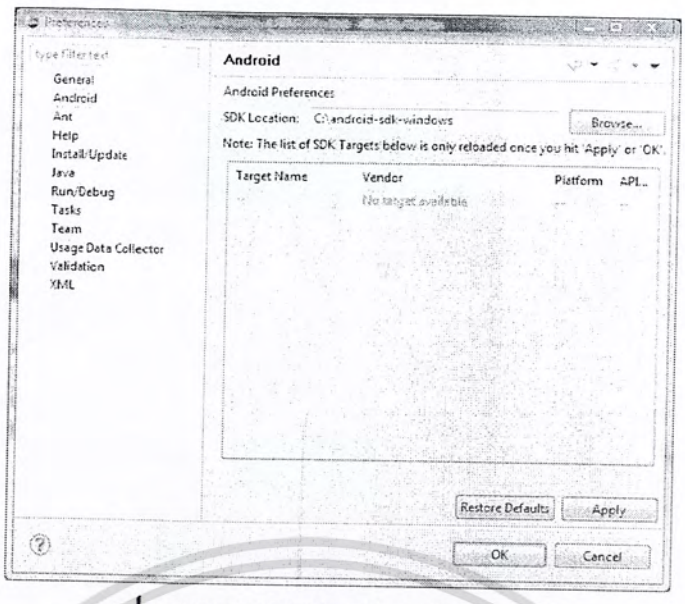
4) หลังรีบูตเสร็จ เราจะเห็นไอคอนแบบนี้ ที่ Eclipse แสดงว่า เราติดตั้ง ADT สมบรูณ์



รูป ก.3 การติดตั้ง ADT เสร็จสมบรูณ์

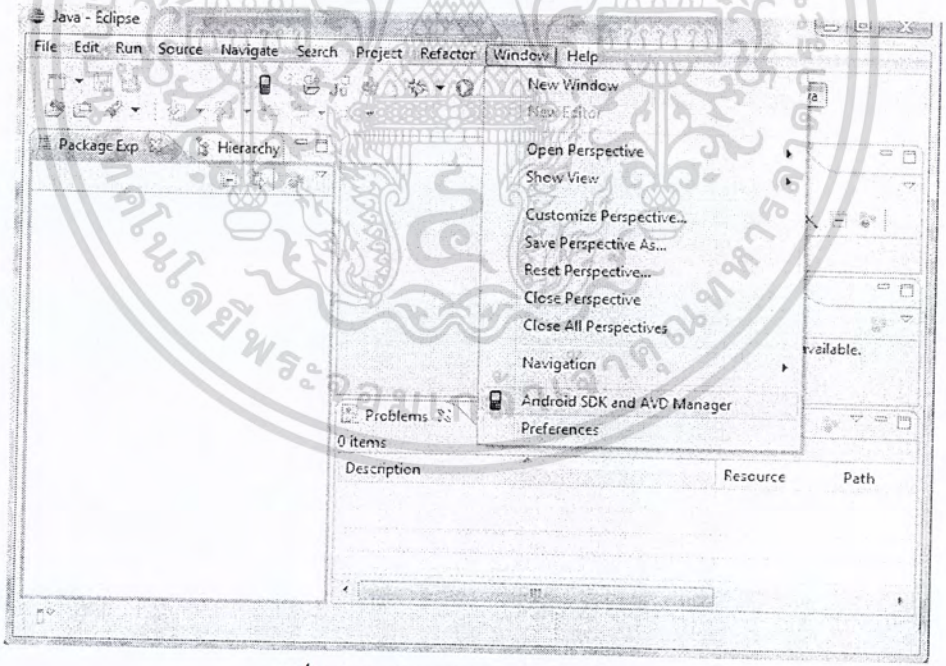
5) หลังจากติดตั้งเสร็จสมบูรณ์แล้ว ไปที่เมนู window ->Preference จากนั้นกด Android Tab ด้านซ้ายมือ มันจะขึ้นเตือนว่า เรายังไม่มีการระบุ path ของ android sdk ให้กดปุ่ม Browse เพื่อระบุ directory path ที่เรา unzip android sdk ก่อนหน้านี้ ในรูปนี้คือ C:\android-sdk-windows แล้วกด Ok เลยเป็นอันเสร็จการ ลิงค์ ADT กับ Android SDK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป ก.4 การลง Path ของ Android ADT

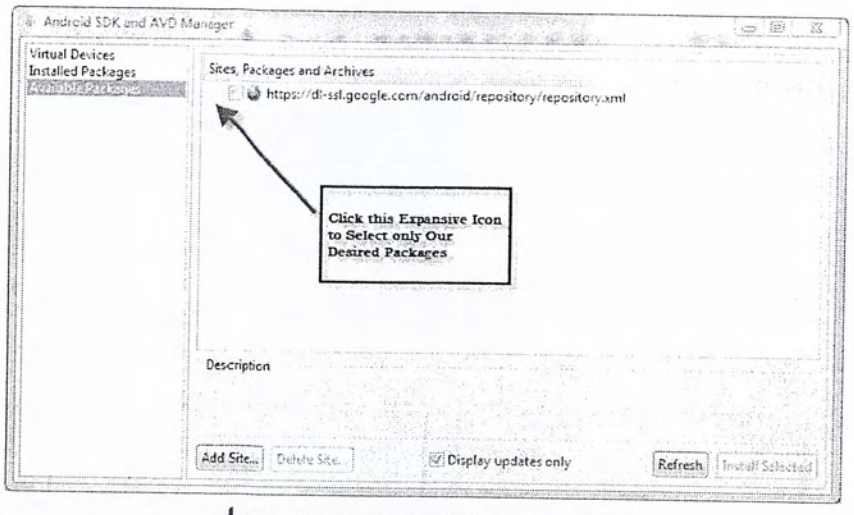
6) ขั้นตอนนี้คือการติดตั้ง Android Platform ที่เราจะใช้ในการพัฒนา ในกรณีนี้เราติดตั้ง Android Platform SDK 2.1 ไปที่ เมนู Window และ เลือก Android SDK and AVD Manager ตามรูป



รูป ก.5 กางลงplatform SDK 2.1

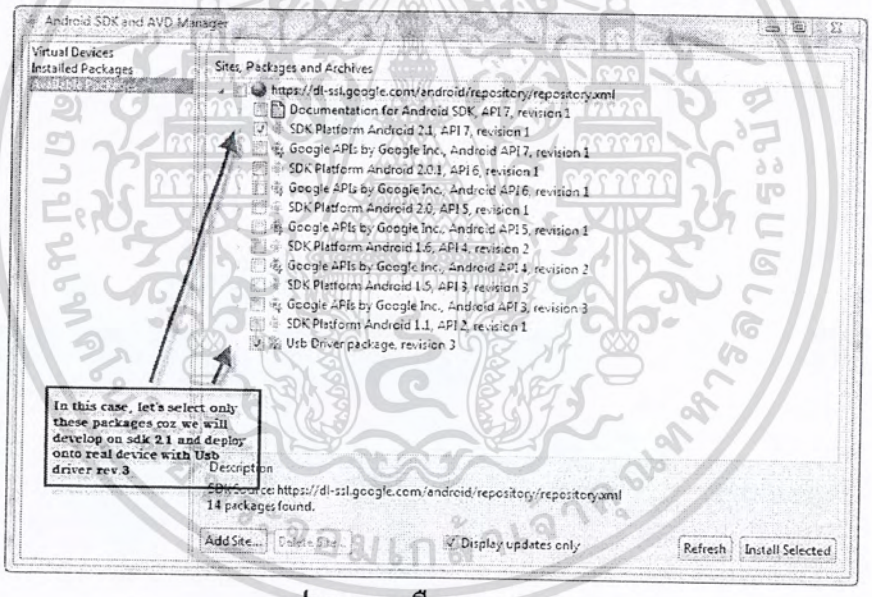
7) ที่ด้านซ้ายมือ เลือก Available Packages เพื่อเตรียมเตรียมเลือก Android SDK Package ที่เราจะลงกด ไอคอน ที่ทำเครื่องหมายในรูป เพื่อเลือก packages ภายใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป ก.6 การลง Android SDK Package

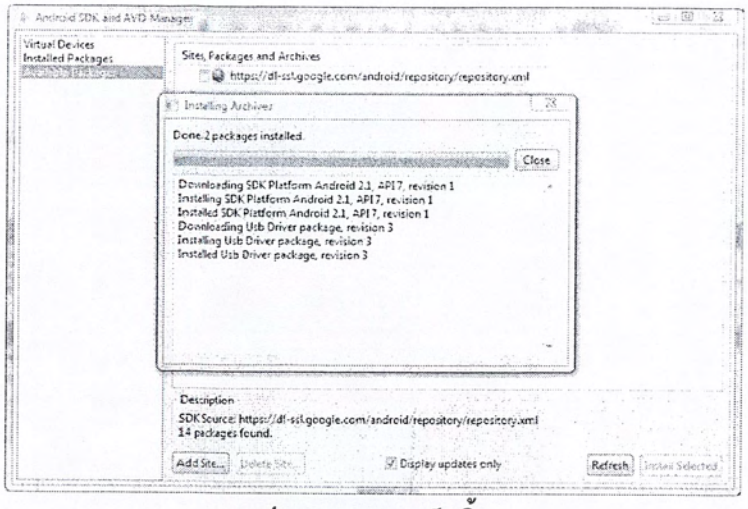
8) เลือก packages ที่เราจะพัฒนา พร้อม กับ Latest USB Driver Version สำหรับติดต่อกับเครื่องมือถือจริงๆ กดปุ่ม Install Selected



รูป ก.7 การเลือก Packages

9) เลือก Accept All แล้วกด Install Accepted รอมัน Install จนเสร็จ ถ้ามันเสร็จแล้ว มันจะมีหน้าตาประมาณนี้ ให้ปิด DialogBox นี้ และ รีบูต Eclipse หนึ่งที เป็นอันเสร็จแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป ก.8 การลงเสร็จสิ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้